IntechOpen

Human Robot Interaction

Edited by Nilanjan Sarkar

Human-Robot Interaction

Edited by
Nilanjan Sarkar

Human Robot Interaction
http://dx.doi.org/10.5772/51
Edited by Nilanjan Sarkar

© The Editor(s) and the Author(s) 2007
The moral rights of the and the author(s) have been asserted.
All rights to the book as a whole are reserved by INTECH. The book as a whole (compilation) cannot be reproduced, distributed or used for commercial or non-commercial purposes without INTECH's written permission.
Enquiries concerning the use of the book should be directed to INTECH rights and permissions department (permissions@intechopen.com).
Violations are liable to prosecution under the governing Copyright Law.

Individual chapters of this publication are distributed under the terms of the Creative Commons Attribution 3.0 Unported License which permits commercial use, distribution and reproduction of the individual chapters, provided the original author(s) and source publication are appropriately acknowledged. If so indicated, certain images may not be included under the Creative Commons license. In such cases users will need to obtain permission from the license holder to reproduce the material. More details and guidelines concerning content reuse and adaptation can be foundat http://www.intechopen.com/copyright-policy.html.

Notice

Statements and opinions expressed in the chapters are these of the individual contributors and not necessarily those of the editors or publisher. No responsibility is accepted for the accuracy of information contained in the published chapters. The publisher assumes no responsibility for any damage or injury to persons or property arising out of the use of any materials, instructions, methods or ideas contained in the book.

First published in Croatia, 2007 by INTECH d.o.o.
eBook (PDF) Published by IN TECH d.o.o.
Place and year of publication of eBook (PDF): Rijeka, 2019.
IntechOpen is the global imprint of IN TECH d.o.o.
Printed in Croatia

Legal deposit, Croatia: National and University Library in Zagreb

Additional hard and PDF copies can be obtained from orders@intechopen.com

Human Robot Interaction
Edited by Nilanjan Sarkar

p. cm.

ISBN 978-3-902613-13-4

eBook (PDF) ISBN 978-953-51-5818-9

We are IntechOpen, the world's leading publisher of Open Access books

Built by scientists, for scientists

4,400+
Open access books available

118,000+
International authors and editors

130M+
Downloads

Our authors are among the

151
Countries delivered to

Top 1%
most cited scientists

12.2%
Contributors from top 500 universities

WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com

Preface

Human-Robot Interaction (HRI) has been an important research area during the last decade. As robots have become increasingly more versatile, they have been useful in a variety of applications where the robots and human are expected to interact very closely. It has thus become necessary to clearly understand the nature of human-robot interaction so that new tools and techniques can be developed to allow human and robot to work together in a seamless, robust and natural manner.

Human-robot interaction research is diverse and covers a wide range of topics. All aspects of human factors and robotics are within the purview of HRI research so far as they provide insight into how to improve our understanding in developing effective tools, protocols, and systems to enhance HRI. For example, a significant research effort is being devoted to designing human-robot interface that makes it easier for the people to interact with robots. Researchers are investigating how to improve the current interfaces so that multiple users can interact with multiple robots simultaneously. There are ongoing efforts to understand how to effectively incorporate multi-modal interaction capabilities within HRI. The role of emotion in HRI is also another important research area so that human and robot can understand each other's affective expressions. Developing suitable control architecture that facilitates versatile human-robot teaming and collaboration is essential for HRI. Such a control architecture proposes the basis for interaction and drives further research into sensor development, communication modes and protocols, and system development. Additionally, a large part of HRI research involves applying and learning lessons from various human-robot applications.

It is neither possible nor is it our intention to cover every important work in this important research field in one volume. HRI is an extremely active research field where new and important work is being published at a fast pace. However, we believe that HRI as a research field has matured enough to merit a compilation of the outstanding work in the field in the form of a book. This book, which presents outstanding work from the leading HRI researchers covering a wide spectrum of topics, is an effort to capture and present some of the important contributions in HRI in one volume. I hope that this book will benefit both experts and novice and provide a thorough understanding of the exciting field of HRI.

Editor
Nilanjan Sarkar
Vanderbilt University
USA

Contents

1

Adaptive Personal Space for Humanizing Mobile Robots

Janaka Chaminda Balasuriya[1], Chandrajith Ashuboda Marasinghe[2] and Keigo Watanabe[1]
[1]*Department of Advanced Systems Control Engineering, Saga University*
[2]*Department of Management and Information Systems Science, Nagaoka University of Technology*
Japan

1. Introduction

Human beings are fascinating creatures. Their behavior and appearance cannot be compared with any other living organism in the world. They have two distinct features with compared to any other living being; unique physical nature and emotions / feelings. Anybody who studies on humans or tries to construct human like machines should consider these two vital facts. When robots are interacting with humans and other objects, they certainly have a safe distance between them and the object. Some of the problems in concern are how can this distance be optimized when interacting with humans; will there be any advantages over achieving this; will it help to improve the condition of robots; can it be a mere constant distance; how will the humans react, etc. In order to "humanize" robots, they (robots) should also have certain understanding of such emotions that we, humans have. The present main objective is to "teach" one such human understanding, commonly known as "personal space" to autonomous mobile robots.

As Simmons et al., 1997 described, recent research in mobile robot navigation has utilized autonomous mobile robots in service fields. To operate them in an environment with people, it requires more than just localization and navigation. The robots should recognize and act according to human social behavior to share the resources without conflict (Nakauchi & Simmons, 2002). Sometimes, even when humans interact with each other, it leads to resource conflict. At such times humans use social rules to maintain order (Sack, 1986).

The comfort level of the humans for which the robot is working will be very important if the robot is to do its job effectively. Extensive research is being performed in the area of robotics to improve the conditions of the robots, advancing the abilities to do specific tasks, motion planning, etc. However, very little work has been performed in trying to understand how people would interact with a robot, how to make them comfortable, factors that make uncomfortable or threatening, methods or ways for robots to indicate their feelings, etc. to analyze the aesthetic qualities of the behavior patterns of robots (Nakauchi & Simmons, 2002).

As in the very beginning of the mobile robotic systems, there had been some kind of distance or space between the robots to any other object in the vicinity. This was just a mere distance for safety for easy maneuvering and for collision avoidance. As Stentz, 1996 and

many others had mentioned, this was just a constant of space. This concept was quite acceptable for the systems such as transporting, surveillance and monitoring, etc. In other words, such kind of safe distance was good for non-human interacting purposes. Can the same be applied for human interaction? Although it will give some results, it will not enhance or optimize the real requirement in need, i.e. to build up harmonious relationship with humans.

When Nakauchi and Simmons, 2002 studied about personal space and applied it to moving robots, it was shown that there were some improvements over the "blind" safe distance. They had experimented using human subjects for "correct distance" or "personal space" in order to have pleasant feeling towards two interacting parties. For the experiment, it was assumed that;

- The size of personal space of the person in front is identical to the personal space of the subject.
- When the same body direction of two people is facing, the personal space towards that direction is the half of the distance between two people.

According to the above two assumptions, the average size of the personal space was estimated. This experimentally derived average personal space had an approximate shape of an oval that is larger towards the front. Although those results were approximate, it had been aligned with the values that were reported in the cognitive science literature (Malmberg, 1980).

Another set of experiments were conducted by Walters et al., 2005 using adults and children with a robot of mechanistic appearance called PeopleBot^R to find the personal space zones, initial distances between robot and humans etc., in the context of the encounters and the human's perception of the robot as a social being. They had found out that the children showed a dominant response to prefer the "social zone" (as shown in Table 1), comparable to distances people adopt when talking to other humans. From the adult studies, they found that, a small number of people preferred the "personal zone" though significant minorities deviate from this pattern.

They also tried to compare human-robot interpersonal distances with human-human interpersonal distances as described by Hall, 1966. According to Hall, 1966 the generally recognized personal space zones between humans are well known and are summarized (for Northern Europeans) in Table 1.

Zone	Range [m]	Interaction
Intimate	0 -- 0.15	Loved ones
Close	0.15 -- 0.45	Close friends
Personal	0.45 -- 1.20	Friends
Social	1.2 -- 3.60	Strangers
Public	3.60 +	Public

Table 1. Personal space zones

In this research project we try to construct a determination system of adaptive personal space (PS) based on adaptive neural fuzzy inference system (ANFIS). In section 2 we analyze some previous work, which encouraged us to pursuit the path of personal space and to find certain parameters. In section 3 suitability of using ANFIS for constructing an adaptive PS and experimental procedure to gather data are discussed. Section 4 describes the input and output variables and the rule structure. Sections 5 and 6 give the detailed

construction of an ANFIS architecture and the procedures of training, checking and testing of it. Section 7 gives some proposal to overcome certain practical limitations during the implementation of the adaptive PS. Section 8 discusses some acceptable way of assigning values for the "appearance" input variable. Finally, section 9 summarizes our efforts giving limitations and identifying future work.

2. Variation of Personal Space

Although it is possible to find a personal space for a specific instance of environment, it is highly volatile depending on the two interaction parties and not definitely a constant. As Walters et al., 2005a suggested, different robot social models, perhaps with very different initial personalities, may be more acceptable to different users (e.g. a discrete servant or even a silent servant, with no obvious initiative or autonomy). They further stated that it probably cannot be assumed that people automatically treat robots socially, apart from simple elements of anthropomorphism as described by Reeves & Nass, 1998. A user-friendly robot should automatically refine and adapt its social model (personality) over a longer period of time, depending on information about and feedback from users and the robots own autonomous learning system. For example, adjustments of social distances according to a user's personality trait will be a promising direction (as proposed by Walters et al., 2005b) towards a true robot companion that needs to be individualized, personalized and adapt itself to the user (Dautenhahn, 2004).

According to Sack, 1986 and Malmberg, 1980, it is reported that the actual size of the personal space at any given instance varies depending on cultural norms and on the task being performed. For a simplified scenario for experimental analysis, appearance (mainly of the robot), previous acquaintance or familiarity of the either parties, gender, age, height of the bodies (specially interaction in the standing position), emission of any sound, emotions on the face, carrying objects, etc. were considered to be important.

Hence in this research project, construction of an automated system to generate a most suitable personal space for any environmental condition is attempted. In order to do that, from the list of above, the following three parameters namely, height (H), appearance (A), and familiarity (F) were considered (as the initial stage for simplicity) to generate an active personal space (PS) and the block diagram is shown in Fig. 1.

Input parameter height analyzes the height of the human who comes closer to a robot for interaction as meeting a very tall or very short person is little bit difficult than an ordinary person. The outer appearance of the robot, whether it looks more like a human or rather like a machine is analyzed in input variable appearance. Familiarity is the closeness that the both parties in interaction have for each other, i.e. if they are more familiar they will keep closer and vice versa.

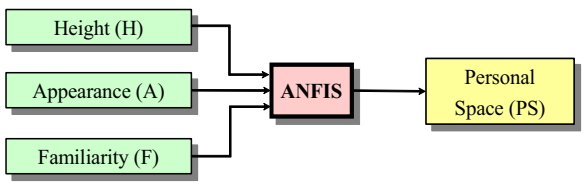


Figure 1. Block diagram of generating PS through adaptive neural fuzzy inference system

3. ANFIS for Personal Space Determination

Adaptive neural fuzzy inference system or simply ANFIS can be used as a basis for constructing a set of fuzzy if-then rules with appropriate membership functions to generate the desired input-output combination (Jang, 1993). It is especially useful when needed to apply a fuzzy inference to already collected input-output data pairs for model building, model following, etc. where there are no predetermined model structures based on characteristics of variables in the system.

3.1 Gathering data

Considering the procedure as Nakauchi & Simmons, 2002 or Walters et al., 2005a to obtain a sense of personal space for robot-human interaction, a similar experimental condition was constructed. Here a robot (or a model) is kept at the end of a scaled line in a room and a human is asked to move closer to it.

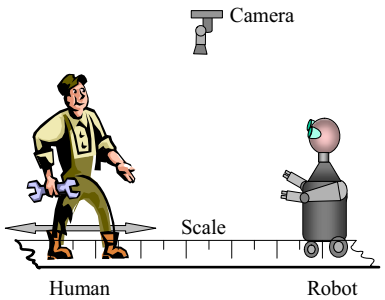


Figure 2. Experimental setup

3.2 Experimental procedure

As the experiment proceeds, one human subject is instructed to move towards the robot as if he needs to talk with it. The human subject is asked to be along the scaled line and look at the robot face and move closer to it until he feels safe enough to make conversation with it as shown in Fig. 2. In the mean time the robotic model was positioned so as to make its face towards the human subject. During the whole time of the experiment, the robot did not do anything and the human subject did all the active tasks of walking, thinking, etc. The robot and the human subject, one person at a time, were supposed to interact at specific duration of time and it ended once the human subject stops in front of the robot. Then the distance between the two parties was obtained by using a camera or by direct human observer (who reached the two parties once they got stabilized). The human subject had no previous experience with the robot and the authors wanted the human subjects to be curious as well as cautioned about the robot that they are going to meet. In other wards human subjects had no idea what kind of robotic system that they are going to face with or any capabilities that it possesses until they meet the robot.

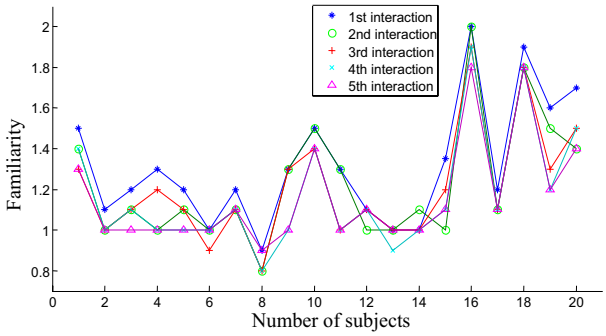


Figure 3. Personal space variation of each interaction with Robot A

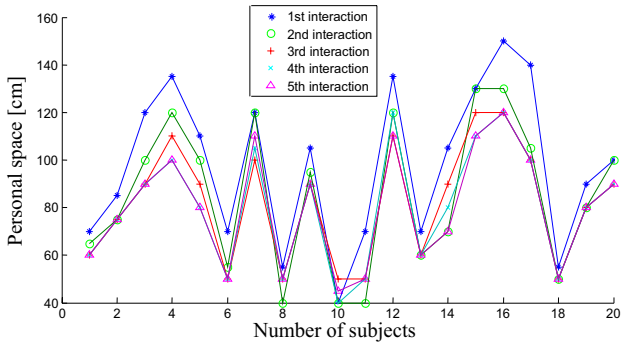


Figure 4. Personal space variation of each interaction with Robot B model

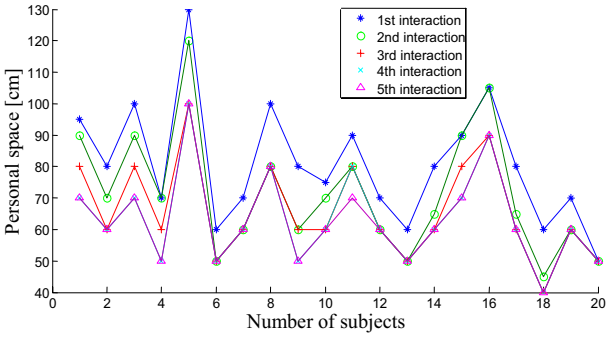


Figure 5. Personal space variation of each interaction with Robot C

Figure 5. Personal space variation of each interaction with Robot C This procedure was repeated for several rounds in order to ascertain any change of personal space due to familiarity of the robot. The data thus obtained clearly indicated the reduction of personal space with many acquaintances with the same robotic system as shown in Figs 3, 4 and 5 for robots A, B and C respectively. Starting from a large space (comparatively) got reduced with each attempt but saturated after some time. That is, after few interactions, the subject tends to keep the distance as the same. (The distance never reached zero with increased number of attempts).

Robot A Robot B Robot C

Figure 6. Robots and models used for the experiment

The robots and a robotic model used in these experiments are Robot A (PA10 robotic manipulator), Robot B (robotic model), and Root C (previously known as Carry Hospital Robot (Jin et al., 1993) re-used with several modifications) as shown in Fig. 6. The first one was a stationary robot with 200 cm in height and 20 cm average diameter, the next was a movable robot model with 100 cm height, 50 cm diameter and around 3 kg, and the last is also a movable robot with 170 cm height, generalized circular diameter of 60 cm and weight of about 25 kg. The data gathered are grouped for training, checking and testing for the ANFIS and are described in later.

4. Input and Output Variables

Out of the three input variables considered, height (of the human) and familiarity were two straightforward measurements while appearance was taken as a collective decision.

4.1 Input variable "height"

The height of the human subject is considered in this input variable. The universe of discourse of the input variable "height (H)" was considered to be 50 cm to 200 cm, having three membership functions "big (B)," "medium (M)," and "small (S)." Those shapes were considered to be bell type.

4.2 Input variable "appearance"

The robot outer appearance with compared to a human (or rather closeness of the robot body to the human body) is considered in this input variable. Humans are more like to reach one of their own looking rather than to that of very peculiar shaped objects. Human like robot gets the closer feeling with respect to the other crude or rather machine looking

robot. The universe of discourse of the input variable "appearance (A)" was considered to be 1 to 5 (without any units), having three membership functions "big (B)," "medium (M)," and "small (S)." Those shapes were considered to be all bell type. Here the appearance value for the Robot A was given as 1, for the Robot B as 2, and Robot C as 5. Although more human like robots were required to get the records, at the time of the experiment, such robots were not available in the laboratory. Hence the universe of discourse was taken as 1 to 5.

4.3 Input variable "familiarity"

As the name implies, the way that a human subject interacts with a robot is analyzed in this variable. Namely, if a human feels more familiar with a certain robot, he will go closer to it. If there are many interactions with the same robot for many times, it is fair to assume that the two interaction parties get more familiar with each other. Due to its complexity of nature of assessing this familiarity for a certain interaction, familiarity value was taken as dividing the interaction distance by 100 for a particular situation. Therefore, more familiar interaction will have a less "familiarity" value and vice versa. Keeping in mind that more interactions mean more familiar with each other, this variable is to be set. The universe of discourse of the input variable "familiarity (F)" was considered to be 0 to 2 (without any units), having three membership functions "big (B)," "medium (M)," and "small (S)," whose forms were considered to be bell type.

R_1 : If H is S and A is S and F is S then PS is PS_1
R_2 : If H is S and A is S and F is M then PS is PS_2
R_3 : If H is S and A is S and F is B then PS is PS_3
R_4 : If H is S and A is M and F is S then PS is PS_4
R_5 : If H is S and A is M and F is M then PS is PS_5
R_6 : If H is S and A is M and F is B then PS is PS_6
R_7 : If H is S and A is B and F is S then PS is PS_7
R_8 : If H is S and A is B and F is M then PS is PS_8
R_9 : If H is S and A is B and F is B then PS is PS_9
R_{10}: If H is M and A is S and F is S then PS is PS_{10}
.
.
.
R_{18}: If H is M and A is B and F is B then PS is PS_{18}
R_{19}: If H is B and A is S and F is S then PS is PS_{19}
.
.
.
R_{27}: If H is B and A is B and F is B then PS is PS_{27}

Figure 7. Rule base for the ANFIS architecture

4.4 Output variable "personal space"

Considering the above three inputs that will generate 27 rules as shown in Fig. 7 in total and each having unity weight for each rule, the output variable "personal space (PS)" of the ANFIS is obtained using the weighted average defuzzification.

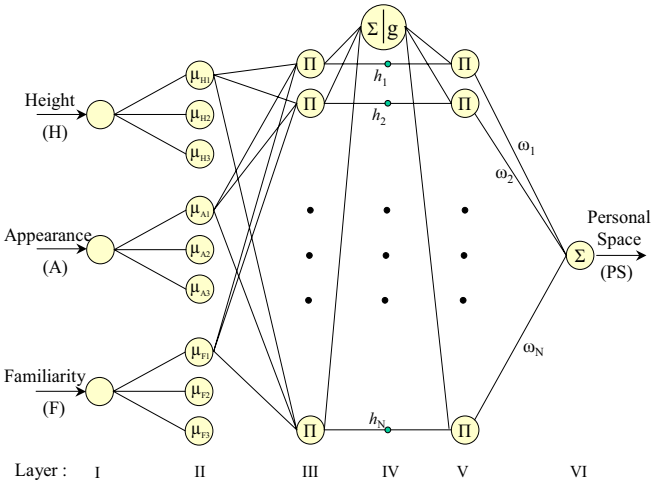


Figure 8. Active PS determination network with ANFIS

5. ANFIS Architecture

The architecture of the active PS determination network is illustrated in Fig. 8. Layer I to layer III represent the antecedent part of the fuzzy neural network, whereas layer V and layer VI represent the consequence part (Jang and Sun, 1995).
As shown in Fig. 08, the domain of discourse of height (H) is described by fuzzy variable H with p number of linguistic values (p = 3), the domain of discourse of appearance (A) is described by fuzzy variable A with q number of linguistic values (q = 3), and the domain of discourse of familiarity (F) is described by fuzzy variable F with r number of linguistic values (r = 3). Hence each input variable is unique in the sense of domain of discourse. It is assumed that each node of the same layer has a similar function, as described below. Here we denote the output of the i^{th} node in layer X as $O_{X,i}$.

Layer I:
Layer I consists of three types of nodes; height (H), appearance (A) and familiarity (F). The current value of height (H), i.e., the crisp input to the height node is H_i, appearance node is A_j and familiarity node is F_k. No computation is carried out at this layer.

Layer II:
This layer acts as the fuzzification layer of the fuzzy neural network. At this layer, the output of a node connected to the current value of input variable acquires the fuzzy membership value of the universe of discourse. Every node i, where $i = \{1,\ldots,p\}$ (or q or r), in this layer is an adaptive node with a node function

$$O_{II,I}=\mu_{Xi,}(x) \tag{1}$$

where x is the input to node i, and X_i is the linguistic label (big, medium, small, etc.) associated with this node function. In other words, $O_{II,i}$ is the membership function of X_i and it specifies the degree to which the given x satisfied the quantifier Xi. Hence the output from the 2nd layer will be:

$$O_{II,p}=\mu_{Hp}(H_i) \tag{2}$$

$$O_{II,q}=\mu_{Aq}(A_j) \tag{3}$$

$$O_{II,r}=\mu_{Fr}(F_k) \tag{4}$$

for height, appearance and familiarity respectively.

Layer III:

In this layer, the nodes labeled as Π compute the T-norm of the antecedent part. Thus the rule evaluates the conditions of the inputs and they are continued to the layer V for normalization. The output of any node t, where $t = \{1,\ldots,N\}$, where $N=p\times q\times r$, in this layer is described by the following equation:

$$O_{III,t}= h_t= \mu_{Hp}(H_i) \times \mu_{Aq}(A_j) \times \mu_{Fr}(F_k) \tag{5}$$

where h_t represents the firing strength of the t^{th} rule and there are N such rules as total.

Layer IV:

The first node of layer IV at fuzzy neural network, which has symbols Σ and g, generates the output through the following function:

$$g(x)=\frac{1}{x} \tag{6}$$

with a linear summed input. Then the output of the first node of layer IV is given by

$$O_{IV,1}=\frac{1}{\sum_{t=1}^{N} h_t} \tag{7}$$

Other nodes just carry forward the outputs of previous nodes to the next layer.

Layer V:

This layer normalizes the fired rule values. Each node labeled as Π in this layer multiplies the value carried forward by previous node with the output of the first node at layer IV. Then the output of any m^{th} node of this layer can be given by the following equation:

$$O_{V,m}=\frac{h_m}{\sum_{t=1}^{N} h_t} \tag{8}$$

Layer VI:

Layer VI is the defuzzification layer of the fuzzy neural network. The node labeled as Σ in this layer calculates the overall output as the summation of all incoming signals. Then the personal space value for certain input variable is given by:

$$O_{VI} = PS = \frac{\sum_{m=1}^{N} W_m h_m}{\sum_{n=1}^{N} h_n} \tag{9}$$

where w_m denotes a constant value in the consequence part of the m^{th} rule. The overall output is the weighted mean of w_m with respect to the weight h_m.
The connection weights are trained by applying the hybrid algorithm. The error tolerance was set to zero.
The error is calculated by comparing the output of the expert knowledge with that of fuzzy neural network for the same input data, x. The adaptation of the m^{th} weight, w_m, at the l^{th} time step is given by the following equation:

$$W_m(l+1) = W_m(l) + \gamma[y_d - y_a]\frac{h_m}{\sum_{n=1}^{N} h_n} \tag{10}$$

where γ represents a small positive learning rate having the initial value of 0.01 which was set to be an adaptive during the training process, and y_d and y_a represent the desired output and actual output respectively for the personal space value selected for the training. The ANFIS was trained setting the error tolerance to zero for forty epochs.

6. Training, Checking and Testing Data Sets

Using the collected data from the experiments, data were rearranged into three groups for training, checking and testing purposes of the active PS with ANFIS. Having considered 5 groups of height, 3 sets of appearances and 5 different interactions make the total data set of 75 (i.e., 5×3×5=75). These are shown in Fig. 9.

6.1 Train data set

Train data set was obtained by grouping the input variable "height (H)." First, the height was categorized into five groups as:

- 161 cm to 165 cm
- 166 cm to 170 cm
- 171 cm to 175 cm
- 176 cm to 180 cm
- 181 cm to 185 cm

Then for a particular height group average is taken for each attempt of interaction.
"Familiarity" was obtained as described below:
The mean familiarity $\overline{F}_i^j$ for each interaction can be defined by

$$\overline{F}_i^j = \frac{1}{N_j}\sum_{l=1}^{N_j} F_i^j(l) \tag{11}$$

where $\overline{F}_i^j(l)$ is a familiarity value to the i^{th} robot designated by the l^{th} subject who was classified into the j^{th} human group with different range of height and N_j is the total number of subjects in the j^{th} group.

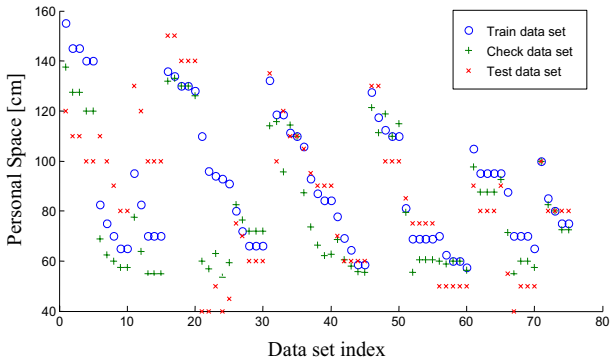


Figure 9. Training, checking and testing data for the active PS using ANFIS

According to the above criteria, obtained results for the height group one (where there were only two human subjects fallen into this category) for each robot are shown in Tables 2, 3, and 4 for robots A, B, and C respectively, where Dist. is the distance between the robot and the subject in [cm], Fam. is the familiarity and Avg. is the averaged value. The complete set of results thus obtained was used as the train data set.

Std. No#	Height [cm]	Int. 1		Int. 2	
		Dist.	Fam.	Dist.	Fam.
18	164	190	1.9	180	1.8
5	165	120	1.2	110	1.1
Avg.	164.5	155	1.6	145	1.5

Int. 3		Int. 4		Int. 5	
Dist.	Fam.	Dist.	Fam.	Dist.	Fam.
180	1.8	180	1.8	180	1.8
110	1.1	100	1	100	1
145	1.5	140	1.4	140	1.4

Table 2. Rearranged data for robot A (Appearance 1), Height group 1 (161 -- 165 [cm])

Std. No#	Height [cm]	Int. 1		Int. 2	
		Dist.	Fam.	Dist.	Fam.
18	164	55	0.6	50	0.5
5	165	110	1.1	100	1
Avg.	164.5	82.5	0.8	75	0.8

Int. 3		Int. 4		Int. 5	
Dist.	Fam.	Dist.	Fam.	Dist.	Fam.
50	0.5	50	0.5	50	0.5
90	0.9	80	0.8	80	0.8
70	0.7	65	0.7	65	0.7

Table 3. Rearranged data for robot B (Appearance 2), Height group 1 (161 -- 165 [cm])

Std. No#	Height [cm]	Int. 1		Int. 2	
		Dist.	Fam.	Dist.	Fam.
18	164	60	0.6	45	0.5
5	165	130	1.3	120	1.2
Avg.	164.5	95	1	82.5	0.8

Int. 3		Int. 4		Int. 5	
Dist.	Fam.	Dist.	Fam.	Dist.	Fam.
40	0.4	40	0.4	40	0.4
100	1	100	1	100	1
70	0.7	70	0.7	70	0.7

Table 4. Rearranged data for robot C (Appearance 5), Height group 1 (161 -- 165 [cm])

6.2 Check data set

In order to optimize the ANFIS once created using the train data set and to overcome the problem of model over fitting during the training process, another data set which does not contain the similar values but close to the original train data output values is required. For this purpose, following equation was considered to generate the check data set. When obtaining the check data set $\{x_c\}$, i.e., considering a particular column, average value x_a is subtracted from the maximum value x_{max} from the gathered data. Then half of that value is added to the minimum value x_{min} of the gathered data in the same column:

$$x_c = \left[\frac{x_{\max} - x_a}{2} \right] + x_{\min} \tag{12}$$

In this process, no change was made to the previous ANFIS structure.

6.3 Test data set

Once the ANFIS system is created using train data and fine tuned with check data, it is necessary to analyze its credibility for correct functioning and desired performance (Matlab). For that purpose another set of data called as test data is used. Construction of test data set was done as follows: Considering the output value of training and checking data sets for a same input entry, very close data value from the original data set of the experiment was selected and grouped to form the test data set. This set was used to test the ANFIS for desired functionality.

The error cost described by

$$e = \left[\frac{aps - ps}{ps} \right] \times 100\% \tag{13}$$

for the trained adaptive PS (*aps*) with that to the original values (*ps*) was calculated and is plotted in Fig. 10. The trained ANFIS output values with the data set values used to train, check, and test of it are shown in Figs. 11, 12, and 13 respectively.

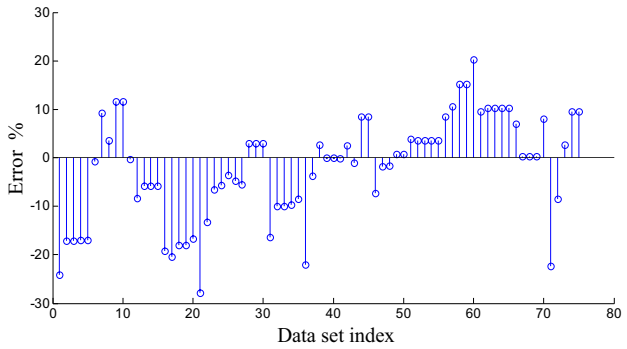


Figure 10. Error percentage of the trained ANFIS output

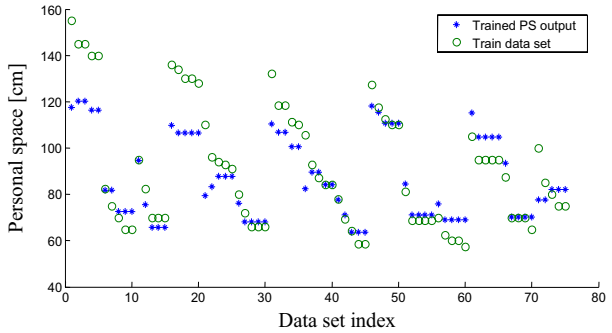


Figure 11. Trained ANFIS output with train data set values

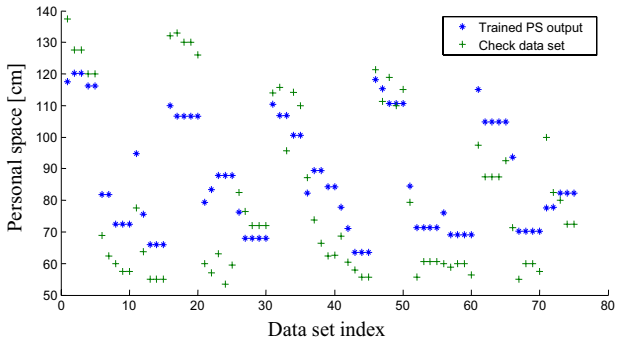


Figure 12. Trained ANFIS output with check data set values

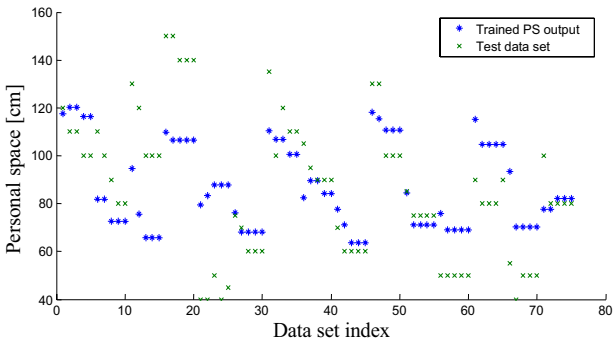


Figure 13. Trained ANFIS output with test data set values

7. Proposal to the Implementation of ANFIS

The ANFIS trained by using the data set as described in the preceding section can not be directly installed into an actual robot, because the input data, appearance (*A*) and familiarity (*F*), to the ANFIS will not be able to be collected through the robot. Therefore, in this section, we propose an ANFIS that will be implementable to a robot using only the incoming data of human height, which will be readily available from the robot by using any camera, together with the average data of appearance and familiarity for each human group with different range of height. This concept is also divided into two classes, depending on the usage of different average data of appearance and familiarity for each robot or on the usage of same average data of them for all robots.

7.1 A case with different mean appearance and familiarity for each robot

For this case, different average data of appearance and familiarity are used for the ANFIS of each robot. So, let $\overline{A}_i^j$ denote the mean appearance that was designated by any subject who was grouped into *j* for the specific robot *i* described by

$$\overline{A}_i^j = \frac{1}{N_j}\sum_{l=1}^{N_j} A_i^j(l) \tag{14}$$

where $A_i^j(l)$ is an appearance value to the i^{th} robot designated by the l^{th} subject who was classified into the j^{th} human group with different range of height and N_j is the total number of subjects in the j^{th} group.

Similarly, the mean familiarity $\overline{F}_i^j$ can be defined by

$$\overline{F}_i^j = \frac{1}{N_r}\sum_{r=1}^{N_r}\left(\frac{1}{N_j}\sum_{l=1}^{N_j} F_i^j(l)\right) \tag{15}$$

where $F_i^j(l)$ is a familiarity value to the i^{th} robot designated by the l^{th} subject who was classified into the j^{th} human group with different range of height, r is the interaction number that l^{th} subject took for the i^{th} robot, and N_r is the total number of such attempts.
By applying these data, we can implement the ANFIS to generate an active personal space, depending on the height of the human and the robot type. Fig. 14 shows the block diagram of ANFIS for a case with different mean appearance and familiarity for each robot.

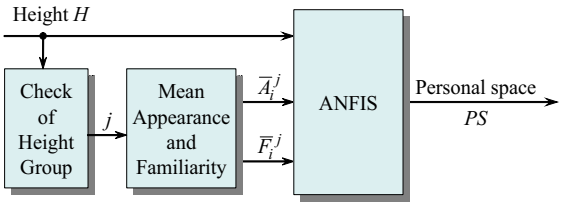


Figure 14. ANFIS for a case with different mean appearance and familiarity for each robot

7.2 A case with same mean appearance and familiarity for all robots

For this case, the identical average data of appearance and familiarity are used for the ANFIS of all robots. So, let $\overline{A}^j$ and $\overline{F}^j$ denote the mean appearance and familiarity averaged by the number of robot types that were designated by any subject who was grouped into j described by

$$\overline{A}^j = \frac{1}{M_j}\sum_{i=1}^{M_j}\overline{A}_i^j \quad \text{and} \quad \overline{F}^j = \frac{1}{M_j}\sum_{i=1}^{M_j}\overline{F}_i^j \tag{16}$$

where $\overline{A}_i^j$ and $\overline{F}_i^j$ are given by Eqs. (14) and (15), and M_j is the total number of robot types that were used for the j^{th} human group with different range of height.

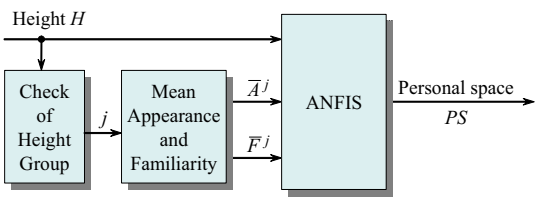


Figure 15. ANFIS for a case with the same mean appearance and familiarity for all robots

Thus, we can implement the ANFIS to generate an active personal space, depending on the height of the human. Fig. 15 shows the block diagram of ANFIS for a case with same mean appearance and familiarity for all robots.

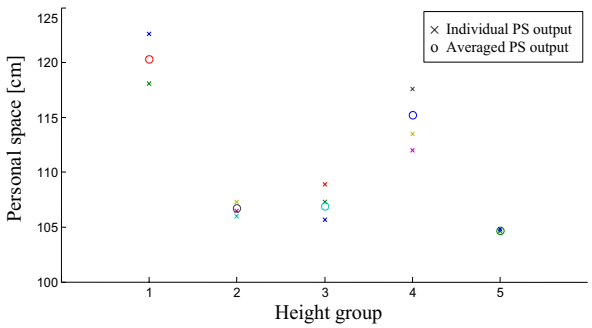


Figure 16. Personal space of each height group for Robot A

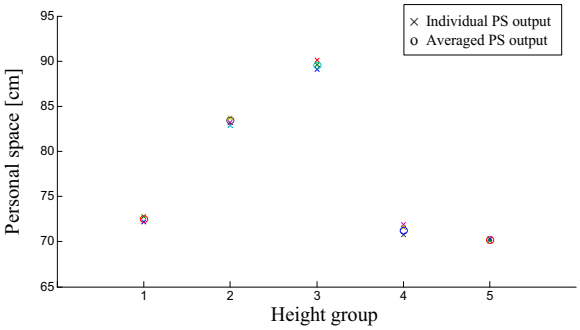


Figure 17. Personal space of each height group for Robot B

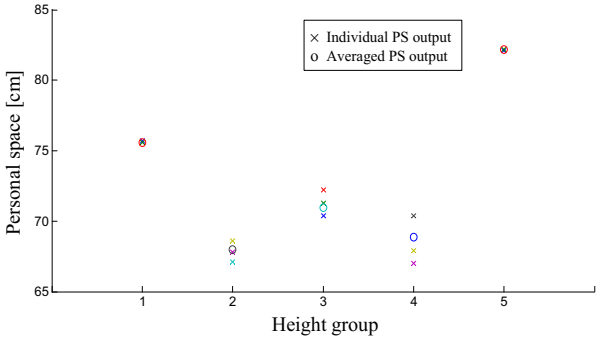


Figure 18. Personal space of each height group for Robot C

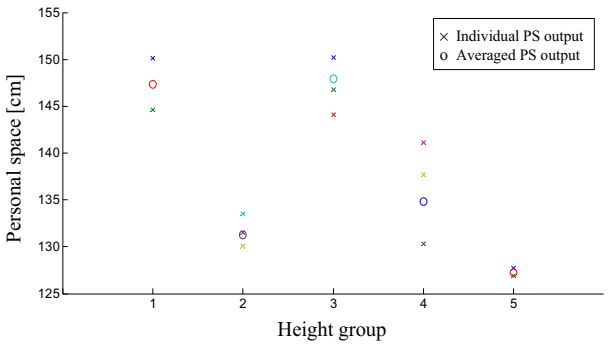


Figure 19. Personal space of each height group for any robot

The results obtained using the above simplification process for each of the height group with averaged value for the same group are plotted for each of the robots. These are shown in Figs. 16, 17, and 18 by applying the proposed method in 7.1 to Robot A, B, and C respectively. Fig. 19 shows the values obtained for all the robots after applying the proposed method in 7.2. Mean error percentage, mean squared error (MSE) and root mean squared error (RMSE) generated by each of the robots in the proposed method in 7.1 with their averaged error are tabulated in Table 5 and the comparison of errors in all the methods including the previous complete attempt is given in Table 6.

According to the error comparison as given in this table, it can be seen that although the mean error percentage of the method in 7.2 is very much higher than the other two, the comparison of MSE or RMSE with each other does not give a much of variation. Complete attempt had the lowest RMSE and the proposed method in 7.2 had the closest value to it. Even the proposed method in 7.1 had a higher value of RMSE than that of the method in 7.2. But all the methods have close RMSE values to each other. Hence it is fairly assumed that the proposed methods in 7.1 and 7.2 can be used to generate the adaptive PS as well.

	Robot A	Robot B	Robot C	Mean value
Mean %	-10.85	5.31	8.47	0.98
MSE	564.04	1251.66	537.63	784.45
RMSE	23.75	35.38	23.18	27.44

Table 5. Error in the proposed method in 7.1

	Complete attempt	Method in 7.1	Method in 7.2
Mean %	-2.09	0.98	55.12
MSE	115.09	784.45	582.97
RMSE	10.73	27.44	24.14

Table 6. Error in all methods

8. Proposal for Appearance Analysis

Input values for the "appearance" were arbitrary selected for this research project. But as it received many critics, it was necessary to apply more appropriate mechanism to obtain

"appearance" values to each of the robots. Since all these about humanizing robots, the most appropriate way to analyze "appearance" of robot was consulting the human subjects once again. That is to ask each of them to rank a robot according to the outer appearance of the robot. This was much valid as it can be applied universally for any of the robot available in the world right now. Hence such an analysis will give a way to construct a "meaningful" measuring unit to rank robot's outer appearance in the future.

8.1 Method

This method analyzes the appearance of a robot at an initial stage and stored the result for the future manipulation. Once an appearance level is obtained for a particular robot, then it will be a constant for future references.

8.2 Procedure

Each of the human subjects was given a paper carrying photographs of several kinds of robots including Robots A, B, and C. They were instructed to analyze the outer appearance of each of these robots with compared to a human and rank each of the robots from a scale of one to ten. That is if a robot is more like a machine, then the rank is one, if a robot is more like a human or have many features that humans have, then the rank will be ten. Each of the human participants was instructed to think alone without discussing with each other and fill a table. Then each of these filled tables was collected and summarized the data to get the ranking of each of the robots. The summarized data is shown in Table 7, where number of votes indicates the number of people who assigned a particular rank to a particular robot.

Rank	1	2	3	4	5	6	7	8	9	10
Robot	Number of votes									
A	6	9	4			1				
B	1	3	4	5	3	2	2			
C	1		2	5	7	1	4			
D	6	1	2	2	2	4	2	1		
E	12	3	2	2		1				
F			1					2	3	14

Table 7. Summarized votes for each robot

8.3 Ranking the robots

According to the results obtained, there were two possible ways of ranking the robots i.e. according to "highest choice" and calculating "linear rank."

- Highest choice is the rank that most of the people selected for a particular robot.
- Linear rank was calculated according to the following equation.

$$R_i = \frac{1}{P_N} \sum_{j=1}^{10} (P_x \times R_j) \tag{16}$$

Linear rank of the robot i where i=*A,B,C,D,E or F*, is given by R_i having P_x number of votes in R_j rank, x is having the values of 1 to total number of participants P_N, and j is having the ranks from 1 to 10.

8.4 Results

Results obtained for the above two methods as with the previous appearance values for the robots A, b, and C are given in Table 8 for comparison.

Robot	A	B	C	D	E	F
Highest choice	2	4	5	1	1	10
Linear Rank	1.8	4	4.8	3.9	1.9	9.3
Previous value	1	2	5	--	--	--

Table 8. Comparison of results for robots

8.5 Summary

After analyzing the results it can be seen that appearance of Robots A and B are increased to higher levels as the general opinion of the participants. But appearance of Robot C was not changed aligning the previous random selection.

9. Conclusion

In this book chapter, a construction method for an automated system to generate a personal space for specific environmental condition has been attempted. Although the considered parameters were limited to only three namely, height, appearance, and familiarity, it will be possible to expand the system for any number of considerable parameters, once a very basic model has been created as performed by this research. The constructed system gave encouraging results as were seen by the comparison of test output values with the trained ANFIS output values for the same set of input environmental conditions (i.e. same input values of height, appearance, and familiarity gave very close output values of original output data values to that of active PS system output data values). Hence this system can be considered as the basic building block of constructing a fully automated, fully functional for any environmental parameters to generate an active personal space determination system.

In the implementation process, although the input "height" can be measured without any doubt, other two inputs may raise arguments. In analyzing robot for their appearances, there should be a "unit" or "measuring scale" that is acceptable to entire robotic manufacturing community. Although there is no such a "scale" at the moment, hope there may be in the future so as to ascertain the robotic outer structure to that of the human appearance making much human looking robots (or humanoids). For the "familiarity" analysis, good human recognition system (or face recognition system) is required. Until it can use such a system, participants must be asked to wear an identity tag that can be read by the robot to recognize them.

In an event of assigning values for the "appearance," compared to the random way of putting values, above proposed method give much accepted procedure. Further, as appearance also depends on the person who looks at a robot, this method may give better results. It is also advisable to find suitable mechanisms to improve this method in the future.

Although this ANFIS cannot be treated as the ultimate solution for finding the personal space for any environment situation for any robotic system currently available in the world, this can be considered as the first step for such final advanced mechanism. But in order to achieve a target as such, many experiments in vast environmental situations should have to be involved. It is a must to obtain similar data with the so-called humanoids to make this experiment complete. Further, more sophisticated supportive equipment such as high speed

processing units for human recognition, memory acquisition and manipulation, image processing, etc. should be coupled. This system gave encouraging results in an offline mode with limited facilities. Authors are planning to make the current system more realistic and get the functioning in a real time mode, and are continuously working on it.

10. References

Simmons R., Goodwin R., Haigh K. Z., Koenig S., and O'Sullivan J. (1997), A layered architecture for office delivery robots, *Proc. of Autonomous Agents*, pp. 245-252, 1997.

Nakauchi Y. and Simmons R. (2002), A social behavioral robot that stands in line, *Autonomous Robots*, vol. 12, pp. 313-324, 2002.

Sack R. (1986), *Human Territory*, Cambridge University Press, UK, 1986.

Stentz A. (1996), Map-based strategies for robot navigation in unknown environments, *Proc. of AAAI,* pp. 110-116, 1996.

Malmberg M. (1980), *Human Territoriality: Survey of behavioral territories in man with preliminary analysis and discussion of meaning*, Mouton Publishers, 1980.

Walters M. L., Dautenhahn K., Koay K. L., Kaouri C., Boekhorst R., Nehaniv C., Werry I., and Lee D. (2005a), Close encounters: spatial distances between people and a robot of mechanistic appearance, *Proc. of 5th IEEE--RAS Int. Conf. on Humanoid Robots*, pp. 450-455, Dec. 2005.

Hall E. T. (1966), *The Hidden Dimension: Man's Use of Space in Public and Private, The Bodley Head* Ltd, London, UK. 1966.

Reeves B. and Nass C. (1998), *The Media Equation: How people treat computers, television and new media like real people and places*, Cambridge University Press 1998, ISBN 157586052x.

Walters M. L., Dautenhahn K., Boekhorst R. Te., Koay K. L., Kaouri C., Woods S., Nehaniv C. L., Lee D., and Werry I. (2005b), The influence of subjects' personality traits on personal spatial zones in a human-robot interaction experiment, *Proc. of the 14th Annual Workshop on Robot and Human Interactive Communication (IEEE ROMAN 2005)*, Tennessee, USA, pp. 347-352. Aug. 2005.

Dautenhahn K. (2004), Robots we like to live with?! - A developmental perspective on a personalized life-long robot companion, *Proc. of the 13th Annual Workshop on Robot and Human Interactive Communication (IEEE ROMAN2004)},* Okayama, Japan, pp. 17-22, Sept. 2004.

Jang J. S. R. (1993), ANFIS: Adaptive-network-based fuzzy inference system, *IEEE Transactions on SMC*, vol. 23, no. 3, pp. 665-685, May/June 1993.

Jin S. H., Kimura I., and Watanabe K. (1993), Controls of servomotors for carry hospital robots, *Journal of Intelligent and Robotic Systems*, vol. 7, pp. 353-369, 1993.

Jang J. S. R. and Sun C. T. (1995), Neuro-fuzzy modeling and control, *Proc. of the IEEE*, vol. 83, no. 3, pp. 378-406, March 1995.

Matlab tutorial, *Anfis and Anfis Editor GUI*, Mathworks Ltd., pp. 2.78-2.105

2

The Potential for Modeling Human-Robot Interaction with GOMS

Jill L. Drury[1], Jean Scholtz[2] and David Kieras[3]
[1]*The MITRE Corporation,* [2]*Pacific Northwest National Laboratory,*
[3]*The University of Michigan*
USA

1. Introduction

Human-robot interaction (HRI) has been maturing in tandem with robots' commercial success. In the last few years HRI researchers have been adopting—and sometimes adapting—human-computer interaction (HCI) evaluation techniques to assess the efficiency and intuitiveness of HRI designs. For example, Adams (2005) used Goal Directed Task Analysis to determine the interaction needs of officers from the Nashville Metro Police Bomb Squad. Scholtz et al. (2004) used Endsley's (1988) Situation Awareness Global Assessment Technique to determine robotic vehicle supervisors' awareness of when vehicles were in trouble and thus required closer monitoring or intervention. Yanco and Drury (2004) employed usability testing to determine (among other things) how well a search-and-rescue interface supported use by first responders. One set of HCI tools that has so far seen little exploration in the HRI domain, however, is the class of modeling and evaluation techniques known as formal methods.

1.1 Difficulties of user testing in HRI

It would be valuable to develop formal methods for use in evaluating HRI because empirical testing in the robotics domain is extremely difficult and expensive for at least seven reasons. First, the state of the art in robotic technology is that these machines are often unique or customized, and difficult to maintain and use, making the logistics of conventional user testing difficult and expensive. Second, if the evaluation is being performed to compare two different user interfaces, both interfaces must be implemented and ported to the robots, which is a significant task. Testing may involve using two robots, and even if they have identical sensors and operating systems, small mechanical differences often result in different handling conditions, especially when using prototyped research platforms. Thus there are serious problems even with preparing the robot systems to be tested.

Third, compared to typical computer usability tests, the task environments used to perform robot usability testing are complex physical environments, difficult to both devise and actually set up. Together with the fairly long time required to perform the tasks in a single situation, the result is that the cost per test user and trial is very high. Fourth, robots are mobile and whether the tasks being evaluated involve moving the entire robot or just moving parts of the robot such as a gripper, the probability that any two users will follow exactly the same path of

movement through the test environment is extremely small. This lack of uniform use of the robots makes comparison between users difficult, especially when the robots get stuck on obstacles or hit structures, seriously disrupting the evaluation. Fifth, if the testing is done outside (necessary for larger robotic vehicles) the same environmental conditions (lighting, rain, snow, etc.) cannot be guaranteed for all participants in the evaluations.
Sixth, training to operate robots is slow and costly; to compare two different interfaces, users need to have approximately the same level of skill in operating the robots. This may take considerable practice time as well as a means of assessing the acquired skills. Seventh, safety issues are always a concern and additional personnel are needed to ensure that no robots, people, or facilities are harmed during the tests, further increasing the cost.
Given the above challenges, it is clear that it is beneficial to obtain as much usability information as possible using methods that do not involve actual user testing. Obviously user testing will be necessary before the user interface can be declared successful, but it will be much less costly if formal methods can be employed prior to user testing to identify at least a subset of usability problems.

1.2 The potential of GOMS models

Perhaps the most widely-used of the formal methods is the Goals, Operations, Methods, and Selection rules (GOMS) technique first presented by Card, Moran, and Newell (1983), and which then developed into several different forms, summarized by John and Kieras (1996a, 1996b). Depending upon the type of GOMS technique employed, GOMS models can predict the time needed for a user to learn and use an interface as well as the level of internal consistency achieved by the interface. GOMS has proven its utility because models can be developed relatively early in the design process when it is cheaper to make changes to the interface. Analysts can use GOMS to evaluate paper prototypes' efficiency, learnability, and consistency early enough to affect the design prior to its implementation in software. GOMS is also used with mature software to determine the most likely candidates for improvement in the next version. Since GOMS does not require participation from end users, it can be accomplished on shorter time scales and with less expense than usability tests. Based on its use as a cost savings tool, GOMS is an important HCI technique: one that bears exploration for HRI.
Very little work has been done so far in the HRI domain using GOMS. A method we used for coding HRI interaction in an earlier study (Yanco et al., 2004) was inspired by GOMS but did not actually employ GOMS. Rosenblatt and Vera (1995) used GOMS for an intelligent agent. Wagner et al. (2006) used GOMS in an HRI study but did so in limited scenarios that did not explore many of the issues specific to HRI. Kaber et al. (2006) used GOMS to model the use of a tele-operated micro-rover (ground-based robot). In an earlier paper (Drury et al., 2007), we explored more types of GOMS than was presented in either Kaber et al. or Wagner et al. within the context of modeling a single interface.

1.3 Purpose and organization of this chapter

This chapter describes a comparison of two interfaces using a Natural GOMS Language (NGOMSL) model and provides a more detailed discussion of GOMS issues for HRI than has been previously published. At this point, we have not conducted a complete analysis of an HRI system with GOMS, which would include estimating some of the parameters involved. Rather, our results thus far are in the form of guidance for using GOMS in future HRI

modeling and evaluation efforts. The primary contribution of this chapter is an illustration of the potential of this approach, which we think justifies further research and application.
The next section discusses GOMS and what is different about using GOMS for HRI versus other computer-based applications. Section 3 contains guidance for adapting GOMS for HRI. We present background information on the two interfaces that we have modeled in Section 4, prior to presenting representative portions of the models in Section 5. Finally, we provide a summary in Section 6 and conclusions and thoughts for future work in Section 7.

2. Why is HRI different with respect to GOMS?

Before discussing the nuances of using GOMS for HRI, we briefly describe GOMS. There is a large literature on GOMS and we encourage the reader who is unfamiliar with GOMS to consult the overviews by John and Kieras (1996a, 1996b).

2.1 The GOMS Family

GOMS is a "family" of four widely-accepted techniques: Card, Moran, and Newell-GOMS (CMN-GOMS), Keystroke Level Model (KLM), Natural GOMS Language (NGOMSL), and Cognitive, Perceptual, and Motor GOMS (CPM-GOMS). John and Kieras (1996a) summarized the four different types of GOMS:

- CMN-GOMS: The original formulation was a loosely defined demonstration of how to express a goal and subgoals in a hierarchy, methods and operators, and how to formulate selection rules.
- KLM: A simplified version of CMN was called the Keystroke-Level Model and uses only keystroke operators — no goals, methods, or selection rules. The modeler simply lists the keystrokes and mouse movements a user must perform to accomplish a task and then uses a few simple heuristics to place "mental operators."
- NGOMSL: A more rigorously defined version of GOMS called NGOMSL (Kieras, 1997) presents a procedure for identifying all the GOMS components, expressed in structured natural language with in a form similar to an ordinary computer programming language. A formalized machine-executable version, GOMSL, has been developed and used in modeling (see Kieras and Knudsen, 2006).
- CPM-GOMS: A parallel-activity version called CPM-GOMS (John, 1990) uses cognitive, perceptual, and motor operators in a critical-path method schedule chart (PERT chart) to show how activities can be performed in parallel.

We used the latter three types of GOMS in Drury et al. (2007). In this chapter we use NGOMSL only, because it emphasizes the interface procedures and their structure. We discuss our selection of NGOMSL in more detail in Section 3.2.

2.2 HRI challenges for GOMS

A first challenge is that traditional GOMS assumes error-free operation on the part of the user and predictable and consistent operation on the part of the computer. The human-error-free assumption for GOMS is often misunderstood, and so needs some discussion. In theoretical terms, one could write GOMS models that describe how users deal with their errors (Card et al., 1983; Wood, 1999, 2000; Kieras, 2005) and even use GOMS to help predict where errors could take place (Wood, 1999; Wood and Kieras, 2002). The reason why GOMS modeling of human error is not routinely done is that (1) the techniques need to be further

developed, and this requires as a foundation a better theory of human error than is currently available; and (2) in many computer user interface design situations, making the interface easy to learn and easy to use (which can already be addressed with GOMS) "automatically" reduces the likelihood of human error, making it less critical to deal with. Safety-critical domains are the obvious exception; clearly, developing techniques for modeling human error should be an area of intensive future research.

However, the second assumption, that of predictable and consistent computer behavior, is much more important in the HRI domain. Even when operated without autonomy, in the search-and-rescue (SAR) domain robots can behave in unexpected ways. In addition, the HRI task environment is such that the user cannot easily predict what the situation will be, or what effects trying to interact with that environment will have. The fact that the user cannot predict the state of the robot or environment in the near future means that models must account for a great range and flexibility of users' responses to any given situation, and the application of the models must be done in a way that takes the great variability of the situation into account. Later in this chapter we illustrate one way of handling this situation: user activity can be segmented into phases and actions whose methods and their predicted execution time can be represented in GOMS, leaving the probability or frequencies of these activities, which GOMS cannot predict, to be dealt with separately.

A second challenge for HRI pertains to the seemingly simple task of maneuvering the robot, which normally occurs with a control device such as a joystick. While GOMS has long modeled the use of pointing devices to move cursors or select different items on the computer display, it has not developed mechanisms to model the types of movements users would employ to continuously or semi-continuously direct a robot's movement with a joystick. This missing element is important because there are fundamental differences in the amounts of time that are spent moving a cursor with a pointing device versus pushing a joystick to steer a robot's motion, and there are likely to be fundamental differences in the cognitive mechanisms involved.

Wagner et al. (2006) applied GOMS to HRI to model mission plan generation but does not include this basic task of driving a robot. GOMS has been used frequently in the aviation domain and so we scoured the literature to find an analogous case, for example when the pilot pulls back on the rudder to change an aircraft's altitude. To our surprise, we found only analyses such as Irving et al. (1994) and Campbell (2002), which concentrated on interactions with the Flight Management Computer and Primary Flight Display, respectively: interactions confined to pushing buttons and verifying the computers' responses.

Kaber et al. (2006) attempted to account for driving times using a simple GOMS model that assumed, in essence, that controlling the robot motion was directly analogous to the computer interface task of finding an object on the screen and pointing to it with a mouse. The resulting model seriously overpredicted driving task times, which is consistent with the possibility that driving tasks are processed quite differently from interface pointing tasks. How to deal with this problem is one of the issues addressed in this chapter.

A third challenge relates to modeling mental operations to incorporate the right amounts of time for the users' thought processes at each stage of using an interface. For example, previous empirical work has shown that it takes a user an average of 1.35 seconds to mentally prepare to perform the next action when executing a routine task in a predictable environment (John and Kieras, 1996b). But robot operations are notoriously non-routine

and unpredictable, as discussed above. Luckily, GOMS has always assumed that application-specific mental operators could be defined as necessary: what is difficult is determining the mental operators that make sense for HRI.

A fourth challenge is that the mental and perceptual operators in GOMS do not account for the effects of having varying qualities of sensor data, either within the same system at different times or on multiple systems that are being compared. For example, if video quality is bad on one system but exceptionally clear on another, it will take more time to extract video-based information via the first system's interface than from the second's interface. Each GOMS operator is normally assigned a single value as its typical time duration, such as the 1.35 seconds cited above for mental preparation. Unless a perceptual operator is assigned one time value in the model for the first system and a shorter time value for the second model (to continue the example), the models will not take into account an important difference that affects performance.

A fifth challenge pertains to different levels of autonomy. We believe it would be very useful, for example, for GOMS models to tell us whether it is more efficient for the robot to prevent the user from getting too close to objects, as opposed to requiring the user to spend time and effort watching out for obstacles immediately around the robot.

As we present our adaptations to GOMS and our example models in the following sections, we provide guidance for overcoming these challenges.

3. Adapting GOMS to HRI

3.1 Procedures vs. perception

The design process for HRI breaks naturally into two major parts: the perceptual content of the displays and the overall procedural operation.

Designers need to define the perceptual content of the displays so that they can be easily comprehended and used for HRI tasks. This design challenge will normally need to be accomplished using traditional interface design wisdom combined with evaluation via user testing; GOMS does not address whether one visual presentation of an item of information is easier to interpret than another. Rather, GOMS assigns a "mental operator" to the part of the task that involves interpreting display information, but this does not shed any light on whether one presentation facilitates users extracting information more quickly than another presentation—the standard mental operator is a simple "one size fits all" estimate. If modelers can define different types of domain- or task-specific mental operators for displays, then competing designs can be examined to see which requires more instances of one type of mental operator versus another. If these operator durations can be measured empirically, then the GOMS model can make a more accurate quantitative contribution.

GOMS can clearly help with the procedural implications of display design. For example, perhaps one design requires actions to toggle the display between two types of information, while another design makes them simultaneously visible; GOMS would highlight this difference. Designers also need to define the overall operation of the user interface in terms of the procedures that users would follow to carry out the task using the interface. Evaluating the procedural design challenge can be done easily and well with GOMS models if the perceptual design challenge can be handled so as not to confound any comparisons that modelers might make between competing designs.

This brings us to our first guideline for modeling HRI using GOMS:

1. Don't get bogged down in modeling the perceptual content of the displays; focus on the procedures instead.

The modeler should focus instead on the step-by-step procedures used when interacting with the interface, keeping in mind that issues in perception might determine and dominate issues regarding procedures. Getting bogged down in the perceptual issues is tempting because this part of the task is obviously important, but current modeling technology doesn't allow modelers to make much progress *a priori*. Many tasks demand that the operator view video or dynamic sensor data. They may need to do this multiple times in order to understand the situation. There is no way to predict the number of times a user may consult a display because doing so is situation-dependent and also dependent upon environmental conditions (such as complexity of the scene and video quality), skill levels, and physical capabilities of the users (eyesight acuity, dexterity, etc.). Even if we could know how many times users would need to refer to a particular part of the displays, it is difficult to assign accurate times to the actions.

GOMS modeling can be used to characterize a single interface, but it becomes much more useful when comparing two interface designs. The difficult-to-model aspects such as perceptual processes can often be held constant between the two designs, enabling the models to pinpoint the procedural differences and highlight their consequences. When improving designs incrementally, however, a modeler can model a single interface to the point where it exposes inconsistencies or inefficiencies that can become the focus of suggested design improvements. The improved design can then be compared to the first design using GOMS to identify the degree of improvement attained.

3.2 Choice of GOMS technique

We present our models using NGOMSL because this form of GOMS is easy to read and understand while still having a relatively high level of expressive power. A further advantage of NGOMSL is that it can be converted relatively easily into the fully executable version, GOMSL notation (see Kieras, 2005; Kieras and Knudsen, 2006). NGOMSL can be thought of as stylized, structured pseudocode. The modeler starts with the highest level goals, then breaks the task into subgoals. Each subgoal is addressed in its own method, which may involve breaking the task further into more detailed subgoals that also are described in their own methods. The lowest-level methods contain mostly primitive operations. Design consistency can be inferred by how often "basic" methods are re-used by other methods. Similarly, efficiency is gained when often-used methods consist of only a few steps. The number of methods and steps is proportional to the predicted learning time. Because NGOMSL lacks the ability to describe actions that the user takes simultaneously, we adopt a bit of syntax from GOMSL, the keyword phrase "Also accomplish goal…", when we need to show that two goals are being satisfied at the same time.

Since all detailed GOMS models include "primitive operators" that each describe a single, atomic action such as a keypress, we discuss primitives next.

3.3 Primitives

At the lowest level of detail, GOMS models decompose a task into sequences of steps consisting of *operators*, which are either motor actions (e.g., home hands on the keyboard) or cognitive activities (e.g., mentally prepare to do an action). As summarized by John and

Kieras (1996a, 1996b), the following primitive operators are each denoted by a one-letter code and their standard time duration:

- **K** to press a key or button (0.28 seconds for average user)
- **B** to press a button under the finger (e.g. a mouse button) (0.1 seconds)
- **M** perform a typical mental action, such as finding an object on the display, or mentally prepare to do an action (1.35 seconds)
- **P** to point to a target on a display (1.1 seconds)
- **H** to home hands on a keyboard or other device (0.4 seconds)
- **W** to represent the system response time during which the user has to wait for the system (variable)

As an example of the **W** operator, the system associated with Interface A (described below) has a delay of about 0.5 seconds before the user can see a response from steering the robot; the value for Interface B (also described below) was 0.25 seconds. This time difference was noticed and commented on by users and so needs to be reflected in the models' timing calculations.

As discussed above, none of these primitives are especially suited to describing manipulating the robot; thus we define a "steer" operator **S** and introduce our second guideline:

2. Consider defining and then assigning a time duration to a robot manipulation operator that is based on typical values for how long the combination of the input devices, robot mechanics, and communications medium (especially for remote operations) take to move the robot a "reference" distance.

This guideline is based on the fact that the time required to manipulate the robot is driven more by the robot mechanics and environment than by the time needed by the human to physically manipulate the steering input device. The ultimately skilled user would perform all perceptual, navigation, and obstacle avoidance subtasks while keeping the robot moving at a constant speed, thus making execution time equal to the time it takes to cover an area at a given speed.

We assigned a reference **S** time of 1 foot/second to the interactions with the two robots modeled in this chapter. In accordance with our guidance, we observed operations and determined that the mechanics of the robot were the dominant factor in determining how quickly, on average, a user would steer the robot to accomplish moving one foot. This is necessarily a somewhat crude approach to assigning a time because the robots could run at various speeds which changed based on lighting conditions, proximity to obstacles, and users deciding to speed up or slow down. When two interfaces are being examined, however, using a single representative speed for each robot should not harm the comparison of their respective GOMS models.

While all the other "standard" primitive operators apply to HRI, the **M** operator requires close scrutiny. As used in modeling typical computer interfaces, **M** represents several kinds of routine bits of cognitive activity, such a finding a certain icon on the screen, recalling a file name, making a routine decision, or verifying that a command has had the expected result. Clearly, using the same operator and estimated time duration for these different actions is a gross simplification, but it has proven to be useful in practice (see John and Kieras, 1996a, 1996b for more discussion). However, consider data being sent to the user from a remote mobile robot that must be continually perceived and comprehended (levels 1 and 2 of Endsley's (1988) definition of situation awareness). This is a type of mental process that is

used to assess the need for further action triggered by external, dynamic changes reflected in the interface (e.g. as seen in a changing video display). Intuitively, this mental process seems qualitatively different, more complex, and more time-consuming from those traditionally represented with **M**. Since this type of mental operation is nontrivial, we propose representing it as a separate operator, and then the analysis can determine if one design versus another requires more instances of this type of mental assessment. For example, if one interface splits up key sensor information on separate screens but another provides them on a fused display, the latter design would require fewer mental operators in general and fewer operators of the type that assesses dynamic changes in the environment.
This leads us to an additional guideline:

3. Without violating guideline number 1, consider defining HRI-specific mental operator(s) to aid in comparing the numbers of instances these operators would be invoked by competing designs.

We define a **C** (Comprehend) operator to refer to a process of understanding and synthesizing complex display information that feeds into the user's continually-changing formulation of courses of action. This operator is expected to take longer than the conventional **M** operator, and will be used to represent the interpretation of the complex displays in this domain.
Once the set of primitives has been finalized, the next step is to assign times to the various operators unique to the HRI domain. To a certain extent, the exact times are not as important as the relative differences in times that result from competing interface designs. The time required for our mental operator **C** for robotics tasks will depend on the quality of the video, the complexity of the situation being viewed, and the design of the map and proximity sensor displays. Thus, if two systems being compared have radically different qualities of sensor data, we suggest the following guideline:

4. Without violating guideline number 1, consider assigning time duration values to HRI-specific mental operators that reflect the consequences of large differences in sensor data presentation.

In the absence of detailed empirical data specific to the system and conditions being modeled, we estimate the time required by the **C** operator to be 2 seconds. This estimate was derived based on observing search-and-rescue operators working with each system in remote operations. Again, this is a crude estimate that varied substantially among the users. In the future, eye tracking may be a possible means of determining how long users gazed at particular part of the display.

4. Example interfaces

This next section presents some sample results from applying GOMS as adapted to HRI. But first, we need to describe the interfaces we modeled. User interface analysts use GOMS to model human activity assuming a specific application interface because the models will be different for each application. Our decision regarding which interfaces to analyze was not important as long as the chosen interfaces contained representative complexity and functionality. We chose two interfaces, illustrated in Figures 1 and 2, that use the same basic set of urban search-and-rescue (USAR) functionality and the same robotic platform.

4.1 Interface "A"

The architecture for the system underlying Interface A was designed to be flexible so that the same interface could be used with multiple robot platforms. We observed the interface operating most frequently with an iRobot ATRV-Jr.: the same one used for Interface B.

Interface A (Figure 1) was displayed on a touch screen. The upper left corner of the interface contained the video feed from the robot. Tapping the sides of the window moved the video camera left, right, up or down. Tapping the center of the window re-centered the camera. Immediately to the right of the video display were pan and tilt indicators. The robot was equipped with two types of cameras that the user could switch between: a color video camera and a thermal camera; the camera selection radio buttons were also to the right of the video area.

The lower left corner contained a window displaying health status information such as battery level, heading, and attitude of the robot. A robot-generated map was placed in the lower central area. In the lower right corner, there was a sensor map that showed red arrows to indicate directions in which the robot's motion was blocked by obstacles.

The robot was controlled through a combination of a joystick and the touch screen. To the right of the sensor map in the bottom right hand corner of the touch screen, there were six mode buttons, ranging from autonomous to tele-operation. Typically, the user touched one of the mode buttons, then used the joystick to steer the robot if not in the fully autonomous mode.

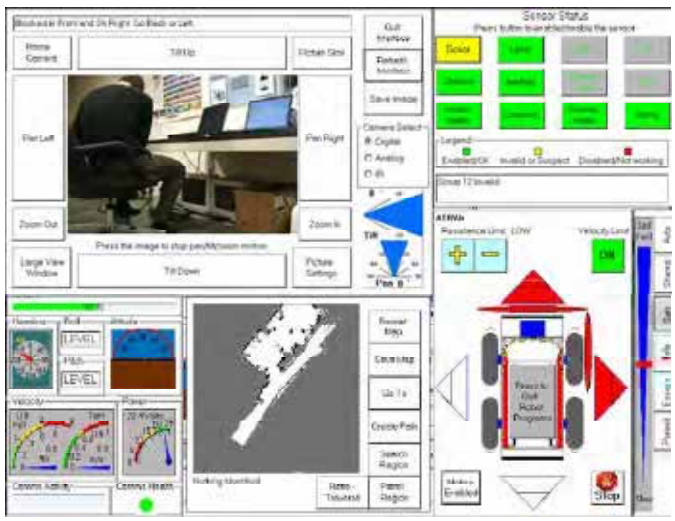


Figure 1. Interface A, one of the two example interfaces that we analyzed

When the user wished to take a closer look at something, he or she touched the video window to pan the camera. For victim identification, the user often switched to the thermal or infrared (IR) sensor (displayed in the same space as the videostream and accessed via a toggle) to sense the presence of a warm body.

The proximity sensors were shown around a depiction of the robot in the bottom right hand side of the display. The triangles turned red to indicate obstacles close to the robot. The small,

outer triangle turned red when the robot first approached objects, then the larger, inner triangle also turned red when the robot moved even closer to an obstacle. The location of the red triangles indicated whether the blockage was to the front, rear, and/or sides.
Note that System A's interface did not incorporate menus. Visual reminders for all possible actions were present in the interface in the form of labels on the touch screen.
While the organization that developed System A has explored other interface approaches since this version, users access almost the same functionality with all interface designs to date. Also, many other USAR robots incorporate similar functionality.

4.2 Interface "B"

The ATRV-Jr. robot used with Interface B was modified to include a rear-facing as well as forward-facing camera. Accordingly, the interface had two fixed video windows (see Figure 2). The larger one displayed the currently selected camera (either front- or rear-facing); the smaller one showed the other video feed and was mirrored to emulate a car's rear-view mirror.
Interface B placed a map at the edge of the screen. The map window could be toggled to show a view of the current laser readings, removing the map from the screen during that time.

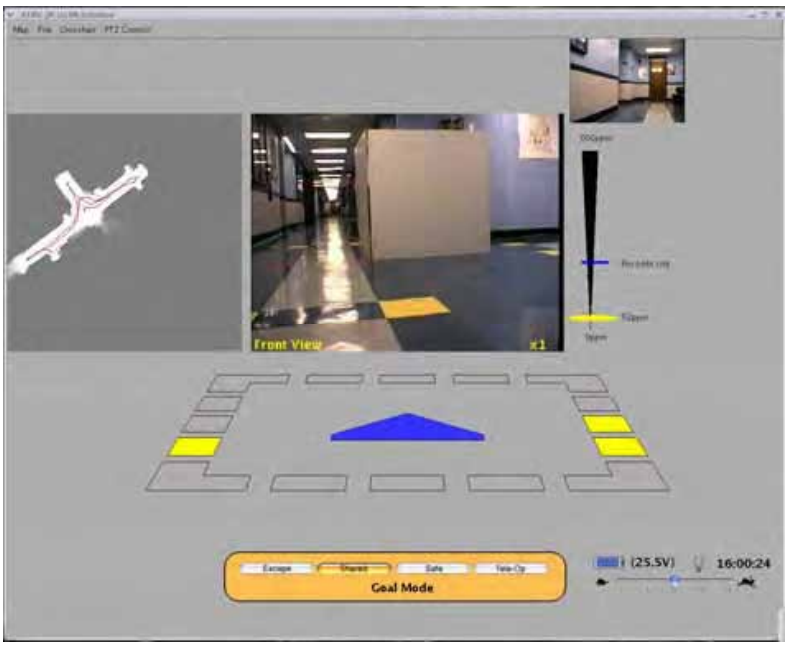


Figure 2. Interface B

Information from the sonar sensors and the laser rangefinder was displayed in the range data panel located directly under the main video panel. When nothing was near the robot, the color of the box was the same gray as the background of the interface, indicating that nothing was there. As the robot approached an obstacle at a one foot distance, the box

turned to yellow, and then red when the robot was very close (less than half a foot away). The ring was drawn in a perspective view, which made it look like a trapezoid. This perspective view was designed to give the user the sensation that they were sitting directly behind the robot. If the user panned the camera left or right, this ring rotated opposite the direction of the pan. If, for example, the front left corner turned red, the user could pan the camera left to see the obstacle, the ring would then rotate right, so that the red box would line up with the video showing the obstacle sensed by the range sensors. The blue triangle, in the middle of the range data panel, indicated the true front of the robot.
The carbon dioxide meter to the right of the primary video screen showed a scale in parts-per-million (PPM) and also indicated the level at which "possible life" was detected as a blue line. (The platform used for Interface A had an IR sensor to serve this same purpose.) The bottom right hand corner showed battery life, whether the light on the front of the robot was illuminated, a clock, and the maximum speed. The level of autonomy was shown in the bottom right hand set of buttons (Shared/Goal Mode is illustrated in the figure).

4.3 Tasks Analyzed

We analyzed tasks that are typical of a search-and-rescue operation: maneuver the robot around an unfamiliar space that is remote from the user, find a potential victim, and confirm the presence of a victim.

5. Example modeling results

Given the novelty of the adaptations, at this point we can present only illustrations of how GOMS can be applied to answer questions about HRI designs. Clearly additional research will be needed to provide a complete assessment of the accuracy and value of these adaptations and any required corrections to them.
In Section 2.2 we pointed out the need to deal with the flexibility and unpredictability of the HRI domain. Now we illustrate a simple approach to this problem: the user's activity is decomposed into segments corresponding to GOMS methods. Often, the activity within a method, and the time to perform it, is fairly well-defined. In many cases, what is unpredictable is simply how likely or how frequently that activity will be required. As long as the likelihood or frequency of an activity is similar for the two interfaces, useful comparisons can then be made from the models.

5.1 Top level model

A major part of creating a model for a task is to characterize the top level of the task. Figure 3 contains a fragment from a preliminary model for the top level of the robot search-and-rescue task. Due to space reasons, we cannot show all of the methods, so we only show those methods that lead to the user determining whether she has spotted a victim. This "thread" of methods is shown in the figure by the bold-face goals.
This top-level model shows the overall assumed task structure. After getting some initial navigation information, the user repeatedly chooses an area to search until all areas have been covered. Each area involves driving around to different locations in that area and looking for victims there. Locating a victim involves repeatedly choosing an area to view, viewing it, and then checking the sensors to see if a victim is present. This last goal will be examined in more detail in the next subsection.

Method for goal: **Perform search and rescue mission**
1. Accomplish goal: obtain global navigation information.
2. Choose next local area.
3. If no more local areas, return with goal accomplished.
4. Accomplish goal: **search local area.**
5. Go to 2.

Method for goal: **search local area**
1. Accomplish goal: drive to new location.

The following step applies 70% of the time.
2. Also accomplish goal: **locate victim.**
3. Return with goal accomplished.

Method for goal: **locate victim**
1. Choose next area of location.
2. If no more areas, return with goal accomplished.
3. Accomplish goal: view area of location.
4. Accomplish goal: **view sensors for indication of victim**.
5. If indication shown, return with goal accomplished.
6. Go to 1.

Figure 3. Top-level methods for search-and-rescue

The top-level method focuses attention on a basic issue in the task, namely the extent to which the user can simultaneously drive the robot to cover the area, and locate a victim using the video and sensors. Both interfaces seem to be compatible with simultaneous operation, compared to some other interface that, for example, used the same joystick for both camera motion control and driving. The method shows this simultaneity assumption with the use of the "Also accomplish goal" operator. However, Yanco and Drury (2004) observed that users were able to drive and look for victims simultaneously only about 70% of the time, and often had to pause to reorient themselves. Currently, GOMS lacks a direct way to express this sort of variability, so we have commented this step in the method as a place-holder.

5.2 Comparing Interfaces A and B

In addition to showing the overall flow of control, the top-level model acts to scope and provide a context for the more detailed modeling, such as the consideration of how different displays might support the goal of viewing sensors for indication of a victim. This is illustrated by critiquing the methods for this goal supplied by Interface A, and then comparing them to Interface B. Note that because the two sensors are different for the two platforms we are comparing the procedure necessary for viewing a thermal (infrared) sensor as illustrated in Interface A with the procedure for viewing a carbon dioxide sensor as illustrated in Interface B.

The method for Interface A is shown in Figure 4A. The method shows that the display must be toggled between the normal video display and the infrared display. Since it may be done frequently, the time cost of this operation could be significant. While the GOMS model cannot predict how often it would be done, the preliminary estimate is that it would take about 2.2 seconds per toggling (two **P**, or Point, operators). Clearly this aspect of the design could use improvement. Hestand and Yanco (2004) are experimenting with a USAR interface that places

infrared data on top of video data. While research is needed to determine if it takes longer for a user to comprehend combined video/infrared data, this model provides a design target: since it takes about 2.2 seconds to toggle the displays in the uncombined form, in order to do better, the combined display should not take any longer to comprehend.

In addition, the infrared display is color-coded in terms of temperature, and there is no on-screen cue about the color that indicates possible life, suggesting that the user will need to perform extra mental work. We have represented this as a step to recall the relevant color code.

In contrast, Interface B shows a different approach for another sensor datum, carbon dioxide level. The method is shown in Figure 4B. There is an on-screen indication of the relevant level, requiring a simple visual position judgment rather than a comparison to a memorized color.

Method for goal: **view sensors for indication of victim**

1. Look at and interpret camera display (**C**).

Using a touchscreen is similar to pointing with a mouse.

2. Point to touchscreen IR button to toggle display (**P**).
3. Recall IR display color-code that indicates possible life (**M**).
4. Look at and interpret IR display (**C**).
5. Decide if victim is present (**M**).

Need to restore display to normal video to support next activity

6. Point to touchscreen Digital button to toggle display (**P**).
7. Return with goal accomplished.

Figure 4A. Fragment of a GOMS model for Interface A

Method for goal: **view sensors for indication of victim**

1. Look at and interpret camera display (**C**).
2. Look at and determine whether carbon dioxide level is above "Possible life" line (**M**).
3. Decide if victim is present (**M**).
4. Return with goal accomplished.

Figure 4B. Fragment of a GOMS model for Inteface B

Interface B's method is shorter than Interface A's for several reasons. Interface A cannot show video and infrared sensor information (to show the presence of body heat) at the same time, incurring costs to switch between them, whereas Interface B can show video and carbon dioxide (present in humans' exhalations) sensor readings simultaneously. Also, Interface B explicitly shows the level above which the presence of nearby human life is likely, whereas users looking at Interface A will need to remember which color-coding in the infrared display indicates heat equivalent to human body temperature. This difference in approaches requires one less operator (to recall the appropriate color) as well as changes the nature of the mental operator (from a **C** to an **M** indicating a simple comparison). For one pass through the method, Interface A requires 2 more steps, two **P** operators, and an additional **C** operator. If we use the estimates for **C**, **P**, and **M** previously discussed, Interface A would require 8.9 seconds for this fragment versus 4.7 seconds for Interface B.

Even though GOMS cannot predict the interpretability of display elements, this example shows that the costs and benefits of different design decisions can be modeled, and even quantified, to some extent. Design decisions must balance the effectiveness of providing different sensors with the work they require on the part of users in order to benefit from having those sensors. If a number of sensors are present and are helpful, then procedures

for viewing all of the sensors and comprehending the information can be modeled and thus account for the time tradeoffs involved.

5.3 Modeling different levels of autonomy

Previously we have stated our contention that GOMS would be useful for showing the impact of differing autonomy levels on the user's efficiency. We illustrate this point by showing the difference in workload between extricating a robot when in tele-operation mode versus in escape mode. In tele-operation mode, the user directs all of the robots' actions in a complete absence of autonomy. In contrast, once the user puts the robot into escape mode, the robot itself figures out how to move away from all obstacles in the immediate environment and then, once clear of all obstacles, stops to await further commands from the user. Our experience is that robots become wedged into tight quarters surprisingly often, which motivated the development of the escape mode.

Figure 5 illustrates the portion of the GOMS model that pertains to getting a robot "unstuck": the unenviable condition where it has few options in how it can move. Figure 5 pertains to Interface A, but the model for Interface B is similar (only a few less steps).

Note that Figure 5 employs an informal means of passing a variable to a method. We denote the passing of a variable by a phrase in square brackets.

Method for goal: get unstuck when tele-operating
1. Accomplish goal: determine direction to move
2. Accomplish goal: drive to new location
3. Accomplish goal: check-movement–related sensors
4. Return with goal accomplished.

Method for goal: determine direction to move
1. Look at and interpret proximity display (**C**)
2. Accomplish goal: move camera in direction of obstacle
3. Return with goal accomplished

Method for goal: move camera [movement direction]
1. Point to touchscreen button for [movement direction] (**P**)
2. Look at and interpret video window (**C**)
3. Decide: if new movement direction is needed (**C**), go to 1.
4. Return with goal accomplished.

Method for goal: drive to new location
1. If hands are not already on joystick, home hands on joystick (**H**)
2. If movement direction not yet known, Accomplish goal: determine direction to move
3. Initiate joystick movement (**W**)
4. Move joystick until new location is reached or until stuck (**S**)
5. If stuck, then accomplish goal: get unstuck when tele-operating
6. Return with goal accomplished.

Method for goal: check movement-related sensors
1. Look at and interpret video window (**C**)
2. Look at and interpret map data window (**C**)
3. Look at and interpret sonar data window (**C**)
4. Return with goal accomplished.

Figure 5. Model Fragment for Tele-Operating Stuck Robots

As might be expected, getting a robot unstuck can be a tedious process. Not counting the shared method for checking movement-related sensors, there are 17 statements in the methods in Figure 5, and it often takes multiple iterations through to completely extricate a robot. Each iteration requires at least 6 **C** operators, a **P**, a **W**, and possible **H**, in addition to the robot movement time included in the **S** operator. These actions will require 15 seconds assuming the robot moves only a foot and only 6 **C** operators are needed: and all of this activity is attention-demanding.

Figure 6 shows the simple method for using an automony feature to extricate the robot. The user changes the autonomy mode from tele-operation to escape and then simply watches the robot use its sensors and artificial intelligence to move itself from a position almost completely blocked by obstacles to one that is largely free of obstacles so that the user can resume tele-operation. In contrast to the Figure 5 methods, the single method shown in Figure 6 lists only 5 statements. This method assumes that the user will do nothing but wait for the robot to finish, but clearly, the user could engage in other activity during this time, opening up other possibilities for conducting the task more effectively.

While this comparison might seem obvious, these models were simple to sketch out, and doing so is a very effective way to assess the possible value of new functionality. Using GOMS could save considerable effort over even simple prototype and test iterations (see Kieras, 2004 for more discussion).

Method for goal: get unstuck when using escape
1. Point to touchscreen button for escape (**P**).
2. Wait for robot to finish (**W**).

Same method called as in manual get unstuck method

3. Accomplish goal: check movement-related sensors
4. Decide: if robot still stuck, go to 1.
5. Return with goal accomplished.

Figure 6. Model fragment for extricating robots using escape mode

6. Summary

In this section we summarize the five primary GOMS-related HRI challenges and how we recommend addressing them. Specifically, using GOMS it is challenging to model:

A. Working with a difficult-to-predict environment or robot state.

The dynamic situations common with many robot applications require flexible modeling techniques. One way to deal with this challenge is to segment user activity into phases and methods whose times can be estimated, leaving the probability or frequencies of these activities to be addressed separately.

B. Maneuvering the robot.

We recommend using a new "steering" operator, **S**, and assigning a standard time to that operator that is characteristic for the robot platform (Guideline 2).

C. Describing users' mental operations as they synthesize complex display information.

While we do not feel it is useful to focus on modeling the perceptual content of displays (Guideline 1), it can be useful to define HRI-specific mental operators (Guideline 3). We recommend defining a mental operator **C** (Comprehend) to refer to understanding and synthesizing complex display information.

D. Accommodating the effects of having sensor data of various qualities.

We recommend assigning time duration values to the HRI-specific mental operators that reflect the different qualities of sensor data presentation (Guideline 4). Determining these time duration values is a topic for future work (see below).

E. Handling different levels of autonomy.

A complete model would need to include different methods that identify users' actions under the various levels of autonomy possible for a particular robotic system. More interestingly, GOMS provides an avenue for investigating the utility of potential new autonomy levels (see below).

7. Conclusions and future work

In this chapter we have shown how GOMS can be used to compare different interfaces for human-robot interaction. GOMS is useful in determining the user's workload, for example when introducing different displays for sensors.

GOMS models are also useful in determining the upper and lower time limits for different procedures. For example, checking movement-related sensors will depend on how many sensor windows the user has to look at and how many of these windows have to be manipulated to be viewed. This can be extremely helpful in deciding what should be visible at what time to maximize the user's efficiency.

GOMS can also be used to evaluate potential new autonomy modes. In our example we used the escape mode on Interface A to show how autonomy modes can be modeled. In actuality, the escape mode was originally incorporated into robots because it was clear to everyone that enabling this robotic behavior could save the user a lot of work and time. In other words, designers did not need a GOMS model to tell them that such functionality would be worthwhile. However, this example shows how GOMS can be used to model other autonomy modes to determine possible human-robot ratios. By examining the maximum expected times for different procedures, it is possible to use Olsen and Goodrich's equations on "fan out" (Olsen and Goodrich, 2003) to determine the upper bound on the number of robots that a single person can control simultaneously.

While GOMS was designed to model user behavior with a particular user interface, what it has done in the escape/tele-operation example is to render explicit the effect of both the robotic behavior and the user interface on the user. GOMS could be used in less obvious cases to "test drive" the effects that new robot behaviors, coupled with their interaction mechanisms, might have on users. To the extent that the effects of the interface design and robot behavior can be teased apart, GOMS can have utility in the process of designing new robot behaviors.

Future work might productively include experimentation with the effect of degraded video on the time necessary for users to perceive and comprehend information. Simulators could introduce a controlled amount of noise into the videostream in a repeatable fashion so that user tests could yield empirical data regarding average times for comprehending video data under various degraded conditions. This data could be codified into a set of "reference" video images. By comparing a system's video quality with the reference images, modelers could assign a reasonable estimate of video comprehension times *a priori*, without further empirical work.

Another future work area might be to examine whether it is useful to employ GOMS to model the robot's performance in interacting with the environment, other robots, and/or humans. Perhaps such an analysis would be desirable to help determine how many autonomous robots would be needed to complete a task under critical time limitations such as a large-scale rescue operation, for example. Such a modeling effort would likely uncover

additional issues and would depend on the nature of the robot's environment, the behaviors that were programmed for the robot, and the mechanical limitations inherent in the robot platform.

As argued in the Introduction, the use of GOMS models for comparing alternative designs can be much less costly than conducting user testing, once time values for domain-specific operators have been estimated. Once our example model has been elaborated to cover all of the critical parts of the task, we feel it would be fairly simple to modify the model to examine a variety of different interface designs and robot functions. The effort needed for modeling can be contrasted with that required for user testing, which necessitates building robust versions of user interfaces, obtaining sufficiently trained users, and having robots that are in operating condition for the number of days needed to conduct the tests.

While open issues exist with applying GOMS models to HRI, we are confident that it will help us develop superior HRI systems more quickly and effectively.

8. Acknowledgments

This work was supported in part by NSF (IIS-0415224) and NIST. We would like to thank Douglas Few and David Bruemmer of Idaho National Laboratories; Elena Messina, Adam Jacoff, Brian Weiss, and Brian Antonishek of NIST; and Holly Yanco and Brenden Keyes of UMass Lowell.

9. References

Adams, J. A. (2005). Human-Robot Interaction Design: Understanding User Needs and Requirements. In *Proceedings of the 2005 Human Factors and Ergonomics Society 49th Annual Meeting,* 2005, Orlando, FL.

Campbell, C. B. (2002). Advanced integrated general aviation primary flight display user interface design, development, and assessment. In *Proceedings of the 21st Digital Avionics Systems Conference*, Vol. 2.

Card, S., Moran, T., and Newell, A. (1983). *The Psychology of Human-Computer Interaction.* Hillsdale, New Jersey: Erlbaum.

Drury, J. L. , Scholtz, J., and Kieras, D. (2007). Adapting GOMS to model human-robot interaction. In *Proceedings of the 2nd ACM Conference on Human-Robot Interaction (HRI 2007).*

Endsley, M. R. (1988). Design and evaluation for situation awareness enhancement. In *Proceedings of the Human Factors Society 32nd Annual Meeting*, Santa Monica, CA, 1988.

Hestand, D. and Yanco, H. A. (2004). Layered sensor modalities for improved human-robot interaction. In *Proceedings of the IEEE Conference on Systems, Man and Cybernetics*, October 2004.

Irving, S., Polson, P., and Irving, J. E. (1994). A GOMS analysis of the advanced automated cockpit. In *Proceedings of the 1994 CHI conference on Human Factors in Computing Systems*, Boston, April 1994.

John, B. E. (1990). Extensions of GOMS analyses to expert performance requiring perception of dynamic visual and auditory information. In *Proceedings of the 1990 Conference on Human Factors in Computing Systems*. New York: ACM, pp. 107 – 115.

John, B. E. and Kieras, D. E. (1996a). Using GOMS for User Interface Design and Evaluation. *ACM Transactions on Human-Computer Interaction*, 3(4), December 1996.

John, B. E. and Kieras, D. E. (1996b). The GOMS Family of User Interface Analysis Techniques: Comparison and Contrast. *ACM Transactions on Human-Computer Interaction*, 3(4), December 1996.

Kaber, D.B., Wang, X., and Kim, S. (2006). Computational Cognitive Modeling of Operator Behavior in Telerover Navigation. In *Proceedings of the 2006 IEEE Conference on Systems, Man, and Cybernetics*, Oct. 8-11, Taipei, Taiwan.

Kieras, D. E. (1997). A Guide to GOMS model usability evaluation using NGOMSL. In M. Helander, T. Landauer, and P. Prabhu (Eds.), *Handbook of Human-Computer Interaction*. (Second Edition). Amsterdam: North-Holland. 733-766.

Kieras, D. E. (2004). Task analysis and the design of functionality. In A. Tucker (Ed.) *The Computer Science and Engineering Handbook* (2nd Ed). Boca Raton, CRC Inc. pp. 46-1 through 46-25.

Kieras, D. E. (2005). Fidelity Issues in Cognitive Architectures for HCI Modeling: Be Careful What You Wish For. In *Proceedings of the 11th International Conference on Human Computer Interaction (HCII 2005),* Las Vegas, July 22-27.

Kieras, D., and Knudsen, K. (2006). Comprehensive Computational GOMS Modeling with GLEAN. In *Proceedings of BRIMS 2006*, Baltimore, May 16-18.

Leveson, N. G. (1986). "Software safety: why, what and how." *ACM Computing Surveys* 18(2): 125 – 162, June 1986.

Olsen, D. R., Jr., and Goodrich, M. A. (2003). Metrics For Evaluating Human-Robot Interactions. In *Proceedings of PERMIS 2003*, September 2003.

Rosenblatt, J. and Vera, A. (1995). A GOMS representation of tasks for intelligent agents. In *Proceedings of the 1995 AAAI Spring Symposium on Representing Mental States and Mechanisms*. M. T. Cox and M. Freed (Eds.), Menlo Park, CA: AAAI Press.

Scholtz, J., Antonishek, B., and Young, J. (2004). Evaluation of a Human-Robot Interface: Development of a Situational Awareness Methodology. In *Proceedings of the 2004 Hawaii International Conference on System Sciences*.

Wagner, A. R., Endo, Y., Ulam, P., and Arkin, R. C. (2006). Multi-robot user interface modeling. In *Proceedings of the 8th International Symposium on Distributed Autonomous Robotic Systems*, Minneapolis, MN, July 2006.

Wood, S. D. (1999). The Application of GOMS to Error-Tolerant Design. Proceedings of the 17th International System Safety Conference, Orlando, FL.

Wood, S.D. (2000). Extending GOMS to human error and applying it to error-tolerant design. *Doctoral dissertation*, University of Michigan, Department of Electrical Engineering and Computer Science.

Wood, S. D. and Kieras, D. E. (2002). Modeling Human Error For Experimentation, Training, And Error-Tolerant Design. In *Proceedings of the Interservice/Industry Training, Simulation and Education Conference*. Orlando, Fl. November 28 – December 1.

Yanco, H. A. and Drury, J. L. (2004). Where Am I? Acquiring Situation Awareness Using a Remote Robot Platform. In *Proceedings of the 2004 IEEE Conference on Systems, Man, and Cybernetics*.

Yanco, H. A., Drury, J. L., and Scholtz, J. (2004). Beyond usability evaluation: analysis of human-robot interaction at a major robotics competition. *Human-Computer Interaction*, Vol. 19, No. 1 & 2, pp. 117 – 149.

3

Supporting Complex Robot Behaviors with Simple Interaction Tools

David J. Bruemmer[1], David I. Gertman[1], Curtis W. Nielsen[1] ,
Douglas A. Few[2] and William D. Smart[2]
[1]Idaho National Laboratory Idaho Falls, [2]Washington University in St. Louis
U. S. A.

1. Introduction

This chapter examines the potential for new mixed-initiative interaction methods and tools to change the way people think about and use robot behaviors. Although new sensors, processing algorithms, and mobility platforms continue to emerge, a remarkable observation is that for the vast majority of fielded systems, the basic inputs and outputs used to communicate between humans and robots have not changed dramatically since mobile robots were used in surprising numbers during World War II. Photographs and old books from the period show that small tracked unmanned ground vehicles were teleoperated for a variety of military missions. Today, video remains the predominant mechanism for providing situation awareness from the robot to the human. Direct control of translational and rotational velocity remains the predominant input from the human to the robot.

The problem is not that researchers have failed to create behaviors and autonomy. A preponderance of semi-autonomous capabilities are now available on laboratory robots (Arkin, 1997; Desai & Yanco, 2005; Maxwell et al., 2004). However, the resulting interaction is often complex and the robot's actions may seem mysterious to robot operators. Autonomy may actually increase the complexity of the task rather than decrease it. Without the right interaction metaphor, users do not know what to expect from their mysterious and complex robot "peers." As behavioral autonomy increases, the answer to the question: "What does it do?" may become increasingly difficult to answer. This is especially true for behaviors that are adaptive or which exhibit emergent effects and/or non-determinism. The trick then, is to make the user comfortable with these behaviors by communicating behavioral intentionality. The interface should make clear what the robot is trying to accomplish with each behavior and should provide an understanding of how the behavior will affect overall mission success. Just as the development of a windows based graphical user interface vastly augmented the impact and value of the personal computer, so likewise, an upsurge in the general utility of robots may follow closely on the heels of change in the human interface.

Until the recent past, robots did not exhibit sophisticated behaviors or take initiative to accomplish complex tasks. More often than not, the underlying theory of robot behavior was so trivial that it may have seemed unnecessary to give it much thought. An example is the basic notion that moving a joystick up and down will cause the robot to drive backwards and forwards. To someone who has already internalized this particular theory of robot

behavior, there is very little confusion. In actuality, the joystick has proved for many applications to be a valuable and effective interaction metaphor. The problem is that it is only appropriate for more direct levels of human-robot interaction. As the level of robot initiative and autonomy increases, the underlying metaphor for interaction must also change, resulting in a need for new and, at times, more sophisticated theories of robot behavior. The goal should not be to simplify the actual robot behaviors – complex environments and tasks require appropriately complex behaviors. Instead, the goal should be to simplify the user's mental model of robot behavior.

This chapter advocates that alongside new autonomous capabilities, the user must also be given interaction tools that communicate a mental model of the robot and the task, enabling the user to predict and understand robot behavior and initiative. The necessary "theory of robot behavior" should not be communicated by providing users with a developer's perspective of the system. Behavioral sciences research indicates that passing on system complexity to the user will increase stress and failure rates while decreasing task efficiency (Gertman et al., 2005). Quite to the contrary, a great deal of craft may be required to filter and fuse the system data, abstracting away from the complexity to support a higher-level, functional understanding. The hard question is how exactly to accomplish this in a principled fashion.

This chapter considers various components of interaction complexity common in human-robot interfaces. With results from several previous HRI experiments, the paper examines means employed to reduce this complexity. These include a) perceptual abstraction and data fusion, b) the development of a common visualization and tasking substrate, and c) the introduction of interaction tools that simplify the theory of robot behavior necessary for the operator to understand, trust and exploit robot behavior. The elements of operator trust are considered in detail with emphasis on how intelligently facilitating information exchange and human and robot initiative can build appropriate trust. New approaches to facilitating human-robot initiative are discussed within the context of several real-world task domains including: robotic event photographer, ground-air teaming for mine detection, and an indoor search and detection task.

2. Background

INL researchers began six years ago developing a suite of robotic behaviors intended to provide dynamic vehicle autonomy to support different levels of user intervention. Designed to be portable and reconfigurable, the Robot Intelligence Kernel (RIK) is now being used on more than a dozen different kinds of robots to accomplish a variety of missions within defense, security, energy, commercial and industrial contexts. RIK integrates algorithms and hardware for perception, world-modeling, adaptive communication, dynamic tasking, and behaviors for navigation, exploration, search, detection and plume mapping for a variety of hazards (i.e. explosive, radiological). Robots with RIK can avoid obstacles, plan paths through cluttered indoor and outdoor environments, search large areas, monitor their own health, find and follow humans, and recognize when anything larger than 10 cm has changed within the environment. Robots with RIK can also recognize when they are performing badly (i.e. experiencing multiple collisions or an inability to make progress towards a goal) and will ask for help.

A series of experiments was performed at the Idaho National Laboratory to assess the potential for autonomous behaviors to improve performance by reducing human error and increasing various measures of task efficiency. These experiments also illustrated the

opportunity for operator confusion regarding robot behavior and initiative. The first of these experiments showed that if operators were not able to predict robot behavior, a fight for control could emerge where the human tried to prevent or counter robot initiative, usually resulting in a significant performance decrement (Marble et al, 2003). Two groups emerged. One group understood and trusted the robot behaviors and achieved significant performance improvements over the baseline teleoperated system. The other group reported that they were confused by the robot taking the initiative and suffered a performance decrement when compared to their performance in the baseline teleoperation setting. This experiment and others like it showed that operator trust was a major factor in operational success and that operator trust was significantly impacted when the user made incorrect assumptions about robot behavior. The key question which emerged from the study was how the interface could be modified to correctly influence the user's assumptions.

Since these early experiments, a research team at the INL has been working not only to develop new and better behaviors, but, more importantly, to develop interaction tools and methods which convey a functional model of robot behavior to the user. In actuality, this understanding may be as important as the performance of the robot behaviors. Results from several experiments showed that augmenting robotic capability did not necessarily result in greater trust or enhanced operator performance (Marble et al., 2003; Bruemmer et al., 2005a; Bruemmer et al, 2005b). In fact, practitioners would prefer to use a low efficiency tool that they understand and trust than a high efficiency tool that they do not understand and do not fully trust. If this is indeed the case, then great care must be taken to explicitly design behaviors and interfaces which together promote an accurate and easily accessible understanding of robot behavior.

3. What Users Need to Know

One of the lessons learned from experimentally assessing the RIK with over a thousand human participants is that presenting the functionality of the RIK in the terms that a roboticist commonly uses (e.g. obstacle avoidance, path planning, laser-based change detection, visual follow) does little to answer the operators' basic question: "What does the robot do?" Rather than requesting a laundry list of technical capabilities, the user is asking for a fundamental understanding of what to expect – a mental model that can be used to guide expectations and input.

4. "What Does it Do?"

To introduce the challenges of human-robot interaction, we have chosen to briefly examine the development and use of a robotic event photographer developed in the Media and Machines Laboratory, in the Department of Computer Science and Engineering at Washington University in St Louis. The event photographer is an intelligent, autonomous robot designed to be used beyond the confines of laboratory in settings where training or education of the user set was nearly impossible. In this project, a mobile robot system acts as an event photographer at social events, wandering about the room, autonomously taking well-framed photographs of people (Byers et al., 2003; Byers et al., 2003; Smart, 2003). The system is implemented on an iRobot B21r mobile robot platform (see figure 1), a bright red cylindrical robot that stands about 4 feet tall. Mounted on top of the robot is a pair of stereo cameras, and a digital still camera (not shown in the figure), at roughly the eye-level of a

(short) human. The robot can rotate in place, and can also move forward and backward. The cameras can pan and tilt independently of the body.

Figure 1. iRobot B21 Intelligent Photographer

The system is completely autonomous, and all computation is performed on-board. This proved to be something of a problem from a human-robot interaction standpoint. Typically, when research robots are deployed in the real world, they are attended by a horde of graduate students. These students are there to make sure the deployment goes smoothly, to fix problems as they arise, and to physically extricate the robot from tricky situations when necessary. They also, however, act as translators and interpreters for members of the public watching the robot. The first question that most people have on seeing a robot operating in the real world is "What is it doing?" The attending graduate students can answer this question, often tailoring the explanation to the level of knowledge of the questioner.

However, since our system worked autonomously and rarely got into trouble, there were often no graduate students nearby. Members of the public had to interact with the system directly, and had to figure out what it was doing for themselves. This proved difficult, since the robot has no body language, none of the external cues that humans often have (such as camera bags), and was unable to answer questions about itself directly. Most of the time, it was impossible to tell if the robot was an event photographer, a security system, or simply wandering aimlessly.

The photographer was first deployed at SIGGRAPH 2002, the major computer graphics conference. Attendees at the conference generally have a technical background, and understand the basics of computer systems, cameras, and computation. Initially, we stationed a single graduate student near the robot, to answer questions about it, and hand out business cards. When asked about the robot, the student would generally start talking about navigation algorithms, automating the rules of photography, and face detection algorithms. While the listener understood each of these component technologies, they typically still did not understand what the robot was trying to accomplish. They lacked the

"big picture" view of the system, and interacted with it as if it was a demonstration of one of the components (face detection, for example). This led to significant unhappiness when, for example, the robot would move away from the human when they were trying to get it to detect their face. Some attendees actually stomped off angrily. They had been given a lecture on robotics capabilities when what they really needed to know was how to interact at a basic level and what to expect.

However, when we supplied the metaphor of "event photographer", the quality of the interaction was completely different. People immediately understood the larger context of the system, and were able to rationalize its behavior in these terms. When the robot moved before taking their picture, it was explained by "it's found someone else to take a picture of." People seemed much more willing to forgive the robot in these cases, and put it down to the fickleness of photographers. They were also much more willing to stand still while the robot lined up the shot, and often joked about the system being "a perfectionist." For the most part, people were instantly able to interact with the robot comfortably, with some sense that they were in control of the interaction. They were able to rationalize the robot's actions in terms of the metaphor ("it doesn't like the lighting here", "it feels crowded there"). Even if these rationalizations were wrong, it gave the humans the sense that they understood what was going on and, ultimately, made them more comfortable.

The use of the event photographer metaphor also allowed us to remove the attending graduate student, since passers-by could now describe the robot to each other. As new people came up to the exhibit, they would look at the robot for a while, and then ask someone else standing around what the robot was doing. In four words, "It's an event photographer", they were given all the context that they needed to understand the system, and to interact effectively with it. It is extremely unlikely that members of the audience would have remembered the exact technical details of the algorithms, let alone bothered to pass them on to the new arrivals. Having the right metaphor enabled the public to explain the robot to themselves, without the intervention of our graduate students. Not only is this metaphor succinct, it is easy to understand and to communicate to others. It lets the observers ascribe intentions to the system in a way that is meaningful to them, and to rationalize the behavior of the autonomous agent.

Although the use of an interaction metaphor allowed people to understand the system, it also entailed some additional expectations. The system, as implemented, did a good job of photographing people in a social setting. It was not programmed, however, for general social interactions. It did not speak or recognize speech, it did not look for social gestures (such as waving to attract attention), and it had no real sense of directly interacting with people. By describing the robot as an event photographer, we were implicitly describing it as being like a human even photographer. Human photographers, in addition to their photographic skills, do have a full complement of other social skills. Many people assumed that since we described the system as an event photographer, and since the robot did a competent job at taking pictures, that it was imbued with all the skills of human photographer. Many people waved at the robot, or spoke to it to attract its attention, and were visibly upset when it failed to respond to them. Several claimed that the robot was "ignoring them", and some even concocted an anthropomorphic reason, ascribing intent that simply wasn't there. These people invariably left the exhibit feeling dissatisfied with the experience.

Another problem with the use of a common interaction metaphor is the lack of physical cues associated with that metaphor. Human photographers raise and lower their cameras, and

have body language that indicates when a shot has been taken. The robot, of course, has none of these external signs. This led to considerable confusion among the public, since they typically assumed that the robot was taking no pictures. When asked why they thought this, they often said it was because the camera did not move, or did not move differently before and after a shot. Again, this expectation was introduced by our choice of metaphor. We solved the problem by adding a flash, which actually fired slightly after the picture was taken. This proved to be enough context to make everyone happy.

5. Developing a "Theory of Robot Behavior"

The need for human and robot to predict and understand one another's actions presents a daunting challenge. If the human has acquired a sufficient theory of robot behavior, s/he will be able to quickly and accurately predict: 1) Actions the robot will take in response to stimuli from the environment and other team members; 2) The outcome of the cumulative set of actions. The human may acquire this theory of behavior through simulated or real world training with the robot. Most likely, this theory of behavior (TORB) will be unstable at first, but become more entrenched with time. Further work with human participants is necessary to better understand the TORB development process and its effect on the task performance and user perception.

It may be helpful to consider another example where it is necessary for the human to build an understanding of an autonomous teammate. In order to work with a dog, the policeman and his canine companion must go through extensive training to build a level of expectation and trust on both sides. Police dog training begins when the dog is between 12 and 18 months old. This training initially takes more than four months, but critically, reinforcement training is continuous throughout the dog's life (Royal Canadian Police, 2002). This training is not for just the dog's benefit, but serves to educate the dog handlers to recognize and interpret the dog's movements which increase the handler's success rate in conducting task.

In our research we are not concerned with developing a formal model of robot cognition, just as a police man need not understand the mechanisms of cognition in the dog. The human must understand and predict the emergent actions of the robot, with or without an accurate notion of how intelligent processing gives rise to the resulting behavior. Many applications require the human to quickly develop an adequate TORB. One way to make this possible is to leverage the knowledge humans already possess about human behavior and other animate objects, such as pets or even video games, within our daily sphere of influence. For example, projects with humanoids and robot dogs have explored the ways in which modeling emotion in various ways can help (or hinder) the ability of a human to effectively formulate a TORB (Brooks et al. 1998). Regardless of how it is formed, an effective TORB allows humans to recognize and complement the initiative taken by robots as they operate under different levels of autonomy. It is this ability to predict and exploit the robot's initiative that will build operator proficiency and trust.

7. Components of Trust

There is no dearth of information experimental or otherwise suggesting that lack of trust in automation can lead to hesitation, poor decision making, and interference with task performance (Goodrich & Boer, 2000; Parasuraman & Riley, 1997; Lee & Moray, 1994; Kaber & Endsley, 2004). Since trust is important, how should we define it? For our purpose, trust

can be defined as a pre-commitment on the part of the operator to sanction and use a robot capability. In general, this precommittment is linked to the user's understanding of the system, acknowledgement of its value and confidence in its reliability. In other words, the user must believe that the robot has sufficient utility and reliability to warrant its use. In terms of robot behavior, trust can be measured as the user's willingness to allow the robot to accomplish tasks and address challenges using its own view of the world and understanding of the task. The behavior may be simple or complex, but always involves both input and output. To trust input, the human must believe that the robot has an appropriate understanding of the task and the environment. One method to build trust in the behavior input is to diagnose and report on robot sensor functionality. Another is to present an intelligible formatting of the robot's internal representation as in the instance of a map. Fostering appropriate distrust may be equally vital. For instance, if the robot's map of the world begins to degrade, trust in the robot's path planning should also degrade.

To trust the output, the human must believe that the robot will take action appropriate to the context of the situation. One example of how to bolster trust in this regard is to continually diagnose the robot's physical capacity through such means as monitoring battery voltage or force torque sensors on the wheels or manipulators. Although trust involves a pre-commitment, it is important to understand that trust undergoes a continual process of reevaluation based on the user's own observations and experiences. In addition to the value of self-diagnostic capabilities, the user will assess task and environment conditions, and monitor the occurrence of type I and type II errors, i.e., "false alarms" and "misses", associated with the robot's decision making.

Note that none of these components of trust require that the operator knows or has access to a full understanding of the robot system. The user does not need to understand every robot sensor to monitor behavior input; nor every actuator to trust the output. Neither does the human need to know volumes about the environment or all of the decision heuristics that the robot may be using. As behaviors become more complex, it is difficult even for developers to understand the fusion of data from many different sensors and the interplay of many independent behaviors. If an algorithmic understanding is necessary, even something as simple as why the robot turned left instead of right may require the developer to trace through a preponderance of debugging data. Put simply, the user may not have the luxury of trust through algorithmic understanding. Rather, the operator develops and maintains a relationship with the robot based on an ability to accomplish a shared goal.

8. Reducing Interaction Complexity

In his 1993 book, *Introduction to the Bootstrap*, Hans Hofmann, states that "The ability to simplify means eliminating the unnecessary so that the necessary may speak." One of the criticisms that the INL research team's early efforts received from colleagues, domain experts, practitioners and novice users was that the interface used to initiate and orchestrate behaviors was simply too complex. There were too many options, too many disparate perspectives and too many separate perceptual streams. This original interface included a video module, a map module, a camera pan – tilt – zoom module, a vehicle status window, a sensor status window and a obstruction module, to name a few.

As developers, it seemed beneficial to provide options, believing that flexibility was the key to supporting disparate users and enabling multiple missions. As new capabilities and behaviors were added, the interface options also multiplied. The interface expanded to

multiple robots and then to unmanned aerial vehicles (UAVs) and unattended ground sensors. Various application payloads were supported including chemical, radiological and explosive hazard detection capabilities. Now these all had to be represented and supported within the interface. In terms of perspectives, the interface needed to include occupancy grids, chemical and radiological plumes, explosive hazards detection, 3D range data, terrain data, building schematics, satellite imagery, real-time aerial imagery from UAVs and 3D representation of arm movement to support mobile manipulation. The critical question was how to support an ever increasing number of perceptions, actions and behaviors without increasing the complexity of the interface. To be useful in critical and hazardous environment such as countermine operations, defeat of improvised explosive devices and response to chemical, biological, radiological or nuclear hazards, the complexity of the task, particularly the underlying algorithms and continuous data streams must somehow not be passed on directly to the human.

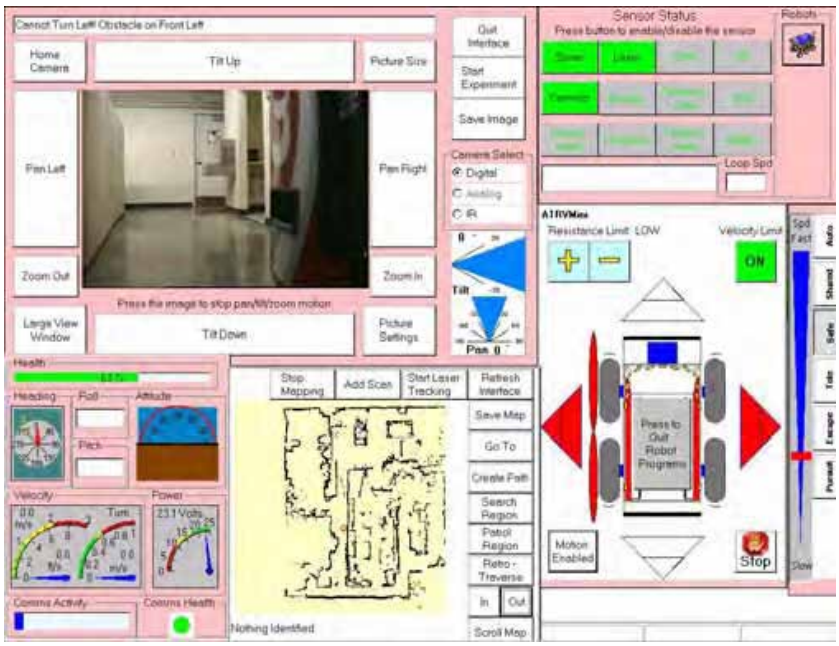


Figure 2: The Original RIK Interface

9. Data Abstraction and Perceptual Fusion

At a low level, robots must process multiple channels of chaotic, multi-dimensional sensor data that stream in from many different modalities. The user should not have to sift through this raw data or expend significant cognitive workload to correlate it. The first step to facilitating efficient human-robot interaction is to provide an efficient method to fuse and filter this data into basic abstractions. RIK provides a layer of abstraction that underlies all

robot behavior and communication. These abstractions provide elemental constructs for building intelligent behavior. One example is the ego-centric range abstraction which represents all range data in terms of an assortment of regions around the robot. Another is the directional movement abstraction which uses a variety of sensor data including attitude, resistance to motion, range data and bump sensing to decide in which directions the robot can physically move. Figure 3 shows how sensor data and robot state is abstracted into the building blocks of behavior and the fundamental outputs to the human.

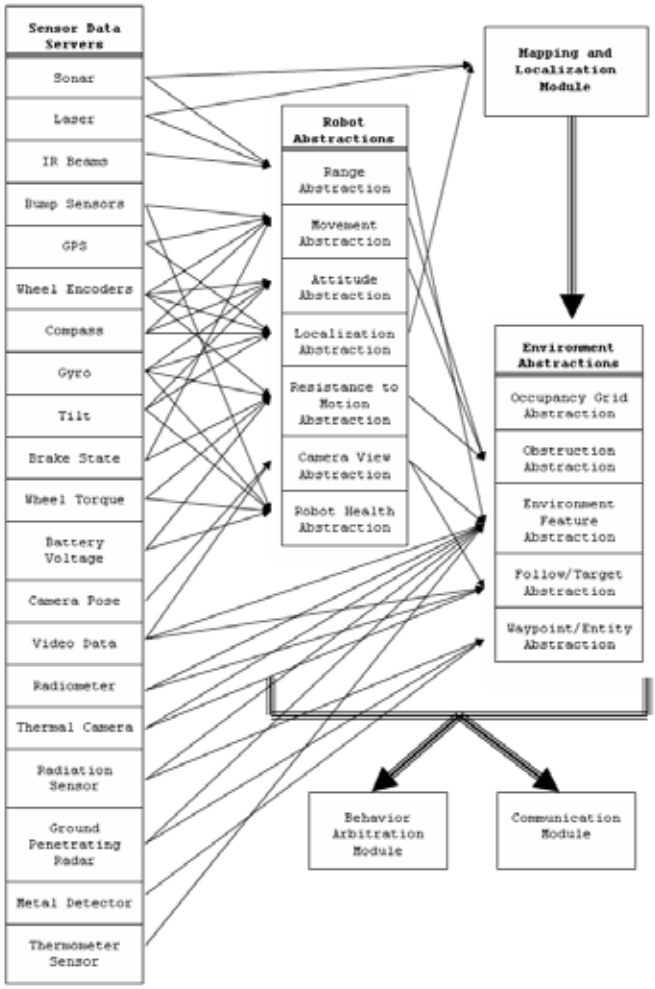


Figure 3. Robot and environment abstractions

The challenges of robot positioning provide an example of how this data fusion takes place as well as show how the abstractions are modulated by the robot to communicate information rather than data. To maintain an accurate pose, it is necessary to probabilistically fuse global positioning, simultaneous mapping and localization, inertial sensors and then correlate this with other data that might be available such as aerial imagery, a priori maps and terrain data. This data fusion and correlation should not be the burden of the human operator. Rather, the robot behaviors and interface intelligence should work hand in hand to accomplish this in a way that is transparent to the user. Figure 4 below shows how the Robot Intelligence Kernel – the suite of behaviors running on the robot fuses position information towards a consistent pose estimate.

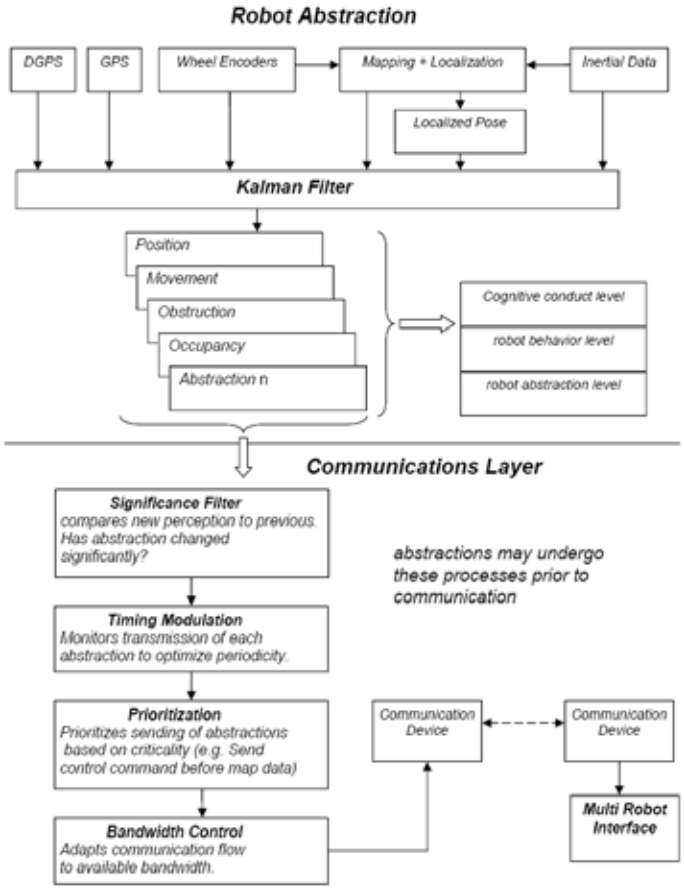


Figure 4. The use of perceptual and robot abstractions within the INL Robot Intelligence Kernel

These abstractions are not only the building blocks for robot behavior, but also serve as the fundamental atoms of communication back to the user. Rather than be bombarded with separate streams of raw data, this process of abstraction and filtering presents the user with only the useful end product. The abstractions are specifically designed to support the needs of situation awareness.

10. Fusing Disparate Perspectives

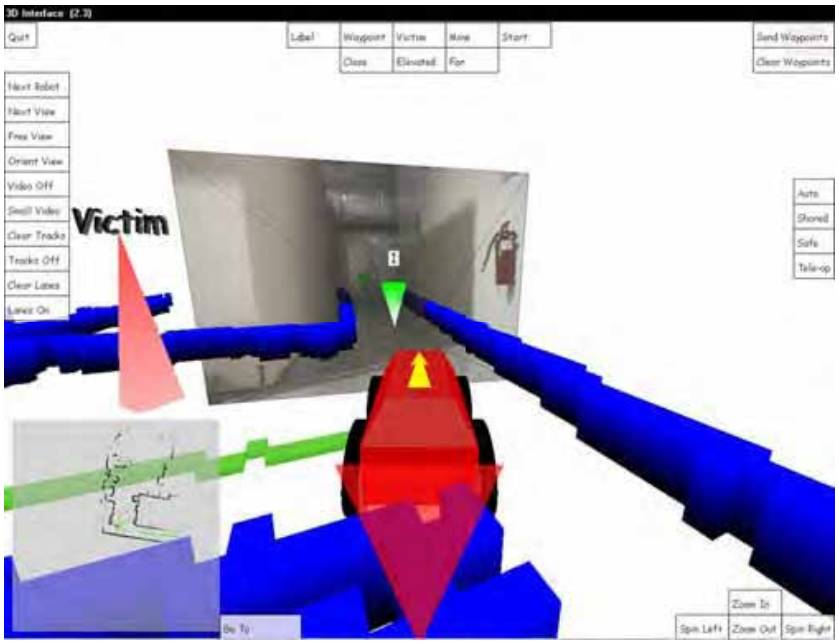


Figure 5. Interface showing fusion of video and map representation

Combining different perspectives into a common reference is another way to reduce complexity. On the interface side, efforts have been undertaken to visually render video, map data and terrain data into a seamless, scalable representation that can be zoomed in or out to support varying levels of operator involvement and the number of robots being tasked. Figure 5 below shows how video can be superimposed over the map data built up by the robot as it navigates. The Interface used with the Robot Intelligence Kernel and shown numerous times throughout this experiment was developed jointly with Brigham Young University and the Idaho National Laboratory (Nielsen & Goodrich 2006). Unlike traditional interfaces that require transmission of live video images from the ground robot to the operator, the representation used for this experiment uses a 3D, computer-game-style representation of the real world constructed on-the-fly. The digital representation is made possible by the robot implementing a map-building algorithm and transmitting the map information to the interface. To localize within this map, the RIK utilizes Consistent Pose Estimation (CPE)

developed by the Stanford Research Institute International (Konolige 1997). This method uses probabilistic reasoning to pinpoint the robot's location in the real world while incorporating new range sensor information into a high-quality occupancy grid map. When features exist in the environment to support localization, this method has been shown to provide approximately +/- 10 cm positioning accuracy even when GPS is unavailable.
Figure 6 shows how the same interface can be used to correlate aerial imagery and provide a contextual backdrop for mobile robot tasking. Note that the same interface is used in both instances, but that Figure 5 is using an endocentric view where the operator has focused the perspective on a particular area whereas Figure 6 shows an exocentric perspective where the operator is given a vantage point over a much larger area. Figure 6 is a snap shot of the interface taken during an experiment where the functionality and performance benefit of the RIK was assessed within the context of a military countermine mission to detect and mark buried landmines.

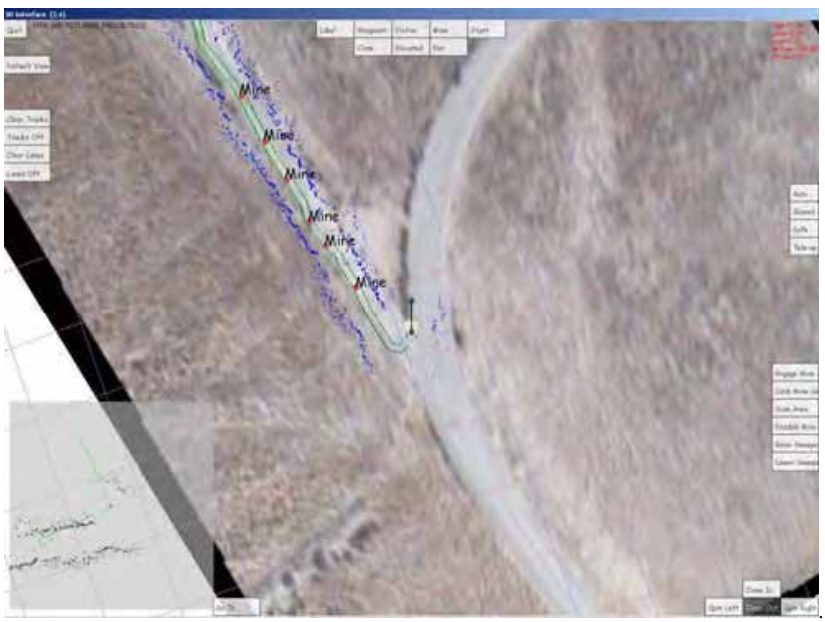


Figure 6. Interface showing fused robot map and mosiaced real-time aerial imagery during a UAV-UGV mine detection task

This fusion of data from air and ground vehicles is more than just a situation awareness tool. In fact, the display is merely the visualization of a collaborative positioning framework that exists between air vehicle, ground vehicle and human operator. Each team member contributes to the shared representation and has the ability to make sense of it in terms of its own, unique internal state. In fact, one lesson learned from the countermine work is that it may not be possible to perfectly fuse the representations especially when error such as positioning inaccuracy and camera skew play a role. Instead, it may be possible to support

collaboration by sharing information about landmarks and key environmental features that can be identified from multiple perspectives such as the corners of buildings or the intersection between two roads. The benefits of this strategy for supporting collaboration will be discussed further in Case Study One.

Simplifying the interface by correlating and fusing information about the world makes good sense. However, sensor fusion is not sufficient to actually change the interaction itself – the fundamental inputs and outputs between the human and the robotic system. To reduce true interaction complexity, there must be some way not only to abstract the robot physical and perceptual capabilities, but to somehow abstract away from the various behaviors and behavior combinations necessary to accomplish a sophisticated operation.

11. Understanding Modes of Autonomy

Another source of complexity within the original interface was the number of autonomy levels available to the user. When multiple levels of autonomy are available the operator has the responsibility of choosing the appropriate level of autonomy. Within the original interface, when the user wished to change the level of initiative that the robot is permitted to take, the operator would select between five different discrete modes. In teleoperation mode the user is in complete control and the robot takes no initiative. In safe mode the robot takes initiative only to protect itself or the environment but the user retains responsibility for all motion and behavior. In shared mode the robot does the driving and selects its own route whereas the operator serves as a backseat driver, providing directional cues throughout the task. In collaborative tasking mode, the human provides only high level intentions by placing icons that request information or task-level behavior (i.e. provide visual imagery for this target location; search this region for landmines; find the radiological source in this region). Full autonomy is a configuration rarely used whereby the system is configured to accept no human input and to accomplish a well-defined task from beginning to end. Table 1 below shows the operator and robot responsibilities for each autonomy mode.

Autonomy Mode	Defines Task Goals	Supervises Direction	Motovates Motion	Prevents Collision
Teleoperation Mode	Operator	Operator	Operator	Operator
Safe Mode	Operator	Operator	Operator	Robot
Shared mode	Operator	Operator	Robot	Robot
Collaborative Tasking Mode	Operator	Robot	Robot	Robot
Autonomous Mode	Robot	Robot	Robot	Robot

Table 1: Responsibility for operator and robot within each of five autonomy modes

Figure 7 below shows how the various behaviors that are used to support tasking in different autonomy modes.

The challenge with this approach is that operators often do not realize when they are in a situation where the autonomy on the robot should be changed and are unable to predict how a change in autonomy levels will actually affect overall performance. As new behaviors and intelligence are added to the robot, this traditional approach of providing wholly separate

modes of autonomy requires the operator to maintain appropriate mental models of how the robot will behave in each mode and how and when each mode should be used. This may not be difficult for simple tasks, but once a variety of application payloads are employed the ability to maintain a functional understanding of how these modes will impact the task becomes more difficult. New responses to this challenge will be discussed in Case Study Two.

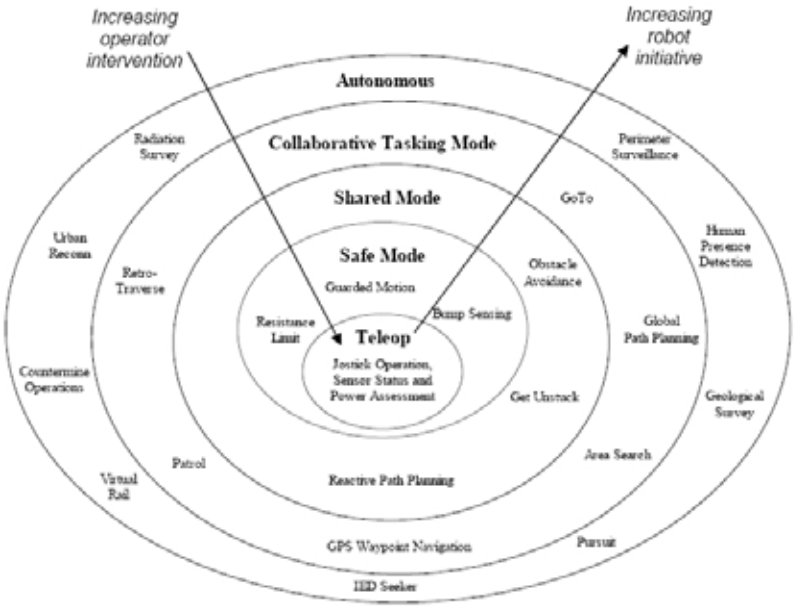


Figure 7. Behaviors associated with each of the five modes of autonomy

12. Case Study One: Robotic Demining

Landmines are a constant danger to soldiers during conflict and to civilians long after conflicts cease, causing thousands of deaths and tens of thousands of injuries every year. More than 100 million landmines are emplaced around the world and, despite humanitarian efforts to address the problem, more landmines are being emplaced each day than removed. Many research papers describe the challenges and requirements of humanitarian demining along with suggesting possible solutions (Nicoud & Habib, 1995; Antonic et. al., 2001). Human mine sweeping to find and remove mines is a dangerous and tedious job. Moreover, human performance tends to vary drastically and is dependent on factors such as fatigue, training and environmental conditions. Clearly, this is an arena where robot behaviors could someday play an important role. In terms of human-robot interaction, the need to locate and mark buried landmines presents a unique opportunity to investigate the value of shared representation for supporting mixed-initiative collaboration. A collaborative representation

is one of the primary means by which it is possible to provide the user with insight into the behavior of the unmanned systems.

13. Technical Challenges

It has long been thought that landmine detection is an appropriate application for robotics because it is dull, dirty and dangerous. However, the reality has been that the critical nature of the task demands a reliability and performance that neither teleoperated nor autonomous robots have been able to provide. The inherent complexity of the countermine mission presents a significant challenge for both the operator and for the behaviors that might reside on the robot. Efforts to develop teleoperated strategies to accomplish the military demining task have resulted in remarkable workload such that U.S. Army Combat Engineers report that a minimum of three operators are necessary to utilize teleoperated systems of this kind. Woods et al. describe the process of using video to navigate a robot as attempting to drive while looking through a 'soda straw'because of the limited angular view associated with the camera (Woods et al., 2004). If teleoperation is problematic for simple navigation tasks, the complexity of trying to use video remotely to keep track of where the robot has been over time as well as precisely gauge where its sensor has covered. Conversely, autonomous solutions have exhibited a low utility because the uncertainty in positioning and the complexity of the task rendered the behaviors less than effective. Given these challenges, it seemed prudent to explore the middle ground between teleoperation and full autonomy.
The requirement handed down from the US Army Maneuver Support Battlelab in Ft. Leonard-Wood was to physically and digitally mark the boundaries of a 1 meter wide dismounted path to a target point, while digitally and physically marking all mines found within that lane. Previous studies had shown that real-world missions would involve limited bandwidth communication, inaccurate terrain data, sporadic availability of GPS and minimal workload availability from the human operator. These mission constraints precluded conventional approaches to communication and tasking. Although dividing control between the human and robot offers the potential for a highly efficient and adaptive system, it also demands that the human and robot be able to synchronize their view of the world in order to support tasking and situation awareness. Specifically, the lack of accurate absolute positioning not only affects mine marking, but also human tasking and cooperation between vehicles.

14. Mixed-Initiative Approach

Many scientists have pointed out the potential for benefits to be gained if robots and humans work together as partners (Fong et. al., 2001; Kidd 1992; Scholtz & Bahrami, 2003; Sheridan 1992). For the countermine mission, this benefit cannot be achieved without some way to merge perspectives from human operator, air vehicle and ground robot. On the other hand, no means existed to support a perfect fusion of these perspectives. Even with geo-referenced imagery, real world trials showed that the GPS based correlation technique does not reliably provide the accuracy needed to support the countermine mission. In most cases, it was obvious to the user how the aerial imagery could be nudged or rotated to provide a more appropriate fusion between the ground robot's digital map and the air vehicle's image. To alleviate dependence on global positioning, collaborative tasking tools were developed that use common reference points in the environment to correlate disparate internal representations (e.g. aerial imagery and ground-based occupancy grids).

As a result, correlation tools were developed that allow the user to select common reference points within both representations. Examples of these common reference points include the corners of buildings, fence posts, or vegetation marking the boundary of roads and intersections. In terms of the need to balance human and robot input, it was clear that this approach required very little effort from the human (a total of 4 mouse clicks) and yet provided a much more reliable and accurate correlation than an autonomous solution. This was a task allocation that provided significant benefit to all team members without requiring significant time or workload.

The mission scenario which emerged included the following task elements.

a) Deploy a UAV to survey terrain surrounding an airstrip.
b) Analyze mosaiced real-time imagery to identify possible minefields.
c) Use common landmarks to correlate UAV imagery & umanned ground vehicle (UGV) occupancy map
d) UGV navigates autonomously to possible minefield
e) UGV searches for and mark mines.
f) UGV marks dismounted lane through minefield.

The behavior decomposition allows each team member to act independently while communicating environmental features and task intent at a high level.

To facilitate initiative throughout the task, the interface must not only merge the perspectives of robotic team members, but also communicate the intent of the agents. For this reason, the tools used in High Level Tasking were developed which allow the human to specify coverage areas, lanes or target locations. Once a task is designed by the operator, the robot generates an ordered waypoint list or path plan in the form of virtual colored cones that are superimposed onto the visual imagery and map data. The placement and order of these cones updates in real time to support the operator's ability to predict and understand the robot's intent. Using a suite of click and drag tools to modify these cones the human can influence the robot's navigation and coverage behavior without directly controlling the robot motion.

15. Robot Design

Figure 8: The Arcturus T-15 airframe and launcher

The air vehicle of choice was the Arcturus T-15 (see Figure 8), a fixed wing aircraft that can maintain long duration flights and carry the necessary video and communication modules. For the countermine mission, the Arcturus was equipped to fly two hour reconnaissance missions at elevations between 200 and 500ft. A spiral development process was undertaken to provide the air vehicle with autonomous launch and recovery capabilities as well as path planning, waypoint navigation and autonomous visual mosaicing. The resulting mosaic can be geo-referenced if compared to a priori imagery, but even then does not provide the positioning accuracy necessary to meet the 10cm accuracy requirements for the mission. On the other hand, the internal consistency of the mosaic is very high since the image processing software can reliably stitch the images together.
Carnegie Mellon University developed two ground robots (see Figure 9) for this effort which were modified humanitarian demining systems equipped with inertial systems, compass, laser range finders and a low-bandwidth, long range communication payload. A MineLab F1A4 detector which is standard issue mine detector for the U. S. Army, was mounted on both vehicles together with an actuation mechanism that can raise and lower the sensor as well as scan it from side to side at various speeds. A force torque sensor was used to calibrate sensor height based on sensing pressure exerted on the sensor when it touches the ground. The mine sensor actuation system was designed to scan at different speeds to varying angle amplitudes throughout the operation. Also, the Space and Naval Warfare Systems Center in San Diego developed a compact marking system that dispenses two different colors of agricultural dye. Green dye was used to mark the lane boundaries and indicate proved areas while red dye was used to mark the mine locations. The marking system consists of two dye tanks, a larger one for marking the cleared lane and a smaller one for marking the mine location.

Figure 9: Countermine robot platform

16. Experiment

The resulting system was rigorously evaluated by the Army Test and Evaluation Command (TECO) and found to meet the Army's threshold requirement for the robotic countermine mission. A test lane was prepared on a 50 meter section of an unimproved dirt road leading off of an airstrip. Six inert A-15 anti tank (AT) landmines were buried on the road at varying depths. Sixteen runs were conducted with no obstacles on the lane and 10 runs had various obstacles scattered on the lane. These obstacles included boxes and crates as well as

sagebrush and tumble weeds. The ARCS was successful in all runs in autonomously negotiating the 50 meter course and marking a proofed 1-meter lane. The 26 runs had an average completion time of 5.75 minutes with a 99% confidence interval of +/- 0.31 minutes. The maximum time taken was 6.367 minutes.

Figure 10: Proofed Lane and Mine Marking

The robot was able to detect and accurately mark, both physically and digitally, 130 out of 135 buried mines. Throughout the experiment there was one false detection. The robot also marked the proved mine-free lanes using green dye. The robot was able to navigate cluttered obstacles while performing various user-defined tasks such as area searches and the de-mining of roads and dismounted lanes.

Figure 11. Interface shows the operator the position of mines detected along a road

17. Discussion

When compared to the current military baseline, the mixed-initiative system produced a fourfold decrease in task time to completion and a significant increase in detection accuracy.This is particularly interesting since previous attempts to create robotic demining systems had failed to match human performance. The difference between past strategies and the one employed in this study is not that the robot or sensor was more capable; rather, the most striking difference was the use of mixed-initiative control to balance the capabilities and limitations of each team member. Without the air vehicle providing the tasking backdrop and the human correlating it with the UGV map, it would not have been possible to specify the lane for the robot to search. The research reported here indicates that operational success was possible only through the use of a mixed-initiative approach that allowed the human, air vehicle and ground vehicle to support one another throughout the mission. These findings indicate that by providing an appropriate means to interleave human and robotic intent, mixed initiative behaviors can address complex and critical missions where neither teleoperated nor autonomous strategies have succeeded.

Another interesting facet of this study is to consider how the unmanned team compares to a trained human attempting the same task. When comparing the robot to current military operations, the MANSCEN at Ft. Leonard Wood reports that it would take approximately 25 minutes for a trained soldier to complete the same task accomplished by the robot, which gives about a four-fold decrease in cycle time without putting a human in harm's way. Furthermore, a trained soldier performing a counter-mine task can expect to discover 80% of the mines. The robotic solution raises this competency to 96% mine detection. Another interesting finding pertained to human input is that the average level of human input throughout the countermine exercises, namely the time to set up and initiate the mission, was less than 2% when calculated based on time. The TECO of the U.S. Army indicated that the robotic system achieved "very high levels of collaborative tactical behaviors."

One of the most interesting HRI issues illustrated by this study is the fact that neither video nor joystick control was used. In fact, due to the low bandwidth required, the operator could easily have been hundreds of miles away from the robot, communicating over a cell phone modem. Using the system as it was actually configured during the experiment, the operator had the ability to initiate the mission from several miles away. Once the aerial and ground perspectives are fused within the interface, the only interaction which the soldier would have with the unmanned vehicles would be initiating the system and selecting the target location within the map.

18. Case Study Two: Urban Search and Rescue

This case study evaluates collaborative tasking tools that promote dynamic sharing of responsibilities between robot and operator throughout a search and detection task. The purpose of the experiment was to assess tools created to strategically limit the kind and level of initiative taken by both human and robot. The hope was that by modulating the initiative on both sides, it would be possible to reduce the deleterious effects referred to earlier in the chapter as a "fight for control" between the human and robot. Would operators notice that initiative was being taken from them? Would they have higher or lower levels of workload? How would overall performance be affected in terms of time and quality of data achieved?

19. Technical Challenge

While intelligent behavior has the potential to make the user's life easier, experiments have also demonstrated the potential for collaborative control to result in a struggle for control or a suboptimal task allocation between human and robot (Marble et al., 2003; Marble et al., 2004; Bruemmer et al., 2005). In fact, the need for effective task allocation remains one of the most important challenges facing the field of human-robot interaction (Burke et al., 2004). Even if the autonomous behaviors on-board the robot far exceed the human operators ability, they will do no good if the human declines to use them or interferes with them. The fundamental difficulty is that human operators are by no means objective when assessing their own abilities (Kruger & Dunning, 1999; Fischhoff et al., 1977). The goal is to gain an optimal task allocation such that the user can provide input at different levels without interfering with the robot's ability to navigate, avoid obstacles and plan global paths.

20. Mixed-Initiative Approach

In shared mode (see table 1), overall team performance may benefit from the robot's understanding of the environment, but can suffer because the robot does not have insight into the task or the user's intentions. For instances, absent of user input, if robot is presented with multiple routes through an area shared mode will typically take the widest path through the environment. As a result, if the task goal requires or the human intends the exploration of a navigable but restricted path, the human must override the robot's selection and manually point the robot towards the desired corridor before returning system control to the shared autonomy algorithms. This seizing and relinquishing of control by the user reduces mission efficiency, increases human workload and may also increase user distrust or confusion. Instead, the CTM interface tools were created to provide the human with a means to communicate information about the task goals (e.g. path plan to a specified point, follow a user defined path, patrol a region, search an area, etc) without directly controlling the robot. Although CTM does support high level tasking, the benefit of the collaborative tasking tools is not merely increased autonomy, but rather the fact that they permit the human and robot to mesh their understanding of the environment and task. The CTM toolset is supported by interface features that illustrate robot intent and allow the user to easily modify the robot's plan. A simple example is that the robot's current path plan or search matrix is communicated in an iconographic format and can be easily modified by dragging and dropping vertices and waypoints. An important feature of CTM in terms of mixed-initiative control is that joystick control is not enabled until the CTM task is completed. The user must provide input in the form of intentionality rather than direct control. However, once a task element is completed (i.e target is achieved or area searched), then the user may again take direct control. Based on this combined understanding of the environment and task, CTM is able to arbitrate responsibility and authority.

21. Robot Design

The experiments discussed in this paper utilized the iRobot "ATRV mini" shown on the left in Figure 12. The robot utilizes a variety of sensor information including compass, wheel encoders, laser, computer camera, tilt sensors, and ultrasonic sensors. In response to laser and sonar range sensing of nearby obstacles, the robot scales down its speed using an event

horizon calculation, which measures the maximum speed the robot can safely travel in order to come to a stop approximately two inches from the obstacle. By scaling down the speed by many small increments, it is possible to insure that regardless of the commanded translational or rotational velocity, guarded motion will stop the robot at the same distance from an obstacle. This approach provides predictability and ensures minimal interference with the operator's control of the vehicle. If the robot is being driven near an obstacle rather than directly towards it, guarded motion will not stop the robot, but may slow its speed according to the event horizon calculation. The robot also uses a mapping, localization system developed by Konolige et al.

Figure 12

22. Experiment

A real-world search and detection experiment was used to compare shared mode where the robot drives, but the human can override the robot at any time, to a Collaborative Tasking Mode (CTM), where the system dynamically constrains user and robot initiative based on the task element. The task was structured as a remote deployment such that the operator control station was located several stories above the search arena so that the operator could not see the robot or the operational environment. Plywood dividers were interspersed with a variety of objects such as artificial rocks and trees to create a 50ft x 50ft environment with over 2000 square feet of navigable space. Each participant was told to direct the robot around the environment and identify items (e.g. dinosaurs, a skull, brass lamp, or building blocks) located at the numbers represented on an a priori map. In addition to identifying items, the participants were instructed to navigate the robot back to the Start/Finish to complete the loop around the remote area. This task was selected because it forced the participants to navigate the robot as well as use the camera controls to identify items at particular points along the path. The items were purposely located in a logical succession in an effort to minimize the affect of differences in the participants' route planning skills.
In addition to the primary task of navigating and identifying objects the participants were asked to simultaneously conduct a secondary task which consisted of answering a series of basic two-digit addition problems on an adjacent computer screen. The participants were instructed to answer the questions to the best of their ability but told that they could skip a problem by hitting the <enter> key if they realized a problem appeared but felt they were

too engaged in robot control to answer. Each problem remained present until it was responded to, or the primary task ended. Thirty seconds after a participant's response, a new addition problem would be triggered. The secondary task application recorded time to respond, in seconds, as well as the accuracy of the response and whether the question was skipped or ignored.

During each trial, the interface stored a variety of useful information about the participant's interactions with the interface. For instance, the interface recorded the time to complete the task to be used as a metric of the efficiency between the methods of control. For the CTM participants, the interface also recorded the portion of time the robot was available for direct control. The interface recorded the number of joystick vibrations caused by the participant instructing the robot to move in a direction in which it was not physically possible to move. The number of joystick vibrations represent instances of human navigational error and, in a more general sense, confusion due to a loss of situation awareness (See Figure 13). The overall joystick bandwidth was also logged to quantify the amount of joystick usage. Immediately after completing a trial, each participant was asked to rank on a scale of 1 to 10 how "in control" they felt during the operation, where 1 signified "The robot did nothing that I wanted it to do" and 10 signified, "The robot did everything I wanted it to do."

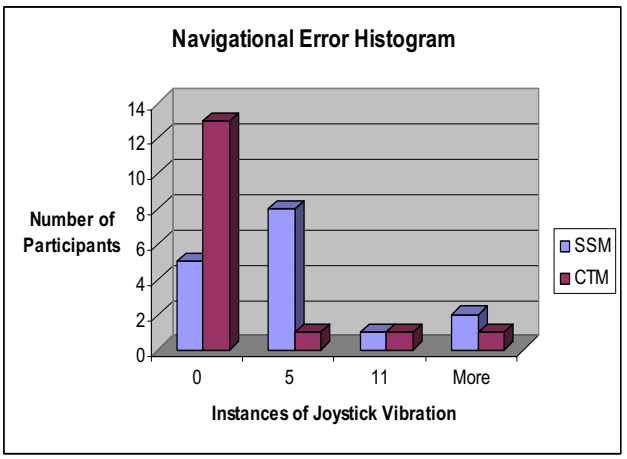


Figure 13. Navigational Error

All participants completed the assigned task. Analysis of the time to complete the task showed no statistically significant difference between the shared mode and CTM groups. An analysis of human navigational error showed that 81% of participants using CTM experienced no instances of operator confusion as compared to 33% for the shared mode participants (see Figure 13). Overall, shared mode participants logged a total of 59 instances of operator confusion as compared with only 27 for the CTM group.

The CTM participants collectively answered 102 math questions, while the shared mode participants answered only 58. Of questions answered, CTM participants answered 89.2% correctly as compared to 72.4% answered correctly by participants using shared mode. To further assess the ability of shared mode and CTM participants to answer secondary task

questions an analysis was performed on the average response time for each group. CTM participants had a statistically significant average response time of 25.1 seconds as compared to 49.2 seconds for those using shared mode. Together these results indicate that the participants using the collaborative tasking tools experienced a substantial decrease in the required workload to complete the task. In addition, CTM participants enjoyed a higher overall feeling of control as compared to shared mode participants (see Figure 14).

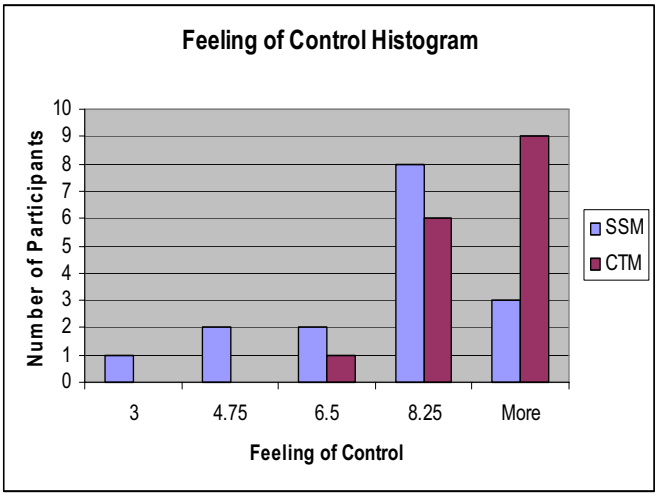


Figure 14. Feeling of Control

23. Discussion

This experiment provides validation of the collaborative tasking tools that have been implemented as part of the RIK. The experiment showed that from an engineering perspective, the blending of guarded motion, reactive obstacle avoidance and global path planning behaviors on board the robot can be used effectively to accomplish a search and detection task. Of greater significance to the Human-Robot Interaction (HRI) community is the fact that this experiment represents a definitive step away from the supervisory control paradigm where the human may accept or decline robot initiative, while remaining at all times in the leadership role for all task elements. Instead, the collaborative tasking tools presented here arbitrate leadership in a facilitative manner to optimize overall team performance. By constraining operator initiative at the right times, CTM reduces human confusion and frustration. Data from this study suggests that the CTM serves in a suprising fashion to increases users' feeling of control by taking control away from them. Something we had not initially predicted. Although the HRI community has long used the phrase "mixed initiative" to describe the goal of team members blending their input together,the findings of this paper imply that rather than "mixing" initiative, human-robot teaming may benefit when initiative is "facilitated" to avoid conflict and optimize task allocation.

24. Conclusion

All too often, increased robot capability has been handed down to the user in the form of increased interaction complexity. As we have seen, autonomous robotic behaviors do not necessarily provide a performance benefit and may lead to operator confusion and distrust if the system does not support a mental model that can be easily adopted or used by operators. As robot behaviors become increasingly complex, it is imperative that we find a means to hide complexity while still keeping users in the know and allowing them to be part of the action. By so doing, the operator can be freed up to successfully oversee the performance of greater numbers of robots while maintaining a greater sense of his or her own situation awareness. We believe that the mixed-initiative tools discussed within this chapter provide evidence that if we are willing to move beyond the existing tools and metaphors, it is possible to craft mixed-initiative interaction methods that can enhance operator and system performance, and decrease operator workload.

Ideally, the interaction metaphor that underlies these tools should be functionally linked to the application such that the human need no longer consider individual actions, perceptions or behaviors, but rather can begin to reason about the task elements that are native to the application domain. Thus, one might say that an effective interaction metaphor provides a readily apparent abstraction from robot behaviors into mission tasking. The more elegant the interaction metaphor, the less the user will have to think about the robot and the more s/he can begin to think about the environment and task in their own terms. It is because of this that an elegant interaction metaphor will, by its very nature, simplify the theory of robot behavior necessary for the operator to efficiently interact with the system. By simplifying the necessary theory of robot behavior it may be possible to reduce uncertainty, bring users' expectations into line, reduce the dimensionality of human-robot communication and narrow the possible outcomes. Ultimately, the overall complexity of the system does not change. Instead of the human performing the cognitive mapping between intent and robot behaviors, the interface must now play a role in accomplishing this mapping. Intelligent interface tools can be used to orchestrate robot behaviors according to the operative interaction metaphor.

24. References

Antonic, D., Ban, Z. & Zagar, M. (2001) Demining robots - requirements and constraints. *Automatika*, 42(3-4)

Arkin, R. C. (1997) *Behavior-Based Robotics*. MIT Press. Cambridge, MA

Brooks, R. ; Brazeal, C., Marjanovic, M., Scassellati, B., & Williamson, M. (1998) The Cog Project: Building a Humanoid Robot. *In Computation for Metaphors, Analogy, and Agents*, Springer Lecture Notes in Computer Science, 1562, C. Nehaniv, Ed., Springer-Verlag, Berlin

Bruemmer, D. J. ; Few, D. A., Walton, M. C., Boring, R. L., Marble, J. L., Nielsen, C. W., & Garner, J. (2005) . Turn off the television!: Real-world robotic exploration experiments with a virtual 3D display . *In Proceedings of the Hawaii International Conference on System Sciences (HICSS*). Waikoloa Village, Hawaii. January 3-6

Bruemmer, D.J. ; Few, D.A., Boring, R. L., Marble, J. L., Walton, M. C. & Nielsen C. W. (2005) Shared Understanding for Collaborative Control. IEEE *Transactions on Systems, Man, and Cybernetics, Part A: Systems and Humans.* Vol. 35, no.4, pp. 505-512. July

Bruemmer, D J; Boring, R. L,. Few. D.A., & Walton M. C. (2004). I Call Shotgun: An Evaluation of a Mixed-Initiative Control for Novice Users of a Search and Rescue Robot *In proceedings of 2004 IEEE International Conference on Systems, Man & Cybernetics*, The Hague, ND October 10–13

Burke, J. L. ; Murphy, R. R., Coovert, M. D. & Riddle D. L. (2004) Moonlight in Miami: A field study of human-robot interaction in the context of an urban search and rescue disaster response training exercise, *Human-Computer Interaction*, 19:85–116

Byers, Z. ; Dixon, M., Goodier, K., Grimm, C. M., & Smart, W. D. (2003) An Autonomous Robot Photographer, In *Proceedings of the IEEE/RSJ International Conference on Robots and Systems (IROS 2003*, Vol 3, pages 2636-2641, Las Vegas, Nevada.

Byers, Z., Dixon, M., Goodier, K., Smart, W. D. & Grimm, C.M. (2003) Say Cheese!: Experiences with a Robot *In Proceedings for the Fifteenth Innovative Applications of Artificial Intelligence Conference (IAAI 2003)*, John Riedl and Randy Hill (editors), pages 65-70, Acapulco, Mexico

Conley, D. T., & Goldman, P. (1994) Facilitative Leadership: How Principals Lead Without Dominating. Eugene, Oregon: Oregon School Study Council, August

Desai. M. & Yanco H. A. Blending Human and Robot Inputs for Sliding Scale Autonomy.(2005) *Proceedings of the 14th IEEE International Workshop on Robot and Human Interactive Communication (Ro-MAN).* Nashville, TN, August

Fischhoff, B. ; Slovic, P., & Lichtenstein S, (1977) Knowing with Certainty: The appropriateness of extreme confidence *Journal of Experimental Psychology: Human Perception and Performance,* 3:552-564

Fong, T. ; Thorpe, C. & Baur, C. (2001) Collaboration, dialogue,and human robot interaction. *In 10th International Symposium of Robotics Research*, Lourne, Victoria, Australia, November

Gertman, D. I., Blackman, H. S., Marble, J. L., Byers, J., & Smith, C. (2005) The SPAR-H Human Reliability Analysis Method, NUREG/CR-6883, US Nuclear Regulatory Commission, Washington, DC

Kidd, P.T. (1992) Design of human-centered robotic systems. In Mansour Rahimi and Waldemar Karwowski, editors, *Human Robot Interaction*, pages 225–241. Taylor and Francis, London, England

Konolige, K. (2004) Large-scale map-making. *In Proceedings of the National Conference on AI (AAAI),* San Jose, CA

Kruger, J., & Dunning, D. (1999). Unskilled and unaware of it: How difficulties in recognizing one's own incompetence lead to inflated self-assessments. *Journal of Personality and Social Psychology*, 77, 1121-1134

Maxwell B. A. ; Smart W. D., Jacoff A., Casper, J., Weiss, B., Scholtz, J., Yanco, H., Micire, M., Stroupe, A., Stormont, D., Lauwers, T., (2004) The 2003 AAAI Robot Competition and Exhibition. *AI Magazine*. Vol. 25, no. 2, pp. 68-80

Marble, J. L. ; Bruemmer, D. J., and Few, D. A. (2003) Lessons Learned from Usability Tests with a Collaborative Cognitive Workspace for Human-Robot Teams. In *Proceedings of the IEEE International Conférence on Systems, Man and Cybernetics*, Washington, D.C., October 5-8

Marble, J. L. ; Bruemmer D. J., Few, D. A. and Dudenhoeffer D. D. (2004) Evaluation of supervisory vs. peer-peer interaction for human-robot teams *In Proceedings of the 37th Annual Hawaii International Conference on Systems Sciences*, Big island, Hawaii, January 4-8

Nicoud, J. D. &. Habib, M. K (1955)The pemex-b autonomous demining robot: perception and navigation strategies. *In Proceedings of intelligent robots and systems*, pages 419–424, Pittsburgh, PA

Nielsen Curtis W. and Michael A. Goodrich. Testing the usefulness of a pan-tilt-zoom (PTZ) camera in humanrobot interactions. In Proceedings *of the Human Factors and Ergonomics Society Meeting*, San Francisco, CA, 2006.

Pacis, E.B., Everett, H. R., Farrington,N., . Bruemmer D. J, (2004) Enhancing Functionality and Autonomy in Man-Portable Robots, In *Proceedings of the SPIE Defense and Security Symposium 2004.* 13 -15 April

Royal Canadian Mounted Police: Police Dog Services, K-9 Recruit Successfully Completes Training, 2002. [Online]. Available: http://www.rcmp-grc.gc.ca/html/dogs.htm

Scholtz, J. & Bahrami, S. (2003) Human-robot interaction: development of an evaluation methodology for the bystander role of interaction. *In Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics*, volume 4, pages 3212–3217

Sellner, B. ; Heger, F. W., Hiatt, L. M, Simmons, R. & Singh, S. (2006) Coordinated multiagent teams and sliding autonomy for large-scale assembly. *Proceedings of the IEEE*, 94(7):1425–1444

Sheridan,T. B. (1992) Telerobotics, automation, and human supervisory control. MIT Press, Cambridge, MA

Smart, W. D. ; Grimm, C. M., Dixon, M. and Byers, Z. (2003) (Not) Interacting with a Robot Photographer,. *In Human Interaction with Autonomous Systems in Complex Environments*: Papers from the 2003 AAAI Spring Symposium, Stanford University, March 24-26,David Kortenkamp and Michael Freed (editors), pages 181-186, Stanford California

Woods, D. D., Tittle, J., Feil, M., F.& Roesler, A. (2004) Envisioning human-robot coordination in future operations. *IEEE Transactions on Systems, Man, and Cybernetics*, Part C, 34(2):210–218, May.

4

Augmented Reality for Human-Robot Collaboration

Scott A. Green[1, 2], Mark Billinghurst[2], XiaoQi Chen[1] and J. Geoffrey Chase[1]
[1]*Department of Mechanical Engineering, University of Canterbury*
[2]*Human Interface Technology Laboratory, New Zealand (HITLab NZ)*
New Zealand

1. Introduction

Although robotics is well established as a research field, there has been relatively little work on human-robot collaboration. This type of collaboration is going to become an increasingly important issue as robots work ever more closely with humans. For example, in space exploration, recent research has pointed out that to reduce human workload, costs, fatigue driven error and risk, intelligent robotic systems will need to be a significant part of mission design (Fong and Nourbakhsh 2005). Fong and Nourbakhsh also observe that scant attention has been paid to joint human-robot teams, and that making human-robot collaboration natural and efficient is crucial to future space exploration. NASA's vision for space exploration stresses the cultivation of human-robotic systems (NASA 2004). In addition, companies such as Honda (Honda 2007), Toyota (Toyota 2007) and Sony (Sony 2007) are interested in developing consumer robots that interact with humans in the home and workplace. Finally, the Cogniron project (COGNIRON 2007), MIT Media lab (Hoffmann and Breazeal 2004) and the Mitsubishi Electric Research Laboratories (Sidner and Lee 2005), among others, are currently conducting research in human-robot interaction (HRI). HRI has become a research field in its own right, as shown by the 2006 inaugural conference for HRI with the theme Toward Human-Robot Collaboration (HRI2006 2006).

Research into human-human communication can be used as a starting point in developing a robust human-robot collaboration system. Previous research with humans has shown that grounding, situational awareness, a common frame of reference and spatial referencing are vital in effective communication. Clearly, there is a growing need for research on human-robot collaboration and models of communication between humans and robotic systems.

Augmented Reality (AR) is a technology for overlaying three-dimensional virtual graphics onto the users view of the real world. It also allows for real time interaction with these virtual graphics, enabling a user to reach into the augmented world and manipulate it directly. Augmented Reality could be used to overlay virtual imagery on a real robot and so display the internal state and intentions of the robot. Thus AR can bridge the divide between human and robotic systems and could enable effective human-robot collaboration.

In this chapter an overview of models of human-human collaboration is provided and how these models could be used to develop a model for human-robot collaboration is investigated. The field of human-robot interaction is reviewed and how it fits into a model

of human-robot collaboration is explored. Augmented Reality is introduced and then the effective use of AR for collaboration is discussed. The potential avenues for creating natural human-robot collaboration through spatial dialog utilizing AR are then investigated. Then the work that has been done in this area is discussed and a holistic architectural design for human-robot collaboration based on Augmented Reality is presented.

2. Communication and Collaboration

In this work, collaboration is defined as "working jointly with others or together especially in an intellectual endeavor". Nass *et al.* (Nass, Steuer et al. 1994) noted that social factors governing human-human interaction equally apply to human-computer interaction. Therefore, before research in human-robot collaboration is described, human-human communication is briefly reviewed and a model for human-human collaboration is presented. This model provides an understanding of the needs of an effective human-robot collaborative system.

2.1 Human-Human Collaboration

There is a vast body of research relating to human–human communication and collaboration. People use speech, gesture, gaze and non-verbal cues to communicate in the clearest possible fashion. In many cases, face-to-face collaboration is also enhanced by, or relies on, real objects or parts of the user's real environment. This section briefly reviews the roles conversational cues and real objects play in face-to-face human-human collaboration. This information is used to derive a set of guidelines for attributes that robots should have to effectively support human-robot collaboration.

A number of researchers have studied the influence of verbal and non-verbal cues on face-to-face communication. Gaze plays an important role by providing visual feedback, regulating the flow of conversation, communicating emotions and relationships, and improving concentration by restriction of visual input (Argyle 1967; Kendon 1967). In addition to gaze, humans use a wide range of non-verbal cues, such as nodding (Watanuki, Sakamoto et al. 1995), gesture (McNeill 1992), and posture (Cassell, Nakano et al. 2001). In many cases, non-verbal cues can only be understood by considering co-occurring speech, such as when using deictic gestures for pointing at something (Kendon 1983). In studying human behaviour it was observed that before conversational partners pointed to an object, they always looked in the direction of the object first (Sidner and Lee 2003). This result suggests that a robot needs to be able to recognize and produce non-verbal communication cues to be an effective collaborative partner.

Real objects and interactions with the real world can also play an important role in collaboration. Minneman and Harrison (Minneman and Harrison 1996) showed that real objects are more than just a source of information, they are also the constituents of collaborative activity, create reference frames for communication and alter the dynamics of interaction. In general, communication and shared cognition are more robust because of the introduction of shared objects. Real world objects can be used to provide multiple representations resulting in increased shared understanding (Clark and Wilkes-Gibbs 1986). A shared visual workspace enhances collaboration as it increases situational awareness (Fussell, Setlock et al. 2003). To support these ideas, a robot should be aware of its surroundings and the interaction of collaborative partners within those surroundings.

Clark and Brennan (Clark and Brennan 1991) provide a communication model to interpret collaboration. Conversation participants attempt to reach shared understanding or common ground. Common ground refers to the set of mutual knowledge, shared beliefs and assumptions that collaborators have. This process of establishing shared understanding, or "grounding", involves communication using a range of modalities including voice, gesture, facial expression and non-verbal body language. Thus, it is evident that for a human-robot team to communicate effectively, all participants will have to be able to easily reach common ground.

2.2 Human-Human Collaboration Model

This chapter investigates human-robot collaboration that is based on a human-human collaboration model, which itself is based on the following three components:

- The communication channels available.
- The communication cues provided by each of these channels.
- The affordances of the technology that affect the transmission of these cues.

There are essentially three types of communication channels available: audio, visual and environmental. Environmental channels consist of interactions with the surrounding world, while audio cues are those that can be heard and visual cues those that can be seen. Depending on the technology medium used communication cues may, or may not, be effectively transmitted between collaborators.

This model can be used to explain collaborative behavior and to predict the impact of technology on collaboration. For example, consider the case of two remote collaborators using text chat to collaborate. In this case, there are no audio or environmental cues. Thus, communication is reduced to one content heavy visual channel: text input. Predictably, this approach has a number of effects on communication: less verbose communication, use of longer phrases, increased time to reach grounding, slower communication and fewer interruptions.

Taking each of the three communication channels in turn, characteristics of an effective human-robot collaboration system can be identified. The robotic system should be able to communicate through speech, recognizing audio input and expressing itself through speech, highlighting a need for an internal model of the communication process. The visual channel should allow the robot to recognize and interpret human non-verbal communication cues and allow the robot to express some non-verbal cues that a human could naturally understand. Finally, through the environmental channel the robot should be able to recognize objects and their manipulation by the human, and be able itself to manipulate objects and understand spatial relationships.

2.3 Summary

This section discussed the general elements of collaboration and then covered how those aspects are seen in human-human collaboration. Human-robot collaboration will require the same fundamental elements, but with different context and avenues. This may well introduce limitations in some channels and increase fidelity in others. The next section, therefore, introduces the robot element and the ways in which robots are used, or might be used, for collaboration with humans.

3. Human-Robot Interaction

3.1 Robots As Tools

The simplest way robots can be used is as tools to aid in the completion of physical tasks. Although there are many examples of robots used in this manner, a few examples are given that highlight human-robot interaction and provide insight into collaboration. For example, to increase the success rate of melon harvesting, a human-robot collaborative system was implemented by (Bechar and Edan 2003). Results indicated that a human operator working with a robotic system with varying levels of autonomy resulted in significantly improved harvesting. Depending on the complexity of the harvesting environment, varying the level of autonomy of the robotic harvester increased positive detection rates by up to 7% from the human operator working alone and by as much as 20% compared to autonomous robot detection alone.

Robots are often used for hazardous tasks. For instance, the placement of radioactive waste in centralized intermediate storage is best completed by robots as opposed to humans (Tsoukalas and Bargiotas 1996). Robotic completion of this task in a totally autonomous fashion is desirable, but not yet achievable due to the dynamic operating conditions. Radiation surveys are initially completed through teleoperation, the learned task is then put into the robots repertoire so the next time the task is to be completed the robot will not need instruction. A dynamic control scheme is needed so that the operator can observe the robot as it completes its task, and when the robot needs help, the operator can intervene and assist with execution. In a similar manner, Ishikawa and Suzuki (Ishikawa and Suzuki 1997) developed a system to patrol a nuclear power plant. Under normal operation the robot is able to work autonomously, however in abnormal situations the human must intervene to make decisions on the robots behalf. In this manner, the system has the ability to cope with unexpected events.

Human-robot teams are used in Urban Search and Rescue (USAR). Robots are teleoperated and used mainly as tools to search for survivors. Studies completed on human-robot interaction for USAR reveal that the lack of situational awareness has a negative effect on performance (Murphy 2004; Yanco, Drury et al. 2004). The use of an overhead camera and automatic mapping techniques improved situational awareness and reduced the number of navigational errors (Scholtz 2002; Scholtz, Antonishek et al. 2005). USAR is conducted in uncontrolled, hazardous environments with adverse ambient conditions that affect the quality of sensor and video data. Studies show that varying the level of robot autonomy and combining data from multiple sensors increases the success rate of identifying survivors (Nourbakhsh, Sycara et al. 2005).

Ohba *et al.* (Ohba, Kawabata et al. 1999) developed a system where multiple operators in different locations control the collision free coordination of several robots in a common work environment. Due to teleoperation time delay and the operators being unaware of each other's intentions, a predictive graphics display was used to avoid collisions. The predictive simulator enlarged the thickness of the robotic arm being controlled by other operators as a buffer to prevent collisions caused by time delay and the remote operators not being aware of each other's intentions. In further work, operator's commands were sent simultaneously to the robot and the graphics predictor to circumvent the time delay (Chong, Kotoku et al. 2001). The predictive simulator used these commands to provide virtual force feedback to the operators and avoid collisions that might otherwise have occurred had the time delay not been addressed. The predictive graphics display is an important means of communicating intentions and increasing situational awareness, thus reducing the number of collisions and damage to the system.

3.2 Guide, Host and Assistant Robots

Nourbakhsh *et al.* (Nourbakhsh, Bobenage et al. 1999) created and installed Sage, an autonomous mobile robot in the Dinosaur Hall at the Carnegie Museum of Natural History. Sage, shown in Fig. 1, interacts with museum visitors through an LCD screen and audio, and uses humor to creatively engage visitors. Sage also exhibits emotions and changes in mood to enhance communication. Sage is completely autonomous and when confronted with trouble will stop and ask for help. Sage shows not only how speech affects communication, but also how the form of speech and non-verbal communication influences how well communication takes place.

Figure 1. Musuem guide robot Sage (Nourbakhsh, Bobenage et al. 1999)

The autonomous interactive robot Robovie, shown in Fig 2, is a humanoid robot that communicates and interacts with humans as a partner and guide (Kanda, Ishiguro et al. 2002). Its use of gestures, speech and eye contact enables the robot to effectively communicate with humans. Results of experiments showed that robot communication behavior induced human communication responses that increased understanding. During interaction with Robovie participants spent more than half of the time focusing on the face of the robot, indicating the importance of gaze in human-robot communication.

Figure 2. Robovie interacting with school children (Kanda, Ishiguro et al. 2002.)

Robots used as guides in museums must interact with people and portray human-like behavior to be accepted. Kuzuoka *et al.* (Kuzuoka, Yamazaki et al. 2004) conducted studies in a science museum to see how humans project when they communicate. The term projection is the capacity to predict or anticipate the unfolding of events. The ability to project was found to be difficult through speech alone because speech does not allow a partner to anticipate what the next action may be in the way that body language (gesture) or gaze can.
Kuzuoka *et al.* (Kuzuoka, Yamazaki et al. 2004) designed a remote instruction robot, Gestureman, to investigate projection. A remote operator controlled Gestureman from a separate room. The operator, through Gestureman's three cameras, had a wider view of the local work space than a person normally would and so could see objects without the robot facing them, as shown in Fig. 3. This dual ecology led to the local human participants being misled as to what the robot was focusing on, and thus not being able to quickly locate what the remote user was trying to identify. The experiment highlighted the importance of gaze direction and situational awareness in effective collaboration.

Figure 3. The Gestureman experiment: Remote operator (left) with wider field of view than robot, identifies object but does not project this intention to local participant (right) (Kuzuoka, Yamazaki et al. 2004)

An assistant robot should exhibit a high degree of autonomy to obtain information about their human partner and surroundings. Iossifidis *et al.* (Iossifidis, Theis et al. 2003) developed CoRa (Cooperative Robot Assistant) that is modeled on the behaviors, senses, and anatomy of humans. CoRa is fixed to a table and interacts through speech, hand gestures, gaze and mechanical interaction, allowing it to obtain information about its surrounding and partner. CoRa's tasks include visual identification of objects presented by its human teacher, recognition of objects, grasping and handing over of objects and performing simple assembly tasks.
Cero (Huttenrauch, Green et al. 2004) is an assistant robot designed to help those with physical disabilities in an office environment. During the iterative development of Cero user studies showed that communicating through speech alone was not effective enough. Users commented that they could not distinguish where the front of the robot was nor could they determine if their commands to the robot were understood correctly. In essence, communication was not being effectively grounded. To overcome this difficulty, a humanoid figure was mounted on the front of the robot that could move its head and arms, see Fig. 4. With the humanoid figure users felt more comfortable communicating with the robot and grounding was easier to achieve (Huttenrauch, Green et al. 2004). These results

highlight the importance of grounding in communication and also the impact that human-like gestures can have on the grounding process.

Figure 4. Cero robot with humanoid figure to enable grounding in communication (Huttenrauch, Green et al. 2004)

Sidner and Lee (Sidner and Lee 2005) show that a hosting robot must not only exhibit conversational gestures, but also must interpret these behaviors from their human partner to engage in collaborative communication. Their robot Mel, a penguin hosting robot shown in Fig. 5, uses vision and speech recognition to engage a human partner in a simple demonstration. Mel points to objects, tracks the gaze direction of the participant to ensure instructions are being followed and looks at observers to acknowledge their presence. Mel actively participates in the conversation and disengages from the conversation when appropriate. Mel is a good example of combining the channels from the communication model to effectively ground a conversation, more explicitly, the use of gesture, gaze direction and speech are used to ensure two-way communication is taking place.

Figure 5. Mel giving a demonstration to a human participant (Sidner and Lee 2005)

3.3 Humanoid Robots

Robonaut is a humanoid robot designed by NASA to be an assistant to astronauts during an extra vehicular activity (EVA) mission. It is anthropomorphic in form allowing an intuitive one to one mapping for remote teleoperation. Interaction with Robonaut occurs in the three roles outlined in the work on human-robot interaction by Scholtz (Scholtz 2003): 1) remote human operator, 2) a monitor and 3) a coworker. Robonaut is shown in Fig. 6. The co-

worker interacts with Robonaut in a direct physical manner and is much like interacting with a human.

Figure 6. Robonaut working with a human (left) and human teleoperating Robonaut (right) (Glassmire, O'Malley et al. 2004)

Experiments have shown that force feedback to the remote human operator results in lower peak forces being used by Robonaut (Glassmire, O'Malley et al. 2004). Force feedback in a teleoperator system improves performance of the operator in terms of reduced completion times, decreased peak forces and torque, as well as decreased cumulative forces. Thus, force feedback serves as a tactile form of non-verbal human-robot communication.

Research into humanoid robots has also concentrated on making robots appear human in their behavior and communication abilities. For example, Breazeal *et al.* (Breazeal, Edsinger et al. 2001) are working with Kismet, a robot that has been endowed with visual perception that is human-like in its physical implementation. Kismet is shown in Fig. 7. Eye movement and gaze direction play an important role in communication aiding the participants in reaching common ground. By following the example of human vision movement and meaning, Kismets' behavior will be understood and Kismet will be more easily accepted socially. Kismet is an example of a robot that can show the non-verbal cues typically present in human-human conversation.

Figure 7. Kismet showing facial expressions present in human communication (Breazeal, Edsinger et al. 2001)

Robots with human social abilities, rich social interaction and natural communication will be able to learn from human counterparts through cooperation and tutelage. Breazeal *et al.* (Breazeal, Brooks et al. 2003; Breazeal 2004) are working towards building socially intelligent cooperative humanoid robots that can work and learn in partnership with people. Robots will need to understand intentions, beliefs, desires and goals of humans to provide relevant assistance and collaboration. To collaborate, robots will also need to be able to infer and reason. The goal is to have robots learn as quickly and easily, as well as in the same manner, as a person. Their robot, Leonardo, is a humanoid designed to express and gesture to people, as well as learn to physically manipulate objects from natural human instruction, as shown in Fig. 8. The approach for Leonardo's learning is to communicate both verbally and non-verbally, use visual deictic references, and express sharing and understanding of ideas with its teacher. This approach is an example of employing the three communication channels in the model used in this chapter for effective communication.

Figure 8. Leonardo activating the middle button upon request (left) and learning the name of the left button (right) (Breazeal, Brooks et al. 2003.)

3.4 Robots in Collaborative Tasks

Inagaki *et al.* (Inagaki, Sugie et al. 1995) proposed that humans and robots can have a common goal and work cooperatively through perception, recognition and intention inference. One partner would be able to infer the intentions of the other from language and behavior during collaborative work. Morita *et al.* (Morita, Shibuya et al. 1998) demonstrated that the communication ability of a robot improves with physical and informational interaction synchronized with dialog. Their robot, Hadaly-2, expresses efficient physical and informational interaction, thus utilizing the environmental channel for collaboration, and is capable of carrying an object to a target position by reacting to visual and audio instruction.

Natural human-robot collaboration requires the robotic system to understand spatial references. Tversky *et al.* (Tversky, Lee et al. 1999) observed that in human-human communication, speakers used the listeners perspective when the listener had a higher cognitive load than the speaker. Tenbrink *et al.* (Tenbrink, Fischer et al. 2002) presented a method to analyze spatial human-robot interaction, in which natural language instructions

were given to a robot via keyboard entry. Results showed that the humans used the robot's perspective for spatial referencing.

To allow a robot to understand different reference systems, Roy *et al.* (Roy, Hsiao et al. 2004) created a system where their robot is capable of interpreting the environment from its perspective or from the perspective of its conversation partner. Using verbal communication, their robot Ripley was able to understand the difference between spatial references such as my left and your left. The results of Tenbrink *et al.* (Tenbrink, Fischer et al. 2002), Tversky *et al.* (Tversky, Lee et al. 1999) and Roy *et al.* (Roy, Hsiao et al. 2004) illustrate the importance of situational awareness and a common frame of reference in spatial communication.

Skubic *et al.* (Skubic, Perzanowski et al. 2002; Skubic, Perzanowski et al. 2004) also conducted a study on human-robotic spatial dialog. A multimodal interface was used, with input from speech, gestures, sensors and personal electronic devices. The robot was able to use dynamic levels of autonomy to reassess its spatial situation in the environment through the use of sensor readings and an evidence grid map. The result was natural human-robot spatial dialog enabling the robot to communicate obstacle locations relative to itself and receive verbal commands to move to or near an object it had detected.

Rani *et al.* (Rani, Sarkar et al. 2004) built a robot that senses the anxiety level of a human and responds appropriately. In dangerous situations, where the robot and human are working in collaboration, the robot will be able to detect the anxiety level of the human and take appropriate actions. To minimize bias or error the emotional state of the human is interpreted by the robot through physiological responses that are generally involuntary and are not dependent upon culture, gender or age.

To obtain natural human-robot collaboration, Horiguchi *et al.* (Horiguchi, Sawaragi et al. 2000) developed a teleoperation system where a human operator and an autonomous robot share their intent through a force feedback system. The human or robot can control the system while maintaining their independence by relaying their intent through the force feedback system. The use of force feedback resulted in reduced execution time and fewer stalls of a teleoperated mobile robot.

Fernandez *et al.* (Fernandez, Balaguer et al. 2001) also introduced an intention recognition system where a robot participating in the transportation of a rigid object detects a force signal measured in the arm gripper. The robot uses this force information, as non-verbal communication, to generate its motion planning to collaborate in the execution of the transportation task. Force feedback used for intention recognition is another way in which humans and robots can communicate non-verbally and work together.

Collaborative control was developed by Fong *et al.* (Fong, Thorpe et al. 2002a; Fong, Thorpe et al. 2002b; Fong, Thorpe et al. 2003) for mobile autonomous robots. The robots work autonomously until they run into a problem they can't solve. At this point, the robots ask the remote operator for assistance, allowing human-robot interaction and autonomy to vary as needed. Performance deteriorates as the number of robots working in collaboration with a single operator increases (Fong, Thorpe et al. 2003). Conversely, robot performance increases with the addition of human skills, perception and cognition, and benefits from human advice and expertise.

In the collaborative control structure used by Fong *et al.* (Fong, Thorpe et al. 2002a; Fong, Thorpe et al. 2002b; Fong, Thorpe et al. 2003) the human and robots engage in dialog, exchange information, ask questions and resolve differences. Thus, the robot has more

freedom in execution and is more likely to find good solutions when it encounters problems. More succinctly, the human is a partner whom the robot can ask questions, obtain assistance from and in essence, collaborate with.
In more recent work, Fong *et al* (Fong, Kunz et al. 2006) note that for humans and robots to work together as peers, the system must provide mechanisms for these peers to communicate effectively. The Human-Robot Interaction Operating System (HRI/OS) introduced enables a team of humans and robots to work together on tasks that are well defined and narrow in scope. The agents are able to use dialog to communicate and the autonomous agents are able to use spatial reasoning to interpret 'left of' type dialog elements. The ambiguities arising from such dialog are resolved through the use of modeling the situation in a simulator.

3.5 Summary

From the research presented, a few points of importance to human-robot collaboration can be identified. Varying the level of autonomy of human-robotic systems allows the strengths of both the robot and the human to be maximized. It also allows the system to optimize the problem solving skills of a human and effectively balance that with the speed and physical dexterity of a robotic system. A robot should be able to learn tasks from its human counterpart and later complete these tasks autonomously with human intervention only when requested by the robot. Adjustable autonomy enables the robotic system to better cope with unexpected events, being able to ask its human team member for help when necessary.
For robots to be effective partners they should interact meaningfully through mutual understanding. Situational awareness and common frames of reference are vital to effective communication and collaboration. Communication cues should be used to help identify the focus of attention, greatly improving performance in collaborative work. Grounding, an essential ingredient of the collaboration model, can be achieved through meaningful interaction and the exchange of dialog.
A robot will be better understood and accepted if its communication behaviour emulates that of humans. The use of humour and emotion can increase the effectiveness of a robot to communicate, just as in humans. A robot should reach a common understanding in communication by employing the same conversational gestures used by humans, such as gaze direction, pointing, hand and face gestures. During human-human conversation, actions are interpreted to help identify and resolve misunderstandings. Robots should also interpret behaviour so their communication comes across as more natural to their human conversation partner. Communication cues, such as the use of humour, emotion, and non-verbal cues, are essential to communication and thus, effective collaboration.

4. Augmented Reality for Human-Robot Collaboration

Augmented Reality (AR) is a technology that facilitates the overlay of computer graphics onto the real world. AR differs from virtual reality (VR) in that it uses graphics to enhance the physical world rather than replacing it entirely, as in a virtual environment. AR enhances rather replaces reality. Azuma *et al.* (Azuma, Baillot et al. 2001) note that AR computer interfaces have three key characteristics:

- They combine real and virtual objects.

- The virtual objects appear registered on the real world.
- The virtual objects can be interacted with in real time.

AR also supports transitional user interfaces along the entire spectrum of Milgram's Reality-Virtuality continuum (Milgram and Kishino 1994), see Fig. 9.

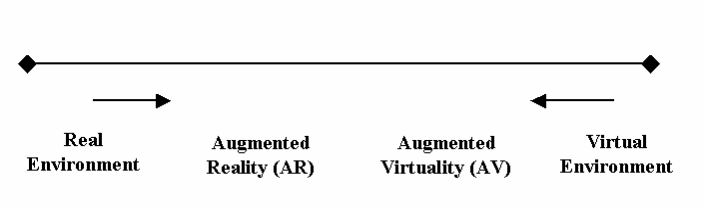


Figure 9. Milgram's reality-virtuality continuum (Milgram and Kishino 1994)

AR provides a 3D world that both the human and robotic system can operate within. This use of a common 3D world enables both the human and robotic system to utilize the same common reference frames. The use of AR will support the use of spatial dialog and deictic gestures, allows for adjustable autonomy by supporting multiple human users, and will allow the robot to visually communicate to its human collaborators its internal state through graphic overlays on the real world view of the human. The use of AR enables a user to experience a tangible user interface, where physical objects are manipulated to affect changes in the shared 3D scene (Billinghurst, Grasset et al. 2005), thus allowing a human to reach into the 3D world of the robotic system and manipulate it in a way the robotic system can understand.

This section first provides examples of AR in human-human collaborative environments, and then discusses the advantages of an AR system for human-robot collaboration. Mobile AR applications are then presented and examples of using AR in human-robot collaboration are discussed. The section concludes by relating the features of collaborative AR interfaces to the communication model for human-robot collaboration presented in section two.

4.1 AR in Collaborative Applications

AR technology can be used to enhance face-to-face collaboration. For example, the Shared Space application effectively combined AR with physical and spatial user interfaces in a face-to-face collaborative environment (Billinghurst, Poupyrev et al. 2000). In this interface users wore a head mounted display (HMD) with a camera mounted on it. The output from the camera was fed into a computer and then back into the HMD so the user saw the real world through the video image, as depicted in Fig. 10.

This set-up is commonly called a video-see-through AR interface. A number of marked cards were placed in the real world with square fiducial patterns on them and a unique symbol in the middle of the pattern. Computer vision techniques were used to identify the unique symbol, calculate the camera position and orientation, and display 3D virtual images aligned with the position of the markers (ARToolKit 2007). Manipulation of the physical markers was used for interaction with the virtual content. The Shared Space application provided the users with rich spatial cues allowing them to interact freely in space with AR content.

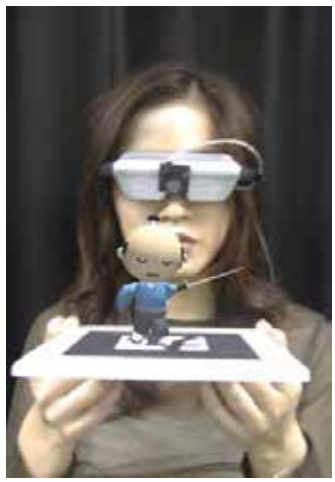


Figure 10. AR with head mounted display and 3D graphic placed on fiducial marker (Billinghurst, Poupyrev et al. 2000)

Through the ability of the ARToolkit software (ARToolKit 2007) to robustly track the physical markers, users were able to interact and exchange markers, thus effectively collaborating in a 3D AR environment. When two corresponding markers were brought together, it would result in an animation being played. For example, when a marker with an AR depiction of a witch was put together with a marker with a broom, the witch would jump on the broom and fly around.

User studies have found that people have no difficulties using the system to play together, displaying collaborative behavior seen in typical face-to-face interactions (Billinghurst, Poupyrev et al. 2000). The Shared Space application supports natural face-to-face communication by allowing multiple users to see each other's facial expressions, gestures and body language, demonstrating that a 3D collaborative environment enhanced with AR content can seamlessly enhance face-to-face communication and allow users to naturally work together.

Another example of the ability of AR to enhance collaboration is the MagicBook, shown in Fig. 11, which allows for a continuous seamless transition from the physical world to augmented and/or virtual reality (Billinghurst, Kato et al. 2001). The MagicBook utilizes a real book that can be read normally, or one can use a hand held display (HHD) to view AR content popping out of the real book pages. The placement of the augmented scene is achieved by the ARToolkit (ARToolKit 2007) computer vision library. When the user is interested in a particular AR scene they can fly into the scene and experience it as an immersive virtual environment by simply flicking a switch on the handheld display. Once immersed in the virtual scene, when a user turns their body in the real world, the virtual viewpoint changes accordingly. The user can also fly around in the virtual scene by pushing a pressure pad in the direction they wish to fly. When the user switches to the immersed virtual world an inertial tracker is used to place the virtual objects in the correct location.

Figure 11. MagicBook with normal view (left), exo-centric view AR (middle), and immersed ego-centric view (right) (Billinghurst, Kato et al. 2001)

The MagicBook also supports multiple simultaneous users who each see the virtual content from their own viewpoint. When the users are immersed in the virtual environment they can experience the scene from either an ego-centric or exo-centric point of view (Billinghurst, Kato et al. 2001). The MagicBook provides an effective environment for collaboration by allowing users to see each other when viewing the AR application, maintaining important visual cues needed for effective collaboration. When immersed in the VR environment, users are represented as virtual avatars and can be seen by other users in the AR or VR scene, thereby maintaining awareness of all users, and thus still providing an environment supportive of effective collaboration.

Prince *et al.* (Prince, Cheok et al. 2002) introduced a 3D live augmented reality conferencing system. Through the use of multiple cameras and an algorithm determining shape from silhouette, they were able to superimpose a live 3D image of a remote collaborator onto a fiducial marker, creating the sense that the live remote collaborator was in the workspace of the local user. Fig. 12 shows the live collaborator displayed on a fiducial marker. The shape from silhouette algorithm works by each of 15 cameras identifying a pixel as belonging to the foreground or background, isolation of the foreground information produces a 3D image that can be viewed from any angle by the local user.

Communication behaviors affect performance in collaborative work. Kiyokawa *et al.* (Kiyokawa, Billinghurst et al. 2002) experimented with how diminished visual cues of co-located users in an AR collaborative task influenced task performance. Performance was best when collaborative partners were able to see each other in real time. The worst case occurred in an immersive virtual reality environment where the participants could only see virtual images of their partners.

In a second experiment Kiyokawa *et al.* (Kiyokawa, Billinghurst et al. 2002) modified the location of the task space, as shown in Fig. 13. Participants expressed more natural communication when the task space was between them; however, the orientation of the task space was significant. The task space between the participants meant that one person had a reversed view from the other. Results showed that participants preferred the task space to be on a wall to one side of them, where they could both view the workspace from the same perspective. This research highlights the importance of the task space location, the need for a common reference frame and the ability to see the visual cues displayed by a collaborative partner.

Figure 12. Remote collaborator as seen on AR fiducial marker (Prince, Cheok et al. 2002)

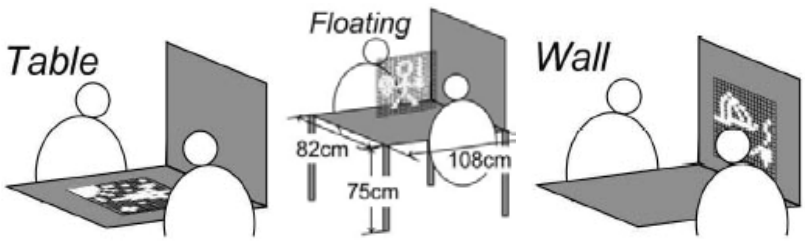


Figure 13. Different locations of task space in Kiyokawa *et al* second experiment (Kiyokawa, Billinghurst et al. 2002

These results show that AR can enhance face-to-face collaboration in several ways. First, collaboration is enhanced through AR by allowing the use of physical tangible objects for ubiquitous computer interaction. Thus making the collaborative environment natural and effective by allowing participants to use objects for interaction that they would normally use in a collaborative effort. AR provides rich spatial cues permitting users to interact freely in space, supporting the use of natural spatial dialog. Collaboration is also enhanced by the use of AR since facial expressions, gestures and body language are effectively transmitted.
In an AR environment multiple users can view the same virtual content from their own perspective, either from an ego- or exo-centric viewpoint. AR also allows users to see each other while viewing the virtual content enhancing spatial awareness and the workspace in an AR environment can be positioned to enhance collaboration. For human-robot collaboration, AR will increase situational awareness by transmitting necessary spatial cues through the three channels of the communication model presented in this chapter.

4.2 Mobile AR

For true human-robot collaboration it is optimal for the human to not be constrained to a desktop environment. A human collaborator should be able to move around in the environment the robotic system is operating in. Thus, mobilily is an important ingredient for human-robot collaboration. For example, if an astronaut is going to collaborate with an

autonomous robot on a planet surface, a mobile AR system could be used that operates inside the astronauts suit and projects virtual imagery on the suit visor. This approach would allow the astronaut to roam freely on the planet surface, while still maintaining close collaboration with the autonomous robot.

Wearable computers provide a good platform for mobile AR. Studies from Billinghurst *et al.* (Billinghurst, Weghorst et al. 1997) showed that test subjects preferred working in an environment where they could see each other and the real world. When participants used wearable computers they performed best and communicated almost as if communicating in a face-to-face setting (Billinghurst, Weghorst et al. 1997). Wearable computing provides a seamless transition between the real and virtual worlds in a mobile environment.

Cheok *et al.* (Cheok, Weihua et al. 2002) utilized shape from silhouette live 3D imagery (Prince, Cheok et al. 2002) and wearable computers to create an interactive theatre experience, as depicted in Fig. 14. Participants collaborate in both an indoor and outdoor setting. Users seamlessly transition between the real world, augmented and virtual reality, allowing multiple users to collaborate and experience the theatre interactively with each other and 3D images of live actors.

Reitmayr and Schmalstieg (Reitmayr and Schmalstieg 2004) implemented a mobile AR tour guide system that allows multiple tourists to collaborate while they explore a part of the city of Vienna. Their system directs the user to a target location and displays location specific information that can be selected to provide detailed information. When a desired location is selected, the system computes the shortest path, and displays this path to the user as cylinders connected by arrows, as shown in Fig. 15.

The Human Pacman game (Cheok, Fong et al. 2003) is an outdoor mobile AR application that supports collaboration. The system allows for mobile AR users to play together, as well as get help from stationary observers. Human Pacman, see Fig. 16, supports the use of tangible and virtual objects as interfaces for the AR game, as well as allowing real world physical interaction between players. Players are able to seamlessly transition between a first person augmented reality world and an immersive virtual world. The use of AR allows the virtual Pacman world to be superimposed over the real world setting. AR enhances collaboration between players by allowing them to exchange virtual content as they are moving through the AR outdoor world.

Figure 14. Mobile AR setup (left) and interactive theatre experience (right) (Cheok, Weihua et al. 2002)

Figure 15. Mobile AR tour guide system (left) and AR display of path to follow(right) (Reitmayr and Schmalsteig 2004)

Figure 16. AR Human Pacman Game (Cheok, Fong 2003)

To date there has been little work on the use of mobile AR interfaces for human-robot collaboration; however, several lessons can be learnt from other wearable AR systems. The majority of mobile AR applications are used in an outdoor setting, where the augmented objects are developed and their global location recorded before the application is used. Two important issues arise in mobile AR; data management, and the correct registration of the outdoor augmented objects. With respect to data management, it is important to develop a system where enough information is stored on the wearable computer for the immediate needs of the user, but also allows access to new information needed as the user moves around (Julier, Baillot et al. 2002). Data management should also allow for the user to view as much information as required, but at the same time not overload the user with so much information that it hinders performance. Current AR systems typically use GPS tracking for registration of augmented information for general location coordinates, then use inertial trackers, magnetic trackers or optical fiducial markers for more precise AR tracking. Another important item to design into a mobile AR system is the ability to continue operation in case communication with the remote server or tracking system is temporarily lost.

4.3 First Steps Using AR in Human-Robot Collaboration

Milgram *et al* (Milgram, Zhai et al. 1993) highlighted the need for combining the attributes that humans are good at with those that robots are good at to produce an optimized human-robot team. For example, humans are good at approximate spatial referencing, such as using 'here' and 'there', whereas robotic systems need highly accurate discrete information. Milgram *et al* pointed out the need for HRI systems that can transfer the interaction mechanisms that are considered natural for human communication to the precision required for machine information. Their approach was to use augmented overlays in a fixed work environment to enable the human 'director' to use spatial referencing to interactively plan and optimize the path of a robotic manipulator arm, see Fig 17.

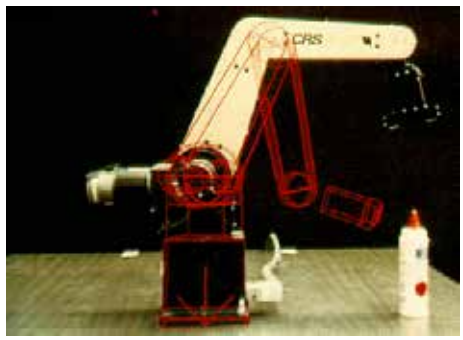


Figure 17. AR overlay in fixed environment for interactive path planning (Milgram, Zhai et al. 1993)

Giesler *et al.* (Giesler, Steinhaus et al. 2004) are working on a system that allows a robot to interactively create a 3D model of an object on-the-fly. In this application, a laser scanner is used to read in an unknown 3D object. The information from the laser scan is overlaid through AR onto the video of the real world. The user interactively creates a boundary box around the appropriate portion of the laser scan by using voice commands and an AR magic wand. The wand is made of fiducial markers and uses the ARToolkit for tracking. Using a combination of the laser scan and video image, a 3D model of a previously unknown object can be created.

In other work Giesler *et al.* (Giesler, Salb et al. 2004) are implementing an AR system that creates a path for a mobile robot to follow using voice commands and the same magic wand from their work above. Fiducial markers are placed on the floor and used to calibrate the tracking coordinate system. A path is created node by node, by pointing the wand at the floor and giving voice commands for the meaning of a particular node. Map nodes can be interactively moved or deleted. The robot moves from node to node using its autonomous collision detection capabilities. As goal nodes are reached, the node depicted in the AR system changes colour to keep the user informed of the robots progress. The robot will retrace steps if an obstruction is encountered and create a new plan to arrive at the goal destination, as shown in Fig. 18.

Although Giesler *et al* (Giesler, Salb et al. 2004) did not mention a user evaluation, they did comment that the interface was intuitive to use. Results from their work show that AR is an

excellent application to visualize planned trajectories and inform the user of the robots progress and intention.

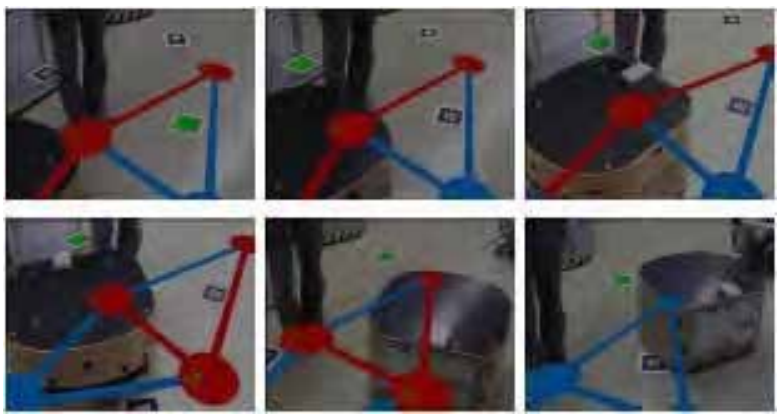


Figure 18. Robot follows AR path nodes, redirects when obstacle in the way (Giesler, Salb et al. 2004)

Bowen *et al* (Bowen, Maida et al. 2004) and Maida *et al* (Maida, Bowen et al. 2006) showed through user studies that the use of AR resulted in significant improvements in robotic control performance. Simlarly, Drury *et al* (Drury, Richer et al. 2006) found that for operators of Unmanned Aerial Vehicles (UAVs) augmenting real-time video with pre-loaded map terrain data resulted in a statistical difference in comprehension of 3D spatial relationships compared to 2D video alone. The AR interface provided better situational awareness of the activities of the UAV. AR has also been used to display robot sensor information on the view of the real world (Collett and MacDonald 2006).

4.4 Summary

Augmented Reality is an ideal platform for human-robot collaboration as it provides many of the features required for robust communication and collaboration. Benefits of the use of AR include:

- The ability for a human to share a remote (ego-centric) view with a robot, thus enabling the ability for the human-robot team to reach common ground.
- The ability for the human to have a world view (exo-centric) of the collaborative workspace, thus affording spatial awareness.
- The use of deictic gestures and spatial dialog by allowing all partners to refer to and interact with the graphic 3D overlaid imagery, supporting the use of natural spatial dialog.
- Collaboration of multiple users, multiple humans can effectively collaborate with multiple robotic systems.
- Seamless transition from the real world to an immersive data space that aids in the grounding process and increases situational awareness.

- Display of visual cues as to what the robots intentions are and it's internal state, greatly enhancing the grounding process and increasing situational awareness.
- Providing the spatial cues necessary for both local and remote collaboration.

A human-robot collaboration system would benefit greatly from the use of AR. AR would enhance the grounding process, provide for increased situational awareness, enable the use of natural spatial dialog, allow for multiple collaborative partners and enable both local and remote collaboration. The result would be a system that allows natural and effective communication and thus collaboration.

5. Research Directions in Human-Robot Collaboration

Given this review of the general state of human-robot collaboration, and the presentation and review of using AR to enhance this type of collaboration, the question is: what are promising future research directions? Two important concepts must be kept in mind when designing an effective human-robot collaboration system. One, the robotic system must be able to provide feedback as to its understanding of the situation and its actions (Scholtz 2002). Two, an effective human-robot system must provide mechanisms to enable the human and the robotic system to communicate effectively (Fong, Kunz et al. 2006). In this section, each of the three communication channels in the model presented is explored, and potential avenues to make the model of human-robot collaboration become a reality are discussed.

5.1 The Audio Channel

There are numerous systems available for automated speech recognition (ASR) and text to speech (TTS) synthesis. A robust dialog management system will need to be developed that is capable of taking human input from the ASR system and converting it into robot commands. The dialog management system will also need to be able to take input from the robot control system and convert this information into suitable text strings for the TTS system to synthesize into human understandable audio output. The dialog manager will thus need to support the ongoing discussion between the human and the robot. The dialog manager will also need to enable a robot to express its intentions and its understanding of the current situation, including responding with alternative approaches to those proposed by the human collaborators or alerting the human team members when a proposed plan is not feasible. This type of clarification (Krujiff, Zender et al. 2006) will require the robotic system to understand the speech, interpret the speech in terms of its surroundings and goals, and express itself through speech. An internal model of the communication process will need to be developed.

The use of humour and emotion will enable the robotic agents to communicate in a more natural and effective manner, and therefore should be incorporated into the dialog management system. An example of the effectiveness of this type of communication can be seen in Rea, a computer generated human-like real estate agent (Cassell, Bickmore et al. 1999). Rea is capable of multi-modal input and output using verbal and non-verbal communication cues to actively participate in a conversation. Audio can also be spatialized, in essence, placing sound in the virtual world from where it originates in the real world. Spatially locating sound will increase situational awareness and thus provide a means to communicate effectively and naturally.

5.2 The Environmental Channel

A robot will need to understand the use of objects by its human counterpart, such as using an object to point or making a gesture. AR can support this type of interaction by enabling the human to point to a virtual object that both the robot and human refer to and use natural dialog such as "go to *this* point", thereby reaching common ground and maintaining situational awareness. In a similar manner the robot would be able to express its intentions and beliefs by showing through the 3D overlays what its internal state, plans and understanding of the situation are. Thus using the shared AR environment as an effective spatial communication tool. Referencing a shared 3D environment will support the use of common and shared frames of references, thus affording the ability to effectively communicate in a truly spatial manner. As an example, if a robot did not fully understand a verbal command, it would be able to make use of the shared 3D environment to clearly portray to its collaborators what was not understood, what further information is needed and what the autonomous agent believes could be the correct action to take.

Real physical objects can be used to interact with an AR application. For human-robot communication this translates into a more intuitive user interface, allowing the use of real world objects to communicate with a robot as opposed to the use of a mouse or keyboard. The use of real world objects is especially important for mobile applications where the user will not be able to use typical computer interface devices.

5.3 The Visual Channel

With the limited speech ability of robotic systems, visual cues will also provide a means of grounding communication. AR, with its ability to provide ego- and exo-centric views and to seamlessly transition from reality to virtuality, can provide robotic systems with a robust manner in which to ground communication and allow human collaborative partners to understand the intention of the robotic system. AR can also transmit spatial awareness though the ability to provide rich spatial cues, ego- and exo-centric points of view, and also by seamlessly transitioning from the real world to an immersive VR world. An AR system could, therefore, be developed to allow for bi-directional transmission of gaze direction, gestures, facial expressions and body pose. The result would be an increased level of communication and more effective collaboration.

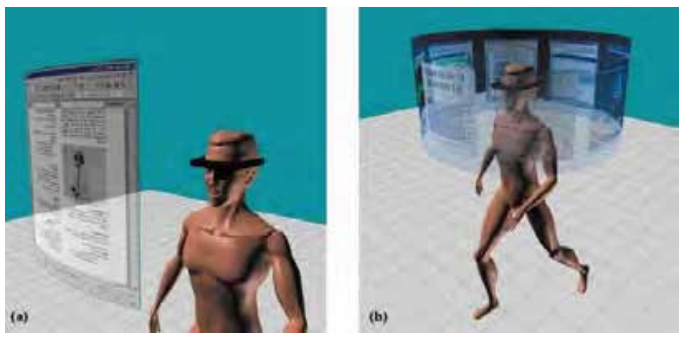


Figure 19. Head stabilised AR information display (left) and body stabilised (right) (Billinghurst, Bowskill et al. 1998)

AR is an optimal method of displaying information for the user. Billinghurst *et al.* (Billinghurst, Bowskill et al. 1998) showed through user tests that spatial displays in a wearable computing environment were more intuitive and resulted in significantly increased performance. Fig. 19 shows spatial information displayed in a head stabilised and body stabilised fashion. Using AR to display information, such as robot state, progress and even intent, will result in increased understanding, grounding and, therefore, enhanced collaboration.

6. Design Guidelines for Human-Robot Collaboration

Given the general overview of the state of human-robot interaction and collaboration, it is possible to identify guidelines for a robust human-robot collaboration system. Humans and robots have different strengths, and to create an effective human-robot collaborative team, the strengths of each member should be capitalized on. Humans are good at using vague spatial references. For example, most people would point out where an object is by using some sort of deictic reference, like "it's over there". Unfortunately robotic systems are not designed to understand these types of spatial references. Therefore, for a human-robot collaboration system to be natural to a human it will have to be able to understand vague spatial references. In the same light, for a collaboration system to be effective for robotic systems it will have to translate vague spatial references into exact spatial coordinates that a robotic system needs to operate.

Humans are good at dealing with unexpected and changing situations. Robotic systems for the most part are not. Robots are good at physical repetitive tasks that can tire human team members. Robots can also be sent into dangerous environments that humans cannot work in. Therefore, for a human-robot team to collaborate at the most effective level the system should allow for varying levels of autonomy enabling robots to do what they do best and humans to do what they do best. By varying the level of autonomy the system would enable the strengths of both the robot and the human to be maximized. Varying levels of autonomy would allow the system to optimize the problem solving skills of a human and effectively balance that with the speed and physical dexterity of a robotic system. A robot should be able to learn tasks from its human counterpart and later complete these tasks autonomously with human intervention only when requested by the robot. Adjustable autonomy enables the robotic system to better cope with unexpected events, being able to ask its human team member for help when necessary.

For robots to be effective partners they should interact meaningfully through mutual understanding. A robotic system will be better understood and accepted if its communication behaviour emulates that of humans. Communication cues should be used to help identify the focus of attention, greatly improving performance in collaborative work. Grounding, an essential ingredient of communication, can be achieved through meaningful interaction and the exchange of dialog. The use of humour and emotion can increase the effectiveness of a robot to communicate, just as in humans. Robots should also interpret behaviour so their communication comes across as more natural to their human conversation partner. Communication cues, such as the use of humour, emotion, and non-verbal cues, are essential to communication and thus, effective collaboration. Therefore, it is evident that for a human-robot team to communicate effectively, all participants will have to feel confident that common ground is easily reached.

Studies have shown that the lack of situational awareness has detrimental effects on a human-robot team. Therefore, a human-robot collaboration system should provide the means for both human and robotic team members to maintain situational awareness. Reference frames are fundamental if a human team member is going to use spatial dialog when communicating with robotic systems. Consequently, a robust collaborative system should effectively allow for human team members to use reference frames at will and translate this information into a usable format for the robotic system. The collaborative system should enable a robot to be aware of its surroundings and the interaction of collaborative partners within those surroundings. If a human-collaboration system entails these design parameters, the result should be an effective natural collaboration between humans and robotic systems.

7. Architectural Design

Employing the lessons learned from this literature review, an architectural design has been developed for Human-Robot Collaboration (HRC). A multimodal approach is envisioned that combines speech and gesture through the use of AR that will allow humans to naturally communicate with robotic systems. Through this architecture the robotic system will receive the discrete information it needs to operate while allowing human team members to communicate in a natural and effective manner by referencing objects, positions, and intentions through natural gesture and speech. The human and the robotic system will each maintain situational awareness by referencing the same shared 3D visuals of the workspace in the AR environment.

The architectural design is shown in Fig. 20. The speech-processing module will recognize human speech and parse this speech into the appropriate dialog components. When a defined dialog goal is achieved through speech recognition, the required information will be sent to the Multimodal Communication Processor (MCP). The speech-processing module will also take information from the MCP and the robotic system and synthesize this speech for effective dialog with human team members. The speech processing will take place using the spoken dialog system Ariadne (Ariadne 2006). Ariadne was chosen for its capability for rapid dialog creation (Denecke 2002).

Gesture processing will enable a human to use deictic referencing and normal gestures to communicate effectively with a robotic system. It is imperative that the system be able to translate the generic references that humans use, such as pointing into 3D space and saying "go here", into the discrete information a robotic system needs to operate. The gesture-processing module will recognize gestures used by a human and pass this information to the MCP. The MCP will combine the speech from the speech processing module, the gesture information from the gesture-processing module and use the Human-Robot Collaboration Augmented Reality Environment (HRC-ARE) to effectively enable the defining of ambiguous deictic references such as here, there, this and that. The disambiguation of the deictic references will be accomplished in the AR environment which is a 3D virtual replication of the robot's world, allowing visual translation and definition of such deictic references. The human will be able to use a real world paddle to reach into and interact with this 3D virtual world. This tangible interaction, using a real world paddle to interact with the virtual 3D content, is a key feature of AR that makes it an ideal platform for HRC. The ARToolKit will be used for tracking in the AR environment.

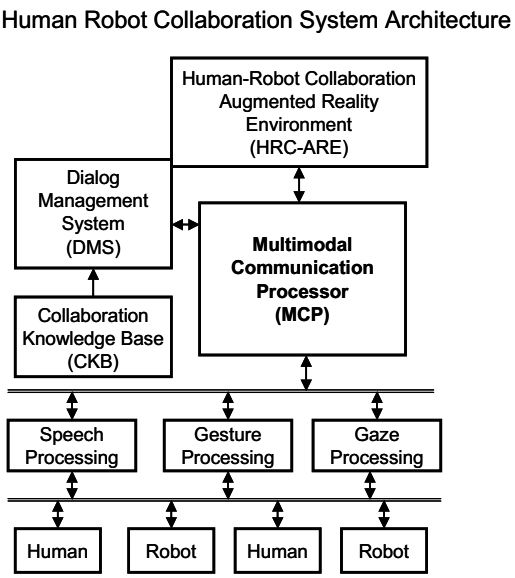


Figure 20. The Human-Robot Collaboration system architecture

The gaze-processing module will interpret the users gaze through the use of a head mounted display. These individual displays will enable each human team member to view the HRC-ARE from his or her own perspective. This personal viewing of the workspace will result in increased situational awareness as each team member will view the work environment from their own perspective and will be able to change their perspective simply by moving around the 3D virtual environment as they would a real world object, or they could move the 3D virtual world and maintain their position by moving the real world fiducial marker that the 3D world is "attached" to. Not only will human team members be able to maintain their perspective of the robotic system's work environment, but they will also be able to smoothly switch to the robot's view of the work environment. This ability to smoothly switch between an exo-centric (God's eye) view of the work environment to an ego-centric (robotic system's) view of the work environment is another feature of AR that makes it ideal for HRC and enables the human to quickly and effectively reach common ground and maintain situational awareness with the robotic system.

The Dialog Management System (DMS) will be aware of the communication that needs to take place for the human and robot to collaboratively complete a task. The MCP will take information from the speech, gesture and gaze processing modules along with information generated from the HRC-ARE and supply it to the DMS. The DMS will be responsible for combining this information and comparing it to the information stored in the Collaboration Knowledge Base (CKB). The CKB will contain information pertaining to what is needed to complete the desired tasks that the human-robot team wishes to complete. The DMS will then respond through the MCP to either human team members or the robotic system,

whichever is appropriate, facilitating dialog and tracking when a command or request is complete.
The MCP will be responsible for receiving information from the other modules in the system and sending information to the appropriate modules. The MCP will thus be responsible for combining multimodal input, registering this input into something the system can understand and then sending the required information to other system modules for action. The result of this system design is that a human will be able to use natural speech and gestures to collaborate with robotic systems.

8. Conclusions

This chapter began by establishing a need for human-robot collaboration. Human-human communication was discussed; a model for human-human collaboration created and this model was used as a reference model for human-robot collaboration. The state of human-robot interaction was reviewed and how this interaction fits into the model of human-robot collaboration was explored. Augmented Reality technology was introduced, reviewed and how AR could be used to enhance human-robot collaboration was explored. Research directions were discussed and then design guidelines were outlined. Finally, a holistic architecture using AR as an enabling device for human-robot collaboration was presented.
The model developed for human communication is based on three components; the communication channels available, the communication cues provided by each of these channels, and the technology that affects the transmission of these cues. There are three channels for communication: visual, audio and environmental. Depending on the transmission medium used, communication cues may not be effectively transmitted. Applying this model to human-robot collaboration, the characteristics of an effective human-robot collaborative system can be analyzed. An effective system should strive to allow communication cues to be transferred in all three channels. Therefore, the robot must be able to understand and exhibit audio, visual and environmental communication cues.
Effective human-robot collaborative systems should make use of varying levels of autonomy. As a result, the system would better capitalize on the strengths of both the human and robot. More explicitly, the system would capitalize on the problem solving skills of a human and the speed and dexterity of a robot. Thus, a robot would be able to work autonomously, while retaining the ability to request assistance when guidance is needed or warranted.
In terms of communication, a robot will be better understood and accepted if its communication behaviour more explicitly emulates that of a human. Common understanding should be reached by using the same conversational gestures used by humans, such as gaze, pointing, and hand and face gestures. Robots should also be able to interpret and display such behaviours so that their communication appears natural to their human conversational partner.
Finally, Augmented Reality has many benefits that will help create a more ideal environment for human-robot collaboration and advance the capability of the communication channels discussed. AR technology allows the human to share an ego-centric view with a robot, thus enabling the human and robot to ground their communication and intentions. AR also allows for an exo-centric view of the collaborative workspace affording spatial awareness. Multiple collaborators can be supported by an AR system; so multiple humans could collaborate with multiple robotic systems. Human-robot

collaborative systems can, therefore, significantly benefit from AR technology because it conveys visual cues that enhance communication and grounding, enabling the human to have a better understanding of what the robot is doing and its intentions. A multimodal approach in developing a human-robot collaborative system would be the most effective, combining speech (spatial dialog), gesture and a shared reference of the work environment, through the use of AR. As a result, the collaboration will be more natural and effective.

9. References

Argyle, M. (1967). The Psychology of Interpersonal Behavior, London, Penguin Books

Ariadne (2006). http://www.opendialog.org/,

ARToolKit (2007). http://www.hitl.washington.edu/artoolkit/, referenced May 2007,

Azuma, R., Y. Baillot, et al. (2001). Recent advances in augmented reality, *IEEE Computer Graphics and Applications*, 21, (6), 34-47

Bechar, A. and Y. Edan (2003). Human-robot collaboration for improved target recognition of agricultural robots, *Industrial Robot*, 30, (5), 432-436

Billinghurst, M., J. Bowskill, et al. (1998). Spatial information displays on a wearable computer, *IEEE Computer Graphics and Applications*, 18, (6), 24-31

Billinghurst, M., R. Grasset, et al. (2005). Designing Augmented Reality Interfaces, *Computer Graphics SIGGRAPH Quarterly, 39(1), 17-22 Feb*,

Billinghurst, M., H. Kato, et al. (2001). The MagicBook: A transitional AR interface, *Computers and Graphics (Pergamon)*, 25, (5), 745-753

Billinghurst, M., I. Poupyrev, et al. (2000). Mixing realities in Shared Space: An augmented reality interface for collaborative computing, *2000 IEEE International Conference on Multimedia and Expo (ICME 2000), Jul 30-Aug 2*, New York, NY

Billinghurst, M., S. Weghorst, et al. (1997). Wearable computers for three dimensional CSCW, *Proceedings of the 1997 1st International Symposium on Wearable Computers, Oct 13-14*, Cambridge, MA, USA, IEEE Comp Soc, Los Alamitos, CA, USA

Bowen, C., J. Maida, et al. (2004). Utilization of the Space Vision System as an Augmented Reality System for Mission Operations, *Proceedings of AIAA Habitation Conference, Houston TX*,

Breazeal, C. (2004). Social interactions in HRI: The robot view, *IEEE Transactions on Systems, Man and Cybernetics* Part C: Applications and Reviews Human-Robot Interactions, 34, (2), 181-186

Breazeal, C., A. Brooks, et al. (2003). Humanoid Robots as Cooperative Partners for People, MIT Media Lab, Robotic Life Group, Submitted for review to *International Journal of Humanoid Robots* December 15,

Breazeal, C., A. Edsinger, et al. (2001). Active vision for sociable robots, *IEEE Transactions on Systems, Man, and Cybernetics Part A:Systems and Humans*, 31, (5), 443-453

Cassell, J., T. Bickmore, et al. (1999). Embodiment in conversational interfaces: Rea, Conference on Human Factors in Computing Systems - Proceedings *Proceedings of the CHI 99 Conference: CHI is the Limit - Human Factors in Computing Systems*, May 15-May 20, 520-527

Cassell, J., Y. Nakano, et al. (2001). Non-Verbal Cues for Discourse Structure, *Association for Computational Linguistics Annual Conference (ACL)*

Cheok, A. D., S. W. Fong, et al. (2003). Human Pacman: A Mobile Entertainment System with Ubiquitous Computing and Tangible Interaction over a Wide Outdoor Area, *Mobile HCI*, 209-223

Cheok, A. D., W. Weihua, et al. (2002). Interactive theatre experience in embodied + wearable mixed reality space, *Proceedings. International Symposium on Mixed and Augmented Reality, ISMAR*

Chong, N. Y., T. Kotoku, et al. (2001). Exploring interactive simulator in collaborative multi-site teleoperation, *10th IEEE International Workshop on Robot and Human Communication, Sep 18-21*, Bordeaux-Paris, Institute of Electrical and Electronics Engineers Inc.

Clark, H. H. and S. E. Brennan (1991). Grounding in Communication, *Perspectives on Socially Shared Cognition*, L. Resnick, Levine J., Teasley, S., Washington D.C., American Psychological Association**:** 127 - 149

Clark, H. H. and D. Wilkes-Gibbs (1986). Referring as a collaborative process, *Cognition*, 22, (1), 1-39

COGNIRON (2007). http://www.cogniron.org/InShort.php, referenced May 2007,

Collett, T. H. J. and B. A. MacDonald (2006). Developer Oriented Visualisation of a Robot Program, *Proceedings 2006 ACM Conference on Human-Robot Interaction, March 2-4*, 49-56

Denecke, M. (2002). Rapid Prototyping for Spoken Dialogue Systems, *Proceedings of 19th International Conference on Computational Linguistics*, 1, 1-7

Drury, J., J. Richer, et al. (2006). Comparing Situation Awareness for Two Unmanned Aerial Vehicle Human Interface Approaches, *Proceedings IEEE International Workshop on Safety, Security and Rescue Robotics (SSRR). Gainsburg, MD, USA August*,

Fernandez, V., C. Balaguer, et al. (2001). Active human-mobile manipulator cooperation through intention recognition, *2001 IEEE International Conference on Robotics and Automation, May 21-26*, Seoul, Institute of Electrical and Electronics Engineers Inc.

Fong, T., C. Kunz, et al. (2006). The Human-Robot Interaction Operating System, *Proceedings of 2006 ACM Conference on Human-Robot Interaction, March 2-4*, 41-48

Fong, T. and I. R. Nourbakhsh (2005). Interaction challenges in human-robot space exploration, *Interactions*, 12, (2), 42-45

Fong, T., C. Thorpe, et al. (2002a). Robot As Partner: Vehicle Teleoperation With Collaborative Control, *Multi-Robot Systems: From Swarms to Intelligent Automata, 01 June*,

Fong, T., C. Thorpe, et al. (2002b). Robot, asker of questions, *IROS 2002, Sep 30*, Lausanne, Switzerland, Elsevier Science B.V.

Fong, T., C. Thorpe, et al. (2003). Multi-robot remote driving with collaborative control, *IEEE Transactions on Industrial Electronics*, 50, (4), 699-704

Fussell, S. R., L. D. Setlock, et al. (2003). Effects of head-mounted and scene-oriented video systems on remote collaboration on physical tasks, *The CHI 2003 New Horizons Conference Proceedings: Conference on Human Factors in Computing Systems, Apr 5-10*, Ft. Lauderdale, FL, United States, Association for Computing Machinery

Giesler, B., T. Salb, et al. (2004). Using augmented reality to interact with an autonomous mobile platform, *Proceedings- 2004 IEEE International Conference on Robotics and Automation, Apr 26-May 1*, New Orleans, LA, United States, Institute of Electrical and Electronics Engineers Inc., Piscataway, United States

Giesler, B., P. Steinhaus, et al. (2004). Sharing skills: Using augmented reality for human-Robot collaboration, *Stereoscopic Displays and Virtual Reality Systems XI, Jan 19-21*, San Jose, CA, United States, International Society for Optical Engineering, Bellingham, WA 98227-0010, United States

Glassmire, J., M. O'Malley, et al. (2004). Cooperative manipulation between humans and teleoperated agents, *Proceedings - 12th International Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems*, HAPTICS 2004, Mar 27-28, Chicago, IL, United States, IEEE Computer Society, Los Alamitos;Massey University, Palmerston, United States;New Zealand

Hoffmann, G. and C. Breazeal (2004). Robots that Work in Collaboration with People, *AAAI Fall Symposium on the Intersection of Cognitive Science and Robotics*, Washington, D.C.

Honda (2007). http://world.honda.com/ASIMO/, accessed May 2007

Horiguchi, Y., T. Sawaragi, et al. (2000). Naturalistic human-robot collaboration based upon mixed-initiative interactions in teleoperating environment, *2000 IEEE International Conference on Systems, Man and Cybernetics, Oct 8-Oct 11*, Nashville, TN, USA, Institute of Electrical and Electronics Engineers Inc., Piscataway, NJ, USA

HRI2006 (2006). http://www.hri2006.org/, referenced May 2007,

Huttenrauch, H., A. Green, et al. (2004). Involving users in the design of a mobile office robot, *IEEE Transactions on Systems, Man and Cybernetics, Part C*, 34, (2), 113-124

Inagaki, Y., H. Sugie, et al. (1995). Behavior-based intention inference for intelligent robots cooperating with human, *Proceedings of the 1995 IEEE International Conference on Fuzzy Systems. Part 3 (of 5), Mar 20-24*, Yokohama, Jpn, IEEE, Piscataway, NJ, USA

Iossifidis, I., C. Theis, et al. (2003). Anthropomorphism as a pervasive design concept for a robotic assistant, *2003 IEEE/RSJ International Conference on Intelligent Robots and Systems, Oct 27-31*, Las Vegas, NV, United States, Institute of Electrical and Electronics Engineers Inc.

Ishikawa, N. and K. Suzuki (1997). Development of a human and robot collaborative system for inspecting patrol of nuclear power plants, *Proceedings of the 1997 6th IEEE International Workshop on Robot and Human Communication, RO-MAN'97, Sep 29-Oct 1*, Sendai, Jpn, IEEE, Piscataway, NJ, USA

Julier, S., Y. Baillot, et al. (2002). Information filtering for mobile augmented reality, *IEEE Computer Graphics and Applications*, 22, (5), 12-15

Kanda, T., H. Ishiguro, et al. (2002). Development and evaluation of an interactive humanoid robot "Robovie", *2002 IEEE International Conference on Robotics and Automation, May 11-15*, Washington, DC, United States, Institute of Electrical and Electronics Engineers Inc.

Kendon, A. (1967). Some Functions of Gaze Direction in Social Interaction, *Acta Psychologica*, 32, 1-25

Kendon, A. (1983). Gesture and Speech: How They Interact, *Nonverbal Interaction*. J. Wiemann, R. Harrison (Eds). Beverly Hills, Sage Publications, 13-46

Kiyokawa, K., M. Billinghurst, et al. (2002). Communication behaviors of co-located users in collaborative AR interfaces, *International Symposium on Mixed and Augmented Reality, ISMAR*

Krujiff, G.-J. M., H. Zender, et al. (2006). Clarification Dialogues in Human-Augmented Mapping, *Proceedings of 2006 ACM Conference on Human-Robot Interaction, March 2-4*, 282-289

Kuzuoka, H., K. Yamazaki, et al. (2004). Dual ecologies of robot as communication media: Thoughts on coordinating orientations and projectability, *2004 Conference on Human Factors in Computing Systems - Proceedings, CHI 2004, Apr 24-29*, Vienna, Austria, Association for Computing Machinery, New York, NY 10036-5701, United States

Maida, J., C. Bowen, et al. (2006). Enhanced Lighting Techniques and Augmented Reality to Improve Human Task Performance, *NASA Tech Paper TP-2006-213724 July*,

McNeill, D. (1992). *Hand and Mind: What Gestures Reveal about Thought,* The University of Chicago Press. Chicago

Milgram, P. and F. Kishino (1994). Taxonomy of mixed reality visual displays, *IEICE Transactions on Information and Systems*, E77-D, (12), 1321-1329

Milgram, P., S. Zhai, et al. (1993). Applications of Augmented Reality for Human-Robot Communication, *In Proceedings of IROS 93: International Conference on Intelligent Robots and Systems*, Yokohama, Japan

Minneman, S. and S. Harrison (1996). A Bike in Hand: A Study of 3D Objects in Design, *Analyzing Design Activity*, N. Cross, H. Christiaans and K. Dorst, Chichester, J. Wiley

Morita, T., K. Shibuya, et al. (1998). Design and control of mobile manipulation system for human symbiotic humanoid: Hadaly-2, *Proceedings of the 1998 IEEE International Conference on Robotics and Automation. Part 2 (of 4), May 16-20*, Leuven, Belgium, IEEE, Piscataway, NJ, USA

Murphy, R. R. (2004). Human-robot interaction in rescue robotics, *Systems, Man and Cybernetics, Part C, IEEE Transactions on*, 34, (2), 138-153

NASA (2004). The Vision for Space Exploration: National Aeronautics and Space Administration, ttp://www.nasa.gov/pdf/55583main_vision_space_exploration2.pdf.

Nass, C., J. Steuer, et al. (1994). Computers are social actors, *Proceedings of the CHI'94 Conference on Human Factors in Computing Systems, Apr 24-28*, Boston, MA, USA, Publ by ACM, New York, NY, USA

Nourbakhsh, I. R., J. Bobenage, et al. (1999). Affective mobile robot educator with a full-time job, *Artificial Intelligence*, 114, (1-2), 95-124

Nourbakhsh, I. R., K. Sycara, et al. (2005). Human-robot teaming for Search and Rescue, *IEEE Pervasive Computing*, 4, (1), 72-77

Ohba, K., S. Kawabata, et al. (1999). Remote collaboration through time delay in multiple teleoperation, *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'99): Human and Environment Friendly Robots with High Intelligence and Emotional Quotients'*, Oct 17-Oct 21, Kyongju, South Korea, IEEE, Piscataway, NJ, USA

Prince, S., A. D. Cheok, et al. (2002). 3-D live: Real time interaction for mixed reality, *The eight Conference on Computer Supported Cooperative Work (CSCW 2002), Nov 16-20*, New Orleans, LA, United States, Association for Computing Machinery

Rani, P., N. Sarkar, et al. (2004). Anxiety detecting robotic system - Towards implicit human-robot collaboration, *Robotica*, 22, (1), 85-95

Reitmayr, G. and D. Schmalstieg (2004). Collaborative Augmented Reality for Outdoor Navigation and Information Browsing, *Proc. Symposium Location Based Services and TeleCartography 2004 Geowissenschaftliche Mitteilungen Nr. 66*,

Roy, D., K.-Y. Hsiao, et al. (2004). Mental imagery for a conversational robot, *Systems, Man and Cybernetics, Part B, IEEE Transactions on*, 34, (3), 1374-1383

Scholtz, J. (2002). Human Robot Interactions: Creating Synergistic Cyber Forces, In A. Schultz and L. Parker, eds., *Multi-robot Systems: From Swarms to Intelligent Automata*, Kluwer,

Scholtz, J. (2003). Theory and evaluation of human robot interactions, *Proceedings of the 36th Annual Hawaii International Conference on System Sciences*

Scholtz, J., B. Antonishek, et al. (2005). A Comparison of Situation Awareness Techniques for Human-Robot Interaction in Urban Search and Rescue, *CHI 2005 | alt.chi, April 2-7*, Portland, Oregon, USA

Sidner, C. L. and C. Lee (2003). Engagement rules for human-robot collaborative interactions, *System Security and Assurance, Oct 5-8*, Washington, DC, United States, Institute of Electrical and Electronics Engineers Inc.

Sidner, C. L. and C. Lee (2005). Robots as laboratory hosts, *Interactions*, 12, (2), 24-26

Skubic, M., D. Perzanowski, et al. (2004). Spatial language for human-robot dialogs, *Systems, Man and Cybernetics, Part C, IEEE Transactions on*, 34, (2), 154-167

Skubic, M., D. Perzanowski, et al. (2002). Using spatial language in a human-robot dialog, *2002 IEEE International Conference on Robotics and Automation, May 11-15*, Washington, DC, United States, Institute of Electrical and Electronics Engineers Inc.

Sony (2007). http://www.sony.net/SonyInfo/QRIO/story/index_nf.html, accessed May 2007

Tenbrink, T., K. Fischer, et al. (2002). Spatial Strategies in Human-Robot Communication, *Korrekturabzug Kuenstliche Intelligenz*, Heft 4/02, pp 19-23, ISSN 0933-1875, arendtap Verla, Bemen,

Toyota (2007). http://www.toyota.co.jp/en/special/robot/, accessed May 2007

Tsoukalas, L. H. and D. T. Bargiotas (1996). Modeling instructible robots for waste disposal applications, *Proceedings of the 1996 IEEE International Joint Symposia on Intelligence and Systems, Nov 4-5*, Rockville, MD, USA, IEEE, Los Alamitos, CA, USA

Tversky, B., P. Lee, et al. (1999). Why do Speakers Mix Perspectives?, *Spatial Cognition Computing*, 1, 399-412

Watanuki, K., K. Sakamoto, et al. (1995). Multimodal interaction in human communication, *IEICE Transactions on Information and Systems*, E78-D, (6), 609-615

Yanco, H. A., J. L. Drury, et al. (2004). Beyond usability evaluation: Analysis of human-robot interaction at a major robotics competition, *Human-Computer Interaction Human-Robot Interaction*, 19, (1-2), 117-149

5

Robots That Learn Language: A Developmental Approach to Situated Human-Robot Conversations

Naoto Iwahashi
National Institute of Information and Communications Technology,
Advanced Telecommunications Research Institute International
Japan

1. Introduction

Recent progress in sensor technologies and in an infrastructure for ubiquitous computing has enabled robots to sense physical environments as well as the behaviour of users. In the near future, robots that change their behaviour in response to the situation in order to support human activities in everyday life will be increasingly common, so they should feature personally situated multimodal interfaces. One of the essential features of such interfaces is the ability of the robot to share experiences with the user in the physical world.
This ability should be considered in terms of spoken language communication, which is one of the most natural interfaces. The process of human communication is based on certain beliefs shared by those communicating (Sperber & Wilson, 1995). Language is one such shared belief and is used to convey meaning based on its relevance to other shared beliefs. These shared beliefs are formed through interaction with the environment and other people, and the meaning of utterances is embedded in such shared experiences. From this viewpoint, spoken language interfaces are important not only because they enable hands-free interaction but also because of the nature of language, which inherently conveys meaning based on shared experiences. For people to take advantage of such interfaces, language processing methods must make it possible to reflect shared experiences.
However, existing language processing methods, which are characterized by fixed linguistic knowledge, do not make this possible (Allen et al., 2001). In these methods, information is represented and processed by symbols whose meaning has been predefined by the machines' developers. In most cases, the meaning of each symbol is defined by its relationship to other symbols and is not connected to perception or to the physical world. The precise nature of experiences shared by a user and a machine, however, depends on the situation. Because it is impossible to prepare symbols for all possible situations in advance, machines cannot appropriately express and interpret experiences in dynamically changing situations. As a result, users and machines fail to interact in a way that accurately reflects shared experiences.
To overcome this problem and achieve natural linguistic communication between humans and machines, we should use methods that satisfy the following requirements.

Grounding: Beliefs that machines have and the relationship among the beliefs must be grounded in the experiences and the environment of each user. The information of language, perception, and actions should be processed in an integrative fashion. The theoretical framework for language grounding was presented by Roy (Roy, 2005). Previous computational studies explored the grounding of the meanings of utterances in conversations in the physical world (Winograd, 1972; Shapiro et al., 2000), but they did not pursue the learning of new grounded concepts.

Scalability: The machines themselves must be able to learn new concepts and form new beliefs that reflect their experiences. These beliefs should then be embedded in the machines' adaptively changing belief systems.

Sharing: Because utterances are interpreted based on the shared beliefs assumed by a listener in a given situation, the shared beliefs assumed by a user and a machine should ideally be as identical or consistent with each other as possible. The machine should possess a mechanism that enables the user and the machine to infer the state of each other's belief systems in a natural way by coordinating their utterances and actions. Theoretical research (Clark, 1996) and computational modelling (Traum, 1994) focused on the sharing of utterance meanings among participants in communication and have attempted to represent it as a procedure- and rule-driven process. However, we should focus on the shared beliefs to be used in the process of generating and understanding utterances in a physical environment. To achieve robust communication, it is also important to represent the formation of shared belief systems with a mathematical model.

All of these requirements show that learning ability is essential in communications. The cognitive activities related to the grounding, scalability, and sharing of beliefs can be observed clearly in the process of language acquisition by infants as well as in everyday conversation by adults. We have been developing a method that enables robots to learn linguistic communication capabilities from scratch through verbal and nonverbal interaction with users (Iwahashi, 2003a; Iwahashi, 2003b; Iwahashi, 2004), instead of directly pursuing language processing.

Language acquisition by machines has been attracting interest in various research fields (Brents, 1996), and several pioneering studies have developed algorithms based on inductive learning using sets of pairs, where each pair consists of a word sequence and non-linguistic information about its meaning. In several studies (Dyer & Nenov, 1993; Nakagawa & Masukata, 1995; Regier, 1997; Roy & Pentland, 2002; Steels & Kaplan, 2001), visual rather than symbolic information was given as non-linguistic information. Spoken-word acquisition algorithms based on the unsupervised clustering of speech tokens have also been described (Gorin et al., 1994; Nakagawa & Masukata, 1995; Roy & Pentland, 2002). Steels examined (Steels, 2003) the socially interactive process of evolving grounded linguistic knowledge shared by communication agents from the viewpoint of game theory and a complex adaptive system.

In contrast, the method described in this chapter focuses on fast online learning of personally situated language use through verbal and nonverbal interaction in the real world. The learning method applies information from raw speech and visual observations and behavioural reinforcement, which is integrated in a probabilistic framework. Through verbal and nonverbal interaction with a user, a robot learns incrementally and online speech units, lexicon (including words referring to objects and words referring to motions of moving objects), grammar, and a pragmatic capability. A belief system including these

beliefs is represented by a dynamic graphical model (e.g., (Jordan & Sejnowski, 2001)) that has a structure that reflects the state of the user's belief system; thus, the learning makes it possible for the user and the robot to infer the state of each other's belief systems. The method enables the robot to understand even fragmentary and ambiguous utterances of users, act upon them, and generate utterances appropriate for the given situation. In addition, the method enables the robot to learn these things with relatively little interaction. This is also an important feature, because a typical user will not tolerate extended interaction with a robot that cannot communicate and because situations in actual everyday conversation change continuously.

This chapter is organized as follows. Section 2 describes the setting in which the robot learns linguistic communication. Section 3 describes the methods of extracting features from raw speech and image signals, which are used in learning processes. Section 4 explains the method by which the robot learns speech units like phonemes or syllables. Section 5 explains the method by which the robot learns words referring to objects, and their motions. Section 6 explains the method for learning grammar. Section 7 addresses the method for learning pragmatic capability. Section 8 discusses findings and plans for future work.

2. Setting for Learning Interaction

The spoken-language acquisition task discussed in this work was set up as follows. A camera, a robot arm with a hand, and the robot's head were placed next to a table. A user and the learning robot saw and moved the objects on the table as shown in Fig. 1. The head of the robot moved to indicate whether its gaze was directed at the user or at an object. The robot arm had seven degrees of freedom and the hand had four. A touch sensor was attached to the robot's hand. A close-talk microphone was used for speech input. The robot initially did not possess any concepts regarding the specific objects or the ways in which they could be moved nor any linguistic knowledge.

Figure 1. Interaction between a user and a robot

The interactions for learning were carried out as follows. First, to help the robot learn speech units, the user spoke for approximately one minute. Second, in learning image concepts of objects and words that refer to them, the user pointed to an object on the table while speaking a word describing it. A sequence of such learning episodes resulted in a set of pairs, each composed of the image of an object and the word describing it. The objects used included boxes, stuffed and wooden toys, and balls (examples are shown in Fig. 2). In each of the episodes for learning motions and words referring to them, the user moved an object while speaking a word describing the motion. Third, in each of the episodes for learning

grammar, the user moved an object while uttering a sentence describing the action. By the end of this learning process, the user and the robot had shared certain linguistic beliefs consisting of a lexicon and a simple grammar, and the robot could understand utterances to some extent. Note that function words were not included in the lexicon. Finally, in the pragmatic capability learning process, the user asked the robot to move an object by making an utterance and a gesture, and the robot responded. If the robot responded incorrectly, the user slapped the robot's hand. The robot also asked the user to move an object, and the user acted in response. Through such interaction, the robot could learn the pragmatic capability incrementally and in an online manner.
These processes of learning lexicon, grammar, and pragmatic capability could be carried out alternately.

Figure 2. Examples of objects used

3. Speech and Image Signal Processing

All speech and visual sensory output was converted into predetermined features. Speech was detected and segmented based on changes in the short-time power of speech signals. The speech features used were Mel-frequency cepstral coefficients (Davis & Mermelstein, 1980), which are based on short-time spectrum analysis, their delta and acceleration parameters, and the delta of short-time log power. These features (25-dimensional) were calculated in 20-ms intervals with a 30-ms-wide window.
The camera contained three separate CCDs, enabling the robot to obtain three-dimensional information about each scene. The information regarding the object's position in terms of the depth coordinates was used in the attention-control process. Objects were detected when they were located at a distance of 50–80 cm from the camera. The visual features used were L*a*b* components (three dimensions) for the colour, complex Fourier coefficients (eight dimensions) of 2D contours for the shape (Persoon and Fu, 1977), and the area of an object (one dimension) for the size. The trajectories of objects were represented by time-sequence plots of their positions on the table (two-dimensional: horizontal and vertical coordinates), velocities (two-dimensional), accelerations (two-dimensional)..

4. Learning Speech Units

4.1 Difficulty

Speech is a time-continuous one-dimensional signal. The robot learns statistical models of speech units from such signals without being provided with transcriptions of phoneme

sequences or boundaries between phonemes. The difficulty of learning speech units is ascribed to the difficulties involved in speech segmentation and the clustering of speech segments into speech units.

4.2 Method using hidden Markov models

It is possible to cope with the difficulty described above by using hidden Markov models (HMMs) and their learning algorithm, the Baum-Welch algorithm (Baum et al., 1970). HMMs are a particular form of dynamic graphical model that statistically represents dynamic characteristics of time-series data. The model consists of unobservable states, each of which has a probability distribution of observed data, and the probabilities of transitions between them. The Baum-Welch algorithm makes it possible to segment speech, cluster speech segments, and learn the HMM parameters simultaneously.

In this method, each speech-unit HMM includes three states and allows for left-to-right transitions. Twenty speech-unit HMMs were connected to one another to construct a whole speech-unit HMM (Fig. 3), in which transitions were allowed from the last states of the speech-unit HMMs to the first states. All parameters of these HMMs were learned using speech data approximately one minute in length without any phoneme transcriptions. After the speech-unit HMMs had been learned, the individual speech-unit HMMs h_1, h_2, h_3,…, and h_{N_p}, were separated from one another by deleting the edges between them, and a speech-unit HMM set was constructed. The model for each spoken word was represented by connecting these speech-unit HMMs.

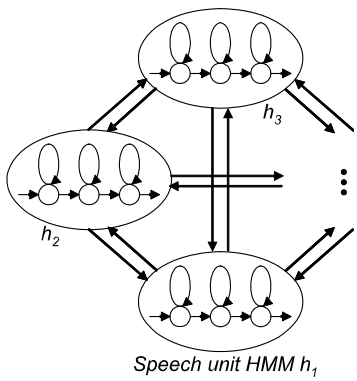


Figure 3. Structure of HMM for learning speech units

4.3 Number of speech units

In the above method, the number, N_p, of speech-unit models was determined empirically. However, ideally it should be learned from speech data. A method of learning the number of speech units and the number of words simultaneously from data comprising pairs that consist of an image of an object and a spoken word describing it has already been presented (Iwahashi, 2003a). It operates in a batch-like manner using information included in both the image and speech observations.

5. Learning Words

The lexicon consists of a set of words, and each word consists of statistical models of speech and a concept. Some words refer to objects and others refer to the motions of moving objects.

5.1 Words referring to objects

In general, the difficulty of acquiring words referring to objects can be ascribed to the difficulties involved in specifying features and applying them to other objects.

Specification: The acoustic features of a spoken word and the visual features of an object to which it refers should be specified using spatiotemporally continuous audio-visual data. For speech, this means that a continuously spoken utterance is segmented into intervals first, and then acoustic features are extracted from one of the segmented intervals. For objects, this means that an object is first selected for a given situation, and then the spatial part of the object is segmented; after that, visual features are extracted from the segmented part of the object.

Extension: To create categories for a given word and its meaning, it is necessary to determine what other features fall into the category to which the specified features belong. This extension of the features of a word's referent to form the word's meaning has been investigated through psychological experiments (Bloom, 2000). When shown an object and given a word for it, human subjects tend to extend the features of the referent immediately in order to infer a particular meaning of the word; this is a cognitive ability called fast mapping (e.g., (Imai & Gentner, 1997)), although such inference is not necessarily correct. For machines, however, the difficulty in acquiring spoken words arises not only from the difficulty in extending the features of referents but also from the difficulty in understanding spoken words. This is because machines are currently much less accurate in recognizing speech than humans; thus, it is not easy for machines to determine whether two different speech segments belong to the same word category.

The learning method described here mainly addresses the problem of extension, in which learning is carried out in an interactive way (Iwahashi, 2004). In learning, the user shows a physical object to the robot and at the same time speaks the name of the object or a word describing it. The robot then decides whether the input word is one in its vocabulary (a *known* word) or not (an *unknown* word). If the robot judges that the input word is an unknown word, it registers it in its vocabulary. If the robot judges that it cannot make an accurate decision, it asks the user a question to determine whether the input word is part of its vocabulary. For the robot to make a correct decision, it uses both speech and visual information about the objects. For example, when the user shows the robot an orange and says the word /ɔrinʒ/ even if the speech recognizer outputs an unknown word /arerʒ/ as the first candidate, the system can modify it to the correct word /ɔrinʒ/ in the lexicon by using visual clues. Such a decision is carried out using a function that represents the confidence that an input pair of image o and speech s belongs to each existing word category w and is adaptively changed online.

Each word or lexical item to be learned includes statistical models, $p(s|w)$ and $p(o|w)$, for the spoken word and for the object image category of its meaning, respectively. The model for each image category $p(o|w)$ is represented by a Gaussian function in a twelve-

dimensional visual feature space (in terms of shape, colour, and size), and learned using a Bayesian method (e.g., (Degroot, 1970)) every time an object image is given.
The Bayesian method makes it possible to determine the area in the feature space that belongs to an image category in a probabilistic way, even if there is only a single sample. The model for each spoken word $p(s \mid w)$ is represented by a concatenation of speech-unit HMMs; this extends a speech sample to a spoken word category.
In experiments, forty words were successfully learned including those that refer to whole objects, shapes, colours, sizes, and combinations of these things.

5.2 Words referring to motions

While the words referring to objects are nominal, the words referring to motions are relational. The concept of the motion of a moving object can be represented by a time-varying spatial relation between a trajector and landmarks, where the trajector is an entity characterized as the figure within a relational profile, and the landmarks are entities characterized as the ground that provide points of reference for locating the trajector (Langacker, 1991). Thus, the concept of the trajectory of an object depends on the landmarks. In Fig. 4, for instance, the trajectory of the stuffed toy on the left moved by the user, as indicated by the white arrow, is understood as *move over* and *move onto* when the landmarks are considered to be the stuffed toy in the middle and the box on the right, respectively. In general, however, information about what is a landmark is not observed in learning data. The learning method must infer the landmark selected by a user in each scene. In addition, the type of coordinate system in the space should also be inferred to appropriately represent the graphical model for each concept of a motion.

Figure 4. Scene in which utterances were made and understood

In the method for learning words referring to motions (Haoka & Iwahashi, 2000), in each episode, the user moves an object while speaking a word describing the motion. Through a sequence of such episodes, the set comprising pairs of a scene O before an action, and action a, $D_m = \{(a_1, O_1), (a_2, O_2), ..., (a_{N_m}, O_{N_m})\}$, is given as learning data for a word referring to a motion concept. Scene O_i includes the set of positions $o^i_{j,p}$ and features $o^i_{j,f}$ concerning colour, size, and shape, $j = 1, ...J_i$, of all objects in the scene. Action a_i is represented by a pair (t_i, u_i) consisting of trajector object t_i and trajectory u_i of its movement. The concepts regarding motions are represented by probability density functions of the trajectory u of moved objects. Four types of coordinate systems

$k \in \{1,2,3,4\}$ are considered. The probability density function $p(u \mid o_{l,p}, k_w, \lambda_w)$ for the trajectory of the motion referred to by word w is represented by HMM λ_w and the type of coordinate system k_w, given positions $o_{l,p}$ of a landmark. The HMM parameters λ_w of the motion are learned while the landmarks l and the type of coordinate system k_w are being inferred based on the EM (expectation maximization) algorithm, in which a landmark is taken as a latent variable as

$$(\mathbf{l}, k_w, \lambda_w) = \arg\max_{\mathbf{l},k,\lambda} \sum_{i=1}^{N_m} \log p\left(u_i \middle| o^{i}_{l_i,p}, k, \lambda\right) , \quad (1)$$

where $\mathbf{l} = [l_1, l_2, ..., l_{N_m}]$. Here, l_i is a discrete variable across all objects in each scene O_i, and it represents a landmark object. The number of states in the HMM is also learned through cross-validation. In experiments, six motion concepts, "move-over," "move-onto," "move-close-to", "move-away", "move-up", and "move-circle", were successfully learned. Examples of inferred landmarks and coordinates in the learning of some motion concepts are shown in Fig. 5. A graphical model of the lexicon containing words referring to objects and motions is shown in Fig. 6.

The trajectory for the motion referred to by word w is generated by maximizing the output probability of the learned HMM, given the positions of a trajector and a landmark as

$$\tilde{u} = \arg\max_{u} p(u \mid o_{l,p}, k_w, \lambda_w) . \quad (2)$$

This maximization is carried out by solving simultaneous linear equations (Tokuda et al., 1995).

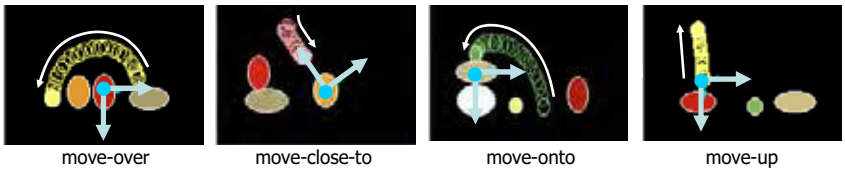


Figure 5. Trajectories of objects moved in learning episodes and selected landmarks and coordinates

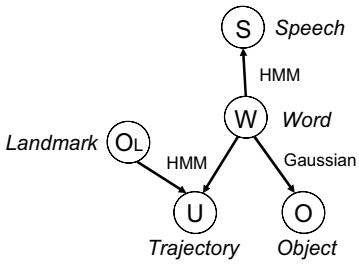


Figure 6. Graphical model of a lexicon containing words referring to objects and motions

6. Learning Grammar

To enable the robot to learn grammar, we use moving images of actions and speech describing them. The robot should detect the correspondence between a semantic structure in the moving image and a syntactic structure in the speech. However, such semantic and syntactic structures are not observable. While an enormous number of structures can be extracted from a moving image and speech, the method should select the ones with the most appropriate correspondence between them. Grammar should be statistically learned using such correspondences, and then inversely used to extract the correspondence.

The set comprising triplets of a scene O before an action, action a, and a sentence utterance s describing the action, $\mathrm{D}_g = \{(s_1, a_1, O_1), (s_2, a_2, O_2), ..., (s_{N_g}, a_{N_g}, O_{N_g})\}$, is given in this order as learning data. It is assumed that each utterance is generated based on the stochastic grammar G based on a conceptual structure. The conceptual structure used here is a basic schema used in cognitive linguistics, and is expressed with three conceptual attributes—[motion], [trajector], and [landmark]—that are initially given to the system, and they are fixed. For instance, when the image is the one shown in Fig. 4 and the corresponding utterance is the sequence of spoken words "*large frog brown box move-onto*", the conceptual structure $z = (W_T, W_L, W_M)$ might be

$$\begin{bmatrix} [\text{trajector}] & : \textit{large frog} \\ [\text{landmark}] & : \textit{brown box} \\ [\text{motion}] & : \textit{move-onto} \end{bmatrix},$$

where the right-hand column contains the spoken word subsequences W_T, W_L, and W_M, referring to trajector, landmark, and motion, respectively, in a moving image. Let y denote the order of conceptual attributes, which also represents the order of the constituents with the conceptual attributes in an utterance. For instance, in the above utterance, the order is [trajector]-[landmark]-[motion]. The grammar is represented by the set comprising the occurrence probabilities of the possible orders as $G = \{P(y_1), P(y_2), ..., P(y_k)\}$.

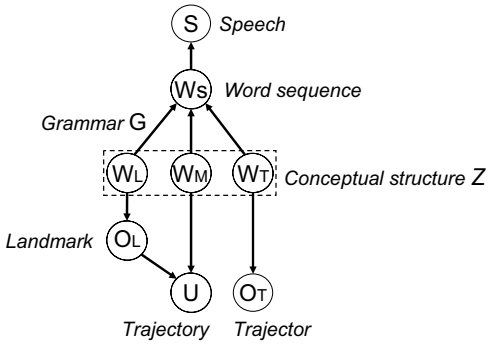


Figure 7. Graphical model of lexicon and grammar

Joint probability density function $p(s,a,O;L,G)$, where L denotes a parameter set of the lexicon, is represented by a graphical model with an internal structure that includes the parameters of grammar G and conceptual structure z that the utterance represents (Fig. 7). By assuming that $p(z,O;L,G)$ is constant, we can write the joint log-probability density function as

$$\begin{aligned}
&\log p(s,a,O;L,G) \\
&\quad = \log \sum_{z} p(s \mid z;L,G)\, p(a \mid x,O;L,G)\, p(z,O;L,G) \\
&\quad \approx \alpha \max_{z,l} \big(\log p(s \mid z;L,G) && \text{[Speech]} \\
&\qquad + \log p\big(u \mid o_{l,p}, W_M;L\big) && \text{[Motion]} \\
&\qquad + \log p\big(o_{t,f} \mid W_T;L\big) + \log p\big(o_{l,f} \mid W_L;L\big)\big), && \text{[Static image of object]}
\end{aligned} \tag{3}$$

where α is a constant value of $p(z,O;L,G)$. Furthermore, t and l are discrete variables across all objects in each moving image and represent, respectively, a trajector object and a landmark object. As an approximation, the conceptual structure $z=(W_T,W_L,W_M)$ and landmark l, which maximizes the output value of the function, are used instead of summing up for all possible conceptual structures and landmarks.

Estimate $\tilde{G}_i$ of grammar G given ith learning data is obtained as the maximum values of the posterior probability distribution as

$$\tilde{G}_i = \arg\max_{G} p\big(G \mid \mathrm{D}_g^i;L\big), \tag{4}$$

where D_g^i denotes learning sample set $\{(s_1,a_1,O_1),(s_2,a_2,O_2),...,(s_i,a_i,O_i)\}$.

An utterance s asking the robot to move an object is understood using lexicon L and grammar G. Accordingly, one of the objects, t, in current scene O is grasped and moved along trajectory u by the robot. Action $\tilde{a}=(\tilde{t},\tilde{u})$ for utterance s is calculated as

$$\tilde{a} = \arg\max_{a} = \log p\big(s,a,O;L,\tilde{G}\big). \tag{5}$$

This means that from among all the possible combinations of conceptual structure z, trajector and landmark objects t and l, and trajectory u, the method selects the combination that maximizes the value of the joint log-probability density function $\log p(s,a,O;L,G)$.

7. Learning Pragmatic Capability for Situated Conversations

7.1 Difficulty

As mentioned in Sec. 1, the meanings of utterances are conveyed based on certain beliefs shared by those communicating in the situations. From the perspective of objectivity, if those communicating want to logically convince each other that proposition p is a shared

belief, they must prove that the infinitely nested proposition, "They have information that they have information that … that they have information that *p*", also holds. However, in reality, all we can do is assume, based on a few clues, that our beliefs are identical to those of the other people we are talking to. In other words, it can never be guaranteed that our beliefs are identical to those of other people. Because shared beliefs defined from the viewpoint of objectivity do not exist, it is more practical to see shared beliefs as a process of interaction between the belief systems held by each person communicating. The processes of generating and understanding utterances rely on the system of beliefs held by each person, and this system changes autonomously and recursively through these two processes. Through utterances, people simultaneously send and receive both the meanings of their words and, implicitly, information about one another's systems of beliefs. This dynamic process works in a way that makes the belief systems consistent with each other. In this sense, we can say that the belief system of one person couples structurally with the belief systems of those with whom he or she is communicating (Maturana, 1978).

When a participant interprets an utterance based on their assumptions that certain beliefs are shared and is convinced, based on certain clues, that the interpretation is correct, he or she gains the confidence that the beliefs are shared. On the other hand, since the sets of beliefs assumed to be shared by participants actually often contain discrepancies, the more beliefs a listener needs to understand an utterance, the greater the risk that the listener will misunderstand it.

As mentioned above, a pragmatic capability relies on the capability to infer the state of a user's belief system. Therefore, the method should enable the robot to adapt its assumption of shared beliefs rapidly and robustly through verbal and nonverbal interaction. The method should also control the balance between (i) the transmission of the meaning of utterances and (ii) the transmission of information about the state of belief systems in the process of generating utterances.

The following is an example of generating and understanding utterances based on the assumption of shared beliefs. Suppose that in the scene shown in Fig. 4 the frog on the left has just been put on the table. If the user in the figure wants to ask the robot to move a frog onto the box, he may say, "*frog box move-onto*". In this situation, if the user assumes that the robot shares the belief that the object moved in the previous action is likely to be the next target for movement and the belief that the box is likely to be something for the object to be moved onto, he might just say "*move-onto*"[1]. To understand this fragmentary and ambiguous utterance, the robot must possess similar beliefs. If the user knows that the robot has responded by doing what he asked it to, this would strengthen his confidence that the beliefs he has assumed to be shared really are shared. Conversely, when the robot wants to ask the user to do something, the beliefs that it assumes to be shared are used in the same way. It can be seen that the former utterance is more effective than the latter in transmitting the meaning of the utterance, while the latter is more effective in transmitting information about the state of belief systems.

[1] Although the use of a pronoun might be more natural than the deletion of noun phrases in some languages, the same ambiguity in meaning exists in both such expressions.

7.2 Representation of a belief system

To cope with the above difficulty, the belief system of the robot needs to have a structure that reflects the state of the user's belief system so that the user and the robot infer the state of each other's belief systems. This structure consists of the following two parts:

The shared belief function represents the assumption of shared beliefs and is composed of a set of belief modules with values (local confidence values) representing the degree of confidence that each belief is shared by the robot and the user.

The global confidence function represents the degree of confidence that the whole of the shared belief function is consistent with the shared beliefs assumed by the user.

Such a belief system is depicted in Fig. 8. The beliefs we used are those concerning speech, motions, static images of objects, and motion-object relationship, and the effect of behavioural context. The motion-object relationship and the effect of behavioural context are represented as follows.

Motion-object relationship $B_R\left(o_{t,f}, o_{l,f}, W_M; R\right)$: The motion-object relationship represents the belief that in the motion corresponding to motion word W_M, feature $o_{t,f}$ of object t and feature $o_{l,f}$ of object l are typical for a trajector and a landmark, respectively. This belief is represented by a conditional multivariate Gaussian probability density function, $p\left(o_{t,f}, o_{l,f} \mid W_M; R\right)$, where R is its parameter set.

Effect of behavioural context $B_H\left(i, q; H\right)$: The effect of behavioural context represents the belief that the current utterance refers to object i, given behavioural context q. Here, q includes information on which objects were a trajector and a landmark in the previous action and which object the user's current gesture refers to. This belief is represented by a parameter set H.

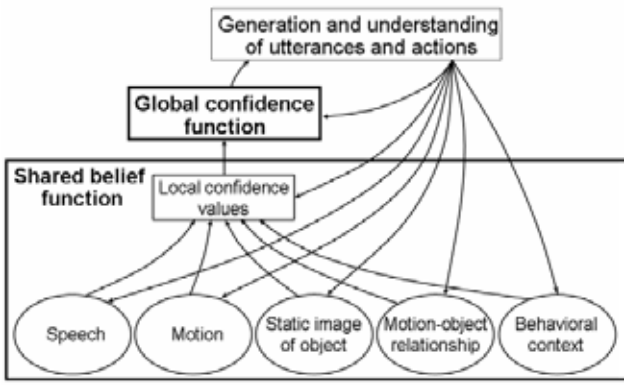


Figure 8. Belief system of robot that consists of shared belief and global confidence functions

7.3 Shared belief function

The beliefs described above are organized and assigned local confidence values to obtain the shared belief function used in the processes of generating and understanding utterances.

This shared belief function Ψ is the extension of $\log p(s,a,O;L,G)$ in Eq. 3. The function outputs the degree of correspondence between utterance s and action a. It is written as

$$\begin{aligned}
&\Psi(s,a,O,q,L,G,R,H,\Gamma) \\
&\quad = \max_{z,l}\Big(\gamma_1 \log p(s \mid z;L,G) && \text{[Speech]} \\
&\qquad + \gamma_2 \log p\left(u \mid o_{l,p},W_M;L\right) && \text{[Motion]} \\
&\qquad + \gamma_2\left(\log p\left(o_{t,f} \mid W_T;L\right) + \log p\left(o_{l,f} \mid W_L;L\right)\right) && \text{[Static image of object]} \\
&\qquad + \gamma_3 \log p\left(o_{t,f},o_{l,f} \mid W_M;R\right) && \text{[Motion-object relationship]} \\
&\qquad + \gamma_4\left(B_H(t,q;H) + B_H(l,q;H)\right)\Big), && \text{[Behavioural context]}
\end{aligned} \tag{6}$$

where $\Gamma = \{\gamma_1,\gamma_2,\gamma_3,\gamma_4\}$ is a set of local confidence values for beliefs corresponding to the speech, motion, static images of objects, motion-object relationship, and behavioural context. Given O, q, L, G, R, H, and Γ, the corresponding action $\tilde{a} = (\tilde{t},\tilde{u})$, understood to be the meaning of utterance s, is determined by maximizing the shared belief function as

$$\tilde{a} = \arg\max_{a} \Psi(s,a,O,q,L,G,R,H,\Gamma). \tag{7}$$

7.4 Global confidence function

The global confidence function f outputs an estimate of the probability that the robot's utterance s will be correctly understood by the user . It is written as

$$f(d) = \frac{1}{\pi}\arctan\left(\frac{d-\lambda_1}{\lambda_2}\right) + 0.5, \tag{8}$$

where λ_1 and λ_2 are the parameters of the function and input d of this function is a margin in the value of the output of the shared belief function between an action that the robot asks the user to take and other actions in the process of generating an utterance. Margin d in generating utterance s to refer to action a in scene O in behavioural context q is defined as

$$d(s,a,O,q,L,G,R,H,\Gamma) = \Psi(s,a,O,q,L,G,R,H,\Gamma) - \max_{A \neq a} \Psi(s,A,O,q,L,G,R,H,\Gamma). \tag{9}$$

Examples of the shapes of global confidence functions are shown in Fig. 9. Clearly, a large margin increases the probability of the robot being understood correctly by the user. If there is a high probability of the robot's utterances being understood correctly even when the margin is small, it can be said that the robot's beliefs are consistent with those of the user. The example of a shape of such a global confidence function is indicated by "strong". In contrast, the example of a shape when a large margin is necessary to get a high probability is indicated by "weak".

When the robot asks for action a in scene O in behavioural context q, it generates utterance $\tilde{s}$ so as to bring the value of the output of f as close as possible to the value of parameter ξ, which represents the target probability of the robot's utterance being understood correctly. This utterance can be represented as

$$\tilde{s} = \arg\min_{s} \left| f\left(d\left(s,a,O,q,L,G,R,H,\Gamma\right)\right) - \xi \right|. \tag{10}$$

The robot can increase its chance of being understood correctly by using more words. On the other hand, if the robot can predict correct understanding with a sufficiently high probability, it can manage with a fragmentary utterance using a small number of words.

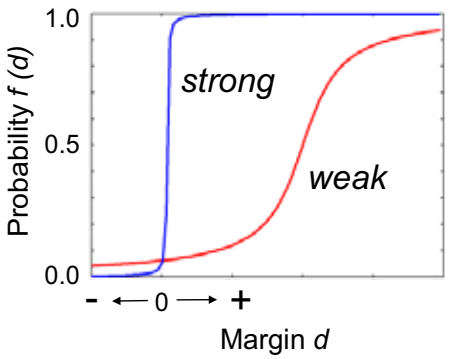


Figure 9. Examples of shapes of global confidence functions

7.5 Learning methods

The shared belief function Ψ and the global confidence function f are learned separately in the processes of utterance understanding and utterance generation by the robot, respectively.

7.5.1 Utterance understanding by the robot

Shared belief function Ψ is learned incrementally, online, through a sequence of episodes, each of which comprises the following steps.

1. Through an utterance and a gesture, the user asks the robot to move an object.
2. The robot acts on its understanding of the utterance.
3. If the robot acts correctly, the process ends. Otherwise, the user slaps its hand.
4. The robot acts in a different way.
5. If the robot acts incorrectly, the user slaps its hand. The process ends.

In each episode, a quadruplet (s,a,O,q) comprising the user's utterance s, scene O, behavioural context q, and action a that the user wants to ask the robot to take, is used. The robot adapts the values of parameter set R for the belief about the motion-object relationship, parameter set H for the belief about the effect of the behavioural context, and local confidence parameter set Γ. Lexicon L and grammar G were learned beforehand, as

described in the previous sections. When the robot acts correctly in the first or second trials, it learns R by applying the Bayesian learning method using the information about features of trajector and landmark objects $o_{t,f}$, $o_{l,f}$ and motion word W_M in the utterances. In addition, when the robot acts correctly in the second trial, it associates utterance s, correct action a, incorrect action A from the first trial, scene O, and behavioural context q with one another and makes these associations into a learning sample. When the ith sample $(s_i, a_i, A_i, O_i, q_i)$ is obtained based on this process of association, H_i and Γ_i are adapted to approximately minimize the probability of misunderstanding as

$$(\tilde{H}_i, \tilde{\Gamma}_i) = \arg\min_{H,\Gamma} \sum_{j=i-K}^{i} w_{i-j} g\left(\Psi(s_j, a_j, O_j, q_j, L, G, R_i, H, \Gamma) - \Psi(s_j, A_j, O_j, q_j, L, G, R_i, H, \Gamma)\right), \quad (11)$$

where $g(x)$ is $-x$ if $x < 0$ and 0 otherwise, and K and w_{i-j} represent the number of latest samples used in the learning process and the weights for each sample, respectively.

7.5.2 Utterance generation by the robot

Global confidence function f is learned incrementally, online through a sequence of episodes, each of which consists of the following steps.

1. The robot generates an utterance to ask the user to move an object.
2. The user acts according to his or her understanding of the robot's utterance.
3. The robot determines whether the user's action is correct.

In each episode, a triplet (a, O, q) comprising scene O, behavioural context q, and action a that the robot needs to ask the user to take is provided to the robot before the interaction. The robot generates an utterance that brings the value of the output of global confidence function f as close to ξ as possible. After each episode, the value of margin d in the utterance generation process is associated with information about whether the utterance was understood correctly, and this sample of associations is used for learning. The learning is done online and incrementally so as to approximate the probability that an utterance will be understood correctly by minimizing the weighted sum of squared errors in the most recent episodes. After the ith episode, parameters λ_1 and λ_2 are adapted as

$$[\lambda_{1,i}, \lambda_{2,i}] \leftarrow (1-\delta)[\lambda_{1,i-1}, \lambda_{2,i-1}] + \delta[\tilde{\lambda}_{1,i-1}, \tilde{\lambda}_{2,i-1}], \quad (12)$$

where

$$(\tilde{\lambda}_{1,i}, \tilde{\lambda}_{2,i}) = \arg\min_{\lambda_1,\lambda_2} \sum_{j=i-K}^{i} w_{i-j} \left(f(d_j; \lambda_1, \lambda_2) - e_j\right)^2, \quad (13)$$

where e_i is 1 if the user's understanding is correct and 0 if it is not, and δ is the value that determines learning speed.

7.6 Experimental results

7.6.1 Utterance understanding by the robot

At the beginning of the sequence for learning shared belief function Ψ, the sentences were relatively complete (e.g., "*green frog red box move-onto*"). Then the lengths of the sentences were gradually reduced (e.g., "*move-onto*") to become fragmentary so that the meanings of the sentences were ambiguous. At the beginning of the learning process, the local confidence values γ_1 and γ_2 for speech, static images of objects, and motions were set to 0.5, while γ_3 and γ_4 were set to 0.

R could be estimated with high accuracy during the episodes in which relatively complete utterances were given and understood correctly. In addition, H and Γ could be effectively estimated based on the estimation of R during the episodes in which fragmentary utterances were given. Figure 10 shows changes in the values of γ_1, γ_2, γ_3, and γ_4. The values did not change during the first thirty-two episodes because the sentences were relatively complete and the actions in the first trials were all correct. Then, we can see that value γ_1 for speech decreased adaptively according to the ambiguity of a given sentence, whereas the values γ_2, γ_3, and γ_4 for static images of objects, motions, the motion-object relationship, and behavioural context increased. This means that non-linguistic information was gradually being used more than linguistic information.

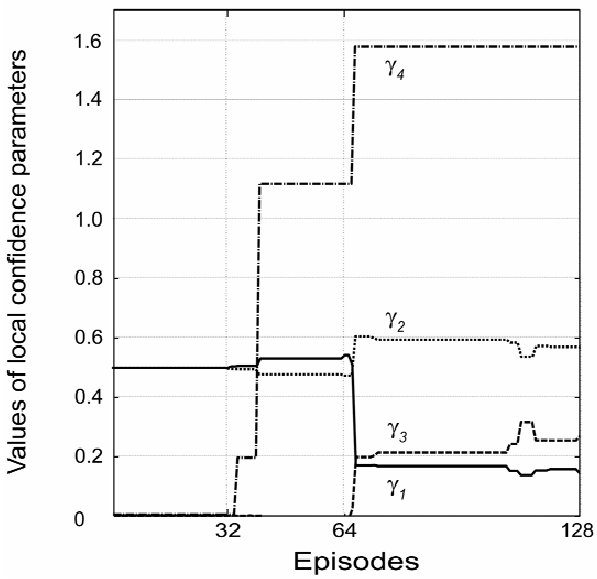
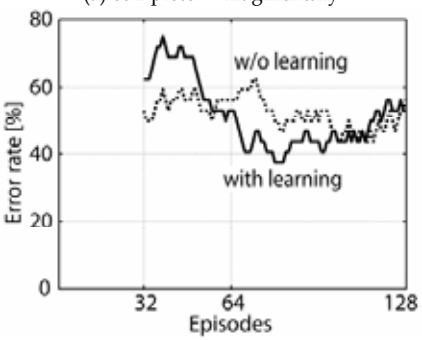


Figure 10. Changes in values of local confidence parameters

Figure 11(a) shows the decision error (misunderstanding) rates obtained during the course of the interaction, along with the error rates obtained for the same learning data by keeping the values of the parameters of the shared belief function fixed to their initial values. We can see that the learning was effective. In contrast, when fragmentary utterances were provided over the whole sequence of the interaction, the robot did not learn well (Fig. 11(b)) because it misunderstood the utterances too often.

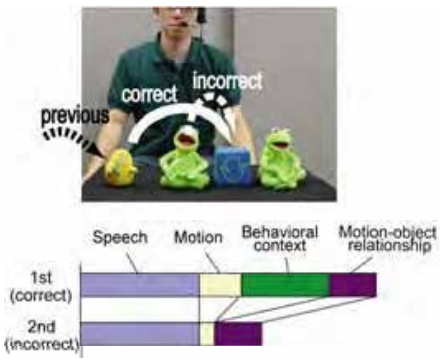
(a) complete → fragmentary



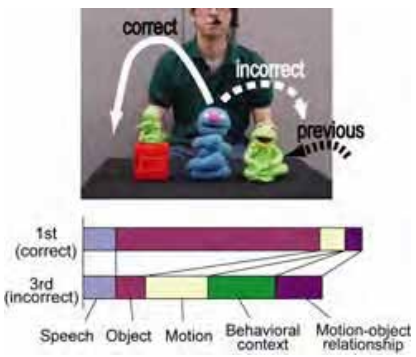
(b) fragmentary → fragmentary

Figure 11. Change in decision error rate

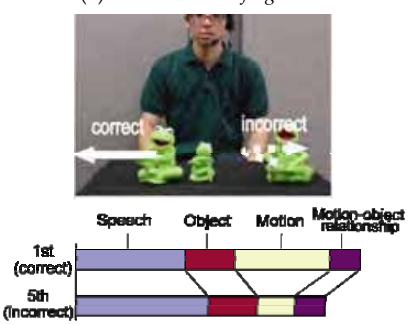
Examples of actions generated as a result of correct understanding are shown together with the output log-probabilities from the weighted beliefs in Figs. 12(a), (b), and (c) along with the second, third, and fifth choices for action, respectively, which were incorrect. It is clear that each non-linguistic belief was used appropriately in understanding the utterances according to their relevance to the situations. The beliefs about the effect of the behavioural context were more effective in Fig. 12(a) , while in Fig. 12(b), the beliefs about the concepts of the static images of objects were more effective than other non-linguistic beliefs in leading to the correct understanding. In Fig. 12(c), even when error occurred in speech recognition, the belief about the motion concept was effectively used to understand the utterance with ambiguous meaning.



(a) "*move-onto*"



(b) "*monster small frog move-over*"

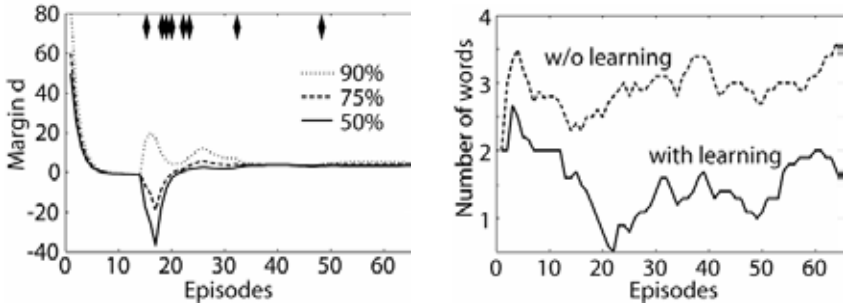


(c) *"large frog small frog move-away"*

Figure 12. Examples of actions generated as a result of correct understanding and weighted output log-probabilities from beliefs, along with second, third, and fifth action choices

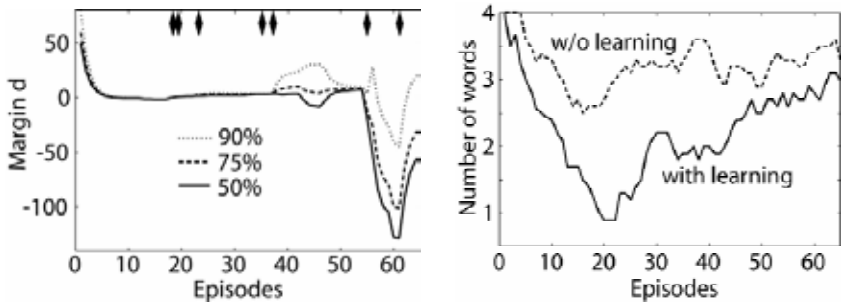
7.6.2 Utterance generation by the robot

In each episode for learning global confidence function f, the robot generated an utterance so as to make the value of the output of the global confidence function as close to $\xi = 0.75$ as possible. Even when the target value ξ was fixed at 0.75, we found that the obtained values were widely distributed around it. The initial shape of the global confidence function was set to make $f^{-1}(0.9) = 161$, $f^{-1}(0.75) = 120$, and $f^{-1}(0.5) = 100$, meaning that a large margin was necessary for an utterance to be understood correctly. In other words, the shape of f in this case represents weak confidence. Note that when all of the values are close to 0, the slope in the middle of f is steep, and the robot makes the decision that a small margin is sufficient for its utterances to be understood correctly. The shape of f in this case represents strong confidence.



(a) (b)

Figure 13. (a) Changes in global confidence function and (b) number of words needed to describe objects in each utterance, $\xi = 0.75$



(a) (b)

Figure 14. (a) Changes in global confidence function and (b) number of words needed in each utterance to describe objects, $\xi = 0.95$

The changes in $f(d)$ are shown in Fig. 13(a), where three lines have been drawn for $f^{-1}(0.9)$, $f^{-1}(0.75)$, and $f^{-1}(0.5)$ to make the shape of f easily recognizable. The

interactions in which utterances were misunderstood are depicted in the upper part of the graph by the black diamonds. Figure 13(b) displays the changes in the moving average of the number of words used to describe the objects in each utterance, along with the changes obtained when f was not learned, which are shown for comparison. After the learning began, the slope in the middle of f rapidly became steep, and the number of words uttered decreased. The function became temporarily unstable with $f^{-1}(0.5) < 0$ at around the 15 th episode. The number of words uttered then became too small, which sometimes led to misunderstanding. We might say that the robot was overconfident in this period. Finally, the slope became steep again at around the 35th episode.

We conducted another experiment in which the value of parameter ξ was set to 0.95. The result of this experiment are shown in Fig. 14. It is clear that, after approximately the 40th interaction, the change in f became very unstable and the number of words became large. We found that f became highly unstable when the utterances with a large margin d were not understood correctly.

8. Discussion

8.1 Sharing the risk of being misunderstood

The experiments in learning a pragmatic capability illustrate the importance of sharing the risk of not being understood correctly between the user and the robot.

In the learning period for utterance understanding by the robot, the values of the local confidence parameters changed significantly when the robot acted incorrectly in the first trial and correctly in the second trial. To facilitate learning, the user had to gradually increase the ambiguity of utterances according to the robot's developing ability to understand them and had to take the risk of not being understood correctly. In the its learning period for utterance generation, the robot adjusted its utterances to the user while learning the global confidence function. When the target understanding rate ξ was set to 0.95, the global confidence function became very unstable in cases where the robot's expectations of being understood correctly at a high probability were not met. This instability could be prevented by using a lower value of ξ, which means that the robot would have to take a greater risk of not being understood correctly.

Accordingly, in human-machine interaction, both users and robots must face the risk of not being understood correctly and thus adjust their actions to accommodate such risk in order to effectively couple their belief systems. Although the importance of controlling the risk of error in learning has generally been seen as an exploration-exploitation trade-off in the field of reinforcement learning by machines (e.g., (Dayan & Sejnowski, 1996)), we argue here that the mutual accommodation of the risk of error by those communicating is an important basis for the formation of mutual understanding.

8.2 Incomplete observed information and fast adaptation

In general, an utterance does not contain complete information about what a speaker wants to convey to a listener. The proposed learning method interpreted such utterances according to the situation by providing necessary but missing information by making use of the assumption of shared beliefs. The method also enabled the robot and the user to adapt such an assumption of shared beliefs to each other with little interaction. We can say that the method successfully

coped with two problems faced by systems interacting with the physical world: the incompleteness of observed information and fast adaptation (Matsubara & Hashida, 1989).
Some previous studies have been done in terms of these problems. In the field of autonomous robotics, the validity of the architecture in which sub-systems are allocated in parallel has been demonstrated (Brooks, 1986). This, however, failed to be applied to large-scale systems because of the lack of a mathematical theory for the interaction among the sub-systems. On the other hand, Bayesian networks (Pearl, 1988) have been studied intensively, providing a probabilistic theory of the interaction among the sub-systems. This can cope with the incompleteness of observed information in large-scale systems but does not address fast adaptation.
Shared belief function Ψ, which is a large-scale system, has the merits of both these approaches. It is a kind of Bayesian network with statistical models of beliefs allocated in parallel in which weighting values Γ are added to these models, as shown in Eq. 6. Based on both the parallel allocation of sub-systems and the probabilistic theory, this method can cope successfully with the incompleteness of observed information and can achieve fast adaptation of the function by changing weighting values.

8.3 Initial setting

The No Free Lunch Theory (Wolpert, 1995) shows that when no prior knowledge about a problem exists, it is not possible to assume that one learning algorithm is superior to another. That is, there is no learning method that is efficient for all possible tasks. This suggests that attention should be paid to domain specificity as well as versatility.
In the methods described here, the initial setting for the learning was chosen by taking into account the generality and efficiency of language learning. The conceptual attributes—[motion], [trajector], and [landmark]—were given beforehand because they are general and essential in linguistic and other cognitive processes. Given this setting, however, the constructions that the method could learn were limited to those like transitive and ditransitive ones. In future work, we will study how to overcome this limitation.

8.4 Abstract meanings

The image concepts of objects that are learned by the methods described in Sec. 5 are formed directly from perceptual information. However, we must consider words that refer to concepts that are more abstract and that are not formed directly from perceptual information, such as "tool," "food," and "pet". In a study on the abstract nature of the meanings of symbols (Savage-Rumbaugh, 1986), it was found that chimpanzees could learn the lexigrams (graphically represented words) that refer to both individual object categories (e.g., "banana", "apple", "hammer", and "key") and the functions ("tool" and "food") of the objects. They could also learn the connection between the lexigrams referring to these two kinds of concepts and generalize it appropriately to connect new lexigrams for individual objects to lexigrams for functions.
A method enabling robots to gain this ability has been proposed (Iwahashi et al., 2006). In that method, the motions of objects are taken as their functions. The main problem is the decision regarding whether the meaning of a new input word applies to a concept formed directly from perceptual information or to a function of objects. Because these two kinds of concepts are allocated to the states of different nodes in the graphical model, the problem becomes the selection of the structures of the graphical model. This selection is made by the Bayesian principle with the calculation of posterior probabilities using the variational Bayes method (Attias, 1999).

8.5 Prerequisites for conversation

Language learning can be regarded as a kind of role reversal imitation (Carpenter et al., 2005). To coordinate roles in a joint action among participants, the participants should be able to read the intentions of the others. It is known that at a very early stage of development, infants become able to understand the intentional actions of others (Behne et al., 2005) and even to understand that others might have beliefs different from their own (Onishi & Baillargeon, 2005).

In the method described in this chapter, the robot took the speech acts of input utterances as descriptive or directive. If the utterance was descriptive in terms of the current situation, the robot learned a lexicon or grammar. If the utterance was directive, the robot moved an object. The distinction between descriptive and directive acts was made by taking account of both the user's behaviour and speech. The simple rule for this distinction was given to the robot and the user beforehand, so they knew it.

A learning method that enables robots to understand the kinds of speech act in users' utterances has been presented (Taguchi et al., 2007). Using this method, robots came to understand their roles in interactions by themselves based on role reversal imitation and came to learn such distinction of speech acts which requests for actions. Eventually the robot came to respond to a request by moving an object, and to answer a question by speaking and pointing. The method was developed by expanding the graphical model described here.

8.6 Psychological investigation

The experimental results showed that the robot could learn new concepts and form a system of beliefs that it assumed the user also had. Because the user and the robot came to understand fragmentary and ambiguous utterances, they must have shared similar beliefs and must have been aware that they shared them. It would be interesting to investigate through psychological experiments the dynamics of belief sharing between users and robots.

9. Conclusion

A developmental approach to language processing for situated human-robot conversations was presented. It meets three major requirements that existing language processing methods cannot: grounding, scalability, and sharing of beliefs. The proposed method enabled a robot to learn language communication capability online with relatively little verbal and nonverbal interaction with a user by combining speech, visual, and behavioural reinforcement information in a probabilistic framework. The adaptive behaviour of the belief systems is modelled by the structural coupling of the belief systems held by the robot and the user, and it is executed through incremental online optimisation during the process of interaction. Experimental results revealed that through a small, but practical number of learning episodes with a user, the robot was eventually able to understand even fragmentary and ambiguous utterances, act upon them, and generate utterances appropriate for the given situation. Future work includes investigating the learning of more complex linguistic knowledge and learning in a more natural environment.

10. Acknowledgements

This work was supported by a research grant from the National Institute of Informatics.

11. References

Allen, J.; Byron, D.; Dzikovska, M.; Ferguson, G.; Galescu, L. & Stent, A. (2001). Toward conversational human-computer interaction. *AI Magazine*, Vol. 22, Issue. 4, pp. 27-38

Attias, H. (1999). Inferring parameters and structure of latent variable models by variational Bayes. *Proceedings of Int. Conf. on Uncertainty in Artificial Intelligence*, pp. 21–30

Baum, L. E.; Petrie, T.; Soules, G. & Weiss, N. (1970). A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains. *Annals of Mathematical Statistics*, Vol. 41, No. 1, pp. 164–171

Behne, T.; Carpenter, M.; Call, J. & Tomasello, M. (2005). Unwilling versus unable – infants' understanding of intentional action. *Developmental Psychology*, Vol. 41, No. 2, pp. 328–337

Bloom, P. (2000). How children learn the meanings of words. MIT Press

Brent, M. R. (1996). Advances in the computational study of language acquisition. *Cognition*, Vol. 61, pp. 1–61

Brooks, R. (1986). A robust layered control system for a mobile robot. *IEEE Journal of Robotics and Automation*, Vol. 1, pp.14–23

Carpenter, M.; Tomasello, M.; Striano, T. (2005). Role reversal imitation and language in typically developing infants and children with autism. *INFANCY*, Vol. 8, No. 3, pp. 253–278

Clark, H. (1996). Using Language. Cambridge University Press

Dayan, P. & Sejnowski, T. J. (1996). Exploration bonuses and dual control. *Machine Learning*, Vol. 25, pp. 5–22

Davis, S. & Mermelstein, P. (1980). Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences. *IEEE Transactions on Acoustics, Speech and Signal Processing*, Vol. 28, No. 4, pp. 357–366

DeGroot, M. H. (1970). Optimal Statistical Decisions. McGraw-Hill

Dyer, M. G. & Nenov, V. I. (1993). Learning language via perceptual/motor experiences. *Proceedings of Annual Conf. of the Cognitive Science Society*, pp. 400–405

Gorin, A.; Levinson, S. & Sanker, A. (1994). An experiment in spoken language acquisition. *IEEE Trans. on Speech and Audio Processing*, Vol. 2, No. 1, pp. 224–240

Haoka, T. & Iwahashi, N. (2000). Learning of the reference-point-dependent concepts on movement for language acquisition. *Tech. Rep. of the Institute of Electronics, Information and Communication Engineers PRMU2000-105*, pp.39-45

Imai, M. & Gentner, D. (1997). A crosslinguistic study of early word meaning – universal ontology and linguistic influence. *Cognition*, Vol. 62**,** pp. 169–200

Iwahashi, N. (2003a). Language acquisition through a human-robot interface by combining speech, visual, and behavioral information. *Information Sciences*, Vol. 156, pp. 109-121

Iwahashi, N. (2003b). A method of coupling of belief systems through human-robot language interaction. *Proceedings of IEEE Workshop on Robot and Human Interactive Communication*, pp. 385-390

Iwahashi, N. (2004). Active and unsupervised learning of spoken words through a multimodal interface. *Proceedings of IEEE Workshop on Robot and Human Interactive Communication*, pp. 437-442

Iwahashi, N.; Satoh, K. & Asoh, H. (2006). Learning abstract concepts and words from perception based on Bayesian model selection. *Tech. Rep. of the Institute of Electronics, Information and Communication Engineers PRMU-2005-234*, pp. 7-12

Jordan, M. I. & Sejnowski, T.J. Eds. (2001). Graphical Models - Foundations of Neural Computation. The MIT Press

Langacker, R. (1991). Foundation of cognitive grammar. Stanford University Press, CA

Matsubara, H. & Hashida, K. (1989). Partiality of information and unsolvability of the frame problem. *Japanese Society for Artificial Intelligence*, Vol. 4, No. 6, pp. 695–703

Maturana, H. R. (1978). Biology of language – the epistemology of reality. In: *Psychology and Biology of Language and Thought – Essay in Honor of Eric Lenneberg*, Miller, G.A., Lenneberg, E., (Eds.), pp.27–64, Academic Press

Nakagawa, S. & Masukata, M. (1995). An acquisition system of concept and grammar based on combining with visual and auditory information. *Trans. Information Society of Japan*, Vol. 10, No. 4, pp. 129–137

Onishi, K. H. & Baillargeon, R. (2005). Do 15-month-old infants understand false beliefs? *Science*, Vol. 308, pp. 225–258

Pearl, J. (1988). Probabilistic reasoning in intelligent systems: Networks of Plausible Inference. Morgan Kaufmann

Persoon, E., Fu, K. S. (1977). Shape discrimination using Fourier descriptors. *IEEE Trans Systems, Man, and Cybernetics*, Vol, 7, No. 3, pp. 170–179

Regier, T. (1997). The Human Semantic Potential. MIT Press

Roy, D. (2005). Semiotic Schemas: A Framework for Grounding Language in Action and Perception. *Artificial Intelligence*, Vol. 167, Issues. 1-2, pp. 170-205

Roy, D. & Pentland, A. (2002). Learning Words from Sights and Sounds: A Computational Model. *Cognitive Science*, Vol. 26, No. 1, pp. 113-146

Shapiro, C. S.; Ismail, O.; Santore, J. F. (2000). Our dinner with Cassie. Proceedings of AAAI 2000 Spring Symposium on Natural Dialogues with Practical Robotic Devices, pp.57–61

Steels, L. (2003). Evolving grounded communication for robots. *Trends in Cognitive Science*, Vol. 7, No. 7, pp. 308–312

Steels, L. & Kaplan, K. (2001). Aibo's first words: the social learning of language and meaning. *Evolution of Communication*, Vol. 4, No. 1, pp.3–32

Savage-Rumbaugh, E. S. (1986). Ape Language – From Conditional Response to Symbol. Columbia Univ. Press

Sperber, D. & Wilson, D. (1995). Relevance (2nd Edition). Blackwell

Taguchi, R.; Iwahashi, N. & Nitta, T. (2007). Learning of question and answer reflecting scenes of the real world. *Tech. Rep. of Japanese Society of Artificial Intelligence*, SIG-SLUD-A603-04, pp. 15-20

Tokuda, K.; Kobayashi, T. & Imai, S. (1995). Speech parameter generation from HMM using dynamic features. In: Proceedings Int. Conf. on Acoustics, Speech and Signal Processing, pp. 660–663

Traum, D. R. (1994). A computational theory of grounding in natural language conversation. Doctoral dissertation, University of Rochester

Winograd, T. (1972). Understanding Natural Language. Academic Press New York

Wolpert, D. H. (1995). The relationship between PAC, the statistical physics framework, the Bayesian framework, and the VC framework. In Wolpert, D.H., ed.: The mathematics of Generalization, pp. 117-214, Addison-Wesley, Reading, MA

6

Recognizing Human Pose and Actions for Interactive Robots

Odest Chadwicke Jenkins[1], Germán González Serrano[2]
and Matthew M. Loper[1]
[1]Brown University, [2]Ecole Polytechnique Fédérale de Lausanne
[1]USA, [2]Switzerland

1. Introduction

Perceiving human motion and non-verbal cues is an important aspect of human-robot interaction (Fong et al., 2002). For robots to become functional collaborators in society, they must be able to make decisions based on their perception of human state. Additionally, knowledge about human state is crucial for robots to learn control policies from direct observation of humans. Human state, however, encompasses a large and diverse set of variables, including kinematic, affective, and goal-oriented information, which has proved difficult to model and infer. Part of this problem is that the relationship between such decision-related variables and a robot's sensor readings is difficult to infer directly.

Our greater view is that socially interactive robots will need to maintain estimates, or beliefs as probabilistic distributions, about all of the components in a human's state in order to make effective decisions during interaction. Humans make decisions to drive their muscles and affect their environment. A robot can only sense limited information about this control process. This information is often partial observations about the human's kinematics and appearance over time, such as images from a robot's camera. To estimate a human's decision making policy, a robot must attempt to invert this partial information back through its model of the human control loop, maintaining beliefs about kinematic movement, actions performed, decision policy, and intentionality.

As a step in this direction, we present a method for inferring a human's kinematic and action state from monocular vision. Our method works in a bottom-up fashion by using a vocabulary of predictive dynamical primitives, learned from previous work (Jenkins & Matarić, 2004a) as "action filters" working in parallel. Motion tracking is performed by matching predicted and observed human movement, using particle filtering (Isard & Blake 1998, Thrun et al., 2005) to maintain nonparametric probabilistic beliefs. For quickly performed motion without temporal coherence, we propose a "bending cone" distribution for extended prediction of human pose over larger intervals of time. State estimates from the action filters are then used to infer the linear coefficients for combining behaviours. Inspired by neuroscience, these composition coefficients are related to the human's cognitively planned motion, or "virtual trajectory", providing a compact action space for linking decision making with observed motion.

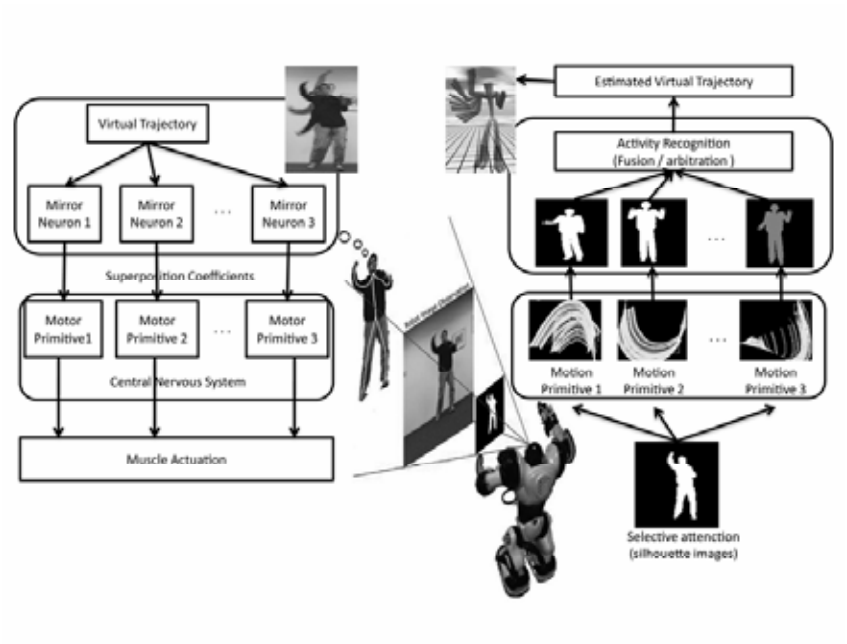


Figure 1. A "toy" example of our approach to human state estimation and movement imitation. The movement of a human demonstrator assumed to be generated by virtual trajectory executed as a weighted superposition of motor primitives, predictive low-dimensional dynamical systems. For movement imitation, a particle filter for each primitive performs kinematic state (or pose) estimation. Pose estimates across the vocabulary are fused at each timestep and concatenated over time to yield an estimate of the virtual trajectory for the robot to execute

We present results from evaluating our motion and action tracking system to human motion observed from a single robot camera. Presented results demonstrate our methods ability to track human motion and action, robust to performer speed and camera viewpoint with recovery from ambiguous situations, such as occlusion. A primary contribution of our work is interactive-time inference of human pose using sparse numbers of particles with dynamical predictions. We evaluate our prediction mechanism with respect to action classification over various numbers of particles and other prediction related variables.
Our work has a broad scope of applications, ranging from robot learning to healthcare. Due to the predictive nature of the motion primitives, the methodology presented can be used to modify control policies according to predicted human actions. The combination of motion primitives and human tracking can also be used in healthcare, analyzing the performance of a given activity by a patient, seeing how much it deviates from a "natural" or "standard" performance due to an illness. An example would be gait-analysis, or rehabilitation

programmes. The analysis of human performance could be extended to sports training, analysing how much a sportsman deviates from the canonical performance described by the motion primitive and how much does that affect his performance.
We highlight the application of our tracking results to humanoid imitation. These results allow us to drive a virtual character, which could be used in videogames or computer animation. The player would be tracked, and his kinematics would be adapted to the closest known pose in the motion primitives. This way, we could correct for imperfect player's performance.

2. Background

2.1 Motor Primitives and Imitation Learning

This work is inspired by the hypotheses from neuroscience pertaining to models of motor control and sensory-motor integration. We ground basic concepts for imitation learning, as described in (Matarić, 2002), in specific computational mechanisms for humanoids. Matarić's model of imitation consists of: 1) a selective attention mechanism for extraction of observable features from a sensory stream, 2) mirror neurons that map sensory observations into a motor repertoire, 3) a repertoire of motor primitives as a basis for expressing a broad span of movement, and 4) a classification-based learning system that constructs new motor skills.
Illustrated in Figure 1, the core of this imitation model is the existence and development of computational mechanisms for mirror neurons and motor primitives. As proposed by (Mussa-Ivaldi & Bizzi, 2000), motor primitives are used by the central nervous system to solve the inverse dynamics problem in biological motor control. This theory is based on an equilibrium point hypothesis. The dynamics of the plant $D(x,\dot{x},\ddot{x})$ is a linear combination of forces from a set of primitives, as configuration-dependent force fields (or attractors) $\phi_i(x,\dot{x},\ddot{x})$:

$$D(x,\dot{x},\ddot{x}) = \sum_{i=i}^{K} c_i \phi_i(x,\dot{x},\ddot{x}) \tag{1}$$

where $\boldsymbol{x}$ is the kinematic configuration of the plant, $\boldsymbol{c}$ is a vector of scalar superposition coefficients, and $\boldsymbol{K}$ is the number of primitives. A specific set of values for $\boldsymbol{c}$ produces stable movement to a particular equilibrium configuration. A sequence of equilibrium points specifies a virtual trajectory (Hogan, 1985) that can be use for control, as desired motion for internal motor actuation, or perception, to understand the observed movement of an external performer.
Matarić's imitation model assumes the firing of mirror neurons specifies the coefficients for formation of virtual trajectories. Mirror neurons in primates (Rizzolatti et al., 1996) have been demonstrated to fire when a particular activity is executed, observed, or imagined. Assuming 1-1 correspondence between primitives and mirror neurons, the scalar firing rate of a given mirror neuron is the superposition coefficient for its associated primitive during equilibrium point control.

2.2 Motion Modeling

While Matarić's model has desirable properties, there remain several challenges in its computational realization for autonomous robots that we attempt to address. Namely, what are the set of primitives and how are they parameterized? How do mirror neurons recognize motion indicative of a particular primitive? What computational operators should be used to compose primitives to express a broader span of motion?

Our previous work (Jenkins & Matarić 2004a) address these computational issues through the unsupervised learning of motion vocabularies, which we now utilize within probabilistic inference. Our approach is close in spirit to work by (Kojo et al., 2006), who define a "proto-symbol" space describing the space of possible motion. Monocular human tracking is then cast as localizing the appropriate action in the proto-symbol space describing the observed motion using divergence metrics. (Ijspeert et al., 2001) encode each primitive to describe the nonlinear dynamics of a specific trajectory with a discrete or rhythmic pattern generator. New trajectories are formed by learning superposition coefficients through reinforcement learning. While this approach to primitive-based control may be more biologically faithful, our method provides greater motion variability within each primitive and facilitates partially observed movement perception (such as monocular tracking) as well as control applications. Work proposed by (Bentivegna & Atkeson, 2001) and (Grupen et al., 1995; Platt et al., 2004) approach robot control through sequencing and/or superposition of manually crafted behaviors.

Recent efforts by (Knoop et al., 2006) perform monocular kinematic tracking using iterative closest point and the latest Swissranger depth sensing devices, capable of precise depth measurements. We have chosen instead to use the more ubiquitous passive camera devices and also avoid modeling detailed human geometry.

Many other approaches to data-driven motion modeling have been proposed in computer vision, animation, and robotics. The reader is referred to other papers (Jenkins & Matarić, 2004a; Urtasun et al., 2005; Kovar & Gleicher, 2004; Elgammal A. M. and Lee Ch. S. 2004) for broader coverage of these methods.

2.3 Monocular Tracking

We pay particular attention to methods using motion models for kinematic tracking and action recognition in interactive-time. Particle filtering (Isard & Blake, 1998; Thrun et al., 2005) is a well established means for inferring kinematic pose from image observations. Yet, particle filtering often requires additional (often overly expensive) procedures, such as annealing (Deutscher et al., 2000), nonparametric belief propagation (Sigal et al., 2004; Sudderth et al., 2003), Gaussian process latent variable models (Urtasun et al., 2005), POMDP learning (Darrell & Pentland, 1996) or dynamic programming (Ramanan & Forsyth, 2003), to account for the high dimensionality and local extrema of kinematic joint angle space. These methods tradeoff real-time performance for greater inference accuracy. This speed-accuracy contrast is most notably seen in how we use our learned motion primitives (Jenkins & Matarić, 2004a) as compared to Gaussian process methods (Urtasun et al., 2005; Wang et al., 2005). Both approaches use motion capture as probabilistic priors on pose and dynamics. However, our method emphasizes temporally extended prediction to use fewer particles and enable fast inference, whereas Gaussian process models aim for accuracy through optimization. Further, unlike the single-action motion-sparse experiments with Gaussian process models, our work is capable of

inference of multiple actions, where each action has dense collections of motion. Such actions are generalized versions of the original training motion and allow us to track new variations of similar actions.

Similar to (Huber & Kortenkamp, 1998), our method aims for interactive-time inference on actions to enable incorporation into a robot control loop. Unlike (Huber and Kortenkamp, 1998), however, we focus on recognizing active motions, rather than static poses, robust to occlusion by developing fast action prediction procedures that enable online probabilistic inference. We also strive for robustness to motion speed by enabling extended look-ahead motion predictions using a "bending cone" distribution for dynamics. (Yang et al., 2007) define a discrete version with similar dynamics using Hidden Markov Model and vector quantization observations. However, such HMM-based transition dynamics are instantaneous with a limited prediction horizon, whereas our bending cone allows for further look-ahead with a soft probability distribution.

3. Dynamical Kinematic and Action Tracking

Kinematic tracking from silhouettes is performed via the steps in Figure 2, those are: 1) global localization of the human in the image, 2) primitive-based kinematic pose estimation and 3) action recognition. The human localization is kept as an unimodal distribution and estimated using the joint angle configuration derived in the previous time step.

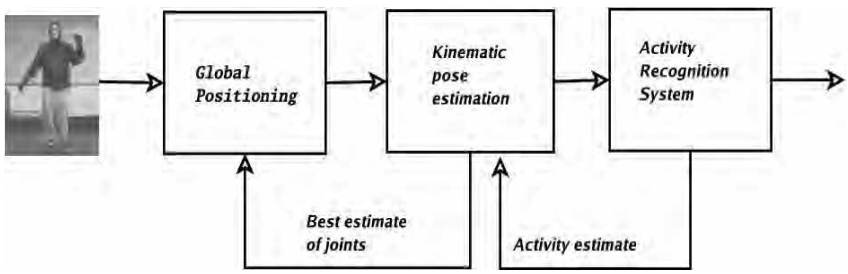


Figure 2. Illustration of the three stages in our approach to tracking: image observations are used to localize the person in 3D, then infer kinematic pose, and finally estimate of activity/action. Estimates at each stage are used to form priors for the previous stage at the next timestep

3.1 Dynamical Motion Vocabularies

The methodology of (Jenkins & Matarić, 2004a) is followed for learning dynamical vocabularies from human motion. We cover relevant details from this work and refer the reader to the citation for details. Motion capture data representative of natural human performance is used as input for the system. The data is partitioned into an ordered set of non-overlapping segments representative of "atomic" movements. Spatio-temporal Isomap (Jenkins & Matarić, 2004b) embed these motion trajectories into a lower dimensional space, establishing a separable clustering of movements into activities. Similar to (Rose et al., 1998),

each cluster is a group of motion examples that can be interpolated to produce new motion representative of the underlying action. Each cluster is speculatively evaluated to produce a dense collection of examples for each uncovered action. A primitive $\boldsymbol{B_i}$ is the manifold formed by the dense collections of poses $\boldsymbol{X_i}$ (and associated gradients) in joint angle space resulting from this interpolation.

We define each primitive $\boldsymbol{B_i}$ as a gradient (potential) field expressing the expected kinematic behaviour over time of the $\boldsymbol{i^{th}}$ action. In the context of dynamical systems, this gradient field $\boldsymbol{B_i(x)}$ defines the predicted direction of displacement for a location in joint angle space $\hat{x}[t]$ at time $\boldsymbol{t}$[1]:

$$\hat{x}_i[t+1] = f_i(x[t], u[t]) = u[t]B_i(x) = u[t]\frac{\sum_{x \in nbhd(x[t])} w_x \Delta_x}{\left\| \sum_{x \in nbhd(x[t])} w_x \Delta_x \right\|} \tag{2}$$

where $\boldsymbol{u[t]}$ is a fixed displacement magnitude, $\boldsymbol{\Delta_x}$ is the gradient of pose $\boldsymbol{x}$ [2], a motion example of primitive $\boldsymbol{i}$, and $\boldsymbol{w_x}$ the weight[3] of $\boldsymbol{x}$ with respect to $\boldsymbol{x[t]}$. Figure 3 shows examples of learned predictive primitives.

Given results in motion latent space dimensionality (Urtasun et al., 2005; Jenkins & Matarić, 2004b), we construct a low dimensional latent space to provide parsimonious observables $\boldsymbol{y_i}$ of the joint angle space for primitive $\boldsymbol{i}$. This latent space is constructed by applying Principal Components Analysis (PCA) to all of the poses $\boldsymbol{X_i}$ comprising primitive $\boldsymbol{i}$ and form the output equation of the dynamical system, such as in (Howe et al., 2000):

$$y_i[t] = g_i(x[t]) = A_i x[t] \tag{3}$$

Given the preservation of variance in $\boldsymbol{A_i}$, it is assumed that latent space dynamics, governed by $\bar{f}_i$, can be computed in the same manner as $\boldsymbol{f_i}$ in joint angle space:

$$\frac{g_i^{-1}(\bar{f}_i(g_i(x[t]), u[t])) - x[t]}{\left\| g_i^{-1}(\bar{f}_i(g_i(x[t]), u[t])) - x[t] \right\|} \approx \frac{f_i(x[t], u[t]) - x[t]}{\left\| f_i(x[t], u[t]) - x[t] \right\|} \tag{4}$$

---

[1] nbhd() is used to identify the k-nearest neighbours in an arbitrary coordinate space, which we use both in joint angle space and the space of motion segments.

[2] The gradient is computed as the direction between $\boldsymbol{y}$ and its subsequent pose along its motion example.

[3] Typically reciprocated Euclidean distance

3.2 Kinematic Pose Estimation

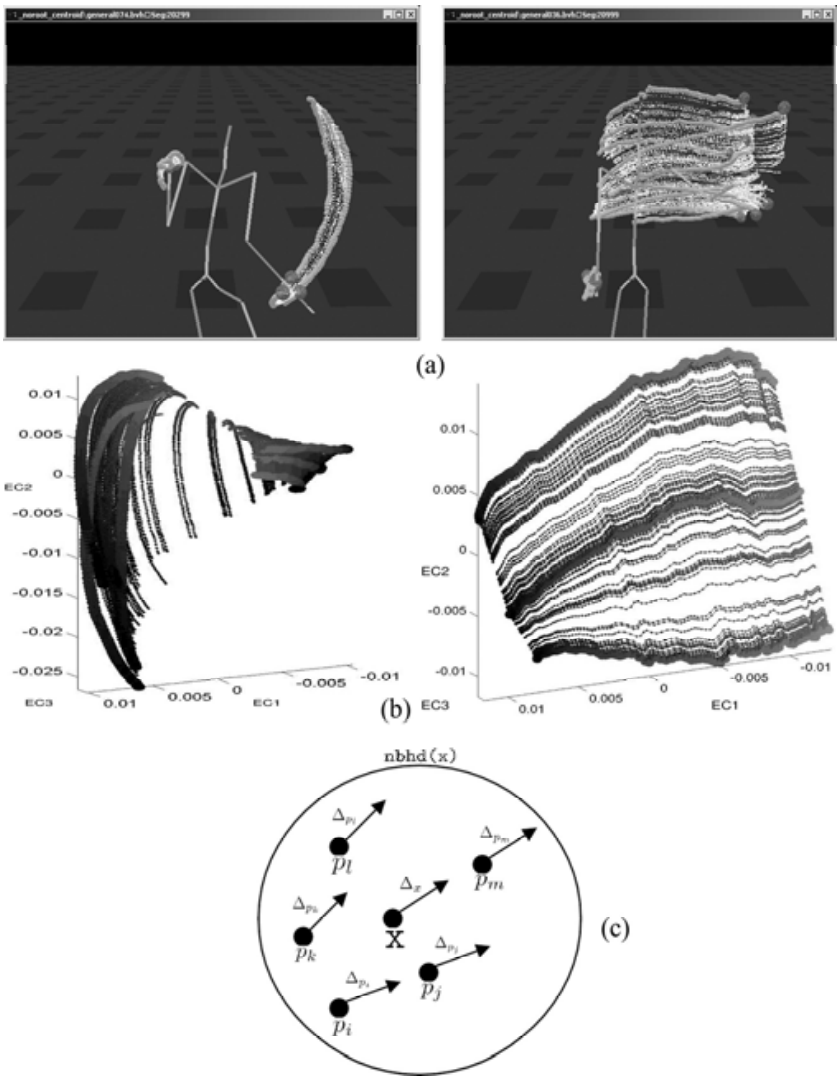


Figure 3. (a) Kinematic endpoint trajectories for learned primitive manifolds, (b) corresponding joint angle space primitive manifolds (view from first three principal components), and (c) an instantaneous prediction example (illustrated as a zoomed-in view on a primitive manifold)

Kinematic tracking is performed by particle filtering (Isard & Blake,1998; Thrun et al., 2005) in the individual latent spaces created for each primitive in a motion vocabulary. We infer with each primitive individually and in parallel to avoid high-dimensional state spaces, encountered in (Deutscher et al., 2000). A particle filter of the following form is instantiated in the latent space of each primitive

$$p(y_i[1:t] \mid z_i[1:t]) \propto p(z[t] \mid g_i^{-1}(y_i[t])) \sum_{y_i} p(y_i[t] \mid y_i[t-1]) p(y_i[1:t-1] \mid z[1:t-1]) \quad (5)$$

where $z_i[t]$ are the observed sensory features at time t and g_i^{-1} is the transformation into joint angle space from the latent space of primitive i.
The likelihood function $p(z[t] \mid g_i^{-1}(y_i[t]))$ can be any reasonable choice for comparing the hypothesized observations from a latent space particle and the sensor observations. Ideally, this function will be monotonic with discrepancy in the joint angle space.

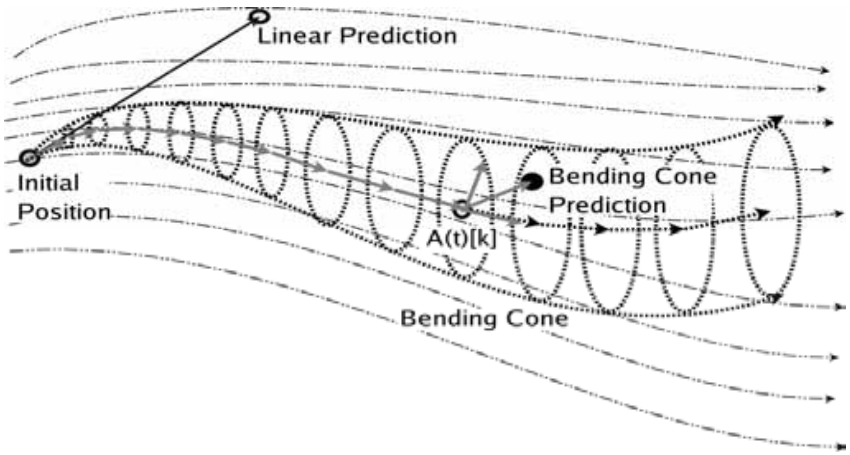


Figure 4. Illustration of the predictive bending cone distribution. The thin dashed black lines indicate the flow of a primitive's gradient field. Linear prediction from the current pose $y_i(t)$ will lead to divergence from the gradient field as the prediction magnitude increases. Instead, we use a bending cone (in bold) to provide an extended prediction horizon along the gradient field. Sampling a pose prediction $y_i(t+1)$ occurs by selecting a cross-section $A(t)[k]$ and adding cylindrical noise

At first glance, the motion distribution $p(z[t] \mid g_i^{-1}(y_i[t]))$ could be given by the instantaneous "flow", as proposed by (Ong et al., 2006), where a locally linear displacement with some noise is expected. However, such an assumption would require temporal coherence between the training set and the performance of the actor. Observations without temporal coherence cannot simply be accounted for by extending the magnitude of the displacement

vector because the expected motion will likely vary in a nonlinear fashion over time. To address this issue, a "bending cone" distribution is used (Figure 4) over the motion model. This distribution is formed with the structure of a generalized cylinder with a curved axis along the motion manifold and a variance cross-section that expands over time. The axis is derived from ***K*** successive predictions $\bar{y}_i[t]$ of the primitive from a current hypothesis $\boldsymbol{y_i[t]}$ as a piecewise linear curve. The cross-section is modelled as cylindrical noise $\boldsymbol{C(a,b,\sigma)}$ with local axis ***a-b*** and normally distributed variance $\boldsymbol{\sigma}$ orthogonal to the axis.

The resulting parametric distribution, equation 6, is sampled by randomly selecting a step-ahead ***k*** and generating a random sample within its cylinder cross-section. Note that ***f(k)*** is some monotonically increasing function of the distance from the cone origin; we used a linear function.

$$p(y_i[t] \mid y_i[t-1]) = \sum_{\bar{y}_i[t]}^{k} C(\bar{y}_i[k+1], \bar{y}_i[k], f(k)) \tag{6}$$

3.3 Action Recognition

For action recognition, a probability distribution across primitives of the vocabulary is created[4]. The likelihood of the pose estimate from each primitive is normalized into a probability distribution:

$$p(B_i[t] \mid z[t]) = \frac{p(z[t] \mid \bar{x}_i[t])}{\sum_B p(z[t] \mid \bar{x}_i[t])} \tag{7}$$

where $\bar{x}_i[t]$ is the pose estimate for primitive ***i***. The primitive with the maximum probability is estimated as the action currently being performed. Temporal information can be used to improve this recognition mechanism by fully leveraging the latent space dynamics over time.

The manifold in latent space is essentially an attractor along a family of trajectories towards an equilibrium region. We consider *attractor progress* as a value that increases as kinematic state progresses towards a primitive's equilibrium. For an action being performed, we expect its attractor progress will monotonically increase as the action is executed. The attractor progress can be used as a feedback signal into the particle filters estimating pose for a primitive ***i*** in a form such as:

$$p(B_i[t] \mid z[t]) = \frac{p(z[t] \mid \bar{x}_i[t], w_i[1:t-1])}{\sum_B p(z[t] \mid \bar{x}_i[t], w_i[1:t-1])} \tag{8}$$

where $\boldsymbol{w_i[1:t-1]}$ is the probability that primitive $\boldsymbol{B_i}$ has been performed over time.

[4] We assume each primitive describes an action of interest.

Figure 5. Robot platform and camera used in our experiments

4. Results

For our experiments, we developed an interactive-time software system in C++ that tracks human motion and action from monocular silhouettes using a vocabulary of learned motion primitives. Shown in Figure 5, our system takes video input from a Fire-i webcam (15 frames per second, at a resolution of 120x160) mounted on an iRobot Roomba Discovery. Image silhouettes were computed with standard background modelling techniques for pixel statistics on colour images. Median and morphological filtering were used to remove noisy silhouette pixels. An implementation of spatio-temporal Isomap (Jenkins & Matarić, 2004b) was used to learn motion primitives for performing punching, hand circles, vertical hand waving, and horizontal hand waving.

We utilize a basic likelihood function, $\boldsymbol{p(z[t] \mid g_i^{-1}(y_i[t]))}$, that returns the similarity $\boldsymbol{R(A,B)}$ of a particle's hypothesized silhouette with the observed silhouette image. Silhouette hypotheses were rendered from a cylindrical 3D body model to an binary image buffer using OpenGL. A similarity metric, $\boldsymbol{R(A,B)}$ for two silhouettes $\boldsymbol{A}$ and $\boldsymbol{B}$, closely related to the inverse of the Generalized Hausdorff distance was used:

$$R(A,B) = \frac{1}{r(A,B) + r(B,A) + \varepsilon} \tag{9}$$

$$r(A,B) = \sum_{a \in A} \left(\min_{b \in B} \|a - b\| \right)^2 \tag{10}$$

This measure is an intermediate between undirected and generalized Hausdorff distance. ε is used only to avoid divide-by-zero errors. An example Hausdorff map for a human

silhouette is shown in Figure 6. Due to this silhouetting procedure, the robot must be stationary (i.e., driven to a specific location) during the construction of the background model and tracking process. As we are exploring in future work, this limitation could be relaxed through the use of other sensor modalities, such as stereo vision or time-of-flight ranging cameras.

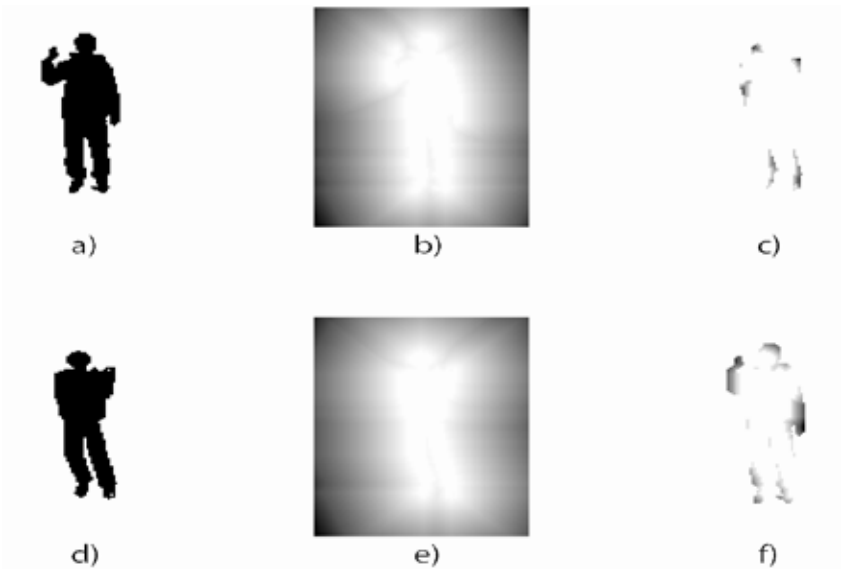


Figure 6. Likelihood function used in the system. (a) is the silhouette *A* extracted from the camera and (d) is the synthetic silhouette *B* generated from a pose hypothesis. (b) and (e) are the respective Hausdorff distance transforms, showing pixels with larger distances from the silhouette as dark. (c) and (f) illustrate the sums *r(A,B)*, how silhouette *A* relates *B*, and *r(B,A)*, silhouette *B* relates to *A*. These sums are added and reciprocated to assess the similarity of *A* and *B*

To enable fast monocular tracking, we applied our system with sparse distributions (6 particles per primitive) to three trial silhouette sequences. Each trial is designed to provide insight into different aspects of the performance of our tracking system.
In the first trial (termed multi-action), the actor performs multiple repetitions of three actions (hand circles, vertical hand waving, and horizontal hand waving) in sequence. As shown in Figures 7, reasonable tracking estimates can be generated from as few as six particles. As expected, we observed that the Euclidean distance between our estimates and the ground truth decreases with the number of particles used in the simulation, highlighting the tradeoffs between the number of particles and accuracy of the estimation.

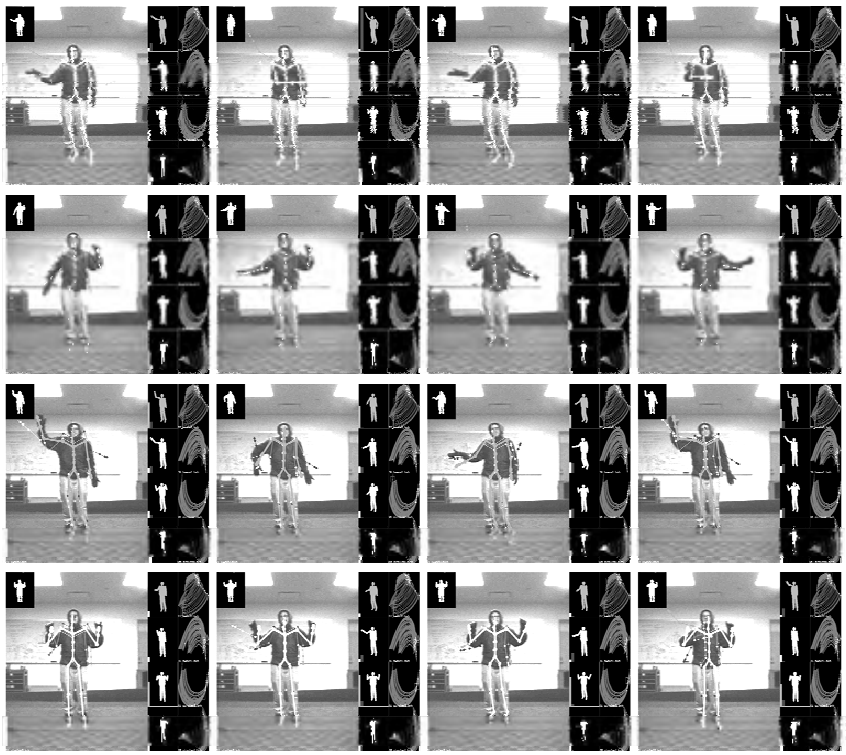


Figure 7. Tracking of a motion sequence containing three distinct actions performed in sequence without stopping. Wach row shows the recognition of individual actions for waving a hand across the body (top row), bottom-to-top in a circular fashion (second and fourth row) and top-to-bottom (third row). The kinematic estimates are shown with a thick-lined stick figure; the color of the stick figures represents the action recognized. Each image contains a visualization of the dynamical systems and pose estimates for each action

To explore the effects of the number of particles, we ran our tracking system on the multi-action trial using powers-of-two number particles between 1 and 1024 for each action. The bending cone for these trials are generated using 20 predictions into the future and the noise aperture is $\Pi/6$, which increases in steps of $20\Pi/6$ per prediction. Shown in Figure 8, the action classification results from the system for each trial were plotted in the ROC plane for each action. The ROC plane plots each trial (shown as a labelled dot) in 2D coordinates where the horizontal axis is the "false positive rate", percentage of frames incorrectly labelled as a given action, and the vertical axis is the "true positive rate", percentage of correctly labelled frames. Visually, plots in the upper left-hand corner are indicative of good performance, points in the lower right-hand corner indicate bad performance, and points along the diagonal indicate random performance.

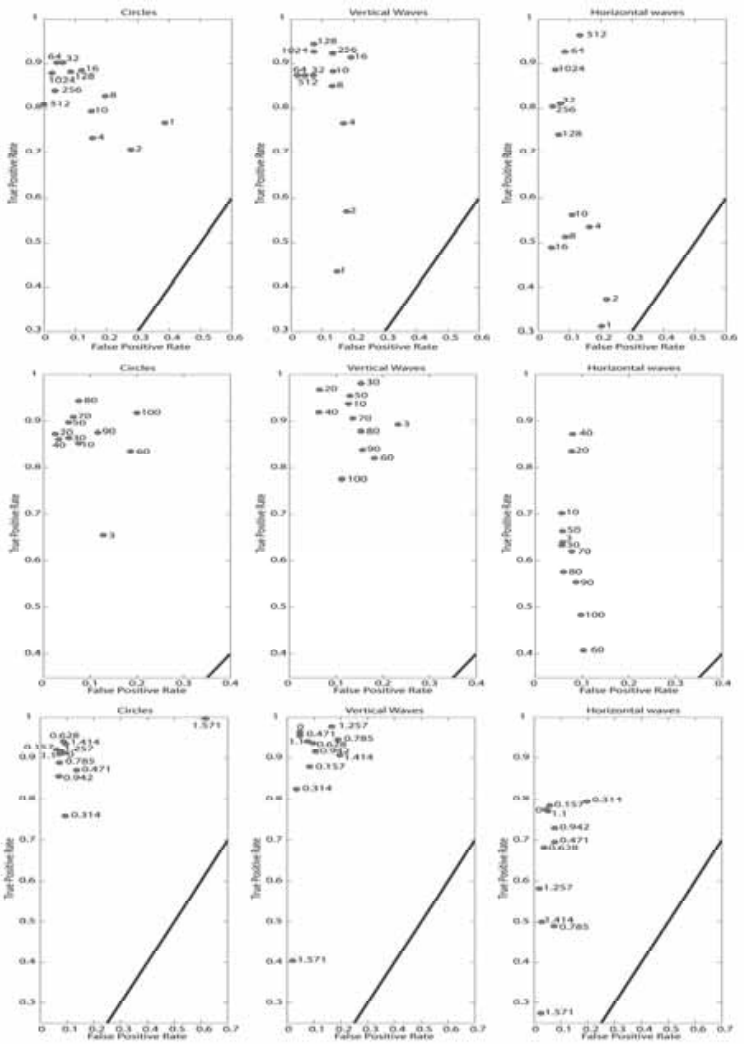


Figure 8. ROC plots of action classification on the multi-action sequence. Columns breakdown by the action performed: Circle action (left), Vertical Waving action (center), and Horizontal Waving action (right). Columns show the effect of varied numbers of particles (top, varied between 1 and 1024), bending cone prediction length (middle, varied between 3 and 100), and bending cone noise aperture (bottom, varied between 0 and $\Pi/2$). Each plot shows for each trial (plotted as a labeled point) the false positive rate (horizontal axis) and true positive rate (vertical axis). Note the difference in scales among rows

Figure 8. Tracking of a fast waving motion. Observed images (top) and pose estimates from the camera view (bottom)

The ROC plots for the numbers of particles indicate the classifier works better with more particles, but not always. Trials using 1024 particles per action are never the closest point to the upper-left corner. The most noticeable benefit to having more particles is when occlusion was present. In particular, the Circle action does not introduce occlusion when performed in profile, where as the Horizontal Waving motion does require occlusion in its performance. Consequently, the Circle trials all plot near the ROC upper-left corner and the Horizontal Waving ROC trials fork bi-modally in classification performance.

ROC plots were also generated for varying bending cone prediction length and noise aperture, also shown in Figure 8. Plotted in the middle row, the variations in bending cone length were varied between 10 and 100 predictions in increments of 10, with an additional trial using 3 predictions. For these trials, the number of particles was fixed to 64 and the noise aperture to $\Pi/6$. Plotted in the bottom row, the variation in the bending cone noise aperture were varied between 0 and $\Pi/2$ in increments of $0.1*\Pi/2$. The number of particles and bending cone length were fixed at 64 and 20, respectively. These plots indicate variations similar to those in the numbers of particles plots. Specifically, good performance results regardless of the bending cone parameters when the action has little or no occlusion ambiguity, but performance drops off when such ambiguity is present. However, in these trials, increased prediction length and noise aperture does not necessarily improve performance. It is surprising that with 0 aperture (that is, staying fixed in the manifold), the classifier does not perform that bad. Instead, there are sweet spots between 20-40 predictions and under 0.15Π noise aperture for the bending cone parameters. Although including more particles is always increase accuracy, we have not yet explored how the sweet spots in the bending cone parameters could change as the numbers of particles vary.

In trial two (fast-wave motion), we analyzed the temporal robustness of the tracking system. The same action is performed at different speeds, ranging from slow (hand moving at ~3 cm/s) to fast motion (hand moving at ~6 m/s). The fast motion is accurately predicted as seen in Figure 9. Additionally, we were able to track a fast moving punching motion (Figure 10) and successfully execute the motion with our physics-based humanoid simulation. Our simulation system is described in (Wrotek et al., 2006).

In trial three (overhead-view), viewpoint invariance was tested with video from a trial with an overhead camera, shown in Figure 11. Even given limited cues from the silhouette, we are able to infer the horizontal waving of an arm. Notice that the arm estimates are consistent throughout the sequence.

Figure 9. Illustrations of a demonstrated fast moving ``punch'' movement (left) and the estimated virtual trajectory (right) as traversed by our physically simulated humanoid simulation

Using the above test trials, we measured the ability of our system to recognize performed actions to provide responses similar to mirror neurons. In our current system, an action is recognized as the pose estimate likelihoods normalized over all of the primitives into a probability distribution, as shown in Figure 12.

Temporal information can be used to improve this recognition mechanism by fully leveraging the latent space dynamics over time. The manifold in latent space is essentially an attractor along a family of trajectories. A better estimator of action would consider *attractor progress*, monotonic progress towards the equilibrium region of an action's gradient

field. We have analyzed preliminary results from observing attractor progress in our trials, as shown in Figure 12. For an action being performed, its attractor progress is monotonically increasing. If the action is performed repeatedly, we can see a periodic signal emerge, as opposed to the noisier signals of the action not being performed. These results indicate that we can use attractor progress as a feedback signal to further improve an individual primitive's tracking performance

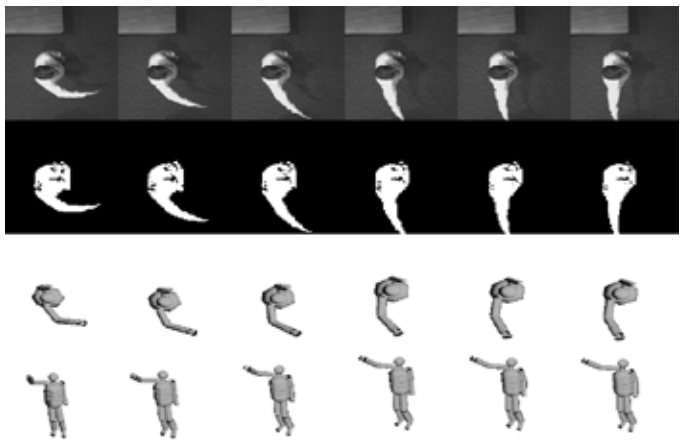


Figure 10. Illustrations of a demonstrated fast moving ``punch'' movement (left) and the estimated virtual trajectory (right) as traversed by our physically simulated humanoid simulation

Because of their attractor progress properties, we believe that we can analogize these action patterns into the firing of idealized mirror neurons. The firings of our artificial mirror neurons provide superposition coefficients, as in (Nicolescu et al., 2006). Given real-time pose estimation, online movement imitation could be performed by directly executing the robot's motor primitives weighted by these coefficients. Additionally, these superposition coefficients could serve as input into additional inference systems to estimate the human's emotional state for providing an affective robot response.

In our current system, we use the action firing to arbitrate between pose estimates for forming a virtual trajectory. While this is a simplification of the overall goal, our positive results for trajectory estimation demonstrate our approach is viable and has promise for achieving our greater objectives. As future work, we will extend the motion dynamics of the vocabulary into basis behaviours using our complementary work in learning behaviour fusion (Nicolescu et al., 2006).

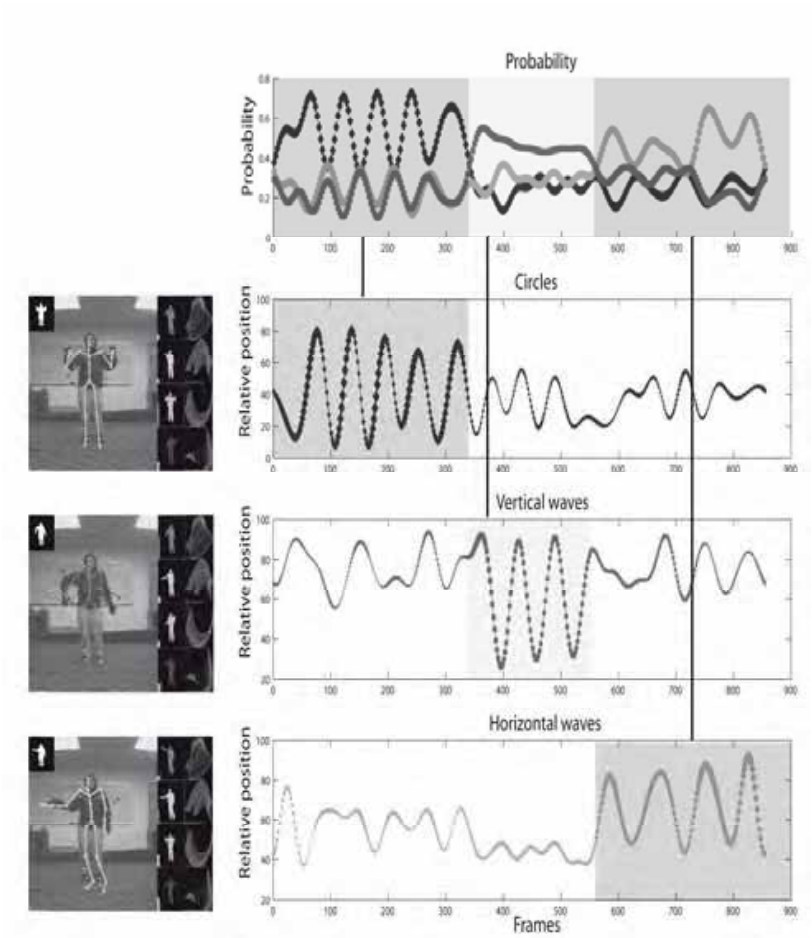


Figure 11. An evaluation of our action recognition system over time with a 3-action motion performing ``hand circles'', ``horizontal waving'', and ``vertical waving'' in sequence. The first row reflects the relative likelihood (idealized as mirror neuron firing) for each primitive with background sections indicating the boundary of each action. Each of the subsequent rows shows time on the x-axis, attractor progress on the y-axis, and the width of the plot marker indicates the likelihood of the pose estimate

5. Conclusion

We have presented a neuro-inspired method for monocular tracking and action recognition for movement imitation. Our approach combines vocabularies of kinematic motion learned offline with online estimation of a demonstrator's underlying virtual trajectory. A modular approach to pose estimation is taken for computational tractability and emulation of structures hypothesized in neuroscience. Our current results suggest our method can perform tracking and recognition from partial observations at interactive rates. Our current system demonstrates robustness with respect to the viewpoint of the camera, the speed of performance of the action, and recovery from ambiguous situations.

6. References

Bentivegna, D. C. and Atkeson, C. G. (2001). Learning from observation using primitives. *In IEEE International Conference on Robotics and Automation*, pp 1988–1993, Seoul, Korea, May 2001, IEEE.

Darrell, T. and Pentland, A. (1996). Active gesture recognition using learned visual attention. *Advances in Neural Information Processing Systems*, 8, pp 858–864, 0-262-20107-0, Denver, CO, USA, November 1995, The MIT Press.

Deutscher J.; Blake, A. and Reid, I. (2000). Articulated body motion capture by annealed particle filtering. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2, pp 126–133, 0-7695-0662-3, Hilton Head, SC, USA, June 2000, IEEE Computer Society.

Elgammal A. M. and Lee Ch. S. (2004). Inferring 3D body pose from silhouettes using activity manifold learning. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2, pp 681-688, 0-7695-0662-3, Washington DC, USA, July 2004, IEEE Computer Society.

Fong, T.; Nourkbaksh, I. and Daoutenhahn, K. (2002). A survey of socially interactive robots: Concepts, design and applications. Carnegie Mellon University Robotics Institute, Pittsburgh, PA Tech. Rep CMU-RI-TR02-29, November 2002.

Grupen, R. A.; Huber, M.; Coehlo Jr. J. A.; and Souccar, K. (1995). A basis for distributed control of manipulation tasks. *IEEE Expert,* 10, 2, (April 1995), pp. 9–14.

Hogan, N. (1985) The mechanics of multi-joint posture and movement control. *Biological Cybernetics*, 52, (September 1985), pp. 315–331, 0340-1200.

Howe, N. R.; Leventon, M. E. and Freeman, W. T. (2000). Bayesian reconstruction of 3D human motion from single-camera video. *Advances In Neural Information Processing Systems, 12*, Denver, CO, USA, 2000, The MIT Press.

Huber, E. and Kortenkamp, D. (1998). A behavior-based approach to active stereo vision for mobile robots. *Engineering Applications of Artificial Intelligence*, 11, (December 1998), pp. 229–243, 0952-1976.

Ijspeert, A. J.; Nakanishi, J. and Schaal, S.(2001). Trajectory formation for imitation with non-linear dynamical systems. *In IEEE Intelligent Robots and Systems*, pp 752–757, Maui, Hawaii, USA, October 2001, IEEE.

Isard, M. and Blake, A. (1998). Condensation – conditional density propagation for visual tracking, *International Journal of Computer Vision*, 29, 1, (August 1998) , pp. 5-28, 0920-5691.

Jenkins, O. C. and Matarić, M. J. (2004a). Performance-derived behavior vocabularies: Data-driven acqusition of skills from motion. *International Journal of Humanoid Robotics*, 1, 2, (June 2004) 237–288, 0219-8436.

Jenkins, O. C. and Matarić, M. J. (2004b). A spatio-temporal extension to isomap non-linear dimension reduction, *Proceedings of the International Conference on Machine Learning (ICML)*, pp. 441–448, Banff, Alberta, Canada, July 2004, Omnipress, Madison, WI, USA.

Knoop, S.; Vacek, S. and Dillmann, R. (2006). Sensor fusion for 3D human body tracking with an articulated 3D body model. *In IEEE International Conference on Robotics and Automation*, pp 1686-1691, Orlando, FL, USA, May 2006, IEEE.

Kojo, N.; Inamura, T.; Okada, K. and Inaba, M.(2006). Gesture recognition for humanoids using proto-symbol space. *Proceedings of the IEEE International Conference on Humanoid Robotics.* pp 76-81, Genova, Italy, December 2006, IEEE,

Kovar, L. and Gleicher, M.(2004). Automated extraction and parameterization of motions in large data sets. *International Conference on Computer Graphics and Interactive Techniques, ACM Siggraph 2004*, pp 559–568, 0730-0301,Los Angeles, California, USA, 2004.

Matarić, M. J. (2002). Sensory-motor primitives as a basis for imitation: Linking perception to action and biology to robotics. *Imitation in Animals and Artifacts.* MIT Press, 0-262-04203-7, Cambridge, Massachussetts, USA.

Mussa-Ivaldi, F. and Bizzi, E.(2000). Motor learning through the combination of primitives. *Philosophical Transactions of the Royal Society: B: Biological Sciences*. 355, pp. 1755-1769 London, UK.

Nicolescu, M.; Jenkins, O. C., and Olenderski A.(2006). Learning behavior fusion estimation from demonstration. *In IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN 2006)*, pp. 340-345, Hatfield, United Kingdom, September 2006, IEEE Computer Society.

Ong, E.; Hilton, A. and Micilotta, A. (2006). Viewpoint invariant exemplar-based 3D human tracking. *Computer Vision and Image Understanding*, 104, 2, (November 2006), pp 178-189, ISSN:1077-3142.

Platt R.; Fagg, A. H. and Grupen, R. R. (2004) Manipulation gaits: Sequences of grasp control tasks. *In IEEE Conference on Robotics and Automation*, pp 801–806, New Orleans, LA, USA, April 2004, IEEE.

Sigal, L.; Bhatia, S.; Roth, S.; Black, M. J. and Isard, M. (2004). Tracking loose-limbed people. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp 421–428, 0-7695-2158-4, Washington, USA, July 2004., IEEE Computer Society .

Sudderth, E. B.; Ihler, A. T.; Freeman, W. T. and Willsky, A. S. (2003). Nonparametric belief propagation. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp 605–612, 0-7695-1900-8, Madison, WI, USA, June 2003, IEEE Computer Society.

Ramanan, D. and Forsyth, D. A. (2003). Automatic annotation of everyday movements. *Advances in Neural Information Processing Systems,16, 0-262-20152-6* , Vancouver, Canada, 2003, The MIT Press.

Rizzolatti, G.; Gadiga, L.; Gallese, V. and Fogassi, L. (1996). Premotor cortex and the recognition of motora ctions. *Cognitive Brain Research*, 3, 2, (March 1996), pp 131–141, 0006-8993.

Rose, C.; Cohen, M. F. and Bodenheimer, B. (1998). Verbs and adverbs: Multidimensional motion interpolation. *IEEE Computer Graphics & Applications*, 18, 5, (September-October 1998), pp. 32–40, 0272-1716.

Thrun, S.; Burgard, W. and Fox, D. (2005). *Probabilistic Robotics*. MIT Press, 0-262-20162-3, Cambridge, Massachusetts, USA.

Urtasun, R.; Fleet, D. J.; Hertzmann, A. and Fua, P. (2005). Priors for people tracking from small training sets. *In International Conference in Computer Vision*, pp 403-410, 0-7695-2334-X, Beijing, China, October 2005, IEEE Computer Society.

Wang, J.; Fleet, D. J. and Hertzmann, A. (2005). Gaussian Process Dynamical Models. *Advances in Neural Information Processing Systems, 18*, Vancouver, Canada, December 2005, The MIT Press.

Wrotek, P.; Jenkins, O. C. and McGuire, M. (2006). Dynamo: Dynamic data-driven character control with adjustable balance. *Proceedings of the Sandbox Symposium on Video Games*, Boston, MA, USA, July 2006, ACM.

Yang, H. D.; Park, A. Y. and Lee, S. W. (2007). Gesture Spotting and Recognition for Human-Robot Interaction. *IEEE transactions on Robotics*, 23, (April 2007), pp 256-270, 1552-3098.

7. Acknowledgments

This research was supported by grants from the Office of Naval Research (Award N000140710141) and National Science Foundation (Award IIS-0534858). The authors are grateful to Chris Jones and iRobot Corporation for support with the Roomba platform, Brian Gerkey the Player/Stage Project, RoboDynamics Corporation for Roomba interfaces, and Prof. Rod Beresford.

7

Development of Service Robot System With Multiple Human User Interface

Songmin Jia and Kunikatsu Takase
University of Electro-Communications, Tokyo
Japan

1. Introduction

Interactive human user interfaces are indispensability because robot has not enough capability in recognition and judgment in performing a service task at facilities or at home. We have been developing a network distributed Human-Assistance Robotic System in order to improve care cost and the QoL (Quality of Life) of elderly people in the population-aging society. Many elderly persons with some chronic disease give up independent living because of difficulty in moving their body to take something such as operating objects in a refrigerator. Implementation of human-assist robotic system enables elderly or disabled people in need for support and care to live independently in their accustomed home environments as long as they wish. This not only fulfils their desire for independence and autonomy, it also helps to improve the costs for the individual treatment. Many service robotic systems have been developed for the Web (World Wide Web) in recent past, as the Internet is low cost and widely available. Schulz et al. (Schulz et al, 2000) and Maeyama (Maeyama et al., 2000) developed museum tour-guide robotic systems that enable ordinary people at home or some other place to remotely view works of art in a museum by manipulating the vision of the robot using a Web browser. Lung N. et al developed an Internet-based robotic system that allows the user to control a robot arm with five degrees of freedom in performing the tedious household task of sorting laundry (Nagi et al., 2002). Coristine et al. developed PumaPaint Project that is an online robot that allows World Wide Web users to remotely create original artwork. This site had thousands of users who consider it entertaining (Coristine et al., 2004; Stein et al., 2000). Our HARSP (Human-Assistance Robotic System Project) project consists on developing a Human-Assistance Robotic distributed System in order to improve care cost and the QoL of the elderly people in the population-aging society (Jia et al., 2002). The proposed system has the potential to provide elderly persons local services in:

a) Intelligent reminding: remind elderly persons about important activities such as taking medical, eating meals, visiting the bathroom, or scheduling medical appointments.
b) Data collection and surveillance: robot can assist the aged or disabled in systematic data collection. Robotic systems may be soon able to inform human caregivers for assistance if they detect that an elderly person has fallen or the other emergency.
c) Daily services: many elderly persons with some chronic disease give up independent living because of difficulty in moving their body to take something such as operating

objects in refrigerators. A mobile robot integrating with a skilful robot arm could help the aged or disabled overcome these barriers and supply them necessary daily services. They can supply meal support, medical support and delivery support.

d) Mobility aid: support the aged or disabled for getting up from the bed or a chair, and implement intelligent walking aid.

We developed multi-robot, and implemented several CORBA application servers, which enabled the user to control the system by using Web browser. In the formerly developed system, the iGPS (indoor Global Positioning System) has been developed to localize an omnidirectional mobile robot (Hada et al., 2001). In this paper, a novel method of localization of mobile robot with a camera and RFID (Radio Frequency Identification) technology is proposed as it is inexpensive, flexible and easy to use in the practical environment. The information of obstacles or environment such as size, colour,, world coordinates can be written in ID tags in advance, which helps mobile robot recognize the obstacle or localization easily and quickly compared with the other method. When the working domain of mobile robot is changed or extended, what needs to be done is just putting the new ID (Identification) tags in new environment and registering these ID tags to database. It is also easy to improve dynamic obstacles recognition (such as chair or person) and occlusion problem that are very difficult to solve for the other system, because the communication between ID Reader and ID Tags uses Radio Frequency. A video/audio conference system is also developed to improve the interaction among the users, switch robot manipulating privilege with the help of a centralized user management server, and enable web-user to get a better understanding of what is going on in the local environment. Considering multi-type user of the developed system, we have implemented multi-type HRI that enable different user to control robot systems easily. Implementation of our developed system enables elderly or disabled people in need for support and care to live independently in their accustomed home environments as long as they wish. A mobile robot integrating with a skilful robot arm could help the aged or disabled overcome these barriers and supply them necessary daily services. This not only fulfils their desire for independence and autonomy, it also improves the problem of the high costs for the individual treatment in nursing homes.

The rest of the paper consists of 6 sections. Section 2 presents the structure of the multi-robot system. Section 3 introduces the proposed method of localization of mobile robot. Section 4 details the developed function of the CORBA application servers. Section 5 explains multiple human robot interfaces for the users interacting. The experimental results are given in Section 6. Section 7 concludes the paper.

2. Multi-Robot Systems

Multi-robot cooperation to perform service tasks for supporting the aged or disabled is indispensable in the population-aging society. The mobile platform equipped with a dexterous manipulator is convenient, but it is very difficult to handle the objects (such as operating objects in refrigerators) because of the difficulty to control the position and orientation of the mobile platform and the manipulator mounted on the mobile platform. In our system, we adopted using a robot arm with five degrees of freedoms cooperating with a mobile robot to implement the service tasks. This method makes it easier to operate the objects such as in refrigerators.

2.1 Robot arm and hand

The Mitsubishi Movemaster Super RV-E3J (5 DOF, Figure 1) is fixed on the place where there are many objects collected, and its manipulability range is approximately 1000 mm in height, 830 mm in radius from -160° to 160°. The maximum speed of the robot arm is about 3500mm/sec; its load weight is about 3.5kgf. The robot arm can manipulate the objects with sufficient dexterity to permit delicate and precise actions. Cameras were mounted on the environment around the robot arm in order to recognize the objects. The communication between the robot arm controller and robot arm control server computer is via RS-232C. To manipulate objects with accuracy and safety and prevent the robot from breaking object it handles, force sensors are affixed to the robot fingers. We designed the CPU control system to measure the grasp force and the servo-driving circuit to drive the fingers (Jia, et al., 2001).

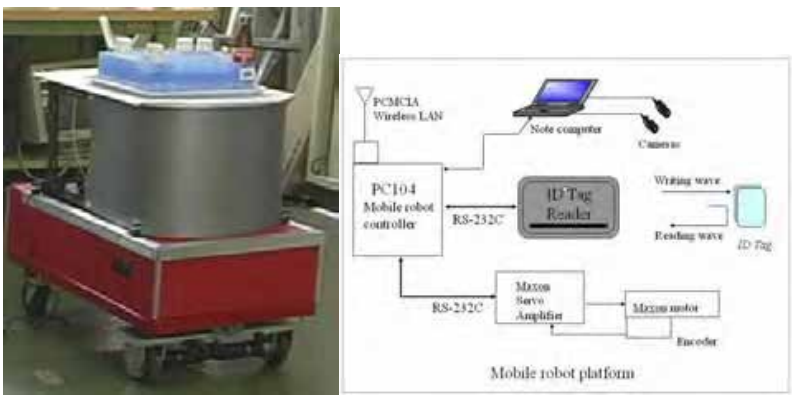
Figure 1. Robot arm and robot hand

2.2 Nonholonomic mobile robot

In the formerly developed system, the omnidirectional mobile robot was used to deliver the objects. Because of the specific structure of its wheel arrangement, it is difficult for the omnidirectional mobile robot to pass over bump or enter a room where there is a threshold. Another important point is to lower costs and decrease the number of motors so that the battery can supply enough electricity for mobile robot to run for a longer time. In our new system, we developed a nonholonomic mobile robot that was remodeled from a commercially available manual cart. The size of the mobile robot is about 700mm x 450mm x 700mm. The structure of the front wheels was changed with a lever balance structure to make mobile robot move smoothly, and the motors were fixed to the two front wheels. It has low cost and is easy to pass over bump or gap between floor and rooms. We selected Maxon EC motor and a digital server amplifier 4-Q-EC 50/5 which can be controlled via RS-232C. For the controller of mobile robot, a PC104 CPU module (PCM-3350 Geode GX1-300 based) is used, on which RT-Linux is running. For communication between a mobile robot and mobile robot control server running on the host computer, the Wireless LAN (PCMCIA-WLI-L111) is used. Figure 2 (a) shows the developed mobile robot. And Figure 2(b) shows the structure of the mobile robot platform.

2.3 RFID system

KENWOOD series was used in the developed system. Tag reader S1500/00 communicates with tags via 2.45GHz radio wave. Since there is a communication area between ID tag and tag reader (the communication between mobile robot controller and tag reader is via RS 232C), so if ID tag comes into the communication area while mobile robot moves to the place close to the ID tags, the ID tag can be detected and the information written in it can simultaneously be read by tag reader mounted on the mobile robot. When the working domain of mobile robot is changed or extended, what needs to be done is just putting the new ID tags in new environment and registering these ID tags to database. It is also helpful to improve dynamic obstacles recognition (such as chair or person).



a) b)

Figure 2. Mobile robot; (a) is the developed mobile; (b) is the structure of mobile robot platform

3. Localization of Mobile Robot

Localization is the most important fundament for a mobile robot to perform a service task in office, at facility or at home. In many previous research works, various methods for localization using many kinds of sensors for mobile robot purpose have been proposed. In this paper, we developed a novel method of localization with a camera and RFID technology to determine the position and orientation of mobile robot as it is inexpensive, flexible and easy to use in the practical environment (Lin et al, 2004). For static obstacles, the user registers ID tags to database beforehand, each ID tag includes the information of the object such as table, bookshelf or feature position like corners, passage crossing or entrance of door. For dynamic or uncertain obstacles with a new ID, the system can detect a new ID tag and read out the information of this ID dynamically, thus decreases the computation of obstacle recognition. It is also helpful to improve dynamic obstacles recognition (such as chair or person) and occlusion problem that are very difficult to solve. This is because the communication between ID reader and ID tags uses Radio Frequency. Figure 3 illustrates the principle of localization of mobile robot.

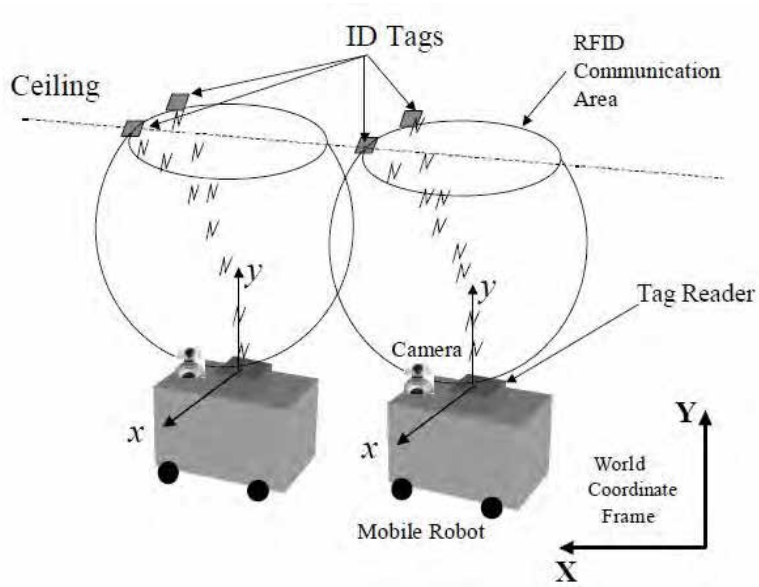


Figure. 3. The principle of localization of mobile robot with RFID and camera

The tag reader and a camera are mounted on mobile robot platform, and ID tags are put on the obstacle or environment of the mobile robot moving. Since there is a communication area between ID tag and tag Reader (the communication between mobile robot controller and tag reader is via RS-232C), so if ID tag comes into the communication area while mobile robot moves to the place close to the ID tags, the ID tag can be detected by tag reader mounted on the mobile robot and the information written in ID tags in advance can be read out at the same time. When the ID tag was detected by reader, the system can judge first that ID tags are obstacle tag or localization tag. As every localization ID tag has a unique ID, so every localization ID tag can indicate an absolute position in the environment. After getting the absolute position of a localization ID tags, we can measure the relative position and orientation of mobile robot to this ID tag using a camera in order to get the world coordinates of the mobile robot.

The camera with 640 x 480 resolution and 30fps frame rate was mounted on the mobile robot to be used in the system to recognize the ID tags then get the relative position of mobile robot. In order to improve the speed of imaging processing, a window function for fast extraction of ID tags has been used. The weighted sum of each window can be calculated by equation (1), (2).

$$Z[m,n] = \sum_{x=0}^{k} \sum_{y=0}^{l} G[i,j]W[x,y] \tag{1}$$

$$G[i,j] = \sqrt{(f[i,j+1] - f[i,j])^2 + (f[i,j] - f[i+1,j])^2} \quad (2)$$

Here, G[i,j] is gradient function of point [i,j], f[i,j] is the gray level of a point [i,j] in the image. K x l are window size, it depends on the size of the ID tags and the background of the experiment environment. W[x,y] is a k x l window array and every elements is 1. If the weighted sum Z[m,n] is smaller than a given threshold (determined by experiments), the dilation processing technique for this window will be done in order to judge that the detected area is ID tag or not. As the image size of camera is 640 x 480, the size of window is 8 x 8, so there are only 80 x 60 elements in Z[m, n]. The selection of window size 8 x 8 was justified by the results of experiments. Selecting a bigger window size can speed up image processing speed, but it lowers the precision of recognition. Conversely, selecting a smaller window size can get better recognition of ID tag, but it increases the computation of image processing.

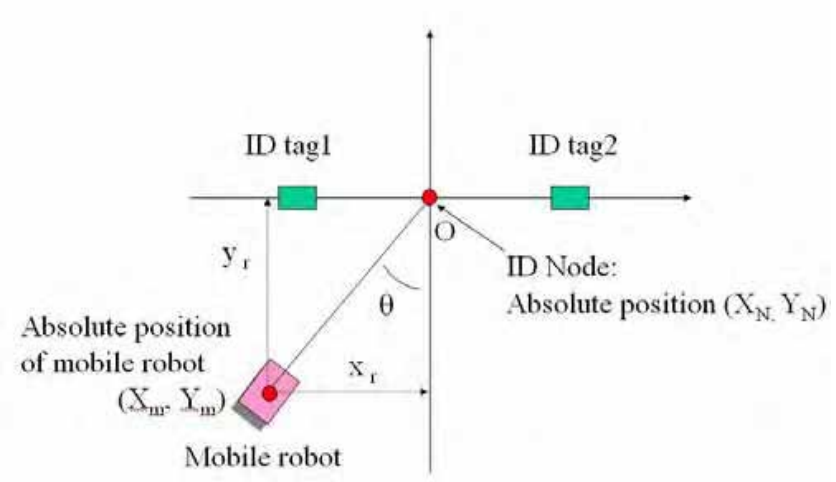


Figure 4. The calculation of the world coordinates of a mobile robot

Two ID tags as a node are affixed on the feature positions of ceiling such as corners, passage crossings, or entrance of door. The middle point of two ID tags is used as the absolute position of ID node in environment. Using camera system to recognize the ID tags can get the relative position of mobile robot with the ID tags. Then the absolute position of mobile robot in world coordinate frame can be calculated by the following equations (equation (3), (4), (5)). Figure 4 illustrates how we can calculate the world coordinates of mobile robot using the proposed method.

$$X_m = X_N - x_r \quad (3)$$

$$Y_m = Y_N - y_r \quad (4)$$

$$\theta = arctan\frac{Y_L - Y_R}{X_L - X_R} = arctan\frac{y_L - y_R}{x_L - x_R} \quad (5)$$

Here, X_N, Y_N are the absolute position of a ID node. When the ID reader detects the ID tag1 and ID tag2, X_N and Y_N can be calculated according to the absolute coordinates of two tags registered beforehand.

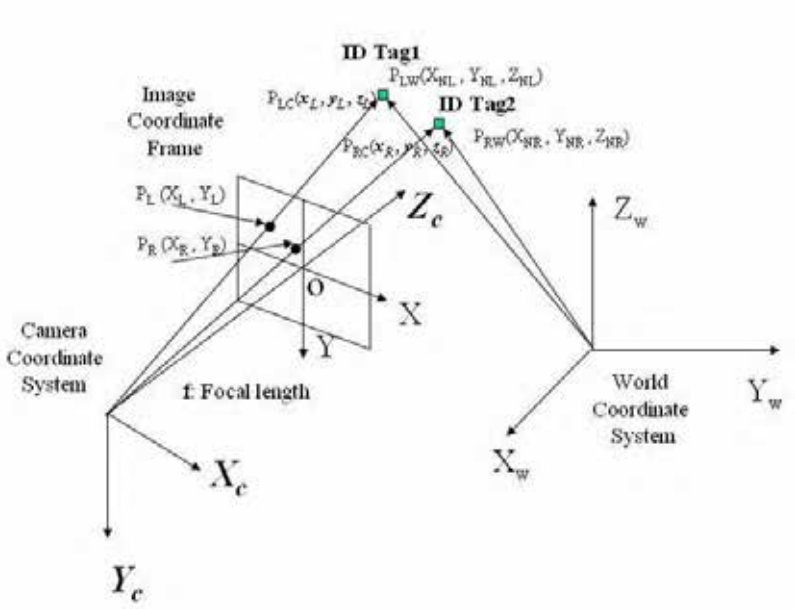


Figure. 5. World coordinate system, camera coordinate system and image coordinate frame

Figure 5 illustrates the relationship among World Coordinate System, Camera Coordinates System and Image Coordinates Frame used in our system. The origin of World Coordinate System was set at the corner of environment the mobile robot moves in. The origin of Camera Coordinates System was set at the focus of CCD camera mounted on the mobile robot. $P_{LW}(X_{NL}, Y_{NL}, Z_{NL})$ and $P_{RW}(X_{NR}, Y_{NR}, Z_{NR})$ are the coordinates of two ID tags in World Coordinate System. $P_{LC}(x_L, y_L, z_L)$ and $P_{RC}(x_R, y_R, z_R)$ are the coordinates of two ID tags in Camera Coordinate System. $P_L(X_L, Y_L)$ and $P_R(X_R, Y_R)$ are the projected coordinates of two ID tags in Image Coordinates Frame. According to coordinate transformation, we can calculate x_L, y_L, x_R, y_R and. X_L, Y_L, X_R, Y_R. Θ is the orientation angle of the mobile robot in the world coordinate frame. x_r, y_r are the position of mobile robot relative to the node, which can be got by recognition with camera. Using these variables, the coordinates of camera (the coordinates of mobile robot X_m, Y_m in World Coordinate System can be calculated by equation (equation (6)).

$$\begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} X_m \\ Y_m \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \quad (6)$$

Here, x and y are the coordinates of the middle point of two ID tags in Camera Coordinate Frame with the coordinates of X, Y in Image Coordinate Frame. X_m and Y_m are the coordinates of camera (the coordinates of mobile robot X_m, Y_m) in World Coordinate System.

As we know, each ID tag has an unique ID, so each ID node can indicate an absolute position in the environment. All the "node" makeup a topologic map of the indoor environment in which the mobile robot moves. For example, if mobile robot moves from START point A to GOAL point F, the moving path can be described with node tree shown in Figure 6. The system searched the shortest path between the START and GOAL node (for example, the shortest path between A and F is A→B→C→D→F) by tracing the branches between them.

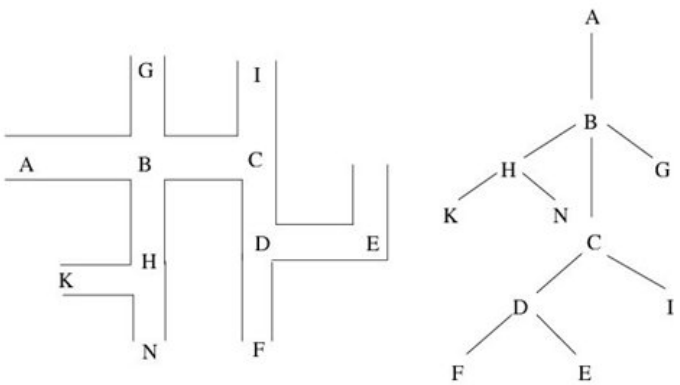


Figure 6. Moving path and its description with node

In order to navigator a mobile robot in an indoor environment, how to build a map is a big problem. In our system, the topological map for mobile robot was used. For building a topological map, the connectivity of ID nodes and the relative direction angles between every two adjacent ID nodes information are necessary. How to represent a node is very important. We represent ID node with a code. A code indicates the location information of this node and its connectivity relationship with surrounding nodes, such as which building, which floor, which room. Now, we represent it with six bit decimal number. For example, a node with 000000 means it is an original node at the first floor. A node with 012221, we can know that it is a sixth level node and the upper level node of it is 012220. Figure 7 depicts the connectivity information from this node to the original node, and node level information.

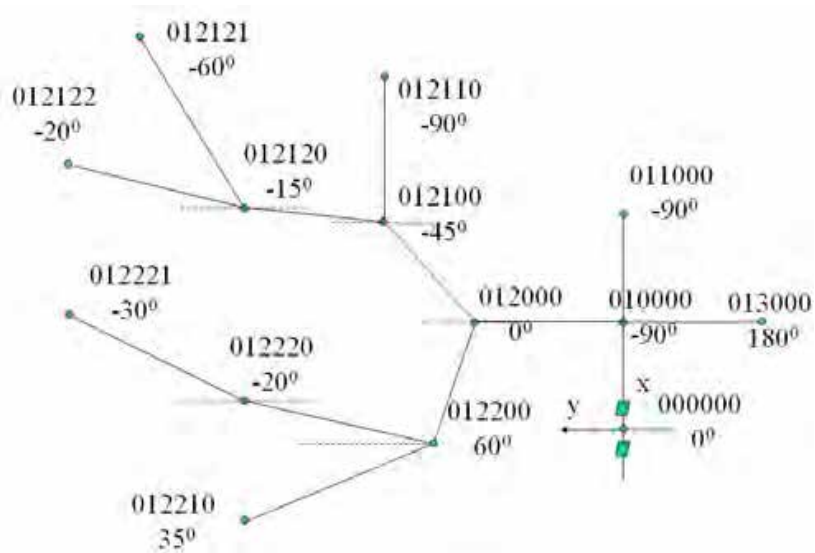


Figure 7. An example of topologic map of the proposed method

4. Application Servers

In our aging society, it would seem particularly important to integrate various kinds of network distributed software and robotic systems for complicated applications aiding the elderly population. Distributed computing technologies that can implement network distributed software sharing and improve the cost of writing and maintaining software is in high demand. The various different distributed computing technologies, termed middleware, include RMI (Remote Method Invocation), DCOM (Distributed Component Object Model), MOM (Messages Oriented Middleware), and CORBA (Common Object Request Broker Architecture). Sun's Java RMI (Java Remote Method Invocation) is a Java-specific RPC middleware that provides a simple and direct model for distributed computation with Java objects. However, its simplicity is enabled by restricting all communication to Java objects only. DCOM is Microsoft's architecture for distributed component-based communication and is based on COM, which is both a specification and implementation developed by Microsoft Corporation. MOM (Message-oriented middleware) provides process-to-process data exchange, enabling the creation of distributed applications. It is analogous to e-mail in the sense that it is asynchronous, requiring the recipients of messages to interpret their meaning and to take appropriate action. MOM is not an industry standard. In contrast to all of these, CORBA (Condie, 1999; Object Management Group; Object Oriented Concepts) focuses on the use of distributed objects to provide for systems integration. CORBA uses GIOPs (General Inter-ORB Protocols), ESIOPs

(Environment Specific Inter-ORB Protocols) and IIOP (Internet Inter-ORB Protocols) to implement a truly heterogeneous distributed system, and makes application and system integration easier. It encourages the writing of open applications, ones that can be used as components of larger systems. Each application is made up of components; integration is supported by allowing other applications to communicate directly with these components. This facilitates network-distributed software sharing and improves the cost of writing and maintaining software. We selected CORBA as communication platform to develop a network-distributed human-assistance robotic system. We implemented User Management Server, Robot Arm Control Server, Mobile Robot Control Server, Real-Time Mobile Robot Positioning Server, and Feedback Image Server, which are independent components and can be distributed on the Internet and executed in parallel. It is possible for the developed system to reduce the number of some used servers, to regroup some of them according to the proposed tasks, and to integrate easily with the other technologies into new comprehensive application systems. The other components of the system can work normally even if there are problems with some of them.

4.1 User Management Server

It implements the management of users' manipulating privilege and the robotic systems manipulating connections between user (caregivers, the aged or disabled), robotic systems and the other CORBA application servers with the help of Authentication/Authorization.

4.2 Service management server

It manages the services that the developed system provides. The caregivers could register new local service tasks and update the information of database. It provides the information about the rationality of service tasks which the user requests. If the user requests a unreasonable task that the system can not provide, the error message will be presented to the user. Additionally, it can autonomously update the database of objects after the robot arm captures the objects.

4.3 Robot Arm Control Server

The task-level robot arm control server allows the remote user to control the remote robot arm at a task level. It receives the task-level requests from the client, performs various kinds of processing and returns the feedback results. When the remote user pushes the command "Juice, Please", the manipulator automatically handles the juice and places it on the tray mounted on the mobile platform. For one method of the task-level robot arm control server, it includes an information part, a task planning part, an implementation part and a communication part (Jia and Takase, 2001). The information part consists of a vision part and a force measure part. It is the source of information for the system. The task planning part receives the information from the information part, recognizes the location and orientation of the tableware scattered on the table, transforms these coordinates to the manipulator's coordinate system, and generates task plan to achieve the goal. Task plan mainly contains how to control the manipulator to achieve the place where the objects to be handle is, and how to grasp it by robot hand. It was implemented autonomously by programming according to the vision and force information (Jia and Takase, 2001). The implementation part executes motion scheduling generated by the task planning part, and it implements the task according to the commands coming from the server computer. The

communication between the server computer, the robot's arm and the robot hand's controller is via RS-232C Links. The details of robot task control algorithm were described in (Jia and Takase, 2001).

4.4 Mobile Robot Control Server

It receives the reasonable requests from the system, and then plans and derives the most appropriate path for the mobile robot to move in order to realize the task what the user issued. It works as:

a) The mobile robot control server receives control commands from the users.
b) ORB intercepts commands and transfers the requests to the mobile robot control server.
c) The mobile robot control server plans and derives a collision-free path that is the shortest distance between the START and GOAL specified by the user.
d) The mobile robot control server programs the mobile robot to move across wireless TCP/IP Ethernet link. According to the results of experiments, the wireless communication between a mobile robot and mobile robot control server running on the host computer is robust and reliable.
e) ORB returns the feedback results to the users.

4.5 Real-Time Mobile Robot Positioning Server

It provides the real-time position and orientation of the mobile robot with respect to the world coordinate system, so that the user could get a better understanding of what the mobile robot is carrying out. In our research, we developed the method of positioning mobile robot using RFID and camera.

4.6 Feedback Image Server

The live image feedback server provides various kinds of live feedback images and control modals for Web users. It receives requests from the user, obtains images from cameras mounted in the environment, and compresses the image into JPEG format. Then, ORB returns the new image with the latest information to the client. The user can see live feedback images of the mobile robot moving, the cooperating with the robot arm, and the state of the rooms of the aged or disabled. The user can also select different control modals to obtain "auto" or "step" live feedback images.

5. Multiple HRI (Human-Robot Interface)

Considering multi-type user of the developed system, we have implemented multi-type user interfaces s that enable different user to control robot systems easily. Robot systems should be able to interact with local user in a natural way and to allow remote users (caregivers, relatives) to understand the environment where the robot systems are working clearly and easily. For a remote user, we developed Web-based user interface. Video stream, a typical way of providing the information of visualizations for the robotic system working, the environment of robot system and the state of the age and disabled, was also provided in order to enable the remote user to get a better understanding of situation. Due to high bandwidth requirements of video stream and necessity to extend the visualizing range, we also developed image feedback server that provides feedback images getting by cameras mounted in the environment according to the users' options.

5.1 Video/Audio Conference System

It is necessary to receive and transmit media streams in real time to improve interaction in network distributed human-assistance robotic system. Additionally, robot manipulation rights should be appropriately allocated by carefully negotiating among the users (caregivers, local users). In order to meet these special requirements of human-assistance robotic system, a private video/audio conference system was proposed (Hou et al., 2002) . Multicast is a clever technique to reduce network bandwidth demand when there are many receivers who want to view or listen to the same source. Therefore, MBone is the best choice for video/audio conference systems. Since RTP provides unreliable end-to-end network delivery services for the real-time data transmission, it is usually adopted over UDP for media streams even if it is network and transport-protocol independent. This is also because the overhead of guaranteeing reliable data transfer slows the overall transmission rate. The architecture of the proposed video/audio conference system is shown in Figure 8. Media Control (MC) and Media Processor (MP) form the video user client site and manage the video sessions via session manager. (Media Control Centre) MCC resembles a multipoint controller H.323 entity on the network, and provides for the control of three or more terminals participating in a multipoint conference. MCC also provides for capability negotiation with all MC in remote video users site, and controls video resources, such as multicast addresses. MC and MP, which are connected with one another via JMF (JavaTM Media Framework) session manager API, work more like another H.323 entity of multipoint processor. They provide for processing, mixing, switching of video/audio streams in a network multipoint conference system. CORBA IIOP is employed as message communication platform between MCC and MC.

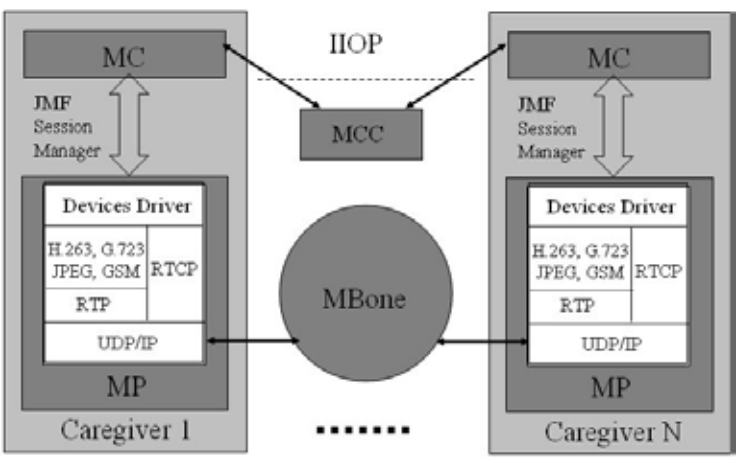


Figure 8. The architecture of the proposed video/audio conference system

5.2 Chat Room Panel

The chat client user interface was developed for changing the manipulating right of robot system. The chat client software is programmed using Java and embedded in the web page as Java applet. Users can login in the chat room, exchange information and transfer robot operating token via this applet. This applet also illustrates the current users who has the manipulating right to operate the multi-robotic system and the log of users operating the robotic system.. After getting the robots manipulating right through the chat room, the other human caregivers can use robotic systems control user interface to control the remote robotic systems to provide services or support for the aged or disabled. In this system, MySQL is used with JDBC. The SQL database MySQL (Yarger et al. 1999) is a popular open source database server in the world. MySQL can be employed to handle large database. The SQL database stores static data (e.g., registry information for users), and dynamic data (e.g., system running states)

5.3 Robot Arm Control and Feedback Image Control Pane

The robot arm control and feedback image control panel including the task-level robot arm control command part, the options and display for different kinds of live feedback images, and control modes (Jia et al., 2002). The task-level robot arm control commands allows the user to submit a task-level request to the system. It consists of meal service commands, drug service commands, and common service commands such as providing a book. Once one task-level is submitted, the other task-level button will be invalid until this task is finished for safety. And the robot arm will recognize and manipulate the objects autonomously. Many options for different kinds of live feedback images have been provided such as the state of the mobile robot cooperating with the manipulator, the rooms of the disabled or aged. In addition, “auto" and "step" control modals of the live image feedback are provided to allow user to see the continuous live feedback images or the "step" image feedback which refreshes the image once after button is pushed.

5.4 Mobile Robot Control Panel

The geometric 2D map is built as a model of the environment of the mobile robotic system's working domain when the user links to this homepage. Its geometric primitives can also be adjusted to add a new obstacle if the environment of the robot system has been changed. The positions where the mobile robot is easy to cooperate with the robot arm, and the mobile robot is easy to pass the objects to the disabled or aged are displayed as marks. The remote user can specify the most appropriate route for the mobile robot to move and directly control the mobile robot to move or rotate in order to cooperate with the manipulator easily if it is necessary. In order to know the state of robotic systems working, the real trajectory of the mobile robot moving was also shown on the user interface. Also, the user can select the live feedback image of the mobile robot, then the user can monitor the state of the mobile robot, the area around it and its pose in real environment.

5.5 Voice User Interface

Natural spoken method is the friendliest way of communication with robot for local user, and it is easy way for the aged or disabled to control robot. We used a commercially available speech system to develop the voice-enabled interface by Visual C++. The task of

speech recognition adapted the syntax of a particular type of BNF (Backus-Naur Form) grammar, is called Speech Recognition Control Language (abbreviated SRCL). A SRCL grammar is defined by enumerating the valid words and phrases. Grammars constructed using SRCL offer an organized view of the words and phrases that are part of the speech grammar, since we define a notation for identifying common phrases, optional phrases and repeated phrases. To improve the recognition rate, we defined 30 SRCL grammars that are relevant to the service tasks the robot systems can provide. The average recognition rate of the developed system is approximately 90% (Wang et al., 2003).

5.6 Touch Panel User Interface

The touch panel user interface has also been developed to make up for the "breakdown" of the speech recognition system, and is helpful to the user who is not convenient to speak to control a robot. When the speech recognition system can not recognize the command the user issued, the system can give user the selection of inputting their request by voice again or using touch interface.

6. Experiments

6.1 Experiments of Localization of mobile robot using RFID and camera

First, we have done the experiments of localization of mobile robot with only RFID technology and compared the results with using odometer which is most widely used. According to the relationship of t_1 (the time tag1 was detected), t_2 (the time tag2 was detected), t_3 (the time tag1 can not be detected), t_4 (the time tag2 can not be detected), we can localize the mobile robot. The experiment is that the mobile robot moves forward about 2.5m from the START position (x=0, y=0, Θ=0) and back. After repeating a number times, in the case of using only odometer the mobile robot's returned positions are far and far from the START position, because odometer has the disadvantage of the slippage problem and inaccuracies of kinematics models with the consequent errors growing with time. The error results of repeating 5 times and 10 time are △x=38.5cm, △y=-18cm, △Θ=-13.0° and △x =88.5cm, △y=35.5cm, △Θ=37.0°. Using RFID localization system, we can get the feedback information about the position and orientation of mobile robot by RFID, then can adjust the mobile robot to return the hopeful position. The same experiments have been done and the error results of repeating 5 times and 10 time are △x=11.5cm, △y=-8.5cm, △Θ=5.5° and △x =5.0cm, △y=-13.5cm, △Θ=7.5°. Although these results are better than that of odometer only, we only got the 10cm and 7.5° resolution to the instability of RFID system and it is not enough to navigate a mobile robot to perform a service task in indoor environment. For improvement of the precision of localization of mobile robot, we mounted a camera on the mobile robot platform, integrating the information of RFID, camera and odometer to determine the position and orientation of mobile robot. The maximum error is about △x =1.9cm, △y=2.5cm and △Θ=2.5°. This result verified that the accuracy of the developed localization system is enough for navigation of mobile robot.

6.2 Video Stream Feedback Experiments

In order to enable remote caregivers to get a better understanding of the local environment, we also provide the live video feedback. The maximum video resolution of the video camera selected is 640 x 480, and its maximum video frame rate is 30 fps (frames per second).

JMF2.1.1 is employed to implement a video/audio transmission. H.263 and JPEG over RTP are implemented for video presentation, and audio encoding select GSM and G.723 over RTP. The performance test of the developed real-time video stream has been done to get live video feedback to monitor the state of the aged or disabled in a campus network. The video server is run on Windows 2000 Professional (Pentium IV, CPU 1.9GHz), and the video client is run on Windows XP (IV, CPU 2.4GHz). The average frame rate is about 19.5fps. The experiments that users transfer robot control token via video/audio conference system have also been done. After entering the multipoint conference, users (e.g., doctor, caregivers) can select media for presentation by double clicking the users names in the middle right of the chat client panel. After a discussion, the robot manipulating token will transfer to an appropriate user. The experiment was successfully done in a campus network. Many options for different kinds of live feedback images have also been provided such as the state of the mobile robot cooperating with the manipulator, the rooms of the disabled or aged. In addition, "auto" and "step" control modals of the live image feedback are provided to allow user to see the continuous live feedback images or the "step" image feedback which refreshes the image once after button is pushed.

6.3 Experiments of User operating the Multi-functional Robot System with HRI

Using CORBA as a communication architecture, we developed a network-distributed multi-functional robotic system to assist the aged and those with impaired mobility, which is very important in the aging society to improve the problem of shortage of persons capable of working and to avoid the high costs for the individual treatment in nursing homes that might otherwise be necessary. We developed a multi-robot, implemented key technologies, and developed CORBA application servers which can be distributed on the Internet and executed in parallel. We also proposed a novel method of localization of mobile robot using RFID system with a camera as it is flexible and easy to use. Considering multi-type user of the system, we have implemented various kinds of user interface that enable different user to control robot system easily. For a remote user, we developed Web-based user interface. Video stream, a typical way of providing the information of visualizations for the local environment was also provided. By remotely controlling a mobile robot to cooperate with a robot arm, the developed system realized successfully some basic services (such as bringing a bottle of water to the aged or disabled) to support the aged and disabled. Figure 9 (a), (b), (c) illustrate some on-line images that the remote Web user is accessing the developed system by using Web use interface.

For a local user (the aged or disabled), they can use natural spoken to control robot systems to realize the local services. If the speech recognition system breaks down, the system can give user the selection of inputting the command again by voice or using touch interface. Touch panel user interface is also helpful to the user who is not convenient to speak to control a robot. Figure 9 (d), (e), (f), (g), (h), (i) illustrates some images that the local user is interacting with mobile robot by speech to instruct the mobile robot to realize a local service task. According to the results of experiments, we know the developed system can provide some daily service to aid the aged or disabled.

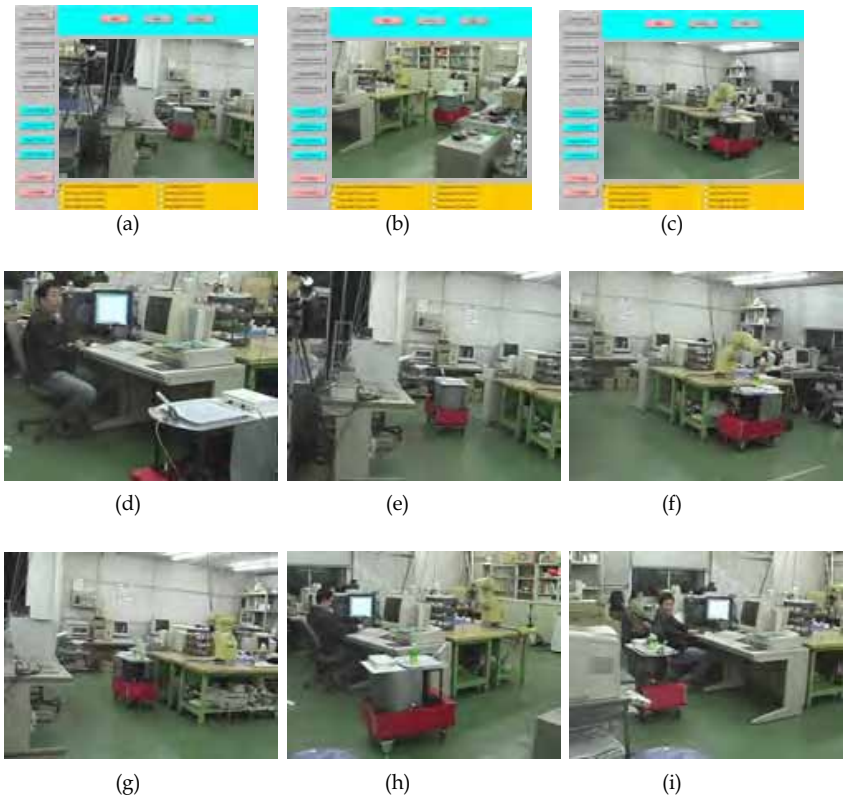


Figure 9. (a), (b), (c) On-line images a remote user interacting with the robotic systems. (d), (e), (f), (g), (h) and (i) are on-line images a local user interacting with mobile robot by speech to instruct the mobile robot to realize a local service task

7. Conclusion

We have been developing a network distributed multi-functional robotic system in order to improve care cost and the QoL of the elderly people. We proposed a novel method of localization of mobile robot using RFID system with a camera as it is flexible and easy to use. Because the information of obstacle or environment can be written in ID tags, the proposed method enables the localization easily and quickly compared with the other method. A video/audio conference system was also developed to improve the interaction among the users and enable web-user o get a better understanding of what is going on in the local environment. Considering multi-type user of the developed system, we have implemented various kinds of user interfaces that enable different users to control robot

system easily. Local user can operate the robot systems by natural speech and touch panel to control a mobile robot cooperating with a skilful robot arm to replace person to operate the objects in refrigerator that is difficult for the aged or disabled, and to supply them necessary daily services. This not only fulfils their desire for independence and autonomy, it also helps to avoid the high costs for the individual treatment in nursing homes that might otherwise be necessary. Caregivers or remote user can support the local user and monitor the state of the aged or disabled and the robotic systems working by video system. Some experimental results verified the effectiveness of the developed system. For future work, improving intelligent and simplifying operation to system, adding services are the main topics.

8. References

Condie, S. (1999). Distributed computing, tomorrow's panacea-an introduction to current technology, *BT Technol J*, Vol. 17, No. 2, pp. 13-23.

Coristine, M., Stein, M. R. (2004), Design of a New PumaPaint Interface and Its Use in One Year of Operation, *Proc. of IEEE Int. Conference on Robotics and Automation}, (ICRA'2004)*, New Orleans, LA., April, pp. 511-516.

Hada, Y. and Takase, K. (2001), Multiple Mobile Robots Navigation Using Indoor Global Positioning System (iGPS), *Proceedings of 2001 IEEE/RSJ Conference on Intelligent Robots and Systems*, pp. 1005-1010.

Chunhai Hou, Songmin Jia, Gang Ye and Kunikatsu Takase, (2002), Manipulation Switching Management for Robots in Internet Telecare Systems', *2002 IEEE International Conference on Industry Technology (ICIT'2002)*, December 11-14, 2002, Thailand pp.886-891, 2002.

Java remote method invocation: http://java.sun.com/products/jdk/rmi/index.html.

JavaTM Media Framework API Guide. Sun Microsystems, Inc., California, USA, 1999.

Jia, S. and Takase, K. (2001), An Internet Robotic System Based Common Object Request Broker Architecture, Proc. *of IEEE Int. Conference on Robotics and Automation, (ICRA'2001)*, Seoul, Korea, pp. 1915-1920.

Jia. S and Takase K. (2001), A CORBA-Based Internet Robotic System, T*he International Journal of Advanced Robotics*, ISSN 0169-1864, Vol. 15, No. 6, pp. 663-673.

Jia S., Hada Y., and Takase K. (2003), Telecare Robotic System for Support Elderly and Disabled People, *Proceedings of IEEE/ASME International Conference on Advanced Intelligent Mechatronics* , pp.1123-1128.

Jia, S. Hada Y., Ye, G. and Takase, K. (2002), Distributed Telecare Robotic Systems Using CORBA as a Communication Architecture, Proc. *of IEEE Int. Conference on Robotics and Automation}, (ICRA'2002)*, Washington, DC, USA, pp. 2002-2007.

Lin, W., Jia, S., Fei, Y., Takase, K. (2004), Topological Navigation of Mobile Robot using ID Tag and WEB Camera, *The 2007 IEEE International Conference on Mechatronics and Automation, pp. 644-649.*

Maeyama, S., Yuta, S. and Harada, (2000), A. Experiments on a Remote Appreciation Robot in an Art Museum, *Proc. of 2000 IEEE/RSJ Conference on Intelligent Robots and Syste*ms, Japan, pp. 1008-1013.

Message-orientated middleware:
http://sims.berkeley.edu/courses/is206/f97/GroupB\mom.

Nagi, N. Newman, W.S., Liberatore, V. (2002), An experiment in Internet-based, human-assisted robotics, *Proc. of IEEE Int. Conference on Robotics and Automation (ICRA'2002)*, Washington, DC, USA, pp.2190-2195.

Object Management Group, http://www.omg.org.

Object Oriented Concepts, Inc., http://www.omg.org.

Schulz, D., Burgard, W., Fox, D. et al.: (2002), Web Interface for Mobile Robots in Public Places, *IEEE Robotics and Automation Magazine*, 7(1), pp. 48-56.

Stein, M. R. Stein, (2000), Interactive Internet Artistry, *IEEE Robotics and Automation Magazine*, 7(1) (2000), pp. 28-32.

R. J. Yarger, G. Reese, T. King and A. Oram, (1999), MySQL and mSQL, Publisher: O'Reilly & Associates, Incorporated.

Wang, K. Jia, S., Lin, W. and Takase, K. (2003), Operation of Mobile Robot by Voice and Touch Panel, *System Integration*, 1I4-4.

8

Human-Robot Interface for end effectors

Marcin Kaczmarski
Technical University, Institute of Automatic Control I-13
Poland

1. Abstract.

This paper focuses on a comprehensive description of the artificial hand and its driving mechanisms. Static and dynamic parameters of the model are detailed. The kinematics of the hand and its application in visual control are presented. Each joint of the designed hand is driven by a pair of McKibben muscles. Due to their parameters – high overloading and stiffness control capabilities, they are very suitable for environment interaction. The chosen actuators - pneumatic muscles, are also a simplified substitute for human muscles. Modeling the work of the human hand can in this case be conducted in a limited range. On the other hand, these limitations simplify the vision analysis, which was adequately considered in the algorithms of image processing. When designing the software, it was attempted to use only that information, which is necessary to accomplish a given task and the interaction of the hand with the steered object. A cue acquired from a biosignal measuring system is the main signal initiating the gripping phase. All miopotentials are recorded by an active surface electrode. The paper presents the problems of artifacts in the measured signal and solutions which allow for reducing their level. It also shows a schematic of laboratory stand, which is used for electromiografy recording and controlling the finger banding process of the artificial hand. The paper ends with collected conclusions from the research projects conducted by our team and the future plans concerning improved replication of human movements and using a stereoscopic view in vision control.

2. Introduction

Evolution of robotic constructions can be observed all over the world. Robots replace human work in many areas. They work in environments that are harmful to humans and are used for tasks requiring precision of movements. To this purpose manipulators of a different kind are being built, starting from structures with a serial kinematical chain to ones with a parallel one which is more commonly used at the present. Unfortunately, these kinds of structures are designed only for one type of tasks. That is why humans try to "spy the nature" to get the inspiration for new, more versatile projects. As always, an unreachable construction goal is the imitation of a human being. Besides of locomotive functions this kind of robot should be able to manipulate in any environment where it is placed. Synthesis of manipulation and locomotion in an anthropomorphic construction is developed by the Honda corporation, where humanoid robots P3 and Asimo were created. Although their external shape is similar to human beings, they can only try to imitate the smoothness of

their natural equivalent. Human smooth gait is a complex process and cannot be replaced by any control algorithm. It is characterized by a dynamic phase which is unreachable by electrical motors. Another construction of anthropomorphic biped robot is proposed by a British company "Shadow". In their project they use artificial McKibben muscles for actuating robot joints. These muscles are characterized by variable stiffness (Feja K., Kaczmarski M., Riabcew P., 2005) which makes smooth and dynamic movement possible. Imperfection of this actuator is the nonlinear force characteristic, which makes control more complex in comparison to electric motors.

Another problem for humanoid robots is the manipulation of objects. To this purpose different kinds of grippers are created. Starting from a simple one with the basic types of grasps, to more complicated ones that provide adaptability to different forms of objects.

Created in the Institute of Automatic Control, the artificial hand is a simplified imitation of the human natural hand. Artificial muscles which were used for actuating the hand are also very simple compared to equivalent natural ones. This of course influences the modeling of movements, making them limited.

In most artificial grippers internal sensors of fingers' positions are used to control movements. Additional force sensors allow recognizing the pressure of fingers during contact with an object and providing grasp adaptation.

This chapter shows that also a vision system can be used to recognize the position of fingers and control the precision and force of grasping. The created artificial hand has no internal sensors, so control is achieved by only the information obtained from the vision system. Although there are no pressure sensors on the finger tips, this kind of control is able to provide precision and regulate the force of grasping which will be shown in the text. Limitations in the construction of the artificial hand allowed to introduce some simplifications into the vision control algorithm. In the further part of the text the focus will be given to the construction properties of the created artificial hand and pneumatic muscles, which are used for actuating the fingers and wrist joints. A mathematical analysis of the kinematic model of the hand and its adaption to vision control for analysis by specific trajectories will be presented. The strategies of grasping and photos presenting the results of the vision control application in an artificial hand control will also be described.

This chapter also presents the biopotential recordings for simple movement driving. For acquisition of the EMG signals a surface-active electrode was constructed. Its purpose was the preliminary preparation of measured signals. It was placed near the selected muscle groups to minimize the interferences. The collected signals were submitted to further processing and to digital acquisition. The created computer algorithm, basing on the myopotential signal, determines the effective (root mean square) value, which in the simplest form is dependant on the tension of the examined muscle. The computer program then controls the electro-valves, which set the appropriate pressure levels in the pneumatic muscles. The muscles actuate active joints in the artificial hand, causing the fingers to tighten. The presented interface for the control of various robotic constructions will be used in future projects in our laboratory.

3. Construction of the artificial hand

The view of an artificial hand is presented on Figure 1.

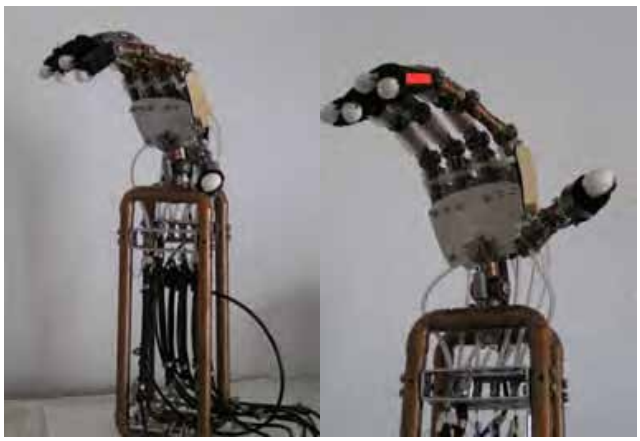


Figure 1. Real view of an artificial hand

The artificial hand can be divided into a number of segments. The biggest one is the structure of the forearm, which holds two layers of pneumatic muscles. The upper one contains two rows of actuators and they are linked in pairs. Each pair is connected to an active finger joint by the Kevlar tendons. Two muscles work in opposition to provide full banding and straightening of each finger. The thumb is actuated by only one pneumatic muscle, because its backward movement is provided by a spring and the gravitational force. The bottom layer contains only two actuators. One is responsible for rotating the hand in the wrist. This movement is similar to the pronation/supination. The second muscle provides the rotation of the thumb in the plane parallel to the surface of the palm. In both situations a spring element generates the force required for backward rotation. This force is proportional to extension of the spring and provides constant inclination of this value during the filling up of the muscle with compressed air. The force characteristic of an artificial muscle is linear in the initial range, so the rotation of the joint is proportion to the air pressure in the muscle.

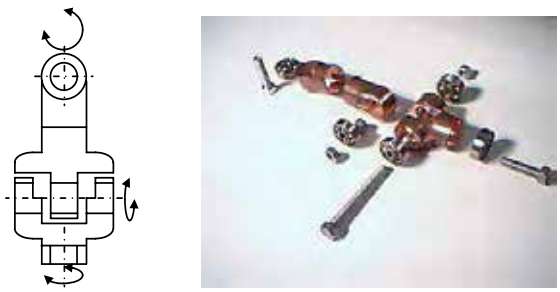


Figure 2. Wrist schematic

The second part of an artificial hand is the wrist. It consists of three independent rotary joints. They are placed near together and oriented in a way providing rotation around XYZ

axis of Cartesian coordinates. The real human wrist has only 2 DOF. The rotation (pronation-supination) in the human hand takes place in the forearm. This type of movement is implemented in the wrist of the artificial hand and increases its object manipulation capabilities. A schematic of the wrist is presented in Figure 2.
The last part of the designed construction is made up of the five fingers on an artificial palm. Every finger (except the thumb) contains three simple rotary joints with 1 DOF, while the thumb contains four. This allows the thumb tip for movements in two parallel planes. The total length of fully extended finger is 120mm. The presented artificial hand model allows fingers to bend completely, which is obtained by simultaneous rotation of all three rotary joints to the same angle. Thanks to this construction it was possible to use only one pair of actuators to drive the first active joint in the finger. Although this kind of solution causes limitations of number of DOF, it reduces the number of actuators required for movement generation.

a)

b)

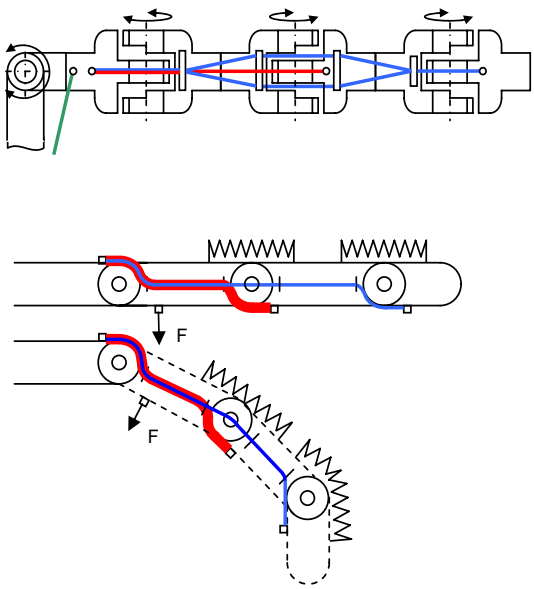


Figure 3. Arrangement of tendons in the fingers a) top view, b) side view

The picture in Fig.3. shows the arrangement of the Kevlar tendons in every finger. All wires being part of the mechanical coupling are attached to the top side of the palm. The other end of the wire is connected to the bottom side of the passive joint. When the active joint rotates, the length of the wire between attaching point and first slide increases while the length between the second end of the wire and the nearest slide decreases simultaneously. This provides the same angle of rotation in every joint of the finger. Every passive joint contains a spring element required for preventing spontaneous descending and providing linear relation of the compressed air pressure in the muscle.

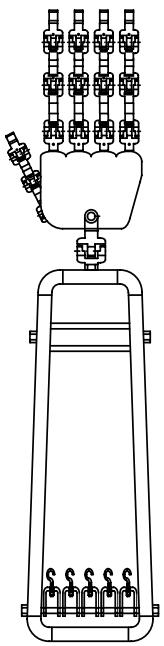
4. System of the driveline

All actuators are connected to the active joints by the Nylon-Kevlar wire. The same kind of tendons are used to transfer the force from the active joints to the passive ones. Their undoubted advantage is a low friction coefficient and high tensile strength, that is the ratio of maximum braking power to the initial cross section field.

5. Characteristic data of the hand

The most important static and dynamic parameters of the created construction are presented below. It shows that the designed gripper with its size and movement possibilities is similar to a natural human limb.

5.1. Dimensions of the artificial hand



Maximum height - 530mm
Maximum width - 105mm
Maximum depth (thumb fully extended) - 240mm
Number of fingers – 5
Wrist – 3 rotations
Total mass – 1,5 kg
Forearm
Height – 285mm
Width – 105mm
Depth – 90mm
- Fingers (1-4)
  - Total length (fully extended) – 120mm
  - Width of the finger – 20mm
  - Number of rotary joints – 3
  - Number of active joints – 1
- Thumb
  - Total length (fully extended) – 160mm
  - Width of the finger – 20mm
  - Number of rotary joints – 4
  - Number of active joints – 2
- Wrist
  - Length – 60mm
  - Width – 30mm
  - Number of Degrees Of Freedom – 3
  - Number of active joints – 3
- Actuator unit
  - Maximum number of pneumatic muscles – 12
  - Number of muscles driving one finger – 2
  - Number of muscles driving the wrist – 3

Figure 4. Schematic view of the designed hand

5.2. Dynamic specifications of the constructed hand

In order to calculate the parameters of the artificial hand, the maximum pressures of one finger in fully extended and folded states were measured. Also, a frequency of the bending-straightening cycle was calculated. All results are presented in the table below.

Maxiumum presure of a finger fully bent	0,64 N
Maximum pressure of a straight finger (one opposing muscle per one finger)	0,78 N
Maximum pressure of a straight finger (two opposing muscles per one finger)	1,37 N
Maximum pressure of three fingers working simultaneously	1,91 N
Maximum frequency of the bending-straightening cycle	0,87Hz
Working frequency of the bending-straightening cycle	0,15Hz

Table 1. Dynamic parameters of the hand

1. Artificial McKibben Muscles

Pneumatic McKibben muscles were used for the first time in 1950 for testing of artificial limb replacements. They consist of a rubber tube covered by an inextensible nylon braid.

Figure 5. Schematic view of the air muscle

These muscles are characterized by a high force to weight ratio, which equals 400:1. For comparison, this ratio for a commonly used electric motor equals 16:1. Pneumatic muscles usually work under pressure between 0 and 5 bars. They can by directly attached to the construction or transfer the force by a system of tendons. Driving a lever is a basic application for artificial muscles. This situation is presented in picture Fig.6.

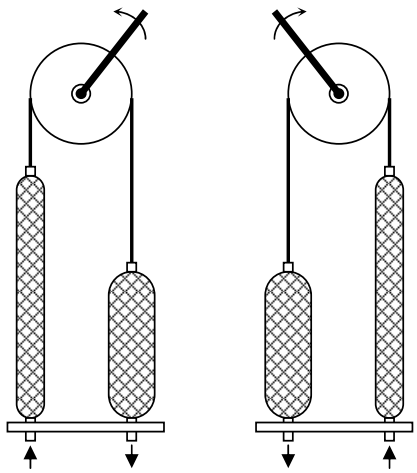


Figure 6. Lever movement driven by a pair of McKibben muscles

Since the artificial muscle can generate only a contracting force similar to real ones, a pair of such actuators is needed. One works as the agonist while the second one as the antagonist force generator

In order to generate movement of the lever to the left, the muscle on the left must be filled with compressed air. The right muscle must be empty during that same time. In order to generate the backward movement, an opposite situation must occur. The right muscle has to be filled while the left one releases air.

7. Considerations on the static parameters ofartificial muscles. (Chou C.P., Hannaford B., 1996)

An artificial muscle is an element in which pneumatic or hydraulic energy is transformed into mechanical energy.

In order to solve the relation between pressure and generating force, the rule of transformation of energy is taken into consideration.

When compressed air presses on the internal walls of the rubber tube, the gas performs work.

$$dW_{in} = \int_{S_i} (P - P_0)\, dl_i ds_i = (P - P_0) \int_{S_i} dl_i ds_i = P' dV \tag{1}$$

Where

P- inner pressure

P0- atmosphere pressure

P′ – absolute pressure

dli – increment of length

Si – total internal surface

dV – increment in volume

External work produces a force combined with theith shortening of the muscle length

$$dW_{out} = -FdL \tag{2}$$

Where:

F - tension

dL - displacementin length.

The principle of conservation of energy shows that the delivered work in the compressed air equals the exterior work.

$$dW_{out} = dW_{in} \tag{3}$$

Transforming equation (3), receive:

$$F = -P' \frac{dV}{dL} \tag{4}$$

In order to calculate the dV/dL some assumptions have to be made. of The first is that when expanding the braid of fibers the Volume of tube depends only on its length. The second is that the active part of the muscle is modeled as a perfect cylinder. This situation is shown on the picture Fig.7.

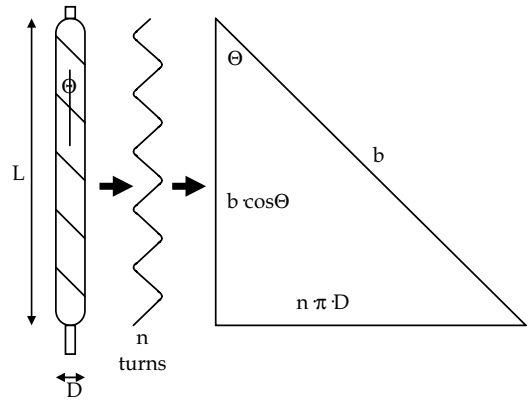


Figure 7. Schematic of the muscle's braid analysis

L – represents the length of the cylinder, D its diameter and the θ represents the angle between the longitudinal axis of the muscle and braid fibers. The character b is the length of the fiber and n is a number of rotations around the rubber tube. The parameters l and D may be calculated by equations

$$l = b\cos\theta \tag{5}$$

$$D = \frac{b\sin\theta}{n\pi} \tag{6}$$

On this basis the volume of the cylinder is represented by the equation (7)

$$V = \frac{1}{4}\pi D^2 L = \frac{b^3}{4\pi n^2}\sin^2\theta\cdot\cos\theta \tag{7}$$

Taking in to consideration the formula (4) and replacing there variable V, the equation described by (8) results in:

$$F = -P\frac{dV}{dL} = -P\frac{dV/d\theta}{dL/d\theta} = \frac{Pb^2\left(2\cos^2\theta - \sin^2\theta\right)}{4\pi n^2} = \frac{Pb^2\left(3\cos^2\theta - 1\right)}{4\pi n^2} \tag{8}$$

With some simplifications the formula of the force may be represented as (9)

$$F = \frac{\pi D_0^2 P'}{4}\left(3\cos^2\theta - 1\right) \tag{9}$$

Where d_0=b/nπ is the diameter, while the angle of the net is 90 degrees.
Taking into analysis equation (9), the maximum shortening occurs when the force F reaches the value 0. This takes place while the θ equals 54,7 degrees.

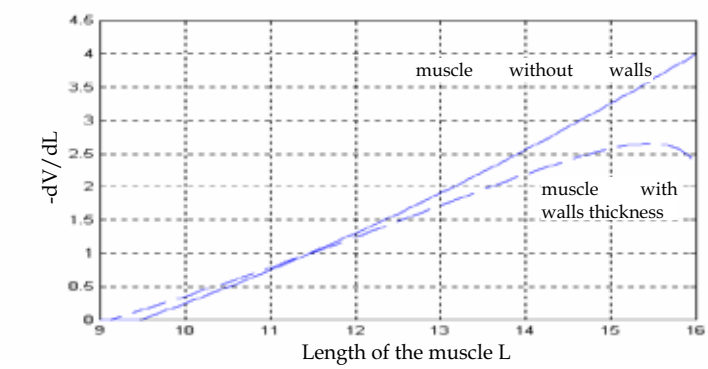


Figure 8. Theoretical characteristic of muscle force with/without walls taken into considerations

Considering the wall thickness as a parameter in all formulas, the final rule describing Force dependant of the pressure, may be represent by the equation (11)

$$V = \frac{1}{4}\pi\left(D - 2t_k\right)^2 L \tag{10}$$

Sample characteristics calculated with a b=16.4cm and n=3.15 are presented in Figure 8.

$$F = -P'\frac{dV}{dL} = \frac{\pi D_0^2 P'}{4}\left(3\cos^2\theta - 1\right) + \pi P'\left[D_0 t_k\left(2\sin\theta - \frac{1}{\sin\theta}\right) - t_k^2\right] \tag{11}$$

Analysis of equations (9)(11) shows that a force generated during statical expansion is proportional to the muscle shortening. This function descends with the shortening in length. It requires the initial condition of full extension of the muscle increasing the range of linear work. Application of artificial muscles in construction of the hand is a reflection of natural method of finger movement generation. Similar to Imitating the nature the presented hand has all muscles placed in a forearm part connected to fingers with tendons. All applying muscles hast their standard length of 150mm in a full extension. They shorten by about 30mm when they are fully filled up with compressed air. Nominal supplying pressure for them is 4 bars producing a force of about 30N. In this work they were supplied with 6 bars. This condition obviously increases the maximum force, but reduce the lifetime of the muscle.

8. Kinematics model of the artificial hand.

Creating the vision system in a control feedback loop is based on the knowledge of fingertip trajectories referred to as the base markers. All curved lines representing the paths of the makers can be calculated with a Denavit-Hartenberg notation (Jezierski E., 2002). This rule allows solving a forward kinematic task. A position of any point of a mechanical structure

can be found only with the knowledge of the angle of rotation joints and the linear translations between them.

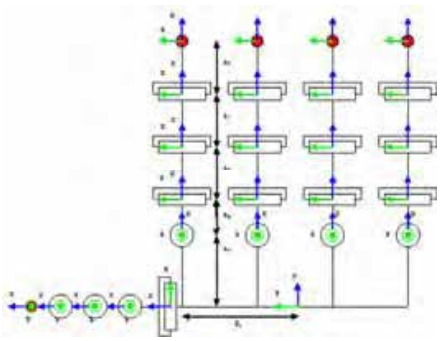


Figure 9. Kinematic model of the hand

The mathematical model presented on Fig.9 reflects the kinematic structure of the constructed hand. The joints of the wrist are ignored because they are not considered in calculations. The created hand has some simplifications in comparison to the presented model because some of the joints are blocked to reduce the number of Degrees Of Freedom.
In order to calculate the 3-D position of every joint, the transition matrix A has to be solved. Transition between two points in 3-D space described by translation and rotation can be calculated by the equation:

$$^{i-1}\hat{r} = A_i \, ^{i}\hat{r} \tag{12}$$

Matrix A represents four mathematical operations:
1. Rotation around Z axis
2. Translation along Z axis
3. Translation along X axis
4. Rotation around X axis

These operations may by represented by equation (13)

$$\begin{aligned} A_i &= \mathrm{Rot}(Z,\theta_i)\mathrm{Trans}(0,0,d_i)\mathrm{Trans}(a_i,0,0)\mathrm{Rot}(X,\alpha_i) = \\ &= \begin{bmatrix} \cos\theta_i & -\sin\theta_i & 0 & 0 \\ \sin\theta_i & \cos\theta_i & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & a_i \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\alpha_i & -\sin\alpha_i & 0 \\ 0 & \sin\alpha_i & \cos\alpha_i & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \end{aligned} \tag{13}$$

The final equation for the A matrix is as follows:

$$A_i = \begin{bmatrix} \cos\theta_i & -\sin\theta_i \cos\alpha_i & \sin\theta_i \sin\alpha_i & a_i \cos\theta_i \\ \sin\theta_i & \cos\theta_i \cos\alpha_i & -\cos\theta_i \sin\alpha_i & a_i \sin\theta_i \\ 0 & \sin\alpha_i & \cos\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (14)$$

Transformation from the 3-D reference position to an effector coordinates may be solved using (15):

$$^0\hat{r} = A_1(q_1)A_2(q_2)...A_n(q_n)^n\hat{r} \quad (15)$$

All dimensions and relations required to determine the finger tip position are presented in Table 1. All translations are given in millimeters and rotations are in radians.

Finger 1				Finger 2			
a_i	α_i	d_i	θ_i	a_i	α_i	d_i	θ_i
85	π/2	42	0	85	π/2	17	0
0	-π/2	0	0	0	-π/2	0	0
45	0	0	0	45	0	0	0
40	0	0	0	40	0	0	0
40	0	0	0	40	0	0	0
Finger 3				Finger 4			
a_i	α_i	d_i	θ_i	a_i	α_i	d_i	θ_i
85	π/2	-17	0	85	π/2	-42	0
0	-π/2	0	0	0	-π/2	0	0
45	0	0	0	45	0	0	0
40	0	0	0	40	0	0	0
40	0	0	0	40	0	0	0

Thumb			
a_i	α_i	d_i	θ_i
1	π/2	45	0
30	-π/2	0	0
45	0	0	0
40	0	0	0
30	0	0	0

Table 2. Kinematic parameters of the finger joints in Denavit-Hartenberg notation

The kinematic model is used to determine the points of finger tip trajectories. These points are calculated in two situations. First one shown in Fig.10.a) represents the side view of the

hand, while the second in Fig.10.b) represents the front view. Those curved lines are useful in vision analysis, because the whole operation of locating fingertip markers can be reduced to searching only along selected lines (Jezierski E. Zarychta D. 1995).
The real curves are only slightly different from the theoretical ones, which are obtained by assuming equal revolutions in all the joints. This difference however cannot be entirely neglected so appropriate corrections have to be made.

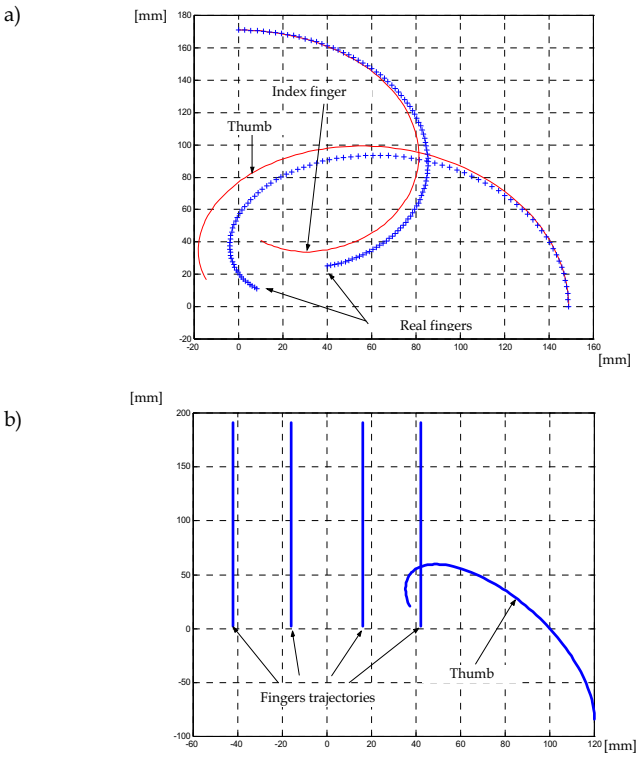


Figure 10. Model characteristics of fingers trajectories in side (a) and front (b) view

Trajectories of the fingertips are presented in Fig. 10. The main disadvantage of the frontal camera view is the lack of possibility to predict perspective distortions. Without these corrections modeling of the fingertips is a simplification and results in obtaining a very inaccurate approximation of the position of the finger tip markers.
Determining the trajectory of the thumb in a general case is complicated because its end traces a curved plane in 3-D space. This is the result of an additional joint, which allows rotation in the plane perpendicular to the palm surface. The shape of the curved plane marked by the thumb tip is presented on Fig.11.

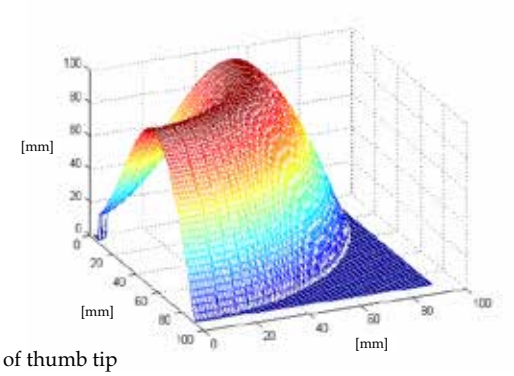


Figure 11. 3-D range of thumb tip

Vision control used for recognition of finger positions scans along specified trajectories instead of analyzing the whole image. This method speeds up the analysis process, allowing it to be implemented on slower computers. The trajectories calculated from the kinematical model are placing on the captured view. They are scaled to the hand size on the picture by two base markers.

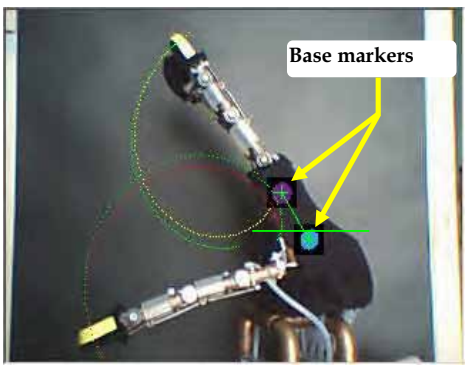


Figure 12. Artificial hand side view with base markers

The distance between these markers on the picture and in reality allows to determine the scaling coefficient (16)

$$k = \frac{\sqrt{(x_{b2} - x_{b1})(x_{b2} - x_{b1}) + (y_{b2} - y_{b1})(y_{b2} - y_{b1})}}{d} \tag{16}$$

$$x' = x \cdot \cos\theta + y \cdot \sin\theta \tag{17}$$

$$y' = -x \cdot \sin\theta + y \cdot \cos\theta \tag{18}$$

$$x'' = x_b + k \cdot x' \tag{19}$$

$$y'' = y_b + k \cdot y' \tag{20}$$

Where :

θ - angle between the line connecting the base markers and the horizontal line passing through the bottom marker

x_{b1},x_{b2},y_{b1},y_{b2} pixel coordinates of base markers on the screen.

Scanning lines are fit to the markers and scaled by the k ratio. Moreover, the base points are useful in calculations of hand orientation. The wrist of the hand can also be moved, so the tilt of the palm may change. Identification of the position of these markers is performed by scanning the area around them. If the cursor changes its position significantly inside the scanning range, a new position of the searching area is determined for next scan. Center of the region overlaps with the center of the base marker. This approach also reduces the number of operations required for vision analysis.

Increasing contrast in the captured view and placing bright makers on a dark background allows quick identification of their position. All trajectories are determined with a 1 degree resolution. Every marker commonly covers more than 2 points of these curves. Calculating the center of the cursor along the scanning line permits identification of every joint's angle.

9. Methods of grasp recognition

Two methods of grasp recognition were utilized (Kaczmarski M., Zarychta D., 2005). The first one focuses on precision and a soft touch. At each image refresh the distance between the edge of a marker and the nearest object point is measured. In the case when the angular distance is less than 4 degrees the signal for the confirmation of the grasp is sent. Such situation however is definitive only in a limited number of cases i.e. in the case of cylinders or cubes. The grasp of e.g. a cone or a pyramid when the base is directed towards the camera can cause a misinterpretation. In a side view only the two-point grasp can be applied. The main cause of this situation is the occluding of the fingers by the nearest one.

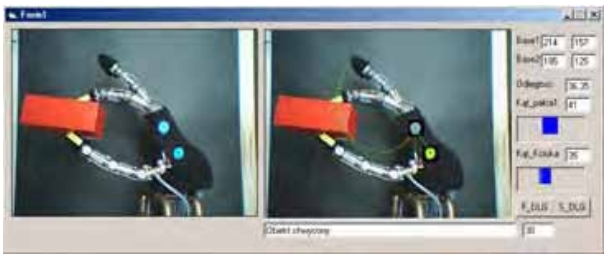


Figure 13. Control program interface

Another approach providing a strong and steady grasp is a time-position analysis. With every refresh of a captured view the positions of all fingers tip markers are calculated. If their position differs from the previous view, the compressed air is pumped to the muscle. If the finger stays in the same position despite the activation for further movement it means that an obstacle is located in its way. After a few ineffective attempts to continue the movement the situation is interpreted as a firm contact. The muscle stops filling up with compressed air. Time analysis allows a correct grasp in the case when the object is not placed directly on the scanning line. Blocking one of the finger sections by the object can

prevent observation ofthe changing of position of the finger tip, which the control algorithm will interpret as the grasping of the obstacle. This also causes the electro-valves to close.
This kind of approach is perfectly suitable for situations where the application of a larger force is needed. It can be used to move small obstacles or to push buttons.

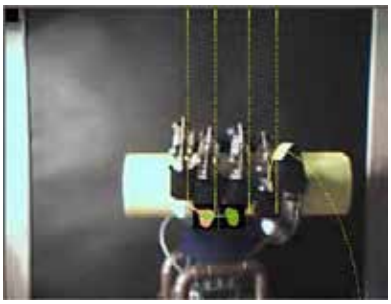


Figure 14. Grasping of a cylinder

Regarding the particular character of the created construction, only the type of grasp with distal segments can be used (Kang S.B., Ikeuchi K., 1992). Presenting considerations referring to the front view, when the camera is placing towards to the hand. This situation allows for shape analysis of the manipulated object, and adapting the finger position to the gripped body.
Foundation of this task was that the object can not rotate during catching, and the object is homogenous. These assumption causes that the grasp must be applied precisely.

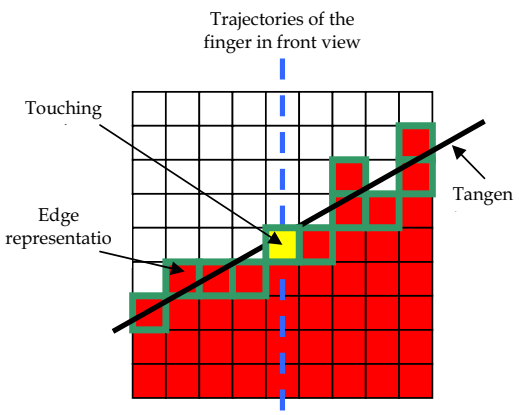


Figure 15. Edge analysis

Perfect for this task is the algorithm that measures the distance between the finger and the object. Applying it causes the fingers to gently touch the edge of the object, without having any influence on its position. Considering the small differences in the muscles and electro-valves constructions and shape of the object, the time of touching the object depends of the

finger construction and the distance between the object and fingertip. Other type of grasp, such as a time-position analysis, could cause a rotation of the grasped body, resulting in a different time of touching and an unbalanced distribution of force. When all fingers gently reach their position the force can be increased, because the distribution of force produces zero momentum of rotation. Only now the grasped object can be lifted.
In calculations of the touching points to the object the distribution of the moments of rotation is considered. The only information obtained from the captured view is the shape of the object and the position of the fingertip markers.
The center of mass of an object view can be calculated under the condition that the grasping body is homogenous. Only this assumption allowed for calculating the 2 dimensional representation of the position of an object's center of mass (Tadeusiewicz R., 1992). Assuming that all the fingers act with the same pressure the counter position of the thumb can be calculated.
Around the contact point, the tangent to the shape edge is calculated (Figure 15). The size of the area where the tangent is calculated is determined by a square with 9x9 pixels. Analysis of the pixels belonging to the object and in the contact point allows for calculating the tangent and the distribution of force. This force produces the torque. In all the calculations the method of the sum of least squares is used. This is represented by equations (21)(22)(23)(24)

$$y = a_0 + a_1 \cdot x \tag{21}$$

Coefficients a_0 and a_1 are determined by (21)(22)

$$a_0 = \frac{1}{D}\begin{vmatrix} \sum y_i & \sum x_i \\ \sum x_i y_i & \sum x_i^2 \end{vmatrix} \tag{22}$$

$$a_1 = \frac{1}{D}\begin{vmatrix} n & \sum y_i \\ \sum x_i & \sum x_i y_i \end{vmatrix} \tag{23}$$

$$D = \begin{vmatrix} n & \sum x_i \\ \sum x_i & \sum x_i^2 \end{vmatrix} \tag{24}$$

Where :
y_i,x_i are the pixel coordinates
n – number of pixels taken to calculations

Those calculations determined for every finger helps to predict the position of the thumb. Distribution of the force during grasping with a position of the thumb for various objects is shown on Fig.16.

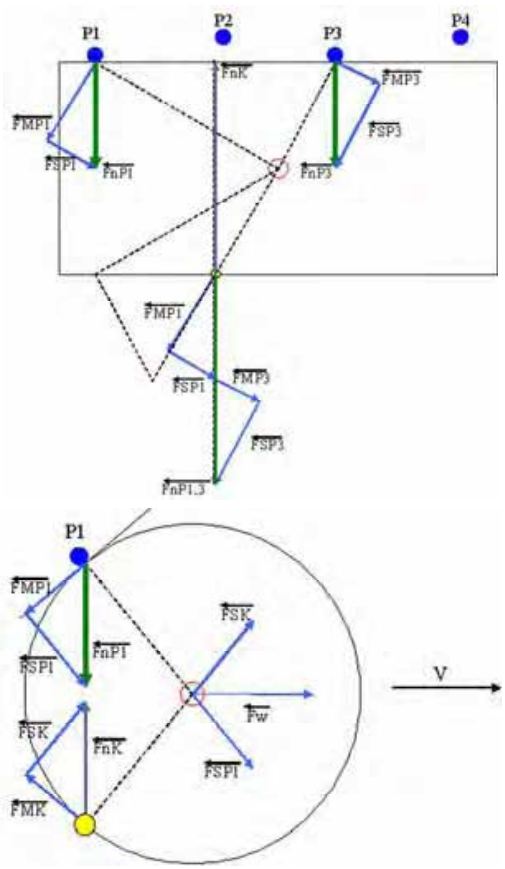


Figure 16. Distribution of forces acting on an object

Analysis of a rounded object with a two finger grasp shows that even if the torque equals zero, there is an unbalanced horizontal force which makes an object slip out from the grasp. To perform correct hold at least three fingers must stay in contact with the object. Moreover, they should be placed symmetrically to the center of mass. Only this kind of arrangement of finger positions allows the object to stays motionless during grasping.

10. Results of experiments

All situations presented on the figures 17-21 show the control program during grasp analysis for objects of different shapes. During this , two, three, or four fingers can participate. Screens show how the shape of the objects influences the fingertip positions. For a single shape only the final stage of grasping is presented.

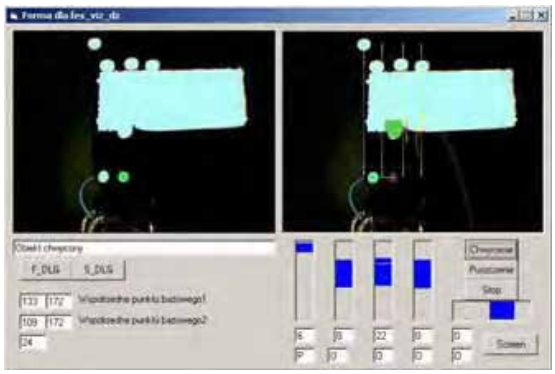


Figure 17. Grasping a cuboid

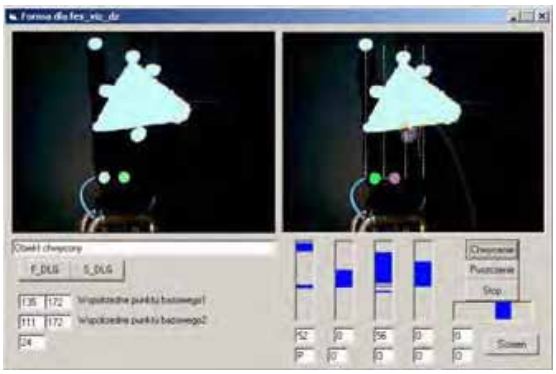


Figure 18. Grasping a cone

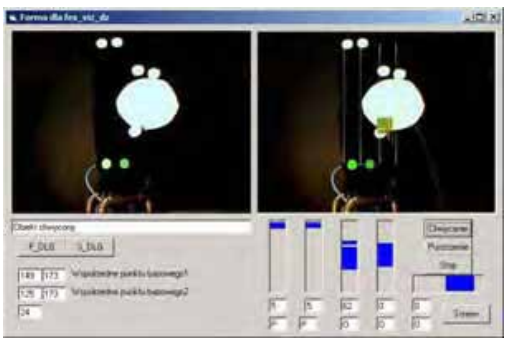


Figure 19. Grasping a sphere

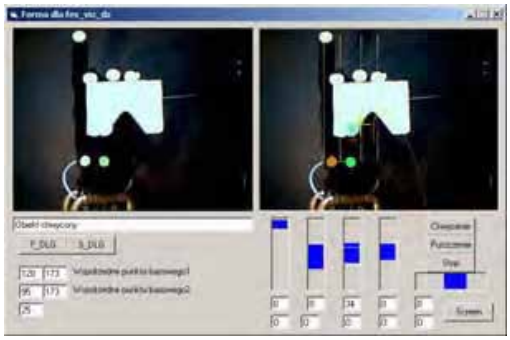


Figure 20. Adaptive grasp

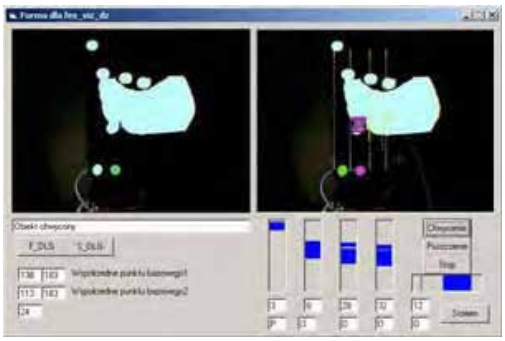


Figure 21. Adaptive grasp

Presented results show that the applied algorithm is well suitable for grasping of objects of unknown shape. It can be used with more than just the typical bodies such as a cuboid, cone, pyramid or a sphere. The presented pictures show that the fingers are able to adapt to the body in the way which doesn't produce a rotation during holding. Vision control which is put into practice has some disadvantages. In this work there was no correction for the perspective distortion of the camera. The deviation produced by the camera didn't significantly affect the control of the artificial hand's fingers' positions, so the distortion could be neglected.

11. Myopotential control for an artificial hand

Human hand including the wrist contains 22 Degrees Of Freedom. This number shows how complex are its manipulation capabilities. Research works are not only aimed at adapting artificial grippers into mobile platforms or humanoid robots, but they are also focused on prosthetic products for amputees of fragments of the upper limb. The mechanical construction of such prosthesis does not cause great problems, however the increasing of the number of degrees of the freedom complicates the control system. The prosthesis of the hand should be characterized by a number of degrees of the freedom being enough for the

realization of basic motor functions of hand, as well as it should be close with the shape and the weight to the lost fragment of the upper limb.
For persons after amputations, electromyographic signals of human muscles are using for steering the bioprosthesis. In most cases 4 classes of states of the hand are determined, but research works at the Technical University of Wroclaw allow for isolation of even 9 classes. Analysis of potential movement possibilities of artificial hand, shows that the construction of a hand containing a number of DOF similar to a real hand is impractical, because using only 9 states classes for the palm some of the joints would be useless. That's why most of the bioprostesis contains only three movable fingers or thumb with a pair containing two fingers in each group. This kind of widely applied three finer bioporsthesis has been made in the Oxford Orthopedic Engineering Center.

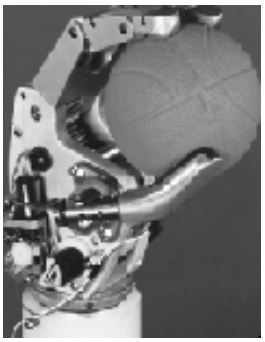


Figure 22. Bioprosthesis from the Oxford Orthopedic Engineering Center

(Oxford University Gazette, 1997) *"Designed to function as a prosthesis for people who do not have a hand, it is controlled by a small computer contained within it. The hand uses low-level signals generated by the user flexing muscles in the forearm, and translates them into action.*
Dr Peter Kyberd, a researcher at the Department of Engineering Science, who led the team which developed the hand, said: `The most important thing about a hand prosthesis is that it should be as easy to wear as a pair of glasses. You just put it on in the morning and use it without thinking about it.'
The user of the hand can give three basic instructions: open, close, and grip. What makes the device unique is that sensors inside it allow it to decide what grip shape to adapt and how hard to squeeze. If it feels an object slipping, it will automatically grip harder without the user's intervention."
The inspection of similar structures includes „Sensor Hand TM" of the company Otto Bock, the artificial DLR hand from Germany, Japanese structure from Complex Systems Engineering from the Hokaido University, and the Mitech Lab bioprosthesis from Italy.
Despite of anthropomorphic construction used for bioprosthesis (Wołczowski A., Kowalewski P., 2005) five fingers hnads for special tasks are also designed. Future robots will work together with humans, so their grippers must be adapted to operating human tools. Therefore their effectors must be similar to and as dexterous as a human hand. An example of such project is the NASA "Robonaut" robot (NASA RoboNaut Project), which is supposed to perform maintenance tasks outside the space station. His artificial hands are similar to human ones, and an advanced control algorithm permits for complex movements such as screwing on nuts.

Figure 23. "RoboNaut" hands – Nasa project

On account of the noninvasive character of muscle mipopotential measurement, an active electrode for biosignals recording has been designed (Clark J.W. 1978). Imperfection of this kind of recordings results from the activation of a wide group of muscles. An invasive method would be able to measure the activation of a single motor unit. Reduction of artifacts in the bioelectric signal is performed by placing contact electrodes close together (10mm), parallel to muscle fibers, that the interferences of other muscles don't significantly affect the measured signal. Additionally, an active electrode is supplied in a third contact point, which reduces the influence of external power line interferences.

12. Interference reduction by Right Leg Drive system.

Common mode voltage in measurement by a two contact electrode can be performed in several ways.

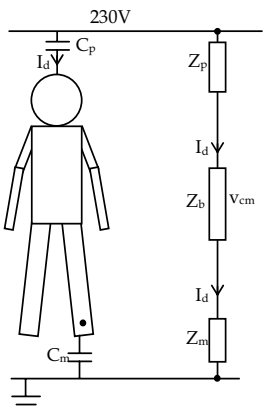


Figure 24. Simple electronic equivalent circuit of human body in electromagnetic field

Electric equivalent circuit of the human body surrounded by an electric field from power lines, shows that a leakage current flowing from the power line to the ground. This current is the consequence of a finite impedance Zp, which can be considered as a capacitance

between a power line and a human body. It flows through impedance of the body Zb and impedance between the body and the ground Zm, causing appearance of a common mode voltage on both surface electrodes. Both impedances Zm and Zp ale larger compared to the value of Zb. The amount of the common mode voltage depends on the proportion between Zp and Zm. In the worst case condition, when breakdown voltage of the power line occurs on the human body the reduction of the shocking impulse current depends only on the isolation Zm between the body and the ground. Unfortunately, to reduce the common mode voltage, this impedance should stay low in order for the potential of the ground of the electric circuit to be comparable to the ground potential of the examined patient. The best solution would be connecting the patient to the potential of the ground. However this situation require special conditions preventing electrical shocking. Moreover in this situation, the instrumentation amplifier's CMRR parameter (Common Mode Reduction Ratio), which should be very high between 100dB-120dB, can only reduce the value of the common mode voltage.

Today the common application for reducing the common mode voltage is the application of a negative feed-back loop (Metting van Rijn A.C., Peper A., Grimbergen C.A., 1990). The patient is connected by a third electrode where a negative and amplified Vcm voltage is presented. This solution allows for reduction of the common mode voltage without decreasing the value of the ground isolation Zm. The negative feed-back loop produces a reference point for biopotential measuring, the level of which is comparable to the ground level of the electric circuit. The electric equivalent circuit is presented on Fig.25.

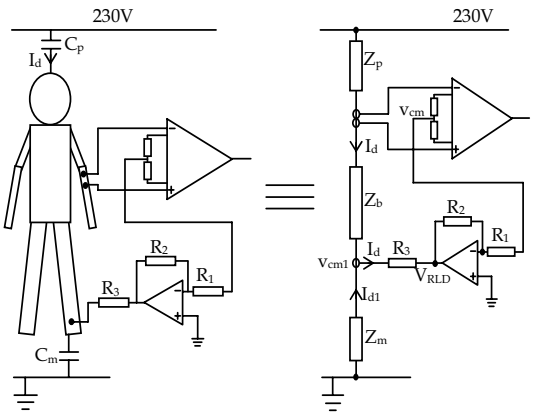


Figure 25. Electronic circuit for interference reduction

Considering relations (25)(26) in this circuit, a rule for reducing a common mode voltage occurs (31). It decreases as much as the gain value of feed back loop increases.

$$V_{cm} = I_d \cdot R_b + V_{cm1} \tag{25}$$

$$V_{cm1} = V_{RLD} + I_d \cdot R_3 \tag{26}$$

$$V_{RLD} = -V_{cm} \frac{R_2}{R_1} \tag{27}$$

$$V_{cm} = I_d \cdot R_b + V_{RLD} + I_d \cdot R_3 \tag{29}$$

$$V_{cm} - V_{RLD} = I_d \cdot (R_b + R_3) \tag{28}$$

$$V_{cm}\left(1 + \frac{R_2}{R_1}\right) = I_d \cdot (R_b + R_3) \tag{30}$$

$$V_{cm} = I_d \cdot (R_b + R_3)\left(\frac{R_1}{R_1 + R_2}\right) \tag{31}$$

Moreover this solution has also a safety advantage. When an electric shock occurs and the patient is isolated from the ground, the feedback amplifier saturates. On its output a negative potential occurs. The shocking current I_d flows through a resistance R_3 to the ground. Typically the value of R_3 is 390kΩ, which reduce this current to a safe value.

13. Description of an active electrode

In order of perform the best collection of the EMG signals, the conditional circuit should be placed as close as possible to the measuring pads. This circuit consist of an instrumentation amplifier and a second order band-pass filter. The high input impedance and a high value of CMRR 110-120dB characterize the used amplifier INA128. These two parameters are the key for a common mode voltage reduction. The band pass filter has a low cut-off frequency set to 10Hz and a high cut-off frequency of 450Hz. This range provides correct measurement in a full bracket. Cutting the low frequencies removes all artifacts which may appear as a consequence of small movements of the electrode pads on the skin. Moreover a high pass filter rejects the offset voltage and provides transformation to a fully bipolar type of the signal. The low-pass filter removes the high frequency signals which are produced by most electronic devices such as D.C. converters. These useless frequencies could be measured by an analog to digital converter and they may affect the frequency spectrum of the biopotentials.

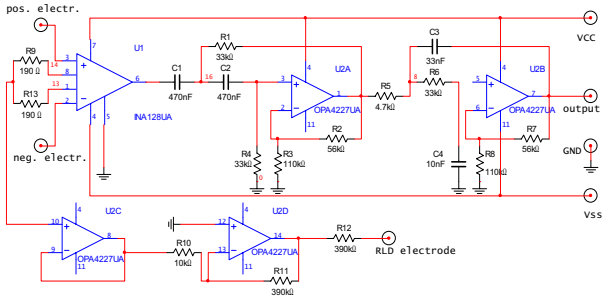


Figure 26. Electronic circuit of surface active electrode for biopotential recordings

Moreover a presented active electrode also contains an RLD circuit which is used for decreasing of the common mode voltage and which is connected to a third pad of the measuring unit. Picture 27 presents real view of the electrode and it's electrical circuit is shown on the figure 26. The size of the electrode is 20mm x 30mm. All 3 silver contact pads are 10mm long and 1mm wide. They are also separated by a distance of 10mm.

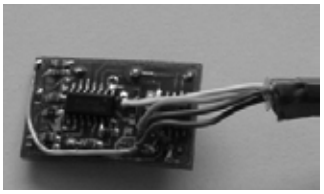


Figure 27. Real view of the active electrode

14. Description of the measuring path.

The active electrode is connected to a measuring path where the biopotential signal is further transformed and relayed to the computer. The most important part of this conditioner is the isolation unit. This device provides a safe level of voltage on the patient side. Its task is to isolate the electrode and the examined patient from potential electrical shocking. Additionally, this measuring path contains another low pass filter to increase the dumping value and a voltage level shifter on the input of the analog to digital converter. Since the amplitude of electomyographic signal is at a level of 500-1000□V , the conditioning path provides an amplification to raise the amplitude to the range of 500-1000mV.
The Amplitude-Frequency characteristic of an EMG measuring path is presented on the picture below.

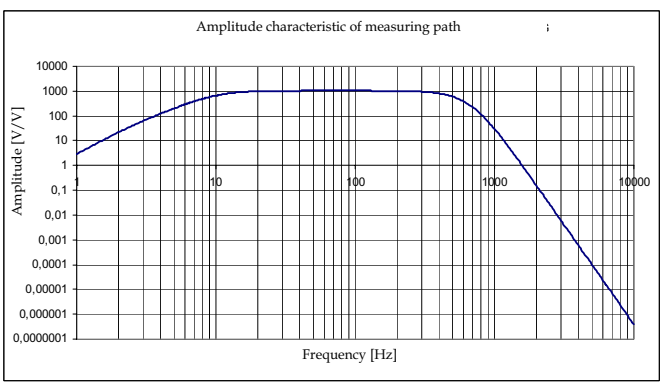


Figure 28. Filter frequency characteristic

Measuring of the biopotential signal with an active electrode put above the flexor digitorum superficialis muscle is presenting on the image Fig.29.

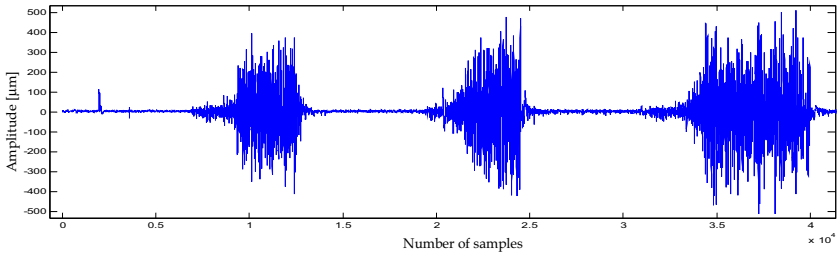


Figure 29. Electromyogram of examined muscle

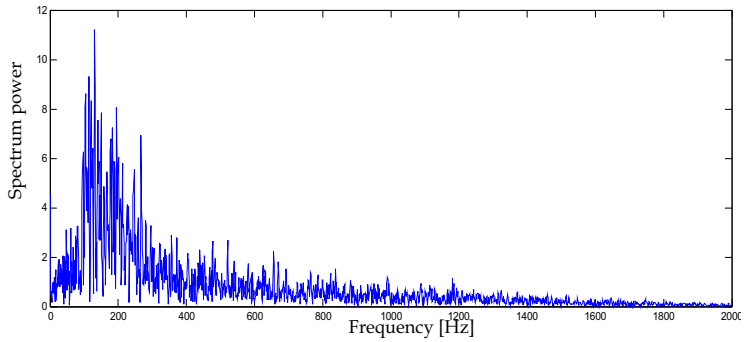


Figure 30. Fourier transformation of the muscle's response to stimulation

The control of the artificial hand by a biopotential signals is performed on the laboratory stand which diagram is presented on Fig.31.

An active electrode placed over the group of examined muscles measures the electric potential. Then the signal is amplified and filtered to the useful range of frequencies 10-450Hz. In the next step, the signal is acquiring by an A/D converter and filtered in order to remove the 50Hz power line frequency. This filtration is performed by a digital algorithm. After these operations, the control program calculates the root mean square value of the received waveform. This calculation is performed after every 10ms of received data. However, a spectrum analysis is performing with a 1s refresh. The RMS value is a basic information showing the level of muscle tension. This information is applied to the function, which converts the level of muscle activation into the air pressure level. Another control algorithm, which uses a two-step regulator with a dead zone, provides stabilization of the air pressure in the McKibben muscles. If the pressure is lower than a minimum threshold, the electrovalves pump air into the muscle. When the pressure exceeds the specified limits another electro-valve releases the air from the muscle. Experiments performed with this control method show that using only one electrode it is possible to obtain a simple close-open movement with a few different bending states.

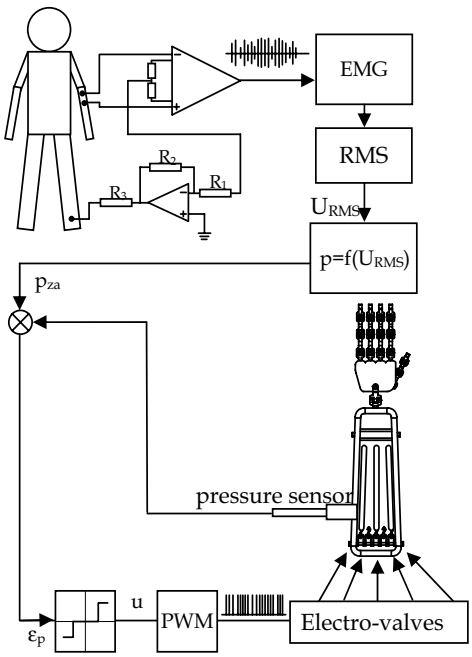


Figure 31. Laboratory stand schematic

15. Conclusions

This work presented aspects connected with the structure and control of a five finger, anthropomorphic gripper, working under the supervision of a vision system. The presented construction is characterized by some simplifications in comparison to the real human hand, as it has only 9 Degrees Of Freedom. However, those limitations do not reduce its grasping possibilities, which was shown in the paper. This hand had no problem with grasping objects of different shapes by adapting the position of fingertips to the shape of body edge. The mechanical construction has an additional DOFs which could make its manipulating possibilities closer to the human hand; however, they are blocked to simplify the algorithms of the vision control feedback loop. Examples presented in the paper based on some simplifications in the vision control algorithm. Perspective distortions were not addressed, because in the scanning line method where the curve trajectories are fitted to the hand view, the deviation value was negligible. The presented observation of the artificial hand for two locations of the camera cause big limitations of control. There is a possibility of the ambiguity of solutions in the thumb position analysis.

The designed and presented research position has essential teaching advantages, which allow for testing algorithms with vision control in the feedback loop in a wide range. Moreover, the artificial hand can be installed on a robotic arm, which increases its

manipulating possibilities. Another expansion of the artificial hand construction may be installation of additional sensors of force and temperature, which would make this hand more similar to a human limb.

The presented control method uses basic electromyographic signal analysis. Although it determines the tension of the examined muscle group it is not sufficient for full analysis of the examined patient's intentions. In the future works an attempt of computerized identification of strength of muscles will be taken. This identification will base on the multipoint EMG measuring system, which allows to determine correlation between the muscle groups in the different finger postures (Krysztoforski K., Wołczowski A., Busz S., 2005). The mathematical model created on this base permits much more precise steering and identifies the patient's intentions.

Figure 32. Real view of hopping robot leg

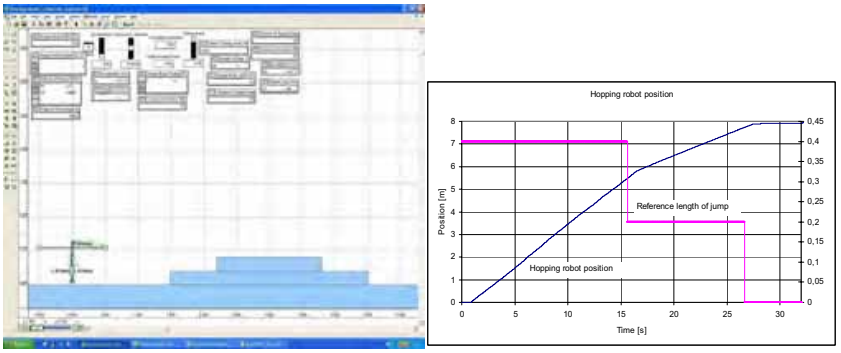


Figure 33. Simulations of jumps in uneven terrain and plot of Hopping robot position

Additionally, an attempt of leg muscle identification during a vertical jump will take place. This information will be used for a jumping robot (Fig.32.)(Fig.33), which task is freely jumping on uneven terrain performing human-like leaps. Tentative simulations show that a mathematical description of the hopping robot permits for balancing jump control of the mechanical equivalent of a human leg (Raibert M.H., 1986). This research was partially financed by Ministry of Science and Higher Education under grant No 3 T11A 023 30 and 3

T11A 024 30. Moreover, the created leg may also be adapted as an exoskeleton, which would support the walking process. For this task myopotential signals are used for identification of walking intensions of an examined patient.

16. References

Clark J.W. (1978), Medical instrumentation, application and design, Houghton Mifflin, Boston.

Jezierski E. (2002), Robotyka kurs podstawowy. Wydawnictwo PŁ, Łódź.

Raibert M.H. (1986), Legged robots that balance, The MIT Press.

Tadeusiewicz R. (1992), Systemy wizyjne robotów przemysłowych. WNT Warszawa.

Chou C.P., Hannaford B. (1996), Measurement and Modeling of McKibben Pneumatic Artificial Muscles. *IEEE Transactions on Robotics and Automation*, Vol. 12, No. 1, pp. 90-102.

Feja K., Kaczmarski M., Riabcew P. (2005), Manipulators driven by pneumatic muscles, CLAWAR *8th International Conference on Climbing and Walking Robots*, pp. 775-782, London UK.

Jezierski E. Zarychta D. (1995), Tracking of moving robot arm using vision system. SAMS, Vol. 18-19, pp. 534-546.

Kaczmarski M., Zarychta D. (2005), Vision control for an artificial hand, CLAWAR 8th *International Conference on Climbing and Walking Robots*, pp. 623-630, London UK.

Kang S.B., Ikeuchi K. (1992), Grasp recognition using the contact web. *Proc. IEEE/RSJ Int'l Conf. on Intelligent Robots and Systems*, Raleigh, NC.

Krysztoforski K., Wołczowski A., Busz S. (2005), Rozpoznawanie postury palców dłoni na podstawie sygnałów EMG, *Procedings of VIII-th National Conference of Robotics*, T2 pp. 213-220, Wrocław.

Metting van Rijn A.C., Peper A., Grimbergen C.A. (1990), High-quality recording of bioelectric events, *Med. & Biol. Eng. & Comput*, 28, 389-397.

Wołczowski A., Kowalewski P. (2005), Konstrukcja antropomorficznego chwytaka dla zręcznej bioprotezy dłoni. *Procedings of VIII-th National Conference of Robotics*, T2 pp. 193-202, Wrocław.

NASA RoboNaut Project. Available from: http://robonaut.jsc.nasa.gov/hands.htm, Accesed 2007-06-15

Oxford University Gazette. Artificial Hand goes on trial . April 24th 1997, Accesed 2007-06-15

9

"From Saying to Doing" – Natural Language Interaction with Artificial Agents and Robots

Christel Kemke
University of Manitoba
Canada

1. Introduction

Verbal communication is one of the most natural and convenient forms of human interaction and information exchange. In order for artificial agents and robots to become adequate and competent partners of humans, they must be able to understand and respond to task-related natural language inputs. A typical scenario, which poses the background of our research, is the collaboration of a human and an artificial agent, in which the human instructs the artificial agent to perform a task or queries the agent about aspects of the environment, e.g. the state of an object. The artificial agent has to be able to understand natural language inputs, as far as they concern actions, which it can perform in the environment. The artificial agent can be a physical robot, or a virtual, simulated agent.

Several projects have explored the development of speech and language interfaces for cooperative dialogues with agent-like systems, in particular TRAINS and TRIPS for cooperative route planning (Traum et al., 1993; Allen et al., 1995; 1996); CommandTalk as spoken language interface for military planning (Stent et al., 1999); Situated Artificial Communicators for construction tasks (SFB-360, 2005; Rickheit & Wachsmuth, 2006) and the CoSy project on human-robot interaction (Kruijff et al., 2007; Kruijff, 2006). Other projects dealing with spoken language interfaces include Verbmobil (Wahlster, 1997) and BeRP (Jurafsky et al., 1994).

Inspired by this work, we developed a basic architecture for natural language interfaces to agent systems for the purpose of human-agent communication in task-oriented settings. Verbal inputs issued by the human user are analyzed using linguistic components, semantically interpreted through constructing a formal representation in terms of a knowledge base and subsequently mapped onto the agent's action repertoire. Since actions are the central elements in task-oriented human-agent communication, the core component of this architecture is a knowledge representation system specifically designed for the conceptual representation of actions. This form of action representation, with an aim to bridge the gap between linguistic input and robotic action, is the main focus of our research. The action representation system is based on taxonomic hierarchies with inheritance, and closely related to Description Logic (Baader et al., 2003), a family of logic-based knowledge representation languages, which is becoming the prevalent approach in knowledge representation.

In order to define the formal action representation system, we combine aspects of action descriptions found in computer science (formal semantics for program verification; functional programming); mathematics (logic, structures, functions); linguistics (verb-oriented semantics; case frames); and artificial intelligence, especially formal methods in knowledge representation (Description Logic), planning and reasoning (STRIPS, Situation Calculus), as well as semantic representation and ontology.
The result of this endeavor is a framework suitable to:

- represent action and object concepts in taxonomic hierarchies with inheritance;
- instantiate object concepts in order to describe the actual world and the environment of the agent system;
- instantiate action concepts to describe concrete actions for execution through the agent/robot;
- provide structural descriptions of concepts in the form of roles and features, to connect to the linguistic processing components through case frames;
- provide a formal semantics for action concepts through precondition and effect formulas, which enables the use of planning and reasoning algorithms;
- support search methods due to the hierarchical arrangement of concepts;
- provide a connection to the action repertoire of artificial agent systems and robots.

We describe the formal structure of the action representation and show how taxonomies of actions can be established based on generic action concepts, and how this is related to object concepts, which are stored in a similar way in the hierarchy. The inheritance of action descriptions and the specialization of action concepts are illustrated in examples, as well as the creation of concrete actions. For further details on the exact formalization, we refer the reader to related literature discussed in section 2. We explain the basics of the natural language interface and the semantic interpretation through case frames. We outline the connection of the knowledge representation, i.e. the action (and object) hierarchy to the natural language interface as well as to the artificial agent system, i.e. arbitrary robots. An important aspect is the grounding of action concepts in the object world, which is crucial in case of physical agents. Other specific advantages of the action taxonomy relevant in this context are the support of retrieval and inference processes, which we will address briefly. In the last sections, we give an illustrative example using the CeeBot language and present some completed projects, which have been developed using the described interface architecture and the action representation methodology. Finally, we provide an informal assessment of the strengths and weaknesses of the approach and an outlook on future work.

2. Knowledge Representation

One core issue of our research is to provide a sound, formal framework for the representation of actions, which can be used as a basis for the development of adaptable speech and language interfaces for agent systems, in particular physical agents, i.e. robots. A crucial point in the development of such interfaces is a representation of the domain actions and objects in a format, which is suitable to model the contents of natural language expressions and to provide a connection to the agent systems as well.

2.1 Background and Related Work

The representational framework we discuss here is based on structured concepts, which are arranged in a taxonomic hierarchy. A formal semantics defines the meaning of the concepts, clarifies their arrangement in the hierarchy, captured in the subclass/superclass relationship, and the inheritance of descriptions between concepts within the hierarchy. The representational approach we use was motivated by KL-ONE, a knowledge representation formalism first proposed by Brachman and colleagues (Brachman & Schmolze, 1985), and is closely related to Term Subsumption Languages (Patel-Schneider, 1990) and the more recently studied class of Description Logic (DL) languages (Baader et al., 2003), which are both derived from KL-ONE. These representation languages are based on the same principles of structured concept representation in taxonomic hierarchies, focusing on a clearly defined formal semantics for concept descriptions, as well as encompassing the classification of concepts and the inheritance of concept descriptions within the hierarchy.

Description Logic languages were conceived in the first instance to represent static concepts and their properties. Dynamic concepts, like actions and events, which are defined through change over time, were considered only later in selected works. The problem of representing action concepts in DL and similar taxonomic representations is to provide a formal description of the changes caused by an action, which can be integrated into the formalism of the base language. Secondly, a clear formal semantics has to be provided and classification and inheritance algorithms have to be defined.

A first approach towards an integration of action concepts into KL-ONE and taxonomic hierarchies in general was developed by the author in the context of a help system for the SINIX operating system; the approach was also motivated through the development of an action ontology for command-based software systems (Kemke, 1987; 2000).

Other relevant work on action concepts in KL-ONE and DL employed a simpler, more restricted formalization of actions similar to STRIPS-operators (Lifschitz, 1987)., describing the effects of an action through ADD- and DELETE-lists; these lists are comprised of sets of literals, which are basic predicate logic formulas (Weida & Litman, 1994; Devanbu & Litman, 1996), which has been embedded into the CLASSIC knowledge representation system. However, there is no direct connection between the actions and an environment model; the representation of actions is not well-integrated into the representational system and thus the semantics of action concepts is not well-grounded. An extension and application of this approach, with similar drawbacks, has been described by (Liebig & Roesner, 1997). Other work on action concepts in DL dealt with composite actions and specified required temporal relations between actions and sub-actions, forming an inclusion or decomposition hierarchy (Artale & Franconi, 1994; 1998). The crucial issues of action classification and inheritance, however, were not addressed in this work. Di Eugenio (1998), in contrast, provided a well-designed, structured representation of action concepts, including relations to object concepts, for describing instructions in the context of tutoring systems. This form of DL based action representation is similar to the one developed independently by Kemke (Kemke, 1987; 1988; 2003), who outlines a formal semantics for action concepts based on the notion of transformation of world models. Baader et al. (2005) suggest another approach to model actions in DL but they start with a description of concrete actions, in a STRIPS-like fashion. While they provide a thorough theoretical analysis regarding the computational complexity of their algorithms, their approach suffers from the same lack of grounding as the earlier work by Litman and colleagues mentioned above.

2.2. Knowledge Representation and Ontology

When working on a certain application of a natural language interfaces to an agent system, the first task is to elicit and model relevant aspects of the domain and describe them in terms of the knowledge representation system. The outcome is a domain ontology, which is used to implement a knowledge base for the application domain. We focus here on knowledge representation through concepts, and distinguish *object concepts*, representing physical or virtual objects in the domain, and *action concepts*, which are modeled conceptually at various levels of complexity or abstraction and provide a connection to the human user and her verbal instructions (complex, abstract actions), as well as to actions of the robotic system (concrete, low level actions).

The structure of the domain is internally represented based on a knowledge representation format closely related to Description Logics, as outlined above. Classes of objects of the domain are modeled through "object concepts" and are represented in a hierarchical fashion, with general concepts, like "physical object", at higher levels, and more specific concepts, like "tool" and "screwdriver" at intermediate levels, and even more special object classes, like "flathead screwdriver size 8" at lower levels.

object:	screwdriver
superclass:	tool
type:	{flathead, philips, pozidriv, torx, hex, Robertson, triwing, torqset, spannerhead}
size:	[1,100]
color:	colors

The hierarchical connection between concepts is a pure superclass/subclass or subsumption/specialization relation. Concepts in such hierarchies typically inherit all properties from their higher level concepts, e.g. "screwdriver" will inherit all descriptions pertaining to "physical object". Therefore, such hierarchies are also referred to as "IS-A" or "inheritance" hierarchies. We prefer to call them "taxonomies".

2.3 Object Concepts

Descriptions of object concepts (and also action concepts) in the representational system contain in addition connections to other concepts in the hierarchy, used to reflect relationships between concepts, called "roles" in DL, consisting of relations between objects from either concepts, and attributes, also called "features", which represent functions applied to an object from one concept, yielding an object from the other concept as value. Examples of roles are the "has-part" relation between a physical object and its components, like "cylinder" as part of a "car-engine", or spatial relations between physical objects, like "besides" or "above". Examples of features of physical objects are "color" or "weight".

Certain roles and features of objects can be marked as fixed and not changeable, while others are modifiable and thus form a basis for action specifications. For example, in an automobile repair system, an engine needs a specific number of cylinders and thus the relation between the engine and its cylinders cannot be changed, even though a cylinder might be replaced with a new one. On the other hand, there are items which are not essential to the operation of the vehicle, like the backseats. The seat could be removed without influencing the car's basic operations. This distinction between essential and non-essential characteristics parallels the notion of fluents in situation calculus, whose values can

change due to the execution of an action. Persistent features are useful as reference points or baseline for assurances about the domain.

2.4 Action Concepts

The formalization of action concepts combines two representational forms, satisfying different demands: a case frame representation, which allows an easy connection to the linguistic input processing module, and a logic-based representation to model preconditions and effects of an action through special features, which is helpful to establish the connection to the agent system, since it supports action selection and execution processes and allows the use of standard planning and reasoning algorithms.

The format we developed for representing action concepts thus provides a frame-like, structural description of actions, describing typical roles pertaining to the action, like the "source" location of a *move*-action, or the "object" to be moved. This kind of information is derived from linguistics and can be easily used to present some form of semantics (Fillmore, 1968; Baker et al., 1998). Current ontologies, like WordNet (CSL Princeton, 2007) or Ontolingua (KSL, 2007), typically use such roles in structural descriptions of verbs or actions.

The example below shows the frame-structure for a *grab*-action, using a grasper as instrument. Parameters to this action are "agent", "object", and "instrument". The concepts to which these parameters pertain are shown in the specification, e.g. the agent is a robot and the instrument is a grasper. The concept "liftable-object" denotes physical objects, which are not attached nor non-moveable but can be picked up.

grab <agent, object, instrument>
agent: robot
object: liftable-object
instrument: grasper

For formal reasoning and planning methods, however, this representation is insufficient. We thus provide in addition a formal semantics of action concepts using preconditions and effects. Actions are then described in a format similar to functional specifications, with a set of parameters, which refer to object concepts, their roles, features and possibly values for them. The functionality is specified through restricted first-order predicate logic formulas, where the precondition-formula constrains the set of worlds, in which the action is applicable, and the effects-formula describes the resulting modification of the earlier world state. These formulas involve the parameter objects as predicates, functions, and constants, and describe changes caused by the action as modification of role or attribute values of the affected object. We complete the *grab*-action from above with those formulas:

The precondition-formula above states that the grasper is not holding any objects, i.e. the relation named "holds" for grasper is empty, and that the grasper is close to the object, modeled through the fuzzy spatial relation "close". Relations are stored explicitly in the knowledge base, and this condition can be checked instantly, if an action is to be executed.

grab <agent, object, instrument>
agent: robot
object: liftable-object
instrument: grasper
precond: holds (grasper, _) = ∅ ∧ close (grasper, liftable-object)
effect: holds (grasper, liftable-object)

The effect of the action is that the grasper holds the object. The terms "grasper" and "object" have the status of variables in these formulas, which will be instantiated with a specific grasper (of the respective robot) and the specific liftable-object to be picked up.
It should be noticed that the detailed modeling of actions, their preconditions and effects, depends on the domain and the specific robot. If, for example, a robot of type "roby" can pick up only items, which weigh less than 500 grams, we can specialize the action above accordingly for a "roby" type robot:

grab <agent, object, instrument>
agent: roby
object: liftable-object
instrument: grasper
precond: holds (grasper, _) = ∅ ∧ close (grasper, liftable-object)
∧ weight (liftable-object) < 0.5kg
effect: holds (grasper, object)

The occurrence of a specific *grab*-action, e.g. "grab <roby-1, sd-1, grasper-2>" involves the specification of the parameters, i.e. a specific robot named "roby-1" of type "roby", a specific liftable-object with identifier "sd-1", and a specific "grasper" named "grasper-2" of roby-1. The referenced object classes or concepts, like "grasper", specify the possible role fillers or function values; this is checked, when the action is instantiated. It will not be possible, for example, to grab an object, which is not classified as "liftable-object", or use a "wheel" instead of a "grasper".
For the action execution, the formula describing the precondition is checked, using the specific instances and values of the parameters. With these values, the execution of the action can be initiated, and the result is determined through the effect-formula. The effect-formula describes the necessary modifications of the knowledge base, needed to update the actual world state.
It is worth mentioning that, in the case above, we do not need to retract the formula "holds(grasper,_) = ∅" as is the case in standard planning or reasoning systems, like STRIPS, since these formulas are evaluated dynamically, on-demand, and the formula "holds(grasper,_)" will not yield the empty set anymore, after the object has been picked up. This is one of the significant advantages of the connection between actions and objects, which our approach provides.
The action representations in the ontology are described in such a way, that some level or levels of actions correspond to actions in the agent system, or can be mapped onto actions of the agent system. Since the action format defined and used in the ontological framework here includes precondition and effect representations, it allows planning and reasoning processes, and thus enables the connection to a variety of agent systems with compatible action repertoires.

2.5 Objects, Actions, and World States Concepts – Grounding and Restricting

As we have seen above, action concepts are arranged in a conceptual taxonomic hierarchy, with abstract action concepts at the higher levels and more concrete and detailed concepts at lower levels. Action concepts themselves are structured through features and roles, which connect them to object concepts. Object concepts model the physical environment and serve as parameters to actions, as in ADL (Pednault, 1989). They refer to those kinds of objects,

which are affected by the execution of an action. The restriction of parameters of actions through references to object concepts also grounds action specifications, i.e. connects them to the physical environment model, and helps to avoid absurd action specifications, which can never be fulfilled.

Instantiated objects are derived from respective object concepts, and they have specific values for features and roles, instead of references to other concepts. They represent concrete domain objects, and are also part of the knowledge base. This part is often referred to as the A-Box (Assertions), which stores information about facts and individuals in the world, in contrast to the T-Box (Terminology), which stores conceptual, terminological knowledge. The set of instantiated objects corresponds to the current world state. Effects of an action can be described through changing the feature or role values of these objects. We use special features for preconditions and effects, which have restricted predicate logic formulas as values. These formulas are constructed using only items from the knowledge representation system, i.e. all predicates, functions, variables, and constants refer to terms in the knowledge base, like concepts, features, roles, and their values.

This allows the description of dynamic effects of actions as changes to the current world state, on a conceptual level for generic action concepts as well as on a concrete level for instantiated actions to be executed.

For further details on the formal semantics of action taxonomies, we refer to (Kemke, 2003), and for a related semantic based classification of action concepts and discussion of the ontology to (Kemke, 2001); implementation examples can be found in (Kemke, 2000; 2004; 2006).

3. Global System Architecture

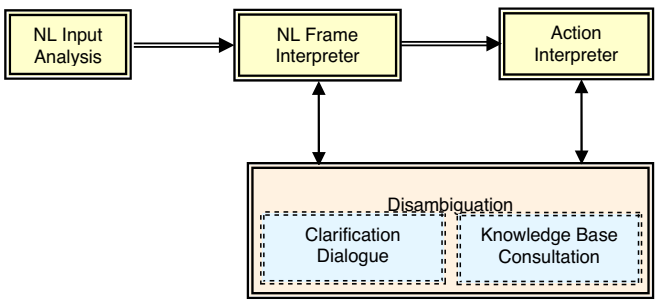


Figure 1. Global System Architecture

The processing within the general system architecture, which connects the natural language interface to the agent system, is illustrated in figure 1. It shows the main components of the system, including the *Natural Language Input Analysis*, the *Natural Language Frame Interpreter*, and the *Action Interpreter*. In addition, two modules for resolving ambiguities in the verbal input or in the stated request are shown: the *Clarification Dialogue* module, and the *Knowledge Base Consultation*.

3.1 Overview of the Processing

The natural language inputs considered here, which manifest the verbal communication with artificial agents, comprise typically commands, questions, and possibly declarative statements, as well as questions and confirmations as part of clarification dialogues between the system and the user.

The natural language input is firstly processed with a standard natural language parser, based on the Earley algorithm. We developed a base grammar, which specifically models the above mentioned sentence types and related sentence structures. The chosen sentence types reflect the essential speech acts needed for the intended human-agent communication, like *command*, *question*, *statement*. They are similar to and can be considered a subset of the *Agent Communication Language (ACL)* suggested by FIPA (2002).

The structural representation of the verbal input is generated by the parser using typical grammatical constructs, which elicit noun phrases, verb phrases, prepositional phrases etc. This structural representation is then processed in a shallow syntactic-semantic analysis, which selects and determines the contents of a case-frame representation of the linguistic input, based on the sentence type and the main syntactic constituents of the sentence, i.e. the subject noun phrase, object noun phrase(s), prepositional phrases for locations etc. plus an indicator for the queried part of the frame in case of questions.

3.2 NL Input Analysis

We discuss the natural language input processing using the following example command sentence:

"*Bring the red screwdriver from the living-room*." (example 1)

A command sentence is typically characterized through a leading verb, followed by complements to this verb, like an object noun phrase; prepositional phrases further describing the object or action, like the object's location or the source or destination of an action, e.g. from where or to where to bring something.

For the example sentence above, the NL Input Analysis yields the following structure:

sentence-type: command
verb: bring
direct-object: NP1 (det *the*) (adj *red*) (noun *screwdriver*)
source: PP1 (prep *from*) (det *the*) (noun *living-room*)

The *parser* accesses during processing a *lexicon*, which stores words relevant to the domain and their synonyms, in order to map different words with similar meanings to a standard form. The major part of the lexicon is constituted by verbs, nouns, adjectives, prepositions, and adverbs. The lexicon contains also information about necessary and optional complements to verbs, reflecting the so-called "verb subcategorization". Verb-categories together with required, optional or no complements are modeled through grammar rules, and this information is taken into account during parsing. For example, verbs like "*bring*" require a direct object or "theme" (what to bring) and an explicitly or implicitly specified destination or "recipient" of the bring-action (where-to or to whom to bring the object); furthermore, "*bring*" can have in addition an (optional) source specification (from where to bring the object).

3.3 Frame Interpreter

This structure is now being used by the *Frame Interpreter* to construct a case-frame representation, which brings the input closer to a semantic representation and a mapping onto the knowledge base. In the example above, which is related to some natural language

action: bring
theme: screwdriver(sd-1) ∧ color(sd-1)=red
location: living-room
destination: location(speaker)

interfaces we developed for simulated household robots (see section 5), the direct-object or *theme* is specified through the phrase *"the red screwdriver"*, the location of the object and source of the *bring*-action is the living room, and the destination is implicitly assumed to be the speaker's location.
The term "sd-1" acts like a logic constant and refers to a specific knowledge base object, representing the screwdriver addressed in the command. Descriptions of objects like "screwdriver" with features like "color" are provided through object concepts and their descriptions through roles and features in the knowledge base; they serve in the action-description as predicates (for concepts and roles) and functions (for features), respectively.

3.4 Disambiguation

The case frame representation might be ambiguous and allow several interpretations, which have to be resolved by the NL Frame Interpreter or the Action Interpreter, in order to derive one unique interpretation of the NL input in terms of the knowledge base. We encounter ambiguities or under-specification in several forms. On the linguistic level, we find lexical and structural ambiguity. Lexical ambiguity refers to the phenomenon that a single word can have different meanings; for example, "bank" can refer to a bench to sit on, or to an institution, which deals with money (like "Royal Bank of Canada"), or to a specific building, which houses a branch of such institution (e.g. the Royal Bank building on Pembina Highway).
In example 1 above, the word "screwdriver" could refer to a tool or to a cocktail drink. Which meaning is the preferred one can depend on the task domain. If our system is supposed to support trades people in the construction phase of a house, the interpretation of "screwdriver" as tool would be obvious. This restriction can be modeled on the lexical level, by using a preferred word set to match the input words, and the lexical mapping mentioned above would take care of this ambiguity by excluding unlikely word meanings. If this is, however, not the case, we can use other context information to resolve the ambiguity. If we assume, for example, that we deal with a homeowner doing renovation work, the interpretation of "screwdriver" as a tool might be more suitable - although this is not necessarily the case, since the homeowner might be stressed out from her renovation work and is in need of an alcoholic beverage.
A second way to resolve this ambiguity is to check the "world", i.e. we can consult the knowledge base and see whether it can find a matching item (drink or tool) in the living-room, or the robot could start a search process to figure out, whether it can find one or the other in the living-room. The additional information from the input sentence that the screwdriver is red can be used to identify the respective object in the current world model, which is the part of the knowledge base consisting of all object instances.

An easier way, which can also be applied if there is more than one "red screwdriver" in the living-room, and the system cannot determine, which one is meant, is a clarification dialogue with the user, in which the system can ask for further information, which it can use to identify the screwdriver.
The phrase "from the living-room" could be considered as ambiguous, since it can be either attached to the object-NP (i.e. the screwdriver from the living-room) or considered as part of the verb-NP, serving as modifier of the bring-action (bring _ from the living-room). Such structural requires a clarification dialogue with the user, where the system asks the user for the correct interpretation, or alternatively the system can pick either of the options.

3.5 Action Interpreter

The system thus constructs a complete instruction in case frame format from the NL input, which can then be translated into an action to be executed by the robotic agent. The processing includes several reasoning steps, and the result is an instantiated frame representation of the action, with references to necessary parameter-objects involved in the performance of the action. This representation can be passed over to the agent-system for execution. The agent system might still have to do further planning or reasoning processes but the action-frame is provided in such a format, that any agent system based on standard action-representations with pre-conditions and effects specified in a logic format should be able to deal with it. In the case of physical agents, like robots, they can be connected using an additional controller or interpreter, which transforms this representation and generates actions in the specific format required by the agent or robot.
Obviously, the action description must be fitted to the robot's action repertoire; further detail may have to be added or the action may have to be decomposed into smaller sub-actions, in order to provide a precise instruction (or set of instructions), which are in the robot's action repertoire and can thus be performed by the robot. We developed an algorithmic approach to work with integrated action abstraction and action/plan decomposition hierarchies (Walker, 2004; Kemke & Walker, 2006), in particular a smart search algorithm, which accesses simultaneously both hierarchies, in order to construct a suitable plan by combining pre-defined plan schemata from the decomposition hierarchy and generic actions from the abstraction hierarchy. The idea is essentially to check actions in the abstraction hierarchy, whether they fulfill a given task or goal description, and to switch over to the plan hierarchy, if no primitive action can be found; vice versa, if a rough plan can be located in the decomposition hierarchy, it can be instantiated with more specific actions taken from the abstraction hierarchy. This approach combines the advantages of using pre-defined plan schemata to model complex actions in an easy and efficient way, and the well-structured organization of abstraction hierarchies, which require significantly less time for search processes than standard lists.

4. CeeBot Example

An example of mapping actions to the instructions of a virtual robot using the CeeBot language is shown below. CeeBot is an educational system, developed by Epsitec SA for teaching programming languages to children and adults of all ages, using virtual robots as application (Epsitec SA, 2007). The CeeBot system is suitable as a test application for

knowledge representation and natural language interfaces to robots, since it is easy to use but still sufficiently complex. It is also very inexpensive compared to real physical robots.
The CeeBot language provides, for example, an action "turn left", which is realized through an instruction to the motors of the left and right wheels. A left turn can be expressed in CeeBot as follows:

action: turn left *instruction:* motor(-1,1)	*left motor reverse, right motor forward*

In our action representation formalism, we can be represent "turn" in various forms, as shown in the small action hierarchy below.
The first action-concept implements a generic *turn*-action with two possible values for the direction of the turn, which can be left or right. The turn changes the orientation of the robot, which is given in degrees in a base coordinate system, with a still unspecified value. In the knowledge base, the orientation would be a part of the description of a robot using a feature (function).

action:	turn <direction>	*generic low level turn-action;* *"direction"-parameter with* *values 'left' or 'right'*
direction:	{left, right}	
precond:	orient (agent) = x	
effect:	orient (agent) = y	

The two action-concepts below are specializations of the turn-action above. The values of the direction have been further restricted, for one action to "left", for the other action to "right". This describes the action concepts "turn left" and "turn right". The effect can now be determined precisely by adding or subtracting 90 degrees from the original value of the robot's orientation.

action:	turn <left>
direction:	left
precond:	orient (agent) = x
effect:	orient (agent) = x+y ∧ y=-90

action:	turn <right>
direction:	right
precond:	orient (agent) = x
effect:	orient (agent) = x+y ∧ y=90

A more general form of representing turns, which can also be used to derive the above specifications, is to allow values between -180 and +180 degrees for a turn.

action:	turn <direction>
direction:	[-180, +180]
precond:	orient (agent) = x
effect:	orient (agent) = x+y

This specification can also be used to describe "fuzzy expressions", for example, a "*turn slightly left*", using a fuzzy mapping yielding a value between 0 and -90 degrees for the direction of the turn.
The representation can also include a decomposition of complex actions, i.e. a representation of complex actions and their decomposition into a sequence of simpler sub-actions as addressed in section 3.4. For example, if the human "master" instructs the robot to bring a certain object, the action generated to describe this task on a higher level could be: bring <robot, object, master>; this can be decomposed into a more detailed, finer grained action sequence, e.g. goto <robot, object>; grab <robot, object>; goto <robot, master>.

action:	bring <robot, object, master>
agent:	robot
theme:	object
recipient:	master
precond:	none
effect:	close (robot, master) ∧ holds (robot, object)

The decomposition includes the following sub-actions:

action#1:	goto <robot, object>
agent:	robot
destination:	location (object)
precond:	none
effect:	close (robot, object)

action#2:	grab <robot, object>
agent:	robot
theme:	object
precond:	close (robot, object)
effect:	holds (robot, object)

action#3:	goto <robot, master>
agent:	robot
destination:	location (master)
precond:	none
effect:	close (robot, master)

The final outcome is that the robot is close to the master "close(robot, master)" and holds the object "holds(robot, object)". The formula "close(robot, object)", which is the effect of the first action, will be overwritten due to the effect of the last action "close(robot, master)".

5. Complete Prototype Systems

Several prototype systems have been developed with different degrees of complexity regarding their natural language capabilities, and varying levels of sophistication regarding their knowledge representation and reasoning components. All these systems have in addition been equipped with a speech input and output facility, except for the toy car - which does not yet speak.

5.1 Interior Design System

This system is a prototype of a combined natural language and visual scene creation system, inspired by the WordsEye system, which generates images based on verbal descriptions (Coyne & Sproat, 2001). The current implementation simulates an interior design system, which models actions and questions related to the positioning of furniture in a room, and includes the addition of furniture to the scene as well as the removal of furniture from the scene. The system works with speech input processed through the DragonTM speech recognition system. It performs a syntactic and semantic analysis of the recognized verbal input and produces a *request-frame* representing the user's query or command. The query or command is processed through accessing a Description Logic style knowledge base.

The system is implemented in two versions, one with a knowledge base implemented in PowerLoom[1], and the other one with a knowledge representation and reasoning module in CLIPS[2]. Both represent domain objects and actions in a conceptual framework based on inheritance hierarchies. In addition, the knowledge representation features spatial relations and respective spatial rules and constraints, like transitivity or asymmetry of spatial

[1] www.isi.edu/isd/LOOM/LOOM-HOME. html

[2] www.ghg.net/clips/CLIPS.html

relations and constraints, e.g. that large objects cannot be put on top of small objects. The system also integrates commonsense rules, like a preference to specify relations in the natural language output with respect to static inventory like windows, doors or a fireplace.

5.2 Virtual Household Agent

One of the earliest prototype projects is a simulated household agent, which interprets and performs tasks issued by a user in written language. The household agent moves in a virtual world, which includes, among other things, a fridge, a dishwasher, a TV, a sofa (for the "Master"), a mop and a bucket (for cleaning the floor), and random dirt. The agent operates in auto-mode, as long as there is no user input, and will do regular chores like cleaning the floor. The agent has some kind of "self-awareness" and returns to its charging station whenever it notices that its batteries are low in power. The user (Master) can issue command statements in natural language to the agent, for example, to switch on the TV or get a drink or food item, like "Bring me a glass of milk".

In case of a verbal command input, the household agent analyzes the natural language instruction and builds a case-frame-like representation. Based on the essential components of this frame-representation, in this example the action "bring", with destination "Master", and the object "glass of milk", the agent finds respective actions in the conceptual action hierarchy and develops suitable plans, if necessary. Using the action and the object hierarchies, the agent determines a higher level action "bring", and fills respective features of this action with information from the case frame. In this example, the object of the bring-action is instantiated as a glass containing milk, e.g. a yet unspecified instance "glass#1" of type "glass", which is a sub-class of the concept "container" with feature "contents = milk". This information can be inferred from the object hierarchy. The bring-action involves two sub-actions: one is to get the respective object (i.e. a glass of milk) and the other one is to deliver it (i.e. to the Master). In order to fulfill the preconditions of the generic bring-action, the agent must have the object: "have(agent, object)" or, with unified variables: "have(agent, glass#1)", and the agent must be at the destination location, i.e. the Master's sofa, to deliver it. In order to fulfill the first condition, the agent performs a get-action, and thus starts a planning process to get a glass of milk, which involves going to the dishwasher, picking a glass from the dishwasher, going to the fridge and filling the glass with milk. The precondition of the deliver-action states that the agent has to be at the Master's location. Thus, the agent plans now a route to the location of the Master (which is the sofa) and moves there. The actions referenced at this level, e.g. "get a glass from the dishwasher", are translated into executable actions for the household agent. The execution of the move itself is based on a planned route involving obstacle avoidance and is similarly assumed to be executable by the robotic agent.

5.3 Speech Controlled Toy Car

A speech interface for a remote controlled toy car has been developed using the DragonTM speech recognition software. The speech interface for the remote controlled toy car allows the user to issue spoken commands like 'forward', 'right', or 'north', 'east' etc. instead of using the manual remote control. The verbal speech commands are hooked to menu items in a window which is again connected to a parallel output port. Signals arriving at this port are transmitted through a custom-made electronic switching circuit to the remote control. This system is supposed to be extended with a more complex natural language interface and

command interpreter, which allows the processing of complex command sentences like "Make a slight turn right and then stop."

6. Conclusions

In this paper we presented a framework for action descriptions and its connection to natural language interfaces for artificial agents. The core point of this approach is the use of a generic action/object hierarchy, which allows interpretation of natural language command sentences, issued by a human user, as well as planning and reasoning processes on the conceptual level, and connects to the level of agent executable actions. The linguistic analysis is guided by a case frame representation, which provides a connection to actions (and objects) represented on the conceptual level. Planning processes can be implemented using typical precondition and effect descriptions of actions in the conceptual hierarchy. The level of primitive actions (leaf nodes of this conceptual hierarchy) connects to the agents' executable actions. The level of primitive actions can thus be adapted to different types of physical agents with varying action sets.
Further work includes the construction of a suitable, general action ontology, based on standard ontologies like FrameNet (ICSI, 2007), Ontolingua (KSL, 2007), Mikrokosmos (CRL, 1996), Cyc (Cycorp, 2007), or SUMO (Pease, 2007; IEEE SUO WG, 2003), which will be enriched with precondition and effect formulas. Other topics to be pursued relate to communication of mobile physical agents (humans) in a "speech-controlled" environment. The scenario is related to the "smart house" but instead of being adaptive and intelligent, the house (or environment) is supposed to respond to verbal instructions and questions by the human user. A special issue, we want to address, is the development of a sophisticated context model, and the use of contextual information to resolve ambiguities in the verbal input and to detect impossible or unreasonable actions.

7. Acknowledgements

The prototype systems described here have been developed and implemented by the author in co-operation with graduate and undergraduate students. Thanks to all of them for their participation and enthusiasm. This work has been partially supported by the Canadian Natural Sciences and Engineering Research Council (NSERC).

8. References

Allen, J. F.; Miller, B. W.; Ringger, E. K. & Sikorski, T. (1996). A Robust System for Natural Spoken Dialogue, *Proceedings of the 34th Annual Meeting of the Association for Computational Linguistics (ACL'96)*, pp. 62-70, Santa Cruz, California, 1996

Allen, J. F. ; et al. (1995). The TRAINS project: A Case Study in Defining a Conversational Planning Agent, *J. of Experimental and Theoretical AI*, 7, 1995, 7-48.

Artale, A. & Franconi, E. (1998). A Temporal Description Logic for Reasoning about Actions and Plans, *Journal of Artificial Intelligence Research*, 9, pp. 463-506

Artale, A. & Franconi, E. (1994). A Computational Account for a Description Logic of Time and Action, *Proceedings of the Fourth International Conference on Principles of Knowledge Representation and Reasoning*, pp. 3-14, Bonn, Germany, 1994

Baader, F.; Calvanese, D.; McGuinness, D.; Nardi, D. & Patel-Schneider, P. (Eds.) (2003). *The Description Logic Handbook*, Cambridge University Press

Baader, F., Milicic, M.; Lutz, C.; Sattler, U. & Wolter, F. (2005). Integrating Description Logics and Action Formalisms: First Results. *2005 International Workshop on Description Logics (DL2005)*, Edinburgh, Scotland, UK, July, 2005, CEUR-Workshop Proceedings Volume 147

Baker, C. F.; Fillmore, C. J. & Lowe, J. B. (1998). The Berkeley FrameNet Project, *Proceedings COLING-ACL*, Montreal, Canada, 1998

Brachman, R. J. & Schmolze, J. G. (1985). An Overview of the KL-ONE Knowledge Representation System, *Cognitive Science*, 9(2), pp. 171-216

Coyne, B. & Sproat, R. (2001). WordsEye: An Automatic Text-to-Scene Conversion System, *Siggraph*, 2001. See also: Semantic Light, www.semanticlight. com

CRL (1996). *Mikrokosmos*, http://crl.nmsu.edu/Research/Projects/mikro/

CSL Princeton (2007). *WordNet*, http://wordnet.princeton.edu/

Cycorp (2007). *Cyc*, http://www.cyc.com/

Devanbu, P. T. & Litman, D. J. (1996). Taxonomic Plan Reasoning, *Artificial Intelligence*, 84, 1996, pp. 1-35

Di Eugenio, B. (1998). An Action Representation Formalism to Interpret Natural Language Instructions, *Computational Intelligence*, 14, 1998, pp. 89-133

Epsitec SA (2007). CeeBot, www.ceebot.com/ceebot/index-e.php, Epsitec SA, Belmont, Switzerland, 2007

IEEE SUO WG (2003). *SUMO Ontology,* IEEE P1600.1 Standard Upper Ontology Working Group (SUO WG), http://ontology.teknowledge.com/

Fillmore, C. J. 1968. The case for case, In: *Universals in Linguistic Theory*, E. Bach & R. Harms, (Eds.), pp. 1-90, Holt, Rhinehart and Winston, New York

FIPA (2002). FIPA ACL Message Structure Specification. Foundation for Intelligent Physical Agents (FIPA), 6 December 2002, www.fipa.org/specs/fipa00061

ICSI (2007). *FrameNet*, http://framenet.icsi.berkeley.edu/

Gomez, F. (1998). Linking WordNet Verb Classes to Semantic Interpretation. *Proceedings COLING-ACL, Workshop on the Usage of WordNet on NLP Systems*. Université de Montréal, Montréal, Canada, 1998

Jurafsky, D.; Wooters, C.; Tajchman, G.; Segal, J.; Stolcke, A.; Fosler, E. & Morgan, N. (1994). The Berkeley Restaurant Project, *Proceedings ICSLP*, pp. 2139-2142, 1994

Kemke, C. (2006). Towards an Intelligent Interior Design System, *Workshop on Intelligent Virtual Design Environments (IVDEs)* at the *Design Computation and Cognition Conference*, Eindhoven, the Netherlands, 9th of July 2006

Kemke, C. (2004). Speech and Language Interfaces for Agent Systems, *Proc. IEEE/WIC/ACM International Conference on Intelligent Agent Technology*, pp.565-566, Beijing, China, September 2004

Kemke, C. (2003). A Formal Approach to Describing Action Concepts in Taxonomical Knowledge Bases, In: *Foundations of Intelligent Systems*, Lecture Notes in Artificial Intelligence, Vol. 2871, N. Zhong, Z.W. Ras, S. Tsumoto, E. Suzuku (Eds.), pp. 657-662, Springer, 2003

Kemke, C. (2001). *About the Ontology of Actions*, Technical Report MCCS -01-328, Computing Research Laboratory, New Mexico State University

Kemke, C. (2000). What Do You Know about Mail? Knowledge Representation in the SINIX Consultant, *Artificial Intelligence Review*, 14, 2000, pp. 253-275

Kemke, C. (1988). Die Darstellung von Aktionen in Vererbungshierarchien (Representation of Actions in Inheritance Hierarchies). In: *GWAI-88, Proceedings of the German Workshop on Artificial Intelligence*, pp. 57-63, Springer, 1988

Kemke, C. (1987). Representation of Domain Knowledge in an Intelligent Help System, Proc. of the Second IFP Conference on Human-Computer Interaction INTER-ACT'87, pp. 215-200, Stuttgart, FRG, 1987

Kemke, C. & Walker, E. (2006). Planning through Integrating an Action Abstraction and a Plan Decomposition Hierarchy, *Proceedings IEEE/WIC/ACM International Agent Technology Conference IAT-2006,* 6 pages, December 2006, Hong Kong

Kruijff, G.-J. M (2006). Talking on the moon. Presentation at the AAAI 2006 Spring Symposium "To Boldly Go Where No Human-Robot Team Has Gone Before", Stanford, California, 2006

Kruijff, G.-J. M.; Zender, H.; Jensfelt, P. & Christensen, H. I. (2007). Situated Dialogue and Spatial Organization: What, Where… and Why? *International Journal of Advanced Robotic Systems*, 4(1), 2007, pp. 125-138

KSL (2007). *Ontolingua*, http://www.ksl.stanford.edu/software/ontolingua/

Liebig, T. & Roesner, D. (1997). Action Hierarchies in Description Logics, *1997 International Workshop on Description Logics (DL'97),* Gif sur Yvette (Paris), France, September, 1997, http://www.lri.fr/~mcr/ps/dl97.html

Lifschitz, V. (1987). On the Semantics of STRIPS, In: *The 1986 Workshop on Reasoning about Actions and Plans*, pp.1-10, Morgan Kaufmann

Patel-Schneider, P. F.; Owsnicki-Klewe, B.; Kobsa, A.; Guarino, N.; MacGregor, R.; Mark, W. S.; McGuiness, D. L.; Nebel, B.; Schmiedel, A. & Yen, J. (1990). Term Subsumption Languages in Knowledge Representation, *AI Magazine*, 11(2), 1990, pp. 16-23

Pease, A. (2007). *Suggested Upper Merged Ontology (SUMO),* http://www.ontologyportal.org

Pednault, E. (1989). ADL: Exploring the middle ground between STRIPS and the situation calculus. *Proceedings of the First International Conference on Principles of Knowledge Representation and Reasoning*, pp. 324-332, 1989

Rickheit, G. & Wachsmuth, I. (Eds.) (2006). *Situated Communication*, Mouton de Gruyter

SFB-360 (2005). *Situated Artificial Communicators*, www.sfb360.uni-bielefeld.de/sfbengl.html

Stent, A.; Dowding, J.; Gawron, M.; Owen Bratt, E. & Moore, R. (1999). The CommandTalk Spoken Dialogue System, *Proceedings of the 37th Annual Meeting of the ACL*, pp. 183-190, University of Maryland, College Park, MD, 1999

Torrance, M. C. (1994). *Natural Communication with Robots*, S.M. Thesis submitted to MIT Department of Electrical Engineering and Computer Science, January 28, 1994

Traum, D.; Schubert, L. K.; Poesio, M.; Martin, N.; Light, M.; Hwang, C.H.; Heeman, P.; Ferguson, G. & Allen, J. F. (1996). Knowledge Representation in the TRAINS-93 Conversation System. *Internat'l Journal of Expert Systems, Special Issue on Knowledge Representation and Inference for Natural Language Processing,* 9(1), 1996, pp. 173-223

Wahlster, W. (1997). VERBMOBIL: *Erkennung, Analyse, Transfer, Generierung und Synthese von Spontansprache*. Report, DFKI GmbH, 1997

Walker, E. (2004). *An Integrated Planning Algorithm for Abstraction and Decomposition Hierarchies of Actions*, Honours Project, Dept. of Computer Science, University of Manitoba, May 2004

Weida, R. & Litman, D. (1994). Subsumption and Recognition of Heterogeneous Constraint Networks, *Proceedings of CAIA-94*, pp. 381-388

10

Can robots replace dogs? Comparison of temporal patterns in dog-human and robot-human interactions

Andrea Kerepesi[1], Gudberg K. Jonsson[2], Enikő Kubinyi[1] and Ádám Miklósi[1]

[1]*Department of Ethology, Eötvös Loránd University,*
[2]*Human Behaviour Laboratory, University of Iceland & Department of Psychology, University of Aberdeen*
[1]*Hungary,* [2]*Iceland*

1. Introduction

Interactions with computers are part of our lives. Personal computers are common in most households, we use them for work and fun as well. This interaction became natural to most of us in the last few years. Some predict (e.g. Bartlett et al 2004) that robots will be as widespread in the not too distant future as PCs are today. Some robots are already present in our lives Some have no or just some degree of autonomy, while others are quite autonomous. Although autonomous robots were originally designed to work independently from humans (for examples see Agah, 2001), a new generation of autonomous robots, the so-called entertainment robots, are designed specially to interact with people and to provide some kind of "entertainment" for the human, and have the characteristics to induce an emotional relationship ("attachment") (Donath 2004, Kaplan 2001). One of the most popular entertainment robots is Sony's AIBO (Pransky 2001) which is to some extent reminiscent to a dog-puppy. AIBO is equipped with a sensor for touching, it is able to hear and recognize its name and up to 50 verbal commands, and it has a limited ability to see pink objects. It produces vocalisations for expressing its 'mood', in addition it has a set of predetermined action patterns like walking, paw shaking, ball chasing etc. Although it is autonomous, the behaviour of the robot depends also on the interaction with the human partner. AIBO offers new perspectives, like clicker training (Kaplan et al. 2002), a method used widespread in dogs' training.

Based on the use of questionnaires Kahn et al (2003) suggested that people at online AIBO discussion forums describe their relationship with their AIBO to be similar to the relationship people have with live dogs. However we cannot forget that people on these kind of on-line forums are actively looking for these topics and the company of those who have similar interests. Those who participated in this survey were probably already devoted to their AIBOs.

It is also interesting how people speak about the robot. Whether they refer to AIBO as a non-living object, or as a living creature? When comparing children's attitudes towards AIBO

and other robots Bartlett et al (2004) found that children referred to AIBO as if it were a living dog, labelled it as "robotic dog" and used rather 'he' or 'she' than 'it' when talked about AIBO. Interviewing children Melson et al (2004) found that although they distinguished AIBO from a living dog, they attributed psychological, companionship and moral stance to the robot. Interviewing older adults Beck et al (2004) found that elderly people regarded AIBO much like as a family member and they attributed animal features to the robot.

Another set of studies is concerned with the observation of robot-human interactions based on ethological methods of behaviour analysis. Comparing children's interaction with AIBO and a stuffed dog Kahn et al (2004) found that children distinguished between the robot and the toy. Although they engaged in an imaginary play with both of them, they showed more exploratory behaviour and attempts for reciprocity when playing with AIBO. Turner et al (2004) described that children touched the live dog over a longer period than the robot but ball game was more frequent with AIBO than with the dog puppy.

Although these observations show that people distinguish AIBO from non-living objects, the results are somehow controversial. While questionnaires and interviews suggest that people consider AIBO as a companion and view it as a family member, their behaviour suggest that they differentiate AIBO from a living dog.

Analysis of dogs' interaction with AIBO showed that dogs distinguished AIBO from a dog puppy in a series of observations by Kubinyi et al. (2003). Those results showed that both juvenile and adult dogs differentiate between the living puppy and AIBO, although their behaviour depended on the similarity of the robot to a real dog as the appearance of the AIBO was manipulated systematically.

To investigate whether humans interact with AIBO as a robotic toy rather than real dog, one should analyze their interaction pattern in more detail. To analyse the structural differences found in the interaction between human and AIBO and human and a living dog we propose to analyze the temporal structure of these interactions.

In a previous study investigating cooperative interactions between the dog and its owner (Kerepesi et al 2005), we found that their interaction consists of highly complex patterns in time, and these patterns contain behaviour units, which are important in the completion of a given task. Analyzing temporal patterns in behaviour proved to be a useful tool to describe dog-human interaction. Based on our previous results (Kerepesi et al 2005) we assume that investigating temporal patterns cannot only provide new information about the nature of dog-human interaction but also about robot-human interaction.

In our study we investigated children's and adults' behaviour during a play session with AIBO and compared it to play with living dog puppy. The aim of this study was to analyse spontaneous play between the human and the dog/robot and to compare the temporal structure of the interaction with dog and AIBO in both children and adults.

2. Method

Twenty eight adults and 28 children were participated in the test and were divided into four experimental groups:

1. Adults playing with AIBO: 7 males and 7 females (Mean age: 21.1 years, SD= 2.0 years)
2. Children playing with AIBO: 7 males and 7 females (Mean age: 8.2 years, SD= 0.7 years)
3. Adults playing with dog: 7 males and 7 females (Mean age: 21.4 years, SD= 0.8 years)
4. Children playing with dog: 7 males and 7 females (Mean age: 8.8 years, SD= 0.8 years)

The test took place in a 3m x 3m separated area of a room. Children were recruited from elementary schools, adults were university students. The robot was Sony's AIBO ERS-210, (dimension: 154mm × 266mm × 274 mm; mass: 1.4 kg; colour: silver) that is able to recognise and approach pink objects. To generate a constant behaviour, the robot was used only in its after-booting period for the testing. After the booting period the robot was put down on the floor, and it "looked around" (turned its head), noticed the pink object, stood up and approached the ball ("approaching" meant several steps toward the pink ball). If the robot lost the pink ball it stopped and „looked around" again. When it reached the goal-object, it started to kick it. If stroked, the robot stopped and started to move its head in various directions. The dog puppy was a 5-month-old female Cairn terrier, similar size to the robot. It was friendly and playful, its behaviour was not controlled in a rigid manner during the playing session. The toy for AIBO was its pink ball, and a ball and a tug for the dog-puppy.
The participants played for 5 minutes either with AIBO or the dog puppy in a spontaneous situation. None of the participants met the test partners before the playing session. At the beginning of each play we asked participants to play with the dog/AIBO for 5 minutes, and informed them that they could do whatever they wanted, in that sense the participants' behaviour were not controlled in any way. Those who played with the AIBO knew that it liked being stroked, that there was a camera in its head enabling it to see and that it liked to play with the pink ball.
The video recorded play sessions were coded by ThemeCoder, which enables detailed transcription of digitized video files. Two minutes (3000 digitized video frames) were coded for each of the five-minute-long interaction. The behaviour of AIBO, the dog and the human was described by 8, 10 and 7 behaviour units respectively. The interactions were transcribed using ThemeCoder and the transcribed records were then analysed using Theme 5.0 (see www.patternvision.com). The basic assumption of this methodological approach, embedded in the Theme 5.0 software, is that the temporal structure of a complex behavioural system is largely unknown, but may involve a set of particular type of repeated temporal patterns (T-patterns) composed of simpler directly distinguishable event-types, which are coded in terms of their beginning and end points (such as "dog begins walking" or "dog ends orienting to the toy"). The kind of behaviour record (as set of time point series or occurrence times series) that results from such coding of behaviour within a particular observation period (here called T-data) constitutes the input to the T-pattern definition and detection algorithms.
Essentially, within a given observation period, if two actions, A and B, occur repeatedly in that order or concurrently, they are said to form a minimal T-pattern (AB) if found more often than expected by chance, assuming as h0 independent distributions for A and B, there is approximately the same time distance (called critical interval, CI) between them. Instances of A and B related by that approximate distance then constitute occurrence of the (AB) T-pattern and its occurrence times are added to the original data. More complex T-patterns are consequently gradually detected as patterns of simpler already detected patterns through a hierarchical bottom-up detection procedure. Pairs (patterns) of pairs may thus be detected, for example, ((AB)(CD)), (A(KN))(RP)), etc. Special algorithms deal with potential combinatorial explosions due to redundant and partial detection of the same patterns using an evolution algorithm (completeness competition), which compares all detected patterns and lets only the most complete patterns survive. (Fig 1). As any basic time unit may be

used, T-patterns are in principle scale-independent, while only a limited range of basic unit size is relevant in a particular study.

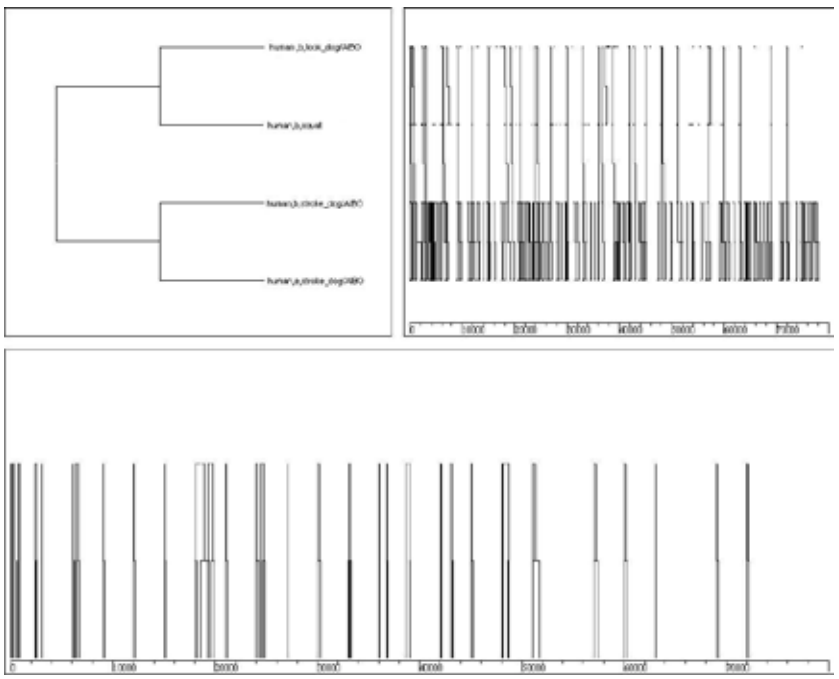


Figure 1. An example for a T-pattern. The upper left box shows the behaviour units in the pattern. The pattern starts with the behaviour unit on the top. The box at the bottom shows the occurrences of the pattern on a timeline (counted in frame numbers)

During the coding procedure we recorded the beginning and the ending point of a behaviour unit. Concerning the search for temporal patterns (T-patterns) we used, as a search criteria, minimum two occurrences in the 2 min. period for each pattern type, the tests for CI was set at p= 0.005, and only included interactive patterns (those T-patterns which contained both the human's and the dog's/AIBO's behaviour units) The number, length and level of interactive T-patterns were analyzed with focusing on the question whether the human or the dog/AIBO initialized and terminated the T-pattern more frequently. A T-pattern is initialized/terminated by human if the first/last behaviour unit in that pattern is human's. A comparison between the ratio of T-patterns initiated or terminated by humans, in the four groups, was carried out as well as the ratio of those T-patterns containing behaviour units listed in Table 1.

Play behaviour		*Activity*		*Interest in partner*	
abbreviation	description	abbreviation	Description	abbreviation	description
Look toy	Dog/AIBO orients to toy	*Stand*	Dog/AIBO stands	*Stroke*	Human strokes the dog/AIBO
Approach toy	Dog/AIBO approaches toy	*Lie*	Dog/AIBO lies	*Look dog*	Human looks at dog/AIBO
Move toy	Human moves the toy in front of dog/AIBO	*Walk*	Dog/AIBO walks (but not towards the toy)		
		Approach toy	Dog/AIBO approaches toy		

Table 1. Behaviour units used in this analysis

Statistical tests were also conducted on the effect of the subjects' age (children vs. adults) and the partner type (dog puppy vs. AIBO) using two-way ANOVA.
Three aspects of the interaction were analyzed. (see Table 1).

1. *Play behaviour* consists of behaviour units referring to play or attempts to play, such as dog/AIBO approaches toy, orientation to the toy and human moves the toy.
2. The partners' *activity during play* includes dog/AIBO walks, stands, lies and approaches the toy.
3. *Interest in the partner* includes humans' behaviour towards the partner and can be described by their stroking behaviour and orientation to the dog/AIBO.

We have also searched for common T-patterns that can be found minimum twice in at least 80% of the dyads. We have looked for T-patterns that were found exclusively in child-AIBO dyads, child-dog dyads, adult-AIBO dyads and adult-dog dyads. We also search for patterns that are characteristic for AIBO (can be found in at least 80% of child-AIBO and adult AIBO dyads), dog (found in child-dog and adult-dog dyads), adult (adult-AIBO and adult-dog dyads) and children (child-dog and child-AIBO dyads)

3. Results

The number of different *interactive T-patterns* was on average 7.64 in adult-AIBO dyads, 3.72 in child-AIBO dyads, 10.50 in adult-dog dyads and 18.14 in child dog-dyads. Their number did not differ significantly among the groups.
Comparing the ratio of *T-patterns initialized by humans*, we have found that adults initialized T-patterns more frequently when playing with dog than participants of the other groups ($F_{3,56}$= 5.27, p= 0.003). Both the age of the human ($F_{1,56}$= 10.49, p= 0.002) and the partner's type ($F_{1,56}$= 4.51, p= 0.038) had a significant effect, but their interaction was not significant.
The partner's type ($F_{1,56}$=10.75, p=0.002) also had a significant effect on the ratio of *T-patterns terminated by humans* ($F_{3,56}$= 4.45, p= 0.007) we have found that both children and adults terminated the T-patterns more frequently when they played with AIBO than when they played with the dog puppy (Fig. 2).

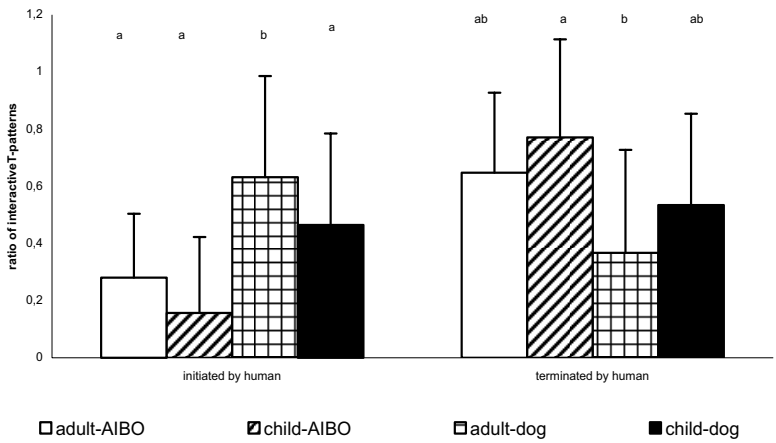


Figure 2. Mean ratio of interactive T-patterns initiated and terminated by humans (bars labelled with the same letter are not significantly different)

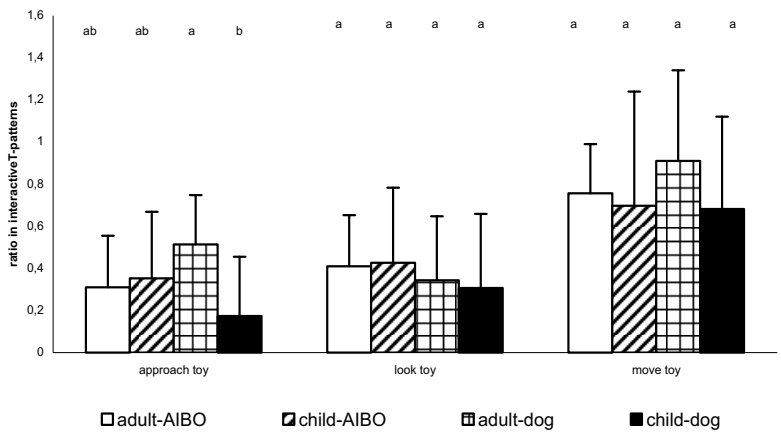


Figure 3. Mean ratio of interactive T-patterns containing the behaviour units displayed by AIBO or dog (Look toy, Approach toy) or Humans (Move toy)

The age of the human had a significant effect on the ratio of T-patterns containing *approach toy* ($F_{1,56}$= 4.23, p= 0.045), and the interaction with the partner's type was significant ($F_{1,56}$= 6.956, p= 0.011). This behaviour unit was found more frequently in the T-patterns of adults playing with dog than in the children's T-patterns when playing with dog. The ratio of *look toy* in T-patterns did not differ among the groups. (Fig. 3)
The ratio of the behaviour unit *stand* also varied among the groups ($F_{3,56}$= 6.59, p< 0.001), there was a lower frequency of such T-patterns when children were playing with dog than in any other case ($F_{1,56}$= 7.10, p= 0.010). However, the ratio of behaviour units *lie* and *walk* in T-patterns did not differ among the groups.
The ratio of humans' behaviour units in T-patterns *(move toy, look dog* and *stroke*) did not vary among the groups.
When searching for common T-patterns we have realized that certain complex patterns were found exclusively to be produced in either child and play subject (AIBO, child-dog) or adult and play subject (AIBO, and adult-dog) interactions. Some pattern types were typical to children and found to occur in both the child-AIBO and child-dog groups (Fig 4.) and others, typical for adults were found in both adult-AIBO and adult-dog groups (Fig 5.)

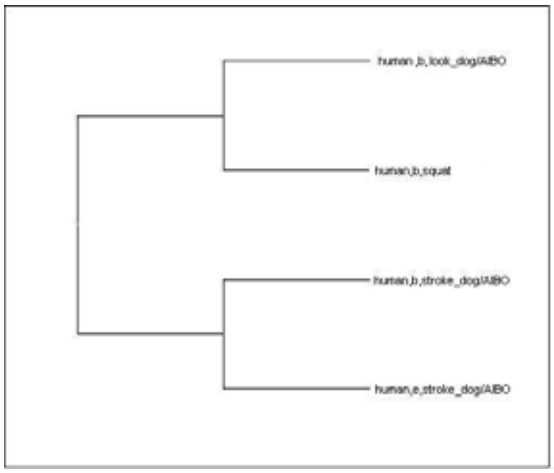


Figure 4. A T-pattern found at least 80% of adults' dyads. The figure shows only the upper left box of the T-pattern The behaviour units in order are: (1) adult begins to look at the dog/AIBO, (2) adult begins to stroke the dog/AIBO, (3) adult begins to squat, (4) adult ends stroking the dog/AIBO, (5) adult ends looking at the dog/AIBO, (6) adult ends squatting

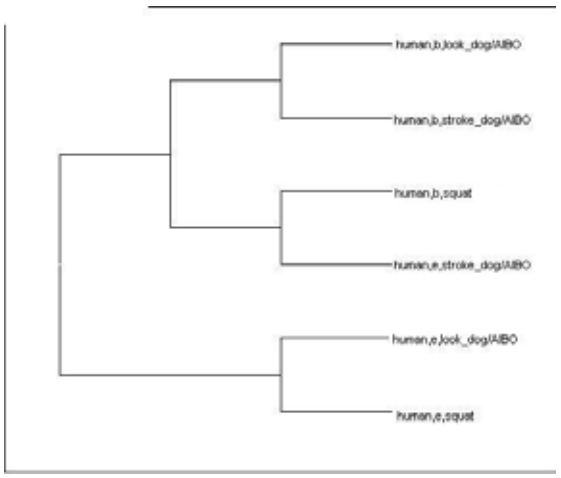


Figure 5. A T-pattern found at least 80% of children's dyads. The figure shows only the upper left box of the T-pattern The behaviour units in order are: (1) child begins to look at the dog/AIBO, (2) child begins to squat, (3) child begins to stroke the dog/AIBO, (4) child ends stroking the dog/AIBO

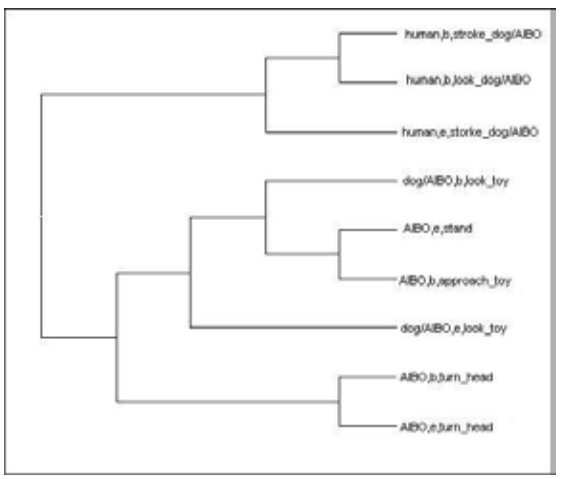


Figure 6. A T-pattern found at least 80% of AIBO's dyads. The figure shows only the upper left box of the T-pattern The behaviour units in order are: (1) child/adult begins to stroke the dog/AIBO (2) , child/adult begins to look at the dog/AIBO, (3) child/adult ends stroking the dog/AIBO (4) dog/AIBO begins to look at the toy , (5) AIBO ends standing, (6) AIBO begins to approach the toy, (7) dog/AIBO ends looking at the toy, (8) AIBO begins to turn around its head (9) AIBO ends turning around its head

All the common T-patters have the same start: adult/child looks at the dog/AIBO and then starts to stroke it nearly at the same moment. The main difference is in the further part of the patterns. The T-patterns end here in case of dogs' interactions. It continues only in AIBO's dyads (Fig 6), when the robot starts to look at the toy, approach it then moving its head around. We have found that children did not start a new action when they finished stroking the AIBO.

4. Discussion

To investigate whether humans interact with AIBO as a non-living toy rather than a living dog, we have analyzed the temporal patterns of these interactions. We have found that similarly to human interactions (Borrie et al 2002, Magnusson 2000, Grammer et al 1998) and human-animal interactions (Kerepesi et al 2005), human-robot interactions also consist of complex temporal patterns**.** In addition the numbers of these temporal patterns are comparable to those T-patterns detected in dog-human interactions in similar contexts.

One important finding of the present study was that the type of the play partner affected the initialization and termination of T-patterns. Adults initialized T-patterns more frequently when playing with dog while T-patterns terminated by a human behaviour unit were more frequent when humans were playing with AIBO than when playing with the dog puppy. In principle this finding has two non-exclusive interpretations. In the case of humans the complexity of T-patterns can be affected by whether the participants liked their partner with whom they were interacting or not (Grammer et al 1998, Sakaguchi et al 2005). This line of arguments would suggest that the distinction is based on the differential attitude of humans toward the AIBO and the dog. Although, we cannot exclude this possibility, it seems more likely that the difference has its origin in the play partner. The observation that the AIBO interrupted the interaction more frequently than the dog suggests that the robot's actions were less likely to become part of the already established interactive temporal pattern. This observation can be explained by the robot's limited ability to recognize objects and humans in its environment. AIBO is only able to detect a pink ball and approach it. If it looses sight of ball it stops that can interrupt the playing interaction with the human. In contrast, the dog's behaviour is more flexible and it has got a wider ability to recognise human actions, thus there is an increased chance for the puppy to complement human behaviour.

From the ethological point of view it should be noted that even in natural situations dog-human interactions have their limitations. For example, analyzing dogs' behaviour towards humans, Rooney et al (2001) found that most of the owner's action trying to initialize a game remains without reaction. Both Millot et al (1986) and Filiatre et al (1986) demonstrated that in child-dog play the dog reacts only at approximately 30 percent of the child's action, while the child reacts to 60 percent of the dog's action. Although in the case of play it might not be so important, other situations in everyday life of both animals and man require some level of temporal structuring when two or more individuals interact. Such kinds of interactions have been observed in humans performing joint tasks or in the case of guide dogs and their owners. Naderi et al (2001) found that both guide dogs and their blind owners initialize actions during their walk, and sequences of initializations by the dog are interrupted by actions initialized by the owner.

Although the results of the traditional ethological analyses (e.g. Kahn et al 2004, Bartlett et al 2004) suggest that people interacting with AIBO in same ways as if it were a living dog puppy, and that playing with AIBO can provide a more complex interaction than a simple

toy or remote controlled robot, the analysis of temporal patterns revealed some significant differences, especially in the formation of T-patterns. Investigating the common T-patterns we can realize that they start the same way: adult/child looks at the dog/AIBO and then starts to stroke it nearly at the same moment. The main difference is in the further part of the patterns. The T-patterns end here if we look for common T-patterns that can be found in at least 80% of the dyads of adults, children and dog. It continues only in AIBO's T-patterns, when the robot starts to look at the toy, approaches it then moving its head around.

By looking at the recordings of the different groups we found that children did not start a new action when they finished stroking the AIBO. Interestingly when they played with the dog, they tried to initiate a play session with the dog after they stopped stroking it, however was not the case in case of AIBO. Adults tried to initiate a play with both partners, however not the same way. They initiated a tug-of-war game with the dog puppy and a ball chase game with the AIBO. These differences show that (1) AIBO has a more rigid behaviour compared to the dog puppy. For example, if it is not being stroked then it starts to look for the toy. (2) Adults can adapt to their partners play style, so they initiate a tug-of-war game with the puppy and a ball-chasing game with the robot. In both cases they chose that kind of play object which releases appropriate behaviour from the play-partner. (3) Children were not as successful to initiate a play with their partners as adults.

Although we did not investigate this in the present study, the differences in initialisation and termination of the interactions could have a significant effect on the human's attitude toward their partner, that is, in the long term humans could get "bored" or "frustrated" when interacting with a partner that has a limited capacity to being engaged in temporally structured interactions.

In summary, contrary to the findings of previous studies, it seems that at a more complex level of behavioural organisation human-AIBO interaction is different from the interactions displayed while playing with a real puppy. In the future more attention should be paid to the temporal aspects of behavioural pattern when comparing human-animal versus human-robot interaction, and this measure of temporal interaction could be a more objective way to determine the ability of robots to be engaged in interactive tasks with humans.

5. References

Agah, A., 2001. Human interactions with intelligent systems: research taxonomy. Computers and Electrical Engineering 27 71-107

Anolli, L., Duncan, S. Magnusson, M.S., Riva G. (Eds.), 2005. The hidden structure of interaction. Amsterdam: IOS Press

Bartlett, B., Estivill-Castro, V., Seymon, S., 2004. Dogs or robots: why do children see them as robotic pets rather than canine machines? 5th Australasian User Interface Conference. Dunedin. Conferences in Research and Practice in Information Technology, 28: 7-14

Beck, A.M., Edwards, N.E., Kahn, P., Friedman, B., 2004. Robotic pets as perceived companions for older adults. IAHAIO People and animals: A timeless relationship, Glasgow, UK, p. 72

Borrie, A., Jonsson, G.K., Magnusson, M.S., 2001. Application of T-pattern detection and analysis in sport research. Metodologia de las Ciencias del Comportamiento 3: 215-226.

Donath, J., 2004. Artificial pets: Simple behaviors elicit complex attachments. In: M. Bekoff (Editor): The Encyclopedia of Animal Behaviour, Greenwood Press.

Filiâtre, J.C., Millot, J.L., Montagner, H., 1986. New data on communication behaviour between the young child and his pet dog. Behavioural Processes, 12: 33-44

Grammer, K., Kruck, K.B., Magnusson, M.S., 1998. The courtship dance: patterns of nonverbal synchronization in opposite-sex encounters. Journal of Nonverbal Behavior, 22: 3-29

Kahn, P.H., Friedmann, B., Perez-Granados, D.R., Freier, N.G., 2004. Robotic pets in the lives of preschool children**.** CHI 2004. Vienna, Austria, pp 1449-1452

Kahn, P. H., Jr., Friedman, B., & Hagman, J., 2003. Hardware Companions? – What Online AIBO Discussion Forums Reveal about the Human-Robotic Relationship. *Conference Proceedings of CHI 2003* New York, NY: Association for Computing Machinery. pp. 273-280

Kaplan, F., Oudeyer, P., Kubinyi, E., Miklósi, Á., 2002. Robotic clicker training Robotics and Autonomous Systems, 38: 197–206

Kaplan, F., 2001. Artificial Attachment: Will a robot ever pass Ainsworth's Strange Situation Test? In: Hashimoto, S. (Ed.), Proceedings of Second IEEE-RAS International Conference on Humanoid Robots, Humanoids. Institute of Electrical and Electronics Engineers, Inc., Waseda University, Tokyo, Japan, pp. 99–106.

Kerepesi, A., Jonsson, G.K, Miklósi Á., Topál, J., Csányi, V., Magnusson, M.S., 2005. Detection of temporal patterns in dog-human interaction. Behavioural Processes, 70(1): 69-79.

Kubinyi, E., Miklósi, Á., Kaplan, F., Gácsi, M., Topál, J., Csányi, V., 2004. Social behaviour of dogs encountering AIBO, an animal-like robot in a neutral and in a feeding situation. Behavioural Processes, 65: 231–239

Magnusson, M.S., 1996. Hidden Real-Time patterns in Intra- and Inter-Individual Behavior Description and Detection. European Journal of Psychological Assessment, 12: 112-123

Magnusson, M.S., 2000. Discovering hidden time Patterns in Behavior: T-patterns and their detection. Behavior Research Methods, Instruments & Computers, 32: 93-110

Melson, G.F., Kahn, P., Beck, A., Friedman, B., Roberts, T., 2004. Children's understanding of robotic and living dog. IAHAIO People and animals: A timeless relationship, Glasgow, UK, p 71.

Millot, J.L., Filiâtre, J.C., 1986. The Behavioural Sequences in the Communication System between the Child and his Pet Dog. Applied Animal Behaviour Science, 16: 383-390

Millot, J.L., Filiatre, J.C., Gagnon, A.C., Eckerlin, A., Montagner, H., 1988. Children and their pet dogs: how they communicate. Behavioural Processes, 17: 1-15

Naderi, Sz., Miklósi, Á., Dóka A., Csányi, V., 2001. Co-operative interactions between blind persons and their dogs. Applied Animal Behaviour Sciences, 74(1): 59-80

Nourbakhsh, I.R., Bobenage, J., Grange, S., Lutz, R. Meyer, R., Soto, A., 1999. An affective mobile robot educator with a full-time job. Artificial Intelligence, 114: 95–124

Pransky, J., 2001. AIBO - the No. 1 selling service robot. Industrial Robot: An International Journal, 28: 24-26

Rooney, N.J. Bradshaw, J.W.S. Robinson, I.H., 2001. Do dogs respond to play signals given by humans? Animal Behaviour, 61. 715–722

Sakaguchi, K, Jonsson, G.K. & Hasegawa, T., 2005. Initial interpersonal attraction between mixed-sex dyad and movement synchrony. In: L. Anolli, S. Duncan, M. Magnusson and G. Riva (Editors) The hidden structure of social interaction: From neurons to culture patterns. IOS Press.

Turner, D.C., Ribi, F.N., Yokoyama, A., 2004. A comparison of children's behaviour toward a robotic pet and a similar sized, live dog over time. IAHAIO People and animals: A timeless relationship, Glasgow, UK, p. 68.

11

A Facial Expression Imitation System for the Primitive of Intuitive Human-Robot Interaction

Do Hyoung Kim†, Kwang Ho An[2], Yeon Geol Ryu[2] and Myung Jin Chung[2]
[1]*Mobile Communication Division, Samsung Electronics Co.*
[2]*Electrical Engineering Division, Korea Advanced Institute of Science and Technology*
Republic of Korea

1. Introduction

The human face has long been considered a representation of humans. According to observations of cognitive psychology, people unconsciously and frequently recognize and identify others from their faces. Many researchers have also emphasized the importance of the human face, e.g., Cicero said "Everything is in a face," Darwin stated that "The human face is the most complex and versatile of all species," and Paul Ekman remarked "The face makes one's behavior more predictable and understandable to others and improves communication" (Kim et al., 2005).

Nobody doubts that the human face is a rich and versatile instrument that serves many different functions and conveys the human's motivational state. Facial gestures can communicate information on their own. Moreover, the face serves several biological functions; for instance, humans generally close their eyes to protect themselves from a threatening stimulus, and they close them for longer periods to sleep (Breazeal, 2004) (Mcneill, 1998) (Ekman et al., 2002).

To interact socially with humans, a robot must be able to do gather information about its surroundings as well as express its state or emotion, so that humans will believe that the robot has beliefs, desires, and intentions of its own. Cynthia Breazeal who is one of pioneers on the natural and intuitive HRI research at MIT mentioned the ideal robotic system like this: "The ideal of the robotic system is for people to interact, play and teach the robot as naturally as they would teach an infant or a very young child. Such interactions provide many different kinds of scaffolding that the robot can potentially use to foster its own learning. As a prerequisite for human-robot interactions, people need to ascribe precocious social intelligence to the robot. However, before people treat the robot as a socially aware being, the robotic system need to convey subjective internal states such as intentions, beliefs, desires, and feelings" (Breazeal, 2004). As a result, facial expressions are critical in a robotic system because they encourage people to treat the robot as an object that can convey internal states, have social intelligence and exploit human-like tendencies.

Our facial expression imitation system is composed of two modules, which are facial expression recognition and facial expression generation (see Figure 1). The system firstly detects human's face in the image. The proposed facial expression recognition algorithm classifies the obtained face into one of P. Ekman's basic facial expressions that include

neutral, happiness, sadness, anger, surprise, disgust and fear (Ekman et al., 2002). From the result of the recognition, our facial expression imitation system knows user's facial expression, and it copies the recognized facial expression through the following procedures: artificial facial expression generation, multiple motor control, and movements of robot's eyelid and mouth.

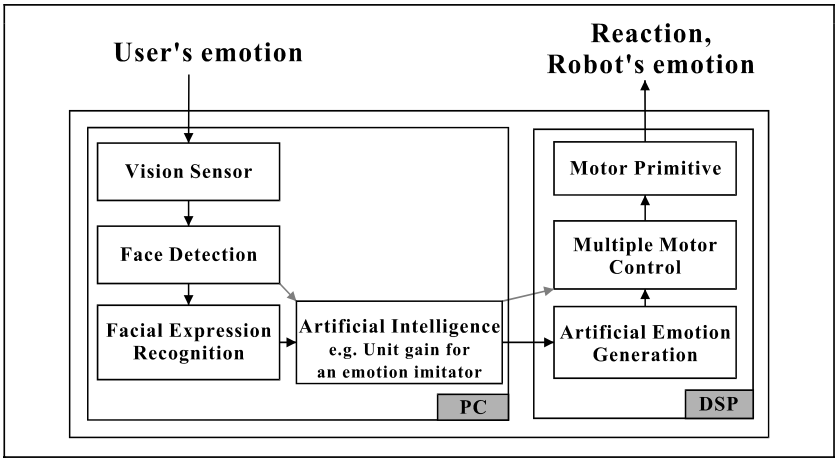


Figure 1. The whole system block diagram. The image pre-processing, face detecion and facial expression recognition algorithm run on a personal computer (PC) with a commercial microprocessor. In addition, the generation of robot's facial expression and motor controller operate in a fixed point digital signal processor (DSP)

For the facial expression recognition, we introduced new types of rectangle features that can distinguish facial expressions efficiently with consideration of more complicated relative distribution of facial image intensities. In order to find the rectangle features that have the highest recognition rate, we firstly selected the efficient rectangle feature types for facial expression recognition from all possible rectangle feature types in a 3×3 matrix form using 1065 facial expression images based on the JAFFE (Japanese Female Facial Expression) database. Each rectangle feature type was selected by the AdaBoost algorithm. We then measured the error rate and chose the top five rectangle feature types that had the least error rate among the total 316 rectangle feature types for each facial expression.

For the facial expression generation, we have designed our system architectures to meet the challenges of real-time visual-signal processing and real-time position control of all actuators with minimal latencies. The motivational and behavioral parts run on a fixed point DSP and 12 internal position controllers of commercial RC servos. 12 actuators are able to simultaneously move to the target positions by introducing a bell-shaped velocity control. The cameras in the eyes are connected to the PC by the IEEE 1394a interface, and all position commands of actuators are sent from PC.

2. Contents of the Chapter

This Chapter has attempted to deal with the issues on establishing a facial expression imitation system for natural and intuitive interactions with humans. Several real-time cognition abilities were implemented such as face detection, face tracking, and facial expression recognition. Moreover, a robotic system with facial components is developed, which is able to imitate human's facial expressions.

As briefly mentioned above, two major issues will be dealt with in this Chapter; one is the facial expression recognition and the other is the facial expression generation. In the recognition part, the following contents will be included.

- The state-of-the-art for automatically recognizing facial expressions of human being.
- Issues for the facial expression recognition in the real-time sense.
- The proposed recognition approach.

In order to recognize human's facial expressions, we suggested a method of recognizing facial expressions through the use of an innovative rectangle feature. Using the AdaBoost algorithm, an expanded version of the process with the Viola and Jones' rectangle feature types has been suggested. We dealt with 7 facial expressions: neutral, happiness, anger, sadness, surprise, disgust, and fear. Real-time performance can be achieved by using the previously trained strong classifier composed by a set of efficient weak classifiers.

In the generation part, the following contents will be included.

- The state-of-the-art for facial robots.
- Issues for designing and developing facial robots; e.g. the mechanics, the system architecture, and the control scheme.
- Issues for image processing and visual tracking.
- Introduction of the developed facial robot and the generation method of facial expressions.

In addition, several experiments show the validity of the developed system. Finally, some conclusions and further works are presented.

3. Facial Expression Recognition

To date, many approaches have been proposed in the facial expression recognition field. According to which factor is considered more important, there are many categorizations (Pantic & Rothkrantz, 2000) (Fasel & Luettin, 2003). Based on facial expression feature extraction, we can classify previous approaches as model-based methods or image-based methods.

As a representative model-based method, Essa and Pentland fitted a 3D mesh model of face geometry to a 2D face image and classified five facial expressions using the peak value of facial muscle movement (Essa & Pentland, 1997). Lanitis et al. used Active Appearance Models (AAM) to interpret the face and Huang and Huang used a gradient-based method to classify face shape using a Point Distribution Model (PDM) (Lanitis et al., 1997) (Huang & Huang, 1997). Zhang et al. fitted the face with sparsely distributed fiducial feature points and distinguished facial expressions (Zhang et al., 1998). Generally, these model-based methods are robust to occlusions. However they are inadequate for a real-time system because they require much time to fit the model to the face image and need high resolution input images to analyze facial expressions.

Among the image-based methods, Lisetti and Rumelhart used the whole face as a feature and Fellenz et al. used Gabor wavelet filtered whole faces (Lisetti & Rumelhart, 1998) (Fellenz et al., 1999). Padgett and Cottrell extracted facial expressions from windows placed around the eyes and mouth and analyzed them using Principle Components Analysis (PCA) (Padgett & Cottrell, 1997). Bartlett distinguished facial expressions using an image intensity profile of the face and Lien et al. used dense flow with PCA and block based density to classify facial expressions (Bartlett, 1998) (Lien et al., 1998).

Recently, Viola and Jones constructed a fast face detection system using rectangle features trained by the AdaBoost algorithm (Viola & Jones, 2001) (Freund & Schapire, 1995). Wang et al. applied this method to facial expression recognition and distinguished 7 class facial expressions in real-time (Wang et al., 2004). However, the systems implemented by AdaBoost used Haar-like rectangle features for face detection or a bit modified rectangle features (Wang et al., 2004) (Littlewort et al., 2004). More analysis and innovative rectangle features should be established for facial expression recognition. Because the rectangle features consider a local region of the face, new rectangle feature types are needed to consider more complicated relative distribution of facial image intensities for facial expression recognition.

Given 7 facial expressions (neutral, happiness, anger, sadness, surprise, disgust, fear), we obtained discernable rectangle feature types for each facial expression among all possible rectangle feature types in 3×3 matrix form. The rectangle features are selected again from the facial image database using the AdaBoost algorithm with the above rectangle feature types, and so improved the recognition rate of a strong classifier and automatically and spontaneously recognized human facial expressions in real-time.

3.1 Viola and Jones' Boosting Method; AdaBoost Algorithm

AdaBoost learning algorithm is a simple learning algorithm that selects a set of efficient weak classifiers from a large number of potential features. Our boosting algorithm is basically the same as Viola and Jones' boosting algorithm (Viola & Jones, 2001). From this procedure, T weak classifiers are constructed and the final strong classifier is a weighted linear combination of the T weak classifiers.

- Consider sample images (x_1, y_1), …, (x_n, y_n), where y_i = 0, 1 for negative and positive samples, respectively.
- Initialize weights $w_{1,i} = \frac{1}{2m}, \frac{1}{2l}$ for y_i = 0, 1, respectively, where m and l are the number of negatives and positives, respectively.
- For t = 1, …, T,

  1) Normalize the weights so that w_t is a probability distribution.

$$w_{t,i} \leftarrow \frac{w_{t,i}}{\sum_{j=1}^{n} w_{t,j}} \tag{1}$$

  2) For each feature, j, train a weak classifier h_j.

$$\varepsilon_j = \sum_i w_i \left| h_j(x_i) - y_i \right| . \tag{2}$$

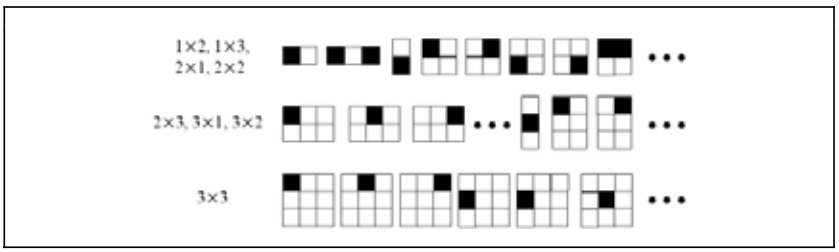


Figure 2. All possible rectangle feature types within up to 3×3 structure size used for training

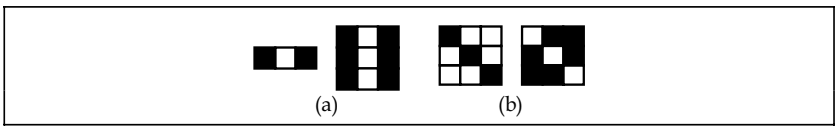


Fig. 3. Examples of overlapped rectangle feature types. (a) The types are independent of size variation, (b) The types consider unsigned feature value.

3) Choose the classifier h_t with the lowest error ε_t.

4) Update the weights.

$w_{t+1,i} = w_{t,i}\beta_t^{1-e_i}$, where $e_i = 0$ if sample x_i is classified correctly, $e_i = 1$ otherwise.

- The final strong classifier is

$$H(x) = \begin{cases} 1 & , if \sum_{t=1}^{T} \alpha_t h_t(x) \geq 0.5 \times \sum_{t=1}^{T} \alpha_t \\ 0 & , otherwise \end{cases}, \quad \alpha_t = \log \frac{1}{\beta_t}. \tag{3}$$

3.2 Procedure for Feature Type Extraction

The process of extracting weak classifiers is identical to Viola and Jones method. Using the selected rectangle feature types, we trained a face training image set and constructed a strong classifier for each facial expression. Before the procedure of extracting classifiers, suitable feature types for facial expression recognition are extracted from all possible rectangle feature types within a 3×3 matrix form as follows.

- Find all possible rectangle feature types within a 3×3 matrix form. (See Figure 2.)
- Among the above rectangle feature types, exclude overlapped feature types.
  1. Size variation

  Though two rectangle feature types in Fig. 3(a) have different shapes, they are overlapped when searching a weak classifier because the size of the rectangle feature can be extended in the x, y directions.

  2. Feature value

  Fig. 3(b) shows rectangle feature types that have the same feature values and opposite polarity. Since the AdaBoost algorithm is a two-class classification method, the two rectangle feature types in Fig. 3(b) have the same function.

- For p = 1, …, # *of facial expression classes*,
  Consider sample images (x_1, y_1), …, (x_n, y_n) where y_i = 1, 0 for the target facial expression images and the other facial expression images, respectively. Q denotes the total number of rectangle feature types.
- For q = 1, …,*Q*,

  Initialize weights $w_i = \frac{1}{2m}, \frac{1}{2l}$ for y_i = 1, 0, respectively, where m and l are the number of the target facial expression images and the other facial expression images.
  - For t = 1, …,T,
    1. Normalize the weights so that w_t is a probability distribution.

$$w_{t,i} \leftarrow \frac{w_{t,i}}{\sum_{j=1}^{n} w_{t,j}}$$

    2. For each the feature, j, based on q^{th} rectangle feature types, train a weak classifier h_j.

$$\varepsilon_j = \sum_i w_i \left| h_j(x_i) - y_i \right| .$$

    3. Choose the classifier $h_{q,t}$ with the lowest error $\varepsilon_{q,t}$.
    4. Update the weights.
       $w_{t+1,i} = w_{t,i}\beta_t^{1-e_i}$, where e_i = 0 if sample x_i is classified correctly, e_i = 1 otherwise.
    5. Calculate $\alpha_{p,q,t}$ as followings:

$$\alpha_{p,q,t} = \log \frac{\varepsilon_{q,t}}{1-\varepsilon_{q,t}} . \tag{4}$$

- Sort $\alpha_{p,q,t}$ in high value order for each p and choose rectangle feature types that have high $\alpha_{p,q,t}$ value.

3.3 Selected Rectangle Features

Fig. 4 shows the rectangular feature types that effectively distinguish the corresponding facial expressions extracted from the results of boosting algorithm. In this figure, we arrange the features in order of the least error from left to right. In Fig. 4, the facial image is place in a position such that the facial expression can be most accurately distinguished. And the best rectangular feature is overlay to the corresponding facial expression image. As you can see in Fig. 4, almost all of the features consider the area just above the mouth, cheek and the white of the eye. This is because we considered the phenomenon that when a person smiles or becomes frustrated or annoyed, he or she tends to alter the expression of the mouth or area around the eyes.

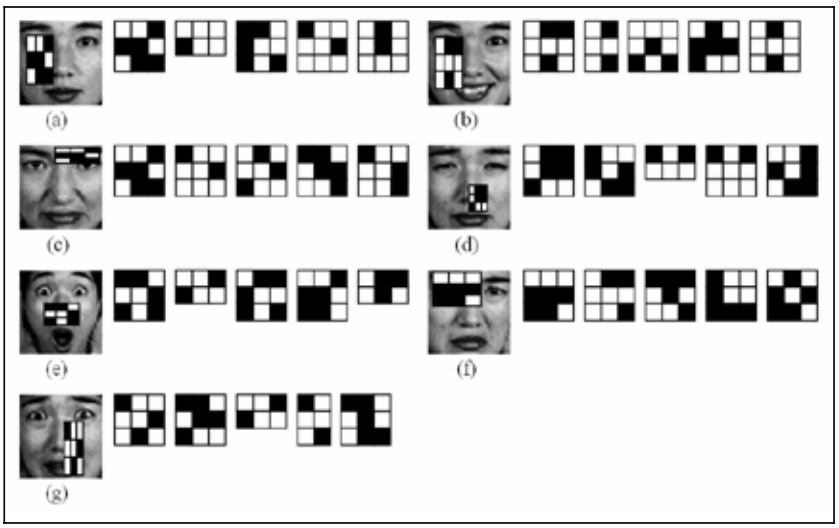


Figure 4. 5 selected rectangle feature types for each facial expression . (a) neutral, (b) happiness, (c) anger, (d) sadness, (e) surprise, (f) disgust, (g) fear

In Fig. 4(a), 5 feature types that distinguish the 'neutral' facial expression from other facial expressions are shown. As you can see in this figure, they consider the area above mouth, cheek and the white of the eye. Fig. 4(b) shows five types that characterize the 'happy' facial expression and the rectangular feature with the least error rate. In Fig. 4(b), the features consider the eye and mouth of the smiling facial expressions. Fig. 4(c) shows the rectangular feature that most effectively distinguishes the expression of 'anger'. The angry facial expression is characterized by a change in the shape of the eyebrows. It has V shape eyebrows and the area between eyebrows appears wrinkled. In Fig. 4(c), the rectangle feature that distinguishes the angry face considers both the eyebrows and the area between eyebrows. Fig. 4(d) shows the rectangular feature that most effectively distinguishes the expression of 'sorrow'. According to Ekman's research, the edge of the lip moves downwards and the inner part of the eyebrows pull toward each other and move upward (Ekman et al., 2002). Meanwhile, the skin area under the eyebrows becomes triangular in shape. However, Fig. 4(d) does not consider these features, because the features that characterize the facial expression of sorrow are very weak and are not adequate to distinguish this expression. Therefore, in Fig. 4(d), we present the rectangular feature obtained by learning that has highest detection rate. Fig. 4(e) presents the best feature that effectively distinguishes the 'surprise' facial expression. There are a number of features that characterize the expression of surprise. In Fig. 4(e) the good features are placed around the nose. In general, when surprised he or she opens his or her eyes and mouth. This pulls the skin under the eyes and around the mouth, making these areas become brighter. Fig. 4(e) shows the selected feature considers this kind of area precisely. The expression of disgust is difficult to distinguish, because it varies from person to person and the degree of the expression is different from one person to another. In particular, it is hard to distinguish this

expression from the expression of anger. In Fig. 4(f), among the rectangular features obtained by boosting algorithm, the one that most effectively distinguishes the expression of 'disgust' is presented. This rectangular feature considers the wrinkles around the eyes and nose. Fig. 4(g) presents the rectangular feature to distinguish the 'fear' facial expression. The expression of fear is the hardest to distinguish. Therefore, we have obtained the fear rectangle feature types through learning. Five rectangular features are shown in Fig. 4(g).
Previously listed rectangle features represent the characters of the facial expressions of Ekman's research (Ekman et al., 2002). Above rectangle feature types consider not only more than two face features but also the character of each emotion.

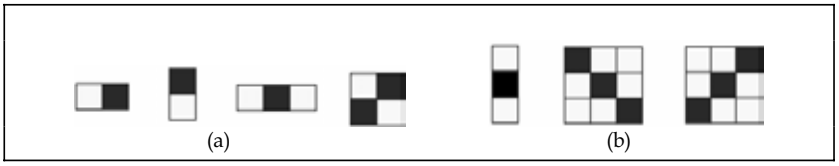


Figure 5. Modified rectangular features. In (a), four types of rectangular features which are proposed by Viola and Jones are shown. Two additional types to focus on diagonal structures and a type for the vertical structure in the face image are shown in (b)

3.4 Selected Feature Analysis

The previous section shows that 35 types of rectangle features are reasonable for classifying facial expressions. This section verifies that the selected rectangle feature types yield better performance than 7 types of rectangle features (four of Viola and Jones (Viola & Jones, 2001) and three of diagonal rectangle feature types in Fig. 5). At first using 7 rectangle feature types, we found the discernable rectangle feature types by using facial expression images in the same manner as that outlined in Section 3.2. Secondly, through application of the best discernable rectangle features to the same images and comparison of the feature values, we conduct a qualitative analysis.

Weak Classifier Comparison

To compare the recognition rate of the weak classifiers, Fig. 6(a) shows a rectangle feature that effectively distinguishes the neutral facial expression among the 7 rectangle feature types and Fig. 6(b) presents the best discernable feature type among the 35 rectangle feature types. These are applied to all facial expression images. Each feature value (the intensity difference between the intensity of the white rectangle and the intensity of the block rectangle) for each facial expression is indicated in Fig. 7. Fig. 7(a) shows the feature values in case of Fig. 6(a) and Fig. 7(b) shows the results of Fig. 6(b). The dotted line in Fig. 7 is the threshold value obtained by using AdaBoost algorithm. As seen in Fig. 7, the method using 7 rectangle feature types doesn't distinguish exactly between neutral facial expression and non-neutral expressions. But the method using 35 rectangle feature types can distinguish between neutral expression and other expressions even if using a weak classifier. However, because this analysis is only for one weak classifier, it can not be sure that the recognition rate of the strong classifier is improved. Nevertheless, if each recognition rate of the weak classifier is improved, we can say that the recognition rate of the strong classifier will be improved.

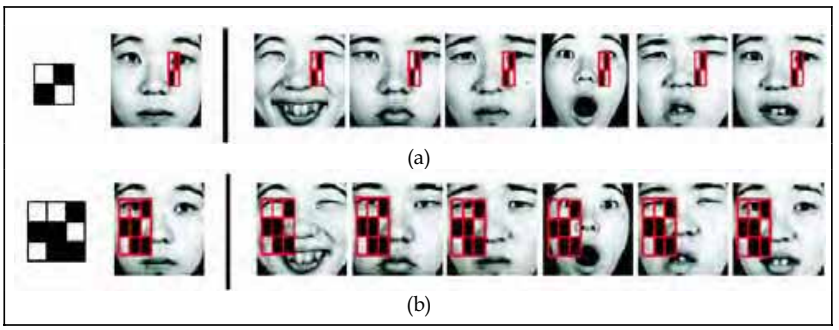


Figure 6. Applying neutral rectangle features to each facial expression: (a) the selected feature that distinguishes a neutral facial expression with the least error from other expressions in using the Viola-Jones features, (b) the selected feature that distinguishes a neutral facial expression with the least error from other expressions using the proposed rectangle features

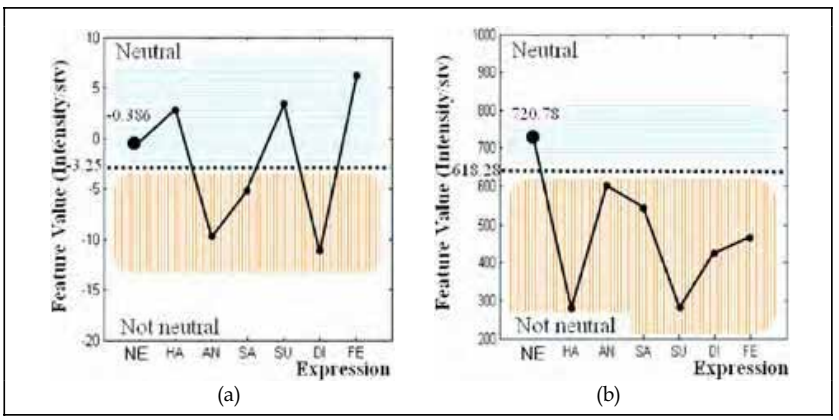


Figure 7. Comparison of feature values in applying a neutral rectangle feature to each facial expression: (a) using the Viola-Jones features, (b) using the proposed features

4. Facial Expression Generation

There are several projects that focus on the development of robotic faces. Robotic faces are currently classified in terms of their appearance; that is, whether they appear real, mechanical or mascot-like. In brief, this classification is based on the existence and flexibility of the robot's skin. The real type of robot has flexible skin, the mechanical type has no skin, and the mascot type has hard skin. Note that although there are many other valuable robotic faces in the world, we could not discuss all robots in this paper because space is limited.

In the real type of robot, there are two representative robotic faces: namely, Saya and Leonardo. Researchers at the Science University of Tokyo developed Saya, which is a

human-like robotic face. The robotic face, which typically resembles a Japanese woman, has hair, teeth, silicone skin, and a large number of control points. Each control point is mapped to a facial action unit (AU) of a human face. The facial AUs characterize how each facial muscle or combination of facial muscles adjusts the skin and facial features to produce human expressions and facial movements (Ekman et al., 2001) (Ekman & Friesen, 2003). With the aid of a camera mounted in the left eyeball, the robotic face can recognize and produce a predefined set of emotive facial expressions (Hara et al., 2001).

In collaboration with the Stan Winston studio, the researchers of Breazeal's laboratory at the Massachusetts Institute of Technology developed the quite realistic robot Leonardo. The studio's artistry and expertise of creating life-like animalistic characters was used to enhance socially intelligent robots. Capable of near-human facial expressions, Leonardo has 61 degrees of freedom (DOFs), 32 of which are in the face alone. It also has 61 motors and a small 16 channel motion control module in an extremely small volume. Moreover, it stands at about 2.5 feet tall, and is one of the most complex and expressive robots ever built (Breazeal, 2002).

With respect to the mechanical looking robot, we must consider the following well-developed robotic faces. Researchers at Takanishi's laboratory developed a robot called the Waseda Eye No.4 or WE-4, which can communicate naturally with humans by expressing human-like emotions. WE-4 has 59 DOFs, 26 of which are in the face. It also has many sensors which serve as sensory organs that can detect extrinsic stimuli such as visual, auditory, cutaneous and olfactory stimuli. WE-4 can also make facial expressions by using its eyebrows, lips, jaw and facial color. The eyebrows consist of flexible sponges, and each eyebrow has four DOFs. For the robot's lips, spindle-shaped springs are used. The lips change their shape by pulling from four directions, and the robot's jaw, which has one DOF, opens and closes the lips. In addition, red and blue electroluminescence sheets are applied to the cheeks, enabling the robot to express red and pale facial colors (Miwa et al., 2002) (Miwa et al., 2003).

Before developing Leonardo, Breazeal's research group at the Massachusetts Institute of Technology developed an expressive anthropomorphic robot called Kismet, which engages people in natural and expressive face-to-face interaction. Kismet perceives a variety of natural social cues from visual and auditory channels, and it delivers social signals to the human caregiver through gaze direction, facial expression, body posture, and vocal babbling. With 15 DOFs, the face of the robot displays a wide assortment of facial expressions which, among other communicative purposes, reflect its emotional state. Kismet's ears have 2 DOFs each; as a result, Kismet can perk its ears in an interested fashion or fold them back in a manner reminiscent of an angry animal. Kismet can also lower each eyebrow, furrow them in frustration, elevate them for surprise, or slant the inner corner of the brow upwards for sadness. Each eyelid can be opened and closed independently, enabling Kismet to wink or blink its eyes. Kismet also has four lip actuators, one at each corner of the mouth; the lips can therefore be curled upwards for a smile or downwards for a frown. Finally, Kismet's jaw has a single DOF (Breazeal, 2002).

The mascot-like robot is represented by a facial robot called Pearl, which was developed at Carnegie Mellon University. Focused on robotic technology for the elderly, the goal of this project is to develop robots that can provide a mobile and personal service for elderly people who suffer from chronic disorders. The robot provides a research platform of social interaction by using a facial robot. However, because this project is aimed at assisting

elderly people, the functions of the robot are focused more on mobility and auditory emotional expressions than on emotive facial expressions.

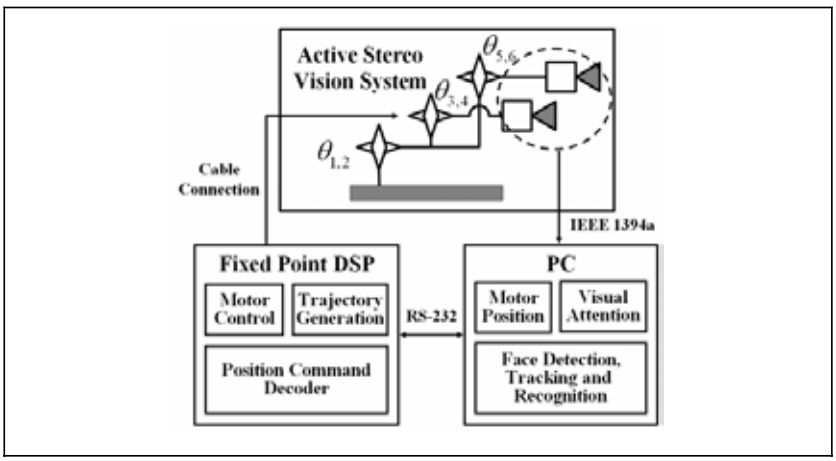


Figure 8. The whole system architecture. The image processing runs on a personal computer (PC) with a commercial microprocessor. In addition, the motion controller operates in a fixed point digital signal processor (DSP). An interface board with a floating point DSP decodes motor position commands and transfers camera images to the PC

Another mascot-like robot called ICat was developed by Philips. This robot is an experimentation platform for human-robot interaction research. ICat can generate many different facial expressions such as happiness, surprise, anger, and sad needed to make the human-robot interactions social. Unfortunately, there is no deep research of emotion models, relation between emotion and facial expressions, and emotional space.

4.1 System Description

The system architecture in this study is designed to meet the challenges of real-time visual-signal processing (nearly 30Hz) and a real-time position control of all actuators (1KHz) with minimal latencies. Ulkni is the name given to the proposed robot. Its vision system is built around a 3 GHz commercial PC. Ulkni's motivational and behavioral systems run on a TMS320F2812 processor and 12 internal position controllers of commercial RC servos. The cameras in the eyes are connected to the PC by an IEEE 1394a interface, and all position commands of the actuators are sent by the RS-232 protocol (see Fig. 8). Ulkni has 12 degrees of freedom (DOF) to control its gaze direction, two DOF for its neck, four DOF for its eyes, and six DOF for other expressive facial components, in this case the eyelids and lips (see Fig. 9). The positions of the proposed robot's eyes and neck are important for gazing toward a target of its attention, especially a human face. The control scheme for this robot is based on a distributed control method owing to RC servos. A commercial RC servo generally has an internal controller; therefore, the position of a RC servo is easily controlled by feeding a signal with a proper pulse width to indicate the desired position, and by then letting the internal controller operate until the current position of the RC servo reaches the desired position.

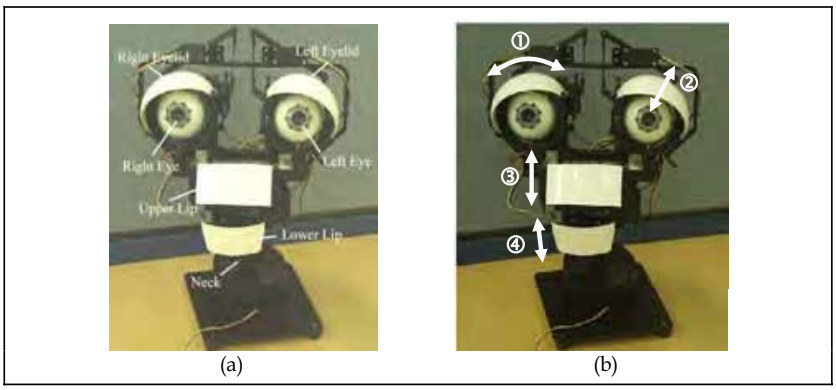


Figure 9. Ulkni's mechanisms. The system has 12 degrees of freedom (DOF). The eyes and the neck can pan and tilt independently. The eyelids also have two DOFs to roll and to blink. The lips can tilt independently. In Ulkni, rotational movements of the eyelids, ①, display the emotion instead of eyebrows. Ulkni's eyelids can droop and it can also squint and close its eyes, ②. Ulkni can smile thanks to the curvature of its lips, ③ and ④

Two objectives of the development of this robot in a control sense were to reduce the jerking motion and to determine the trajectories of the 12 actuators in real time. For this reason, the use of a high-speed, bell-shaped velocity profile of a position trajectory generator was incorporated in order to reduce the magnitude of any jerking motion and to control the 12 actuators in real time. Whenever the target position of an actuator changes drastically, the actuator frequently experiences a severe jerking motion. The jerking motion causes electric noise in the system's power source, worsening the system's controller. It also breaks down mechanical components. The proposed control method for reducing the jerking motion is essentially equal to a bell-shaped velocity profile. In developing the bell-shaped velocity profile, a cosine function was used, as a sinusoidal function is infinitely differentiable. As a result, the control algorithm ensures that the computation time necessary to control the 12 actuators in real time is achieved.

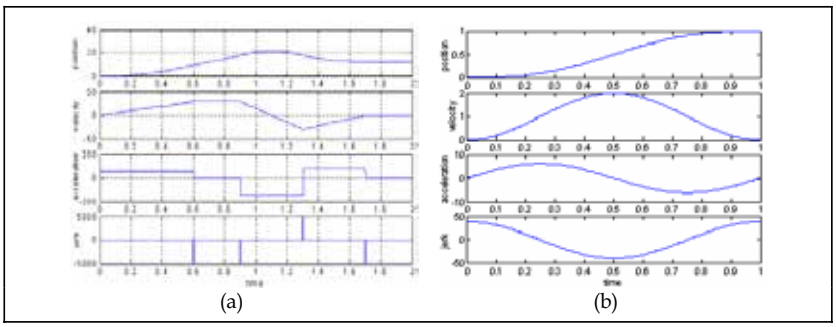
4.2 Control Scheme

Our control scheme is based on the distributed control method owing to RC servos. A commercial RC servo generally has an internal controller. Therefore, the position of a RC servo is easily controlled by feeding the signal that has the proper pulse width, which indicates a desired position, to the RC servo and then letting the internal controller operate until the current position of an RC servo reaches the desired position.

As mentioned above, if the target position of an actuator is changed drastically, severe jerking motion of the actuator will occur frequently. Jerking motions would cause electric noise in the power source of the system, worsen the controller of the system, and break down mechanical components. Therefore, our goal is to reduce or eliminate all jerking motion, if possible.

There have been some previous suggestions on how to solve the problems caused by jerking motion. Lloyd proposed the trajectory generating method, using blend functions (Lloyd & Hayward, 1991). Bazaz proposed a trajectory generator, based on a low-order spline method

(Bazaz & Tondu, 1999). Macfarlane proposed a trajectory generating method using an s-curve acceleration function (Macfarlane & Croft, 2001). Nevertheless, these previous methods spend a large amount of computation time on calculating an actuator's trajectory. Therefore, none of these methods would enable real-time control of our robot, Ulkni, which has 12 actuators.



(a) (b)

Figure 10. Comparison between the typical velocity control and the proposed bell-shaped velocity control: (a) trajectories of motor position, velocity, acceleration, and jerking motion with conventional velocity control, (b) trajectories of those with the proposed bell-shaped velocity control using a sinusoidal function

The two objectives of our research are to reduce jerking motion and to determine the trajectories of twelve actuators in real-time. Therefore, we propose a position trajectory generator using a high-speed, bell-shaped velocity profile, to reduce the magnitude of any jerking motion and to control twelve actuators in real-time. In this section, we will describe the method involved in achieving our objectives.

Fast Bell-shaped Velocity Profile

The bigger the magnitude of jerking motion is, the bigger the variation of acceleration is (Tanaka et al., 1999). Therefore, we can say that reducing the jerking motion is that the function of acceleration should be differentiable.

Presently, nearly all of the position control methods use a trapezoid velocity profile to generate position trajectories. Such methods are based on the assumption of uniform acceleration. Uniform acceleration causes the magnitude of jerking motion to be quite large. If the function of acceleration is not differentiable in any period of time an almost infinite magnitude of jerking motion will occur. Therefore, we should generate a velocity profile with a differentiable acceleration of an actuator (see Fig. 10).

Currently, researchers working with intelligent systems are trying to construct an adequate analysis of human motion. Human beings can grab and reach objects naturally and smoothly. Specifically, humans can track and grab an object smoothly even if the object is moving fast. Some researchers working on the analysis of human motion have begun to model certain kinds of human motions. In the course of such research, it has been discovered that the tips of a human's fingers move with a bell-shaped velocity profile when a human is trying to grab a moving object (Gutman et al., 1992). A bell-shaped velocity is generally differentiable. Therefore, the magnitude of jerking motion is not large and the position of an actuator changes smoothly.

We denote the time by the variable t, the position of an actuator at time t by the variable $p(t)$, the velocity of that at time t by the variable $v(t)$, the acceleration of that at time t by the variable $a(t)$, the goal position by the variable p_T, and the time taken to move the desired position by the variable T.
In (5), a previous model which has a bell-shaped velocity profile is shown.

$$\dot{a}(t) = -9\frac{a(t)}{T} - 36\frac{v(t)}{T^2} + 60\frac{\left(p_T - p(t)\right)}{T^3} \tag{5}$$

The basic idea of our control method to reduce jerking motion is equal to a bell-shaped velocity profile. The proposed algorithm is used to achieve the computation time necessary to control twelve actuators in real-time. Therefore, we import a sinusoidal function because it is infinitely differentiable. We developed a bell-shaped velocity profile by using a cosine function. Assuming the normalized period of time is $0 \le t \le 1$, (6) shows the proposed bell-shaped velocity, the position, the acceleration, and the jerking motion of an actuator (see Fig. 10).
As seen in (6), the acceleration function, $a(t)$, is a differentiable, as well as the velocity function. Therefore, we can obtain a bounded jerk motion function. To implement this method, the position function, $p(t)$, is used in our system.

$$\begin{aligned} v(t) &= 1 - \cos 2\pi t \\ p(t) &= \int_0^t v(t)\,dt = t - \frac{1}{2\pi}\sin 2\pi t \\ a(t) &= \frac{dv(t)}{dt} = 2\pi \sin 2\pi t \\ \dot{a}(t) &= \frac{da(t)}{dt} = 4\pi^2 \cos 2\pi t \end{aligned} \tag{6}$$

$$, \textit{ where } \; 0 \le t \le 1.$$

Finally, the developed system can be controlled in real-time even though target positions of 12 actuators are frequently changed. Some experimental results will be shown later.

4.3 Basic Function; Face Detection and Tracking

The face detection method is similar to that of Viola and Jones. Adaboost-based face detection has gained significant attention. It has a low computational cost and is robust to scale changes. Hence, it is considered as state-of-the-art in the face detection field.

AdaBoost-Based Face Detection

Viola et al. proposed a number of rectangular features for the detection of a human face in real-time (Viola & Jones, 2001) (Jones & Viola, 2003). These simple and efficient rectangular features are used here for the real-time initial face detection. Using a small number of important rectangle features selected and trained by AdaBoost learning algorithm, it was possible to detect the position, size and view of a face correctly. In this detection method, the value of rectangle features is calculated by the difference between the sum of intensities

within the black box and those within the white box. In order to reduce the calculation time of the feature values, integral images are used here. Finally, the AdaBoost learning algorithm selects a small set of weak classifiers from a large number of potential features. Each stage of the boosting process, which selects a new weak classifier, can be viewed as a feature selection process. The AdaBoost algorithm provides an effective learning algorithm on generalization performance.

The implemented initial face detection framework of our robot is designed to handle frontal views (-20~+20 [deg]).

Face Tracking

The face tracking scheme proposed here is a simple successive face detection within a reduced search window for real-time performance. The search for the new face location in the current frame starts at the detected location of the face in the previous frame. The size of the search window is four times bigger than that of the detected face.

4.4 Visual Attention

Gaze direction is a powerful social cue that people use to determine what interests others. By directing the robot's gaze to the visual target, the person interacting with the robot can accurately use the robot's gaze as an indicator of what the robot is attending to. This greatly facilitates the interpretation and readability of the robot's behavior, as the robot reacts specifically to the thing that it is looking at. In this paper, the focus is on the basic behavior of the robot - the eye-contact - for human-robot interaction. The head-eye control system uses the centroid of the face region of the user as the target of interest. The head-eye control process acts on the data from the attention process to center on the eyes on the face region within the visual field.

In an active stereo vision system, it is assumed for the purposes of this study that joint angles, $\boldsymbol{\theta} \in \mathbf{R}^{n\times 1}$, are divided into those for the right camera, $\boldsymbol{\theta}_r \in \mathbf{R}^{n_1\times 1}$, and those for the left camera, $\boldsymbol{\theta}_l \in \mathbf{R}^{n_2\times 1}$, where $n_1 \le n$ and $n_2 \le n$. For instance, if the right camera is mounted on an end-effector which is moving by joints 1, 2, 3, and 4, and the left camera is mounted on another end-effector which is moving by joints 1, 2, 5, and 6, the duplicated joints would be joints 1 and 2.

Robot Jacobian describes a velocity relation between joint angles and end-effectors. From this perspective, it can be said that Ulkni has two end-effectors equipped with two cameras, respectively. Therefore, the robot Jacobian relations are described as $\mathbf{v}_r = \mathbf{J}_r\dot{\boldsymbol{\theta}}_r$, $\mathbf{v}_l = \mathbf{J}_l\dot{\boldsymbol{\theta}}_l$ for the two end-effectors, where $\mathbf{v}_r$ and $\mathbf{v}_l$ are the right and the left end-effector velocities, respectively. Image Jacobian relations are $\dot{\mathbf{s}}_r = \mathbf{L}_r\mathbf{v}_r$ and $\dot{\mathbf{s}}_l = \mathbf{L}_l\mathbf{v}_l$ for the right and the left camera, respectively, where $\dot{\mathbf{s}}_r$ and $\dot{\mathbf{s}}_l$ are velocity vectors of image features, $\mathbf{L}_r$ and $\mathbf{L}_l$ are image Jacobian, and $\mathbf{v}_r$ and $\mathbf{v}_l$ may be the right and the left camera velocity if the cameras are located at the corresponding end-effectors. The image Jacobian is calculated for each feature point in the right and the left image, as in (7). In (7), Z is rough depth value of features and (x, y) is a feature position in a normalized image plane.

$$\mathbf{L}_{i=r,l} = \begin{pmatrix} -1/Z & 0 & x/Z & xy & -(1+x^2) & y \\ 0 & -1/Z & y/Z & 1+y^2 & -xy & -x \end{pmatrix} \tag{7}$$

In order to simplify control of the robotic system, it was assumed that the robot is a pan-tilt unit with redundant joints. For example, the panning motion of the right (left) camera is moved by joints 1 and 3(5), and tilt motion by joint 2 and 4(6). The interaction matrix is re-calculated as follows:

$$\dot{\mathbf{s}}_i = \tilde{\mathbf{L}}_i \dot{\boldsymbol{\theta}}_i,\ \tilde{\mathbf{L}}_i = \begin{pmatrix} xy & -(1+x^2) \\ 1+y^2 & -xy \end{pmatrix} \tag{8}$$

The visual attention task is identical to the task that is the canter of the detected face region to the image center. The visual attention task can be considered a regulation problem, $\dot{\mathbf{s}}_i = \mathbf{s}_i - \mathbf{0} = \mathbf{s}_i$. As a result, the joint angles are updated, as follows:

$$\begin{aligned} \dot{\boldsymbol{\theta}}_i &= \tilde{\mathbf{L}}_i^{+} \dot{\mathbf{s}}_i = \tilde{\mathbf{L}}_i^{+} \mathbf{s}_i \\ &= \frac{1}{x^2+y^2+1} \begin{pmatrix} -xy & 1+x^2 \\ \left(1+y^2\right) & xy \end{pmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \\ &= \lambda\left(\mathbf{s}_i\right) \begin{pmatrix} y \\ -x \end{pmatrix} \end{aligned} \tag{9}$$

Two eyes are separately moved with a pan-tilt motion by the simple control law, as shown in (8). The neck of the robot is also moved by the control law in (9) with a different weight.

5. Experimental Results

5.1 Recognition Rate Comparison

The test database for the recognition rate comparison consists of a total of 407 frontal face images selected from the AR face database, PICS (The Psychological Image Collection at Stirling) database and Ekman's face database. Unlike the trained database, the test database was selected at random from the above facial image database. Consequently, the database has not only normal frontal face images, but also slightly rotated face images in plane and out of plane, faces with eye glasses, and faces with facial hair. Furthermore, whereas the training database consists of Japanese female face images, the test database is comprised of all races. As a result, the recognition rate is not sufficiently high because we do not consider image rotation and other races. However, it can be used to compare the results of the 7 rectangle feature types and those of the 42 rectangle feature types. Fig. 11 shows the recognition rate for each facial expression and the recognition comparison results of the 7 rectangle feature case and the 42 rectangle feature case using the test database. In Fig. 11, emotions are indicated in abbreviated form. For example, NE is neutral facial expression and HA is happy facial expression. As seen in Fig. 11, happy and surprised expressions show facial features having higher recognition rate than other facial expressions.

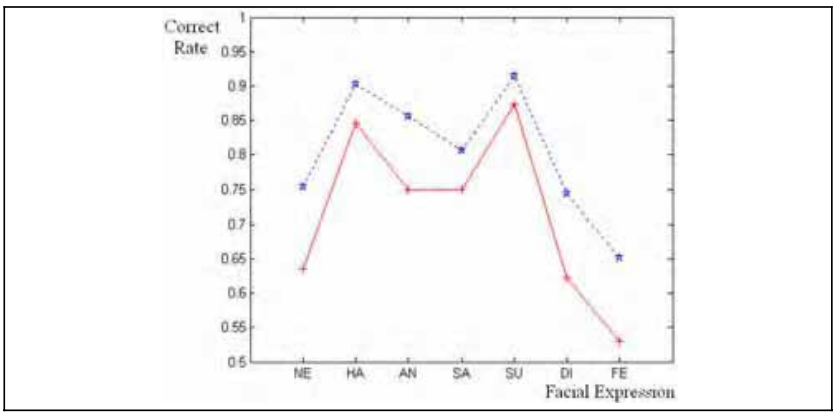


Figure 11. Comparison of the rate of facial expression recognition; '+' line shows the correct rate of each facial expression in case of 7 (Viola's) feature types, '*' line shows the correct rate of each facial expression in case of the proposed 42 feature types

In total, the 7 rectangle feature case has a lower recognition rate than the 42 rectangle feature case. For emotion, the difference ranges from 5% to 10% in the recognition rate. As indicated by the above results, it is more efficient to use 42 types of rectangle features than 7 rectangle features when training face images.

5.2 Processing Time Results

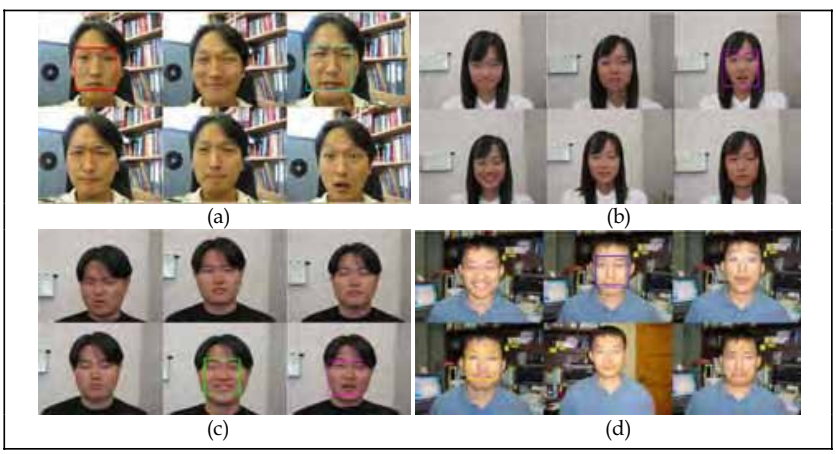


Figure 12. Facial expression recognition experiment from various facial expressions

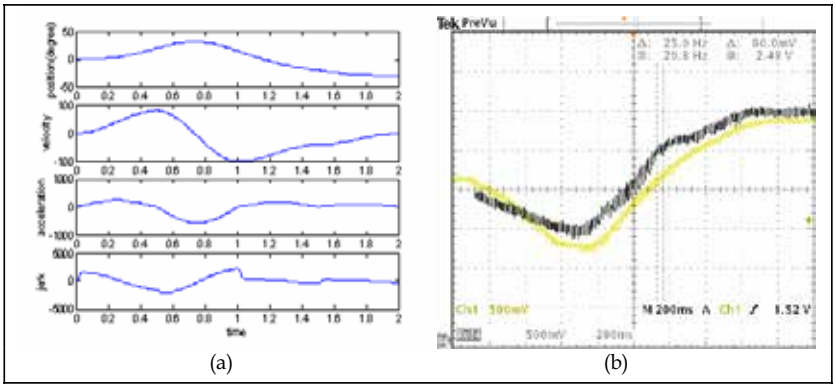
Our system uses a Pentium IV, 2.8GHz CPU and obtains 320×240 input images from a camera. 250~300ms are required for a face to be detected for the case of an input image

where the face is directed forward. In the process of initial face detection, the system searches the entire input image area (320×240 pixels) and selects a candidate region and then performs pattern classification in the candidate region. As such, this takes longer than face tracking. Once the face is detected, the system searches the face in the tracking window. This reduces the processing time remarkably. Finally, when the face is detected, the system can deal with 20~25 image frames per second.

Figure 13. Facial expression recognition results. These photos are captured at frame 0, 30, 60, 90, 120, 150, 180 and 210 respectively (about 20 frames/sec)

5.2 Facial Expression Recognition Experiments

In Fig. 12, the test images are made by merging 6 facial expression images into one image.



(a) (b)

Figure 14. The initial position of an actuator is zero. Three goal positions are commanded at times 0, 1, and 1.5 seconds respectively: $p_T(0)=40$, $p_T(1)=10$, $p_T(1.5)=-30$. The velocity function can be obtained by merging the three bell-shaped velocity profiles, which are created at 0, 1, and 1.5 seconds, respectively: (a) simulation results and (b) experimental results

We then attempted to find specific facial expressions. Neutral and disgusted facial expressions are shown in Fig. 12(a), neutral and surprised facial expressions in Fig. 12(b),

happy and surprised facial expressions in Fig. 12(c) and angry and sad facial expressions in 12(d). It can be seen that the target facial expressions are detected. Fast processing time can be an essential factor for a facial expression recognition system since the facial expression may last only in a few seconds. Thus, we did the experiments for the sequential facial expression images. Fig. 13 shows the process of recognizing 7 facial expressions (neutral, happiness, anger, sadness, surprise, disgust, fear) in order.

5.3 Facial Expression Generation

First, we evaluated the proposed jerk-minimized control method to one joint actuator. The initial position of an actuator is zero, $p(0)=0$. Then, we commanded three goal positions: 40, 10, and -30, at 0 second, 1 second, and 1.5 seconds respectively. That is, $p_T(0)=40$, $p_T(1)=10$, $p_T(1.5)=-30$. Depicting this situation, Figure 14(a) shows the simulation results, which are calculated by Matlab, and Figure 14(b) shows the experimental results, with a comparison between the previous method, using a trapezoid velocity profile, and the proposed method, using a fast bell-shaped velocity profile. Figure 14(a) illustrates that there is less noise than in the previous method and the position trajectory of the motor is smoother than that of the previous method. In addition, an infinite magnitude of jerking motion is not found in Figure 14(b). That is, the magnitude of jerking motion is bounded within the limited area.

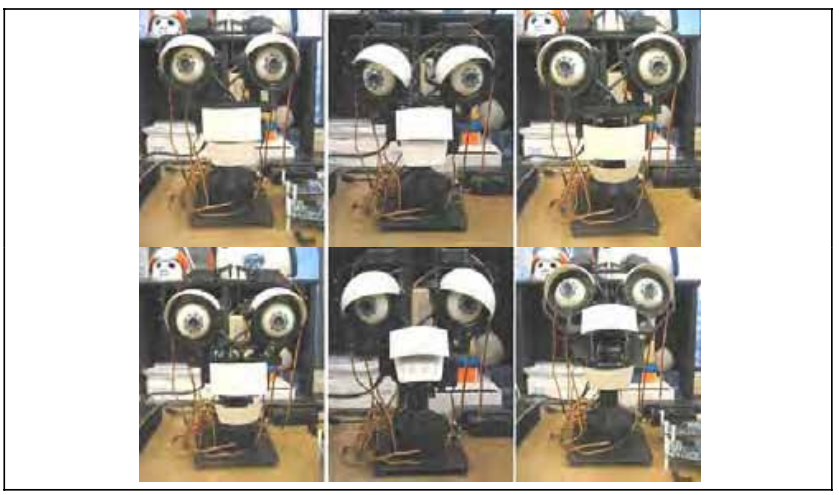


Figure 15. Ulkni's various facial expressions. There are 6 facial expressions which show his status; neutral, anger, happiness, fear, sadness, and surprise from the left-top to the right-bottom

Ulkni is composed of 12 RC servos, with four degrees of freedom (DOF) to control its gaze direction, two DOF for its neck, and six DOF for its eyelids and lips. Figure 15 shows the experimental results of the facial expression generation, such as normality, surprise,

drowsiness, anger, happiness, winsomeness. To express these kinds of facial expressions, we used the distributed control structure and the proposed fast bell-shaped velocity profiler. Our control scheme can simultaneously move the 12 actuators of our robot, Ulkni, in real-time and minimize the jerking motion of the actuators. The controller can update a desired change of position easily and smoothly, even if an actuator has not reached the previous goal position yet.

6. Conclusion

This Chapter has attempted to deal with the issues on establishing a facial expression imitation system for natural and intuitive interactions with humans. Several real-time cognition abilities are implemented to a robotic system such as face detection, face tracking, and facial expression recognition. Moreover, a robotic system with facial components is developed, which is able to imitate human's facial expressions.

A method of recognizing facial expressions is proposed through the use of an innovative rectangle feature. Using the AdaBoost algorithm, an expanded version of Viola and Jones' method has been suggested as a new approach. We deal with 7 facial expressions: neutral, happiness, anger, sadness, surprise, disgust, and fear. For each facial expression, we found five suitable rectangle features using the AdaBoost learning algorithm. These 35 rectangle features and 7 rectangle features were used to find new weak classifiers for facial expression recognition. A real-time performance rate can be achieved through constructing the strong classifier while extracting a few efficient weak classifiers by AdaBoost learning.

In addition, an active vision system for social interaction with humans is developed. We proposed a high-speed bell-shaped velocity profiler to reduce the magnitude of jerking motion and used this method to control 12 actuators in real-time. We proved our distributed control structure and the proposed fast bell-shaped velocity profiler to be practical. Several basic algorithms, face detection and tracking, are implemented on the developed system.

By directing the robot's gaze to the visual target, the person interacting with the robot can accurately use the robot's gaze as an indicator of what the robot is attending to. This greatly facilitates the interpretation and readability of the robot's behavior, as the robot reacts specifically to the thing that it is looking at. In order to implement visual attention, the basic functionality mentioned above, e.g. face detection, tracking and motor control, is needed.

Finally, we introduced an artificial facial expression imitation system using a robot head. There are a number of real-time issues for developing the robotic system. In this Chapter, one solution for developing it is addressed. Our final goal of this research is that humans can easily perceive motor actions semantically and intuitively, regardless of what the robot intends. However, our research lacks a sound understanding of natural and intuitive social interactions among humans. Our future research will focus on perceiving the mental model of human to apply it to the robotic system. It is expected that the suitable mental model for the robots will convey robot's emotion by facial expressions.

7. References

Bartlett, M. S. (1998). *Face Image Analysis by Unsupervised Learning and Redundancy Reduction*, PhD thesis, Universiy of California, San Diego.

Bazaz, S. & Tondu, B. (1999). Minimum Time On-line Joint Trajectory Generator Based on Low Order Spline Method for Industrial Manipulators, *Robotics and Autonomous Systems*, Vol. 29, pp. 257-268.

Breazeal, C. (2002). *Designing Sociable Robots*, MIT Press, Cambridge, MA, USA

Ekman, P.; Friesen, W. V. & Hager, J. C. (2002). *Facial Action Coding System*, Human Face, Salt Lake, Utah, USA

Ekman, P. & Friesen, W. V. (2003). *Unmasking the Face*, Malor Books, Cambridge, MA, USA.

Essa I. A. & Pentland A. P. (1997). Coding, Analysis, Interpretation, and Recognition of Facial Expressions, *IEEE Transaction on Pattern Analysis and Machine Intelligence*, Vol. 19, No. 7, pp. 757-763.

Fasel, B. & Luettin, J. (2003). Automatic Facial Expression Analysis: A Survey, *Pattern Recognition*, Vol. 36, No. 1, pp. 259-275.

Fellenz, W. A.; Taylor, J. G.; Tsapatsoulis, N. & Kollias, S. (1999). Comparing Template-Based, Feature-Based and Supervised Classification of Facial Expressions form Static Images, *Proceedings of Circuits, Systems, Communications and Computers*, pp. 5331-5336.

Freund, Y. & Schapire, R. E. (1995). A Decision-theoretic Generalization of On-line Learning and an Application to Boosting, *Computational Learning Theory: Eurocolt'95*, pp. 23-37.

Gutman, S. R.; Gottlieb, G. & Corcos, D. (1992). Exponential Model of a Reaching Movement Trajectory with Nonlinear Time, *Comments Theoretical Biology*, Vol. 2, No. 5, pp. 357-383.

Hara, F.; Akazawa, H. & Kobayashi, H. (2001). Realistic Facial Expressions by SMA Driven Face Robot, *Proceedings of IEEE Int. Workshop on Robot and Human Interactive Communication*, pp. 504-511.

Huang, C. L. & Huang, Y. M. (1997). Facial Expression Recognition using Model-Based Feature Extraction and Action Parameters Calssification, *Journal of Visual Communication and Image Representation*, Vol. 8, No. 3, pp. 278-290.

Jones, M. & Viola, P. (2003). Fast Multi-View Face Detection. *Technical Report of Mitsubishi Electric Research Laboratory*: TR-2003-96, USA.

Kim, D. H.; Lee, H. S. & Chung, M. J. (2005). Biologically Inspired Models and Hardware for Emotive Facial Expressions, *Proceedings of IEEE Int. Workshop Robot and Human Interactive Communication,* pp. 680-685, Tennessee, USA

Lanitis, A.; Taylor, C. J. & Cootes, T. F. (1997). Automatic Interpretation and Coding of Face Images using Flexible Models, *IEEE Transaction on Pattern Analysis and Machine Intelligence*, Vol 19, No, 7, pp. 743-756.

Lien, J. J.; Kanade, T.; Cohn, J. F. & Li, C-C. (1998). Automated Facial Expression Recognition Based FACS Action Units, *Proceedings of IEEE International Conference on Automatic Face and Gesture Recognition*, pp. 390-395.

Lisetti, C. L. & Rumelhart, D. E. (1998). Facial Expression Recognition using a Neural Network, *Proceedings of the International Flairs Conference*. AAAI Press.

Littlewort, G. C.; Bartlett, M. S.; Chenu, J.; Fasel, I.; Kanda, T.; Ishiguro, H. & Movellan, J. R. (2004). Towards Social Robots: Automatic Evaluation of Human-Robot Interaction by Face Detection and Expression Classification, In: *Advances in Neural Information Processing Systems*, S. Thrun, L. Saul and B. Schoelkopf, (Eds.), Vol 16, pp. 1563-1570, MIT Press.

Lloyd, J. & Hayward, V. (1991). Real-Time Trajectory Generation Using Blend Functions, *Proceedings of IEEE Int. Conf. Robotics and Automation*, pp. 784-789.

Macfarlane, S. & Croft, E. (2001). Design of Jerk Bounded Trajectories for On-line Industrial Robot Applications, *Proceedings of IEEE Int. Conf. Robotics and Automation*, pp. 979-984.

Miwa, H.; Okuchi, T.; Takanobu, H. & Takanishi, A. (2002). Development of a New Human-like Head Robot WE-4, *Proceedings of IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, pp. 2443-2448.

Miwa, H.; Okuchi, T.; Itoh, K.; Takanobu, H. & Takanishi, A. (2003). A New Mental Model for Humanoid Robots for Human Friendly Communication, *Proceedings of IEEE Int. Conf. on Robotics and Automation*, pp. 3588-3593.

Mcneill, D. (1998). *The Face*, Little, Brown & Company, Boston, USA

Padgett, C. & Cottrell, G. W. (1997). Representing Face Image for Emotion Classification, In: *Advances in Neural Information Proceessing Systems*, M. Mozer, M. Jordan, and T. Petsche, (Eds.), Vol. 9, pp. 894-900, MIT Press.

Pantic, M. & Rothkrantz, L. J. M. (2000). Automatic Analysis of Facial Expression: the State of Art, *IEEE Transaction on Pattern Analysis and Machine Intelligence*, Vol. 22, No. 12, pp. 1424-1445.

Tanaka, Y.; Tsuji, T. & Kaneko, M. (1999). Bio-mimetic Trajectory Generation of Robotics using Time Base Generator, *Proceedings of IEEE Int. Conf. Intelligent Robots and Systems*, pp. 1301-1315.

Viola, P. & Jones, M. (2001). Rapid Object Detection using a Boosted Cascade of Simple Features, *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition*, pp. 511-518.

Wang, Y.; Ai, H.; Wu, B. & Huang, C. (2004). Real Time Facial Expression Recognition with Adaboost, *Proceedings of IEEE International Conference on Pattern Recognition*, vol. 3, pp. 926-929.

Zhang, Z.; Lyons, M.; Schuster, M. & Akamatsu, S. (1998). Comparision between Geometry-Based and Gabor-Wavelets-Based Facial Expression Recognition using Multi-Layer Perceptron, *Proceedings of IEEE International Conference on Automatic Face and Gesture Recognition*, pp. 454-459.

12

Evaluating Emotion Expressing Robots in Affective Space

Kolja Kühnlenz, Stefan Sosnowski and Martin Buss
Institute of Automatic Control Engineering, Technische Universität München
Germany

1. Introduction

Research on human emotions has been an area of increased interest in the field of human-robot interaction in the last decade. Subfields reach from usability studies over emotionally enriched communication to even social integration in human-robot groups. Prominent aims are the investigation of the impact of emotional responses, perception of emotions, and emotional decision making on the efficiency and robustness of the interaction process. Intuitive communication and easy familiarization are other factors of major interest.

In order to facilitate emotionally enriched communication, means of expressing "emotional states" of a robot are necessary, i.e. expressive features, which can be used to induce emotions in the human or simply to provide additional cues on the progression of the communication or interaction process. A common approach is the integration of facial expression elements in the robot artefact as very elaborated frameworks on human facial expressions exist, which can be utilized, e.g. (Blow et al., 2006; Breazeal, 2002a; Grammer & Oberzaucher, 2006; Hara & Kobayashi, 1996; Sosnowski et al., 2006a; Zecca et al., 2004).

The design and control of such expressive elements have a significant impact on how the represented emotional state of the robot is perceived by the human counterpart. Particularly, the controlled posture is an important aspect and a well investigated issue in human nonverbal communication considering facial expressions. Common frameworks are works using the Facial Action Coding System (FACS) (Ekman & Friesen, 1977) and variants establishing the link between muscular activations and facial expressions, i.e. the quantitative contribution of muscular group poses to perceived emotions, e.g. (Grammer & Oberzaucher, 2006). Such a design approach is dimensional (continuous) in nature as a continuous representation of the emotional state space composed of the dimensions valence/pleasure, arousal, and dominance/stance is used and the contribution of muscular group poses to these components is provided.

The choice of concept for the evaluation of displayed facial expressions is an issue of equal importance. A comprehensive evaluation is essential as the actuating elements of the robot (motors, joints, transmission elements, etc.) differ significantly from those of the human. Thus, although elaborated frameworks as e.g. FACS are used in the design process a significant deviation of the intended and perceived expression can be expected. Common evaluation procedures use a categorical approach where test participants may choose best fits from a set.

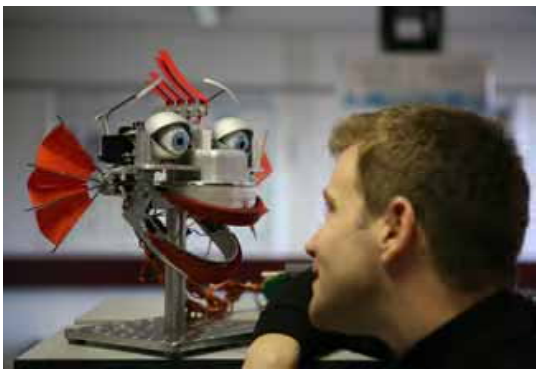


Figure 1. Robot head EDDIE with actuated features for displaying facial and non-facial expressions (Sosnowski et al., 2006a)

Design and evaluation are, thus, commonly conducted using different models. In consequence evaluations are of low test-theoretical validity as a psychological test not only examines the subject but also the theory in which the subject is embedded. Another shortcoming is the lack of feedback for design improvements, e.g. guidelines for posture adjustments of the expressive features.

This chapter discusses the use of dimensional approaches for evaluation of emotion expressing robots. In the first part a theoretical framework is established and guidelines for evaluation are derived. In the second part these guidelines are exemplarily implemented for illustration purposes and two user studies are presented evaluating the robot head EDDIE.

An additional contribution of the framework discussed in the second part is the use of dimensional evaluation approaches as a generic tool for integrating expressive elements of arbitrary design (e.g. animal-like) for which no common framework as e.g. FACS exists. This is illustrated by a third pilot user study evaluating the impact of EDDIE's ears and crown on the dimensions of the emotion model.

The chapter is organized as follows: Section 2 introduces and discusses common dimensional measures exemplified in the field of emotional expressions; Section 3 presents the application of a semantic differential approach to the evaluation of a facial expression robot head; conclusions are given in Section 4.

2. Dimensional Evaluation Approaches

2.1 Introduction of Quality Measures for Tests

By definition, a test is a specific psychological experiment. The goal of this experiment is to obtain comparative judgments about different subjects and their attitudes, impressions, or psychological and physiological variables (Ertl, 1965). The variables measured in a subject can be divided into two groups: latent and manifest variables. Manifest variables are easily observable like the height or the weight of a person. Latent variables like attitudes, feelings or personal traits are not directly observable and, thus, have to be derived from the answering behavior in a test. The idea of using a test in order to obtain information of a latent variable is depicted in Figure 2.

Figure 2. Correlation of answering behavior in a test and latent variables

Thereby, it is assumed that the latent variable influences the answering behavior in a test. After having obtained the answers, the latent variable is deduced from the observed answers of the test (mostly a questionnaire or an interview). This deduction is the most difficult part of the construction of a test since the correlation of latent variables and answering behavior in the test cannot be formulated easily. For that reason, it is very important that the deduction of the latent variable shall be embedded in a profound theory of how true feelings and attitudes of people can be obtained by using tests. In the following paragraph the most common and mostly used theory of measuring feelings and attitudes is described.

2.2 The Semantic Differential

Osgood et al. tried to connect scaled measurement of attitudes to the connotative meaning of words (Osgood et al., 1957). In their classical experiment they discovered that the whole semantic space of words can be described by just three dimensions. These dimensions are evaluation (e.g. good vs. bad), potency (e.g. strong vs. weak), and activity (e.g. aroused vs. calm). The measurement technique they used is called semantic differential (approx. 20-30 bipolar adjectives on a seven-point Likert-scale). By using this method, a person's attitude can be plotted in a semantic space and different subjects' attitudes towards a product or object can be compared. Due to the wide range of usage, it became a standard tool in marketing, advertising, and attitude research. Applying this knowledge to the field of emotion research, (Ertl, 1965) and (Mehrabian and Russell, 1974) showed that also emotional adjectives can be reduced to a three-dimensional (affective) space with the dimensions valence/pleasure, arousal, and dominance/stance, see Figure 3a. Thereby, the three dimensions found by (Osgood, 1957) can easily by transformed into the dimensions found by (Mehrabian and Russell, 1974).

Valence can be interpreted as the evaluation dimension, arousal as the activity dimension, and potency can be referred to as the dominance dimension. From his dimensional paradigm (Mehrabian & Russell, 1974), Mehrabian developed a test system called Pleasure-Arousal-Dominance (PAD) emotion model (Mehrabian, 1998). The use of Mehrabian's PAD-test to measure affective experiences represents a generic way to gather self-report-based user data regarding emotions. In the following paragraph the PAD-test is described in more detail.

2.3 The PAD-Emotion Model by Mehrabian

The test is divided into three scales: a 16-item pleasure-displeasure scale, a 9-item arousal-non-arousal scale, and a 9-item dominance-submissiveness scale. The items of the PAD-test are also in the format of a semantic differential. The duration of this test is approximately 7 minutes (Mehrabian, 2007). Alternatively, a short 12-item version exists. Each emotion expressing robot can, thus, be rated in 2-3 minutes. The reliability (internal consistency) of

the full-length version is (α_c: Cronbach's alpha) α_c=0.97 for the pleasure scale, α_c=0.89 for the arousal scale, and α_c=0.80 for the dominance scale. The internal consistency for the abbreviated version is α_c=0.95 for pleasure scale, α_c=0.83 for the arousal scale, and α_c=0.78 for the dominance scale (Mehrabian & Russell, 1974).

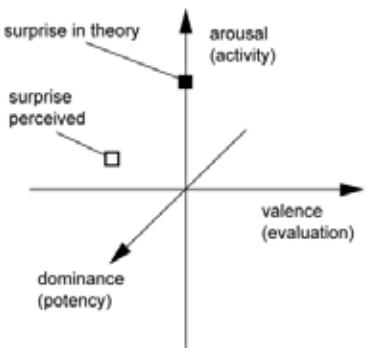


Figure 3a. Results of the semantic differential

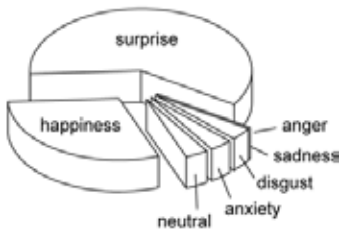


Figure 3b. Results of a categorical test

As a result of the fact that the affective space (with the dimensions valence, arousal and dominance) (Mehrabian & Russell, 1974) or the derived two-dimensional version of the circumplex model of emotion (Russell, 1997) are a common theoretical basis for building emotion expressing robots, only the PAD-model has been introduced as a method to obtain data from affective experiences. There are several other tests measuring attitudes and feelings, e.g. the PANAS (Watson et al., 1988), which, however, are not as easily transferable into the theoretical model of emotion expressing robots.

2.4 Advantages of Dimensional Approaches as the Semantic Differential

Generally, two approaches to evaluate emotion expressing robots exist. On the one hand, dimensional approaches like the semantic differential can be used representing emotion states in continuous affective space. On the other hand, emotions can be treated as quasi-independent categories by asking the subjects to choose the perceived emotion state of the robot from a list of possible emotions (Breazeal, 2002b).
In this chapter, dimensional approaches like the PAD-model are proposed as a generic tool for dimensionally designed emotion expressing robots due to the following advantages:

2.4.1 Test-Theory

A test is conducted in order to examine the performance of the emotion expressing robot. Thus, the test also examines the theory in which the robot is embedded. In order to do so, the test has to be derived from the same theoretical framework as the robot. So, if the robot is embedded in the framework of the circumplex model of Russell (Russell, 1997) or the affective space of Mehrabian (Mehrabian & Russell, 1974) then the test has to be embedded in the same framework. In case of the affective space, which is widely used as a starting point for development, the PAD-test meets this requirement.

2.4.2 Test-Construction

As discussed above, there has to be a substantiated description of the conjunction between latent variable and answering behavior in the test. Since the PAD-model is based on the established and well-grounded model of the semantic space (Osgood et al., 1957), it meets this premise.

2.4.3 Guidelines for Improvements

The result of an evaluation experiment should provide concrete instructions how the evaluated device can be improved. If the robot is theoretically embedded in affective space the semantic differential provides well interpretable data on the quality of the emotion expression on all three dimensions. Figure 3a shows an example with fictive data for the perceived emotion 'surprise' and the theoretical position of 'surprise' in affective space. Due to the fact that certain activation units are linked directly to certain dimensions/areas of the affective space and, moreover, the activation units are linked to certain artificial muscles of the emotion expressing robot, it is possible to conclude the contribution of the artificial facial muscle from the measured position in affective space. Thus, by using the semantic differential more of the gathered information can be processed and interpreted.
Contrarily, if a list of categorical emotions is used for evaluation, only data with "correct hits" is gathered. Such a categorical test would only state that the emotion 'surprise' is identified correctly by a certain percentage. Furthermore, from this result it would be unclear which affective dimension of the displayed emotion is identified well or poorly. In consequence, no conclusion on how to adjust the corresponding joint positions in order to improve the displayed emotion can be drawn. Figure 3b exemplarily shows the fictive result of a categorical evaluation method. It can be noted that about 50% of the participants evaluate the facial expression of the robot as 'surprise', around 30% perceive it as 'happiness' and the rest as other emotions. To this point, no method exists to derive guidelines for improvement of the robot from this evaluation.

2.4.4 Weighting Algorithm

In (Breazeal, 2002b) it is argued that if ten items in a multiple choice test are provided then the probability of chance for each item to be picked is ten percent and, thus, the expected chance probability of each emotion to be picked in such a test would be also ten percent. However, this assumption does not hold. Due to the fact that some emotions share more activation units than others (Grammer & Oberzaucher, 2006) and some lie closer together in the affective space, the probability of each emotion to be picked has to be weighted accounting for these issues. Thus, the expected probability of ten percent of the mentioned example would to be increased for similar emotions and decreased for dissimilar emotions. Yet, an algorithm for weighting the expected probability has not been developed. Such an algorithm, however, would be needed for a categorical test since these expected probabilities are used in statistical tests to analyze whether a displayed emotion is classified sufficiently correctly.

2.4.5 Reliability and Validity

Each test, which is used should provide data on its quality. Otherwise, the quality of the result of the test cannot be assessed. The value of a test can be rated by quality measures of classical test theory: reliability, validity, and objectivity. These have been evaluated for dimensional tests, e.g. for the PAD-test. Furthermore, this test has been used in other

studies, e.g. (Valdez & Mehrabian, 2006; Mehrabian et al., 1997), and evaluated towards other tests, which also test affective experiences (Mehrabian, 1997).

2.4.6 Expressive Features of Arbitrary Design

Emotions of biological entities are expressed by a variety of expressive elements from facial muscles over limb motions to color adjustments and acoustic signals. To date, only a framework for the integration of facial elements for expression of emotional states of artificial entities exists (e.g. FACS). However, this framework assumes actuators, which correspond exactly to those of the biological paradigm. If these are not matched well, then the controlled expressive element provides a different contribution to the emotion state to be expressed and, thus, a different emotion will be perceived as is intended. A framework for adjustments of expressive element motions in order to match the intended emotion state better has not been established yet. Dimensional approaches like the PAD-model can be utilized not only to accomplish the desired adjustment, but also to evaluate and adjust the impact of expressive features of arbitrary design (e.g. animal-like) due to the provided guidelines for design improvement (Bittermann et al., 2007).

3. Application of the Guidelines

3.1 Comparative Evaluation Study

In order to illustrate the advantages of dimensional evaluation approaches as the semantic differential for evaluation of emotion expressing robots, two user studies have been conducted exemplarily evaluating the robot head EDDIE, which has been developed at the authors' lab (Sosnowski et al., 2006a). These studies are only intended to show up the differences of both approaches and do not account for demographical issues of the test participants.

The first study uses a test based on a dimensional model of emotions and evaluates the performance of the emotional facial expressions. The second study uses a test based on a categorical emotion model evaluating the same robot. In the following, the system setup is presented in brief, the studies are described, and a discussion is given.

3.1.1 Face Robot EDDIE

EDDIE is a robot head designed for displaying facial expressions, particularly, emotional expressions realizing 13 of the 21 action units of FACS relevant to emotional expressions.. In addition to the facial elements, animal-like features, the crown of a cockatoo and the ears of a dragon lizard with special folding mechanisms, are integrated.

EDDIE is encapsulated accepting commands from a higher-level decision and control unit via a serial communication protocol. The desired displayed emotion state can be transmitted based on the three-dimensional affective space representation and feedback is given in affective and joint space representations. An embedded controller manages the transformation between affective space, action units, and joint space. More details on design and control of EDDIE can be found in (Sosnowski et al., 2006a).

3.1.2 Categorical Evaluation Study

A study with 30 participants (researchers of Technische Universität and the University of the Armed Forces, München, 11 female, 19 male, age-mean 30 years) has been conducted.

Six basic emotions have been presented to each individual. The subjects' ratings of the displayed emotions were rated with a multiple-choice test with a seven-item-scale. The result of this study can be seen in Figure 4. On the abscissa the six displayed emotions are shown. For each displayed emotion the amount of assumed emotions by the subjects is presented. For example, 90% of the participants agree in seeing a sad face if a sad face is displayed. For the displayed emotions 'anger' and 'anxiety' about 50% of the answers were correct, the remaining 50% are shared between the other emotions. Evaluating these results, a scientist should be able to draw conclusions in order to improve the robot. Yet, no framework exists in order to formulate new guidelines for robot improvement considering the incorrect answers. Furthermore, it is questionable not to use the data of the incorrect answers as information would be disregarded.

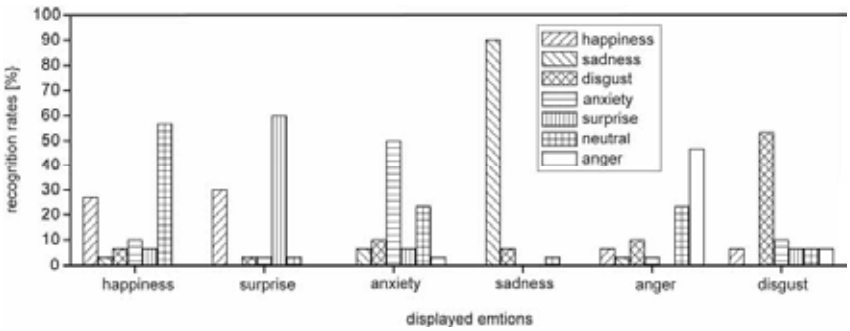


Figure 4. Results of a categorical evaluation study on EDDIE

3.1.3 Dimensional Evaluation Study

A study with 30 participants (students and researchers of the Ludwig-Maximilians-Universität, Munich, 15 female, 15 male, age-mean 25 years) has been conducted. A number of 30 different facial expressions corresponding to certain emotion states has been presented to each subject separately in random order. The subjects' impressions of the shown emotions have been acquired by using a German translation of the semantic differential of Mehrabian. The results of the study are presented in Figure 5 showing the expected emotion values (ground truth) and the values obtained from the study (measurements) in affective space (dimension dominance is not displayed). The results of the study clearly show how each perceived emotion is empirically located in affective space.

3.1.4 Discussion

From these results and the knowledge of the action units actually needed for each specific emotion (Grammer & Oberzaucher, 2006) the quality of the realization of each action unit in the mechatronical robot face can be concluded. Additionally, steps for improvement can be derived from the results. For example, the results in Figure 5 show that the displayed emotion 'fear' has been perceived as a nearly neutral emotion with small values on each dimension. By analyzing this problem, one can compare the amount of action units needed for the intended displayed emotion and the amount of action units realized in the robot. Fear consists of action units 1, 2, 4, 20 and 26. In EDDIE the action units 1 and 2 are

combined for technical reasons and cannot work separately. Furthermore, the action unit 4 was not implemented in the display. Yet, action unit 4 reflects the brow lowerer (musculus corrugatro supercilii) and is important for the emotion 'fear'. Furthermore, the range of action unit 20 (lip stretcher) could have been to small to show a motion, which people expect from their experience with real human emotions. Based on these conclusions direct tasks of improvement can now easily be derived.

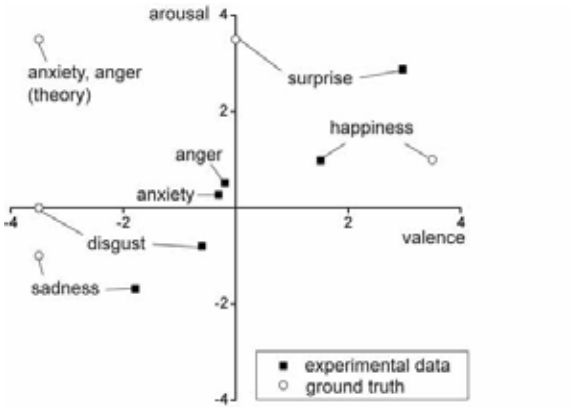


Figure 5. Results of dimensional evaluation study on EDDIE based on the semantic differential approach (without ears and comb)

Not only single emotions but also global deviations of the robot can be assessed. In Figure 5 a global "positive valence shift" can be noted. By comparing the theoretical values of the emotions and the ones actually found, it is obvious that the displayed emotions are all shifted by one or two units to the positive side of the valence dimension. A possible reason for this shift can be noted in Figure 1. The lips of EDDIE are designed in such a way that it seems to smile slightly in most displayed emotion states. Guidelines for improvement are, thus, clear from this point: the lips have to be redesigned. This example shows how even the summarized results can be used to provide new insight into the overall quality of the robot. This is a clear advantage of this method. Taking all the different aspects into consideration, it can be stated that dimensional evaluation methods as the semantic differential approach provide a powerful tool for evaluation of expressive robots by backprojection of joint space into affective space via human perception.

3.2 Integration of Animal-like Features

In a more general context the semantic differential approach can be utilizes as a generic means to evaluate the influence of actuated expressive elements of an arbitrary kind on the perceived emotion. The knowledge gained from such evaluation procedures can then be used for the derivation of control commands for those expressive elements. Thereby, actuated expressive features of arbitrary design can be systematically controlled in order to intensify or attenuate the displayed emotion in a particular selected dimension, e.g. valence, arousal, dominance in case of the affective space.

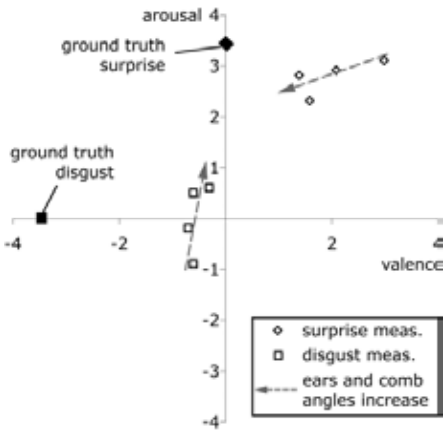


Figure 6. Results for displayed 'surprise' and 'disgust' of a dimensional evaluation study of EDDIE with varied angles of the ears and the comb

3.2.1 A Pilot Study

This is exemplarily shown in an experimental pilot study with 30 participants (15 females, 15 males) evaluating the influence of two animal-like features (crown of a cockatoo and ears of a dragon lizard). In a 2x2 ANOVA design with repeated measures (1. Factor: crown, 2. Factor: ears) it is analyzed whether these two factors shift the observed six basic emotions in affective space. Each factor is realized in four conditions (from fully stilted to dismantled). All six basic emotions are displayed with each combination of the two factors. Afterwards, the participants have to rate each displayed emotion on the verbal semantic differential scale. Every subject has participated in one third of the 96 possible combinations. All data is tested with a Mauchly-test for sphericity. All values are greater than 0.1. Thus, no Greenhouse-Geisser correction is necessary. Missing values are substituted by linear interpolation. Due to incorrect answering behavior some data has to be excluded. For that reason no F-Test can be calculated for 'joy' and 'surprise' (dimension dominance) and 'fear' and 'sadness' (dimensions arousal, dominance). The significant results of the ANOVA are shown in Table 1.

The results suggest that the ears and the crown may have an influence, in particular, for the emotions joy, surprise, anger and disgust. As can be seen in Table 1, mostly the interaction effect between crown and ears becomes significant. For the emotions anger and disgust the animal-like features effect the evaluation of the subjects on all dimensions in affective space. Interestingly, the additional features have a different effect on the evaluation depending on the emotion the robot is expressing. This can also be seen in Figure 6 clearly showing the tendency of the propagation of selected emotion states in affective space while adjusting the attitude angle of the ears and crown simultaneously. While for some emotions a shift towards positive valence and higher arousal side can be noted (e.g. for 'disgust'), the impact of ears and crown is rather reverse for other emotions (e.g. for 'surprise'). The results of the pilot study, thus, suggest further investigations of the usage of new non-humanlike features in emotion expressing robots.

Evaluating the propagation of the perceived emotion 'surprise' in Figure 6 a straightforward control command in order to improve the expression of 'surprise' using the ears and crown in addition to the facial elements is obvious. If ears and crown are fully extended then the perceived emotion 'surprise' moves towards the ground truth. Considering again the global positive shift due to the lip design it can be expected that ground truth 'surprise' is nearly perfectly achievable after a lip redesign using the animal-like features ears and crown. This is a nice result considering that robot and human facial action units differ substantially. The use of additional features, thus, contributes to improve the quality of the artificial emotion expression.
It has successfully been shown that arbitrary expressive features can easily be integrated in a control concept of emotion expressing robots using the semantic differential approach.

emotion	dimension	factor	F-value	p-value
1, 2	V	comb	F(3,12)=4.013	0.034
1, 2	V	ears * comb	F(9.36)=3.631	0.003
1, 2	A	ears * comb	F(9,54)=3.258	0.003
3, 4	V	ears * comb	F(9,18)=5.843	0.001
5, 6	V	ears	F(3,6)=4.835	0.048
5, 6	V	ears * comb	F(9,18)=4.132	0.005
5, 6	A	ears	F(3,6)=67.582	0.000
5, 6	A	comb	F(3,6)=11.987	0.006
5, 6	D	ears	F(3,62)=46.724	0.000
5, 6	D	ears * comb	F(9,18)=9.463	0.000

Table 1. Results of 2x2 ANOVA, repeated measures. (1: hapiness, 2: surprise, 3: anxiety, 4: sadness, 5: anger, 6: disgust; V: valence, A: arousal, D: dominance)

4. Conclusions

In this chapter the use of dimensional approaches for evaluation of an emotion expressing robots is discussed. The advantages of dimensional evaluation studies are pointed out and exemplarily illustrated in evaluation studies on face robot EDDIE using a categorical test and the PAD test based on a semantic differential approach. Main arguments supporting the use dimensional models are provided guidelines for design improvement obtained from the evaluation results and a possible integration of expressive elements of arbitrary design. The impact of additional expressive features of EDDIE (ears and comb), which are not covered by a common design framework as, e.g. the Facial Action Coding System, is exemplarily evaluated in a pilot study showing a significant influence of these features on most emotion states. Based on these results the derivation of control commands for integrating these expressive features in a general framework is discussed.
Future work will cover the improvement of the performance of EDDIE by conducting comprehensive studies accounting also for demographical issues and integrating the additional expressive elements based on the framework established in this chapter.

5. Acknowledgments

The authors would like to thank Dr. Albert Mehrabian (University of California, Los Angeles) for inspiring discussions.
This work is supported in part within the DFG excellence initiative research cluster *Cognition for Technical Systems – CoTeSys*, see also www.cotesys.org.

6. References

Bittermann, A., Kühnlenz, K. & Buss, M. (2007). On the Evaluation of Emotion Expressing Robots, *Proceedings of the IEEE/RAS International Conference on Robotics and Automation (ICRA)*, pp. 2138-2143, 2007, Rome, Italy

Breazeal, C. L. (2002a). *Designing Sociable Robots*, MIT Press, Cambridge, MA, USA

Breazeal, C. L. (2002b). Emotion and Sociable Humanoid Robots, *International Journal of Human-Computer Studies*, Vol. 59, No. 1-2, 2002, pp. 119-155

Blow, M., Dautenhahn, K., Appleby, A., Nehaniv, C. L. & Lee, D. C. (2006). Perception of Robot Smiles and Dimensions for Human-Robot Interaction Design, *Proceedings of the IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)*, pp. 469-474, 2006, Hatfield, UK

Ekman, P. (1982). Methods for measuring facial action, In: *Handbook of methods in nonverbal behavior research*, Scherer, K. R. & Ekman, P., (Eds.), pp. 44-90, Cambridge University Press, Cambridge and New York, UK

Ekman, P. & Friesen, W. V. (1977). *Facial Action Coding,* Consulting Psychologists Press, Palo Alto, CA, USA

Ertl., S. (1965). Standardization of a semantic differential, *Zeitschrift für Experimentelle und Angewandte Psychologie*, Vol. 12, No. 1, 1965, pp. 22-58

Grammer, K. & Oberzaucher, E. (2006). The reconstruction of facial expressions in embodied systems: New approaches to an old problem, *ZIF Mitteilungen*, Vol. 2, 2006, pp. 14-31

Hamm, A. O. & Vaitl, D. (1993). Emotional induction by visual cues, *Psychologische Rundschau*, Vol. 44, 1993, pp. 143-161

Hara, F. & Kobayashi, H. (1996). A Face Robot Able to Recognize and Produce Facial Expression, *Proceedings of the IEEE International Conference on Intelligent Robots and Systems (IROS)*, pp. 1600-1607, 1996, Osaka, Japan

Marx, W. (1997). Semantische Dimensionen des Wortfeldes der Gefühlsbegriffe, *Zeitschrift für Experimentelle Psychologie*, Vol. 44, 1997, pp. 478-494

Mehrabian, A. (1997). Comparison of the PAD and PANAS as models for describing emotions and for differentiating anxiety from depression, *Journal of Psychopathology and Behavioral Assessment*, Vol. 19, 1997, 331-357

Mehrabian, A. (1998). *Manual for a comprehensive system of measures of emotional states: The PAD model*, (Available from Albert Mehrabian, 1130 Alta Mesa Road, Monterey, CA, USA 93940)

Mehrabian, A. (2007). *The PAD Comprehensive Emotion (Affect, Feeling) Tests*, http://www.kaaj.com/psych/scales/emotion.html

Mehrabian, A. & Russell, J. A. (1974). *An approach to environmental psychology*, MIT Press, Cambridge, MA, USA

Mehrabian, A., Wihardja, C. & Lyunggren, E. (1997). Emotional correlates of preferences for situation-activity combinations in everyday life, *Genetic, Social, and General Psychology Monographs*, Vol. 123, 1997, pp. 461-477

Minato, T., Shimada, M., Ishiguro, H. & Itakura, S. (2004). Development of an Android Robot for Studying Human-Robot Interaction, *Proceedings of the International Conference on Industrial and Engineering Applications of Artificial Intelligence and Expert Systems (IEA/AIE)*, pp. 424-434, 2004

Osgood, C. E., Suci, G. J. & Tannenbaum, T. H. (1957). *The measurement of meaning*, University of Illinois Press, Oxford, UK

Rost, J. (1996). *Lehrbuch Testtheorie, Testkonstruktion*, 1. Aufl., Huber, Bern, Austria

Russell, J. A. (1997). Reading emotions from and into faces: Resurrecting a dimensional-contextual perspective, In: *The Psychology of Facial Expression*, Russell, J. A. & Fernandes-Dols, J., (Eds.), pp. 295-320, Cambridge University Press, Cambridge, UK

Sosnowski, S., Kühnlenz, K. & Buss, M. (2006a). EDDIE – An Emotion-Display with Dynamic Intuitive Expressions, *Proceedings of the IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)*, pp. 569-574, 2006, Hatfield, UK

Sosnowski, S., Kühnlenz, K., Bittermann, A. & Buss, M. (2006b). Design and Evaluation of Emotion-Display EDDIE, Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2006, Beijing, China

Valdez, P. & Mehrabian, A. (1994). Effects of color on emotions, *Journal of Experimental Psychology*, Vol. 123, 1994, pp. 394-409

Watson, D., Clark, L. A. & Tellegen, A. (1988). Development and validation of brief measures of positive and negative affect: The PANAS scales, *Journal of Personality and Social Psychology*, Vol. 54, 1988, pp. 1063-1070

Zecca, M., Roccella, S., Carrozza, M. C., Miwa, H., Itoh, K., Cappiello, G., Cabibihan, J.-J., Matsumoto, M., Takanobu, H., Dario, P. & Takanishi, A. (2004). On the Development of the Emotion Expression Humanoid Robot WE-4RII with RCH-1, *Proceedings of the IEEE International Conference on Humanoid Robots (Humanoids)*, pp. 235-252, 2004, Los Angeles, CA, USA

13

Cognitive Robotic Engine: Behavioral Perception Architecture for Human-Robot Interaction

Sukhan Lee, Seung-Min Baek and Jangwon Lee
Sungkyunkwan University, Suwon
Republic of Korea

1. Introduction

For personal or domestic service robots to be successful in the market, it is essential for them to have the capability of natural and dependable interaction with human. However, such a natural and dependable human-robot interaction (HRI) is not so easy to accomplish, as it involves a high level of robotic intelligence for recognizing and understanding human speech, facial expression, gesture, behavior, and intention as well as for generating a proper response to human with artificial synthesis. It is our view that the first key step toward a successful deployment of HRI is to level up the dependability of a robot for recognizing the intention of the human counterpart. For instance, to date, robotic recognition of human speech, as well as human gestures, facial expressions, let alone human intention, is still quite unreliable in a natural setting, despite the tremendous effort by researchers to perfect the machine perception. We observe that the robustness and dependability human enjoys in human-human interaction may not merely come from the fact that human has powerful perceptual organs such as eyes and ears but human is capable of executing a series of behaviors associated with a perceptual goal, for instance, the behaviors related to the collection of additional evidences till the decision is sufficiently credible. In analogy, we claim here that the dependability of robotic recognition of human intention for HRI may not come from the perfection of the individual capabilities for recognizing speech, gesture, facial expression, etc. But, it comes with the automatic generation of robotic behaviors that makes sure of reaching a credible decision for the given perceptual goal.

We present here "Cognitive Robotic Engine (CRE)" that automatically generates such perceptual behaviors as selecting and collecting an optimal set of evidences, for dependable and robust recognition of human intention under a high level of uncertainty and ambiguity. CRE is to demonstrate that the dependability of robotic perception may not come from "the perfection of individual components for perception," but from "the integration of individual components into dependable system behaviors, no matter how imperfect and uncertain individual components may be." CRE presents a novel robotic architecture featuring 1) the spontaneous establishment of ad-hoc missions in connection to perceptual goals, 2) the determination of an optimal set of evidences to be selected and/or collected for processing based on in-situ monitoring of the current situation, 3) the integration of such behavioral building blocks as mission management, evidence selection, evidence

collection, evidence fusion and filtering for decision-making in an asynchronous and concurrent architecture, and 4) the implementation of behavioral personality of a robot under CRE framework. We applied CRE to a robot identifying a caller in a crowed and noisy environment. The experimental results demonstrate the great enhancement of the dependability of robotic caller identification through the proposed behavioral perception approach to HRI based on CRE.

1.1. Issues involved in Conventional Approaches to Human-Robot Interaction

A natural HRI has been an interesting subject of research in robotic community for sometime as a means of establishing viable robotic service to human (Font T, et al., 2003). Accordingly, there have been developed various technologies for understanding human expressions such as speech recognition, gesture recognition, understanding human facial expression, etc. Many of the recent research results show significant advancement in their recognition capabilities for individual sake (Sakaue, et al., 2006, Betkowska, et al., 2007).

However, the advancement of individual components such as face and speech recognition themselves does not guarantee the dependability of HRI required for natural human robot interaction. For instance, recognition of human face or speech becomes problematic should robot be in dark, noisy, or dynamic environment. And, it is a growing consensus that efforts to perfect such individual components may be neither fruitful nor justified. Instead, it makes more sense to integrate individual components into a meaningful cognitive process or behavior that guarantees the required dependability.

In an attempt to generate a more user friendly robot, the EU "Cogniron" Project has adopted the multi-modal interaction framework where multiple human-robot interaction components are fused in order to reduce the dependence of the decision on, potentially uncertain, individual components. (Fritsh, et al., 2005 and Li, et al., 2005, 2007). Similarly, the integration of audio-visual cues based on Bayesian theorem has been conducted in order to deal with uncertainties in speaker localization (Choi, et al., 2006). For the development of a robot photographer, face and gesture detection components are integrated to identify the caller among a crowd of people (Ahn, et al., 2006). However, these approaches stop at the level of reducing uncertainties instead of reaching the level of a cognitive process for dependability in which robot self defines its own missions and generates behaviors for automatically selecting and collecting evidences. It may be interesting to note the work of Bin & Masahide where behavioral status is used to determine visual attention as a cognitive process for automatically selecting optimal evidences (Bin & Masahide, 2007).

Cognitive Robotic Engine (CRE) presented in this chapter aims at establishing a biologically inspired cognitive process that provides dependable and robust recognition of human intention in real noisy environments.

2. Cognitive Robotic Engine (CRE): Conceptual Overview

As an introduction of the concept of CRE, let us consider how a human identifies a caller, if there is, dependably despite the adverse condition of, say, a crowded and noisy party environment. Upon hearing a novel but highly uncertain nature of sound that may indicate someone calling, one immediately registers in his/her consciousness an ad-hoc mission of verifying if there is a caller, if any. This ad-hoc mission will remain in his/her consciousness till the verification is done with a sufficient level of confidence an individual set. With the

registered mission producing a stress, a flurry of asynchronous and concurrent perceptual processing takes place inside in such a way as to reduce the uncertainty as efficiently as possible.

The asynchronous and concurrent perceptual processing starts with laying out an asynchronous and concurrent flow of perceptual building blocks spontaneously, from which a sufficient amount of evidences are collected for the registered ad-hoc mission. The basic perceptual building blocks represent the perception units of various levels of abstraction and complexity, ranging from a lower level of elementary sensing primitives to a higher level of conceptual recognition units. These building blocks are assumed predefined, although individuals may have a different number, type, and quality of these building blocks. In the above example, a sufficient amount of evidences may be quickly assembled from multi-modal sensing cues, including both auditory and visual cues such as calling hand gestures and/or calling facial expressions, generated by an asynchronous and concurrent flow of auditory and visual perception building blocks. Potentially, there may exist a large number of possibilities for laying out an asynchronous and concurrent flow of building blocks, with individual sensors as origins, and for acquiring distinct evidences from different paths of the chosen asynchronous concurrent flow. However, to explore all the possible flows of evidences may be neither possible due to a limit in computational resources, nor desirable for efficiency in time and energy. The key may be to understand an optimal way of constructing an asynchronous concurrent flow of perceptual building blocks for decision, dynamically to the real-time variation of situations and, perhaps, similarly to the way our brain functions.

An asynchronous and concurrent flow of perceptual building blocks is connected to the actions to be taken proactively to collect sensing data of less uncertainty or of additional evidences. Human seldom depends passively on what is sensed only for a decision, if the sensed information is incomplete or uncertain. Rather, human tends to take appropriate actions for gathering a better quality of or a new addition of information. For instance, in the above example, to reduce uncertainty, one may resort to look around to see if he/she can find someone making a calling gesture or to generate a confirmation call, like " is there anybody calling? ", to get a reply. Human dependability of perception is thus deeply linked to the proactive actions for assisting and guiding perception by incorporating action building blocks into an asynchronous and concurrent flow of perceptual building blocks. Action building blocks range from an actuator level of action primitives to a task level of functional units. To explore all the possible ways of incorporating action blocks into a perceptual flow may be prohibitive or infeasible due to a potentially large number of possibilities that action blocks can be incorporated, due to the existence of possible conflicts among action blocks, or due to the cost in time and energy that is associated with taking actions. The key is how to choose action blocks to be incorporated into an asynchronous and concurrent flow of perceptual building blocks in such a way as to achieve an optimal overall efficiency in reaching the decision. This requires evaluating an action block as an information source against the cost in time and energy to exercise it.

Summarizing the above, human dependability in perception may be conjectured as the result of the following exercises:

1. The spontaneous and self-establishment of ad-hoc perceptual missions in connection to particular sensing that drive the subsequent perceptual processes till satisfied.

2. The choice of particular asynchronous and concurrent flow architecture of perceptual building blocks out of a potentially huge number of possible flow architectures as the basis for deriving evidences to be fused together.
3. The incorporation of action blocks into the chosen asynchronous and concurrent flow architecture of perceptual building blocks as a means of proactively collecting sensing data of less uncertainty and of new evidence, which triggers a dynamic reorganization of the asynchronous and concurrent flow architecture of perceptual building blocks.
4. The optimal process control in terms of the choice of a particular asynchronous and concurrent flow architecture of perceptual building blocks to follow as well as of the choice of particular action blocks to be invoked at each sampling time, where the optimality is defined in terms of the time and energy to be consumed for completing the ad-hoc mission, which is in turn a function of the amount of uncertainty reduction and the time and computational resources required for completing the perceptual and action building blocks to be processed. Note the control strategy may differ by individuals since some heuristics are involved in the strategy, due to the complexity of search space leading no definite optimal solution is feasible. However, it is interesting to see that this heuristics actually represent a personality of an individual or a robot that we can exploit for creating robots with personality.

The environment or toolkit that enables the above asynchronous and concurrent flow of a perceptual process, or, in general, a robotic process, is referred to here as Cognitive Robotic Engine (CRE). In what follows, we present a more details on how to implement the above concept in computer by describing 1) an asynchronous and concurrent architecture for CRE with the definition of perceptual and action building blocks, the representation of search space with the partial order and fusion relation of perceptual building blocks as well as with the exclusion relation and organized actions for action building blocks, 2) a method of connecting perceptual and action building blocks, 3) an optimal control of CRE with self-establishment of ad-hoc missions, of choosing a particular flow architecture with the optimality in terms of speed and time, and, finally, 4) a demonstration of the value of CRE by a caller identification experimentation with a robot .

3. CRE Architecture

Overall architecture of CRE system is shown in Fig. 1. CRE consists of three major parts, perceptual, control and action parts. Perceptual part is composed of sensors, perceptual processes which are processed asynchronously and concurrently, precedence and evidence fusion relations through which a robot perceives the environment like a human. Control part takes charge of invoke mission or mission transition and it controls behavior selection or behavior changing. Finally, action part is in charge of robot action such as searching, approaching, and gazing. The system operating procedures are as follows: 1) the sensors receive and transmit external data, 2) the perceptual processes analyze the information, 3) the control part gathers all the information from perceptual processes, and then make a decision, 4) if there is any necessity the action part makes the robot to act. Note that system operates asynchronously.

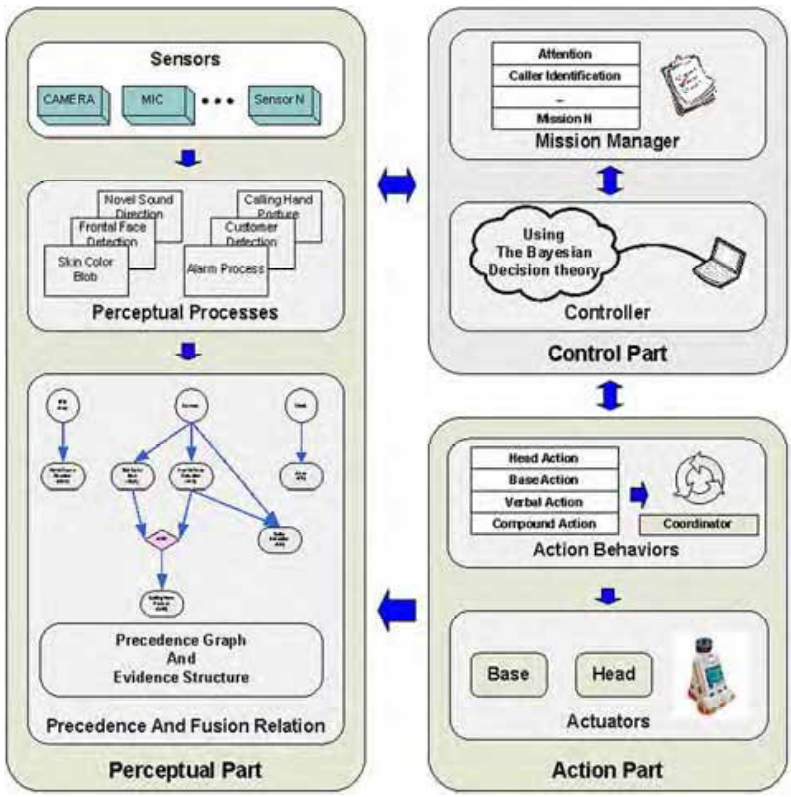


Figure 1. Overall Architecture of Cognitive Robotic Engine

3.1 Perceptual Process and Precedence Relation

Present CRE architecture and all perceptual processes have been organized to accomplish the caller identification mission. The perception process of CRE means basic building block for the entire perception. Table I represents the specification of all perceptual processes – Novel Sound Detection (NSD), Frontal Face Detection (FFD), Skin Color Blob (SCB), Calling Hand Posture (CHP), Color Detection (CD), and Alarm (AL). Normally, the output of individual perceptual process has calculated certainty (CF), spatial probability distribution (SP), action candidates that can improve the certainty factor (AC), processing time (PT), and packet recording time (RT).

NSD	Def.	When the sound volume exceeds the threshold, estimates the direction of source
	Source	Mic array (3 channel)
	Input	Raw data of sound
	Output	Direction of novel sound Calculated Certainty (CF) Spatial probability distribution (SP) Candidate of Action (AC) Processing Time (PT) Packet recording Time (RT)
FFD	Def.	Finds face region by image feature
	Source	Camera
	Input	Raw image from Camera
	Output	Coordinate, and size of detected face CF, SP, AC, PT, RT
SCB	Def.	Distinguishes skin region by RGB condition and makes others black in image
	Source	Camera
	Input	Raw image from Camera
	Output	Image of skin color segmentation Most probable direction that callers exist in. CF, SP, AC, PT, RT
CHP	Def.	Estimates calling hand by skin color in face adjacent area
	Source	FFD, SCB
	Input	Coordinate and size of detected face Skin segmented image
	Output	Direction, and distance of caller CF, SP, AC, PT, RT
CD	Def.	Estimates clothing color of a person who is detected by FFD process.
	Source	Camera, FFD
	Input	Coordinate and size of detected face
	Output	Estimated clothing color (Red/Blue) CF, SP, AC, PT, RT
AL	Def.	Send alarm signal at reservation time
	Source	Time check Thread
	Input	Current time
	Output	Alarm signal Information of reserved user CF, SP, AC, PT, RT

Table 1. Description of Perceptual Processes

If the outputs of one or more processes are necessary as an input or inputs of another for processing, a relationship between the processes defines precedence relation. Each process is

assumed independent as long as they are not under precedence restrictions. Fig. 2 shows the precedence relation of all perceptual processes of system.

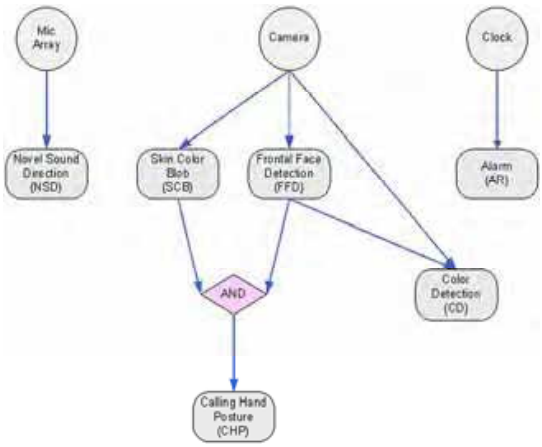


Figure 2. The precedence relation of all perceptual processes – All the relations without AND mean OR

4. In-Situ Selection of an optimal set of evidences

4.1 Evidence Structure for the Robot Missions

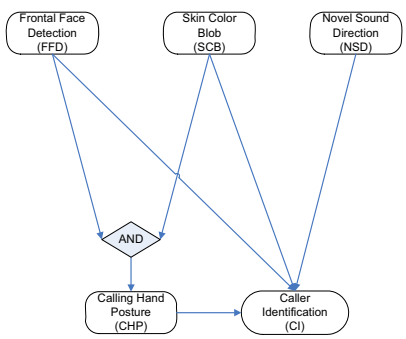


Figure 3. Evidence Structure For Caller Identification Mission

CRE aims at combining or fusing multiple evidences in time for dependable decision. In order to integrate multiple evidences, we needed another relation graph for certainty estimation. Although, above mentioned precedence relation graph shows the input-output relation of each perceptual process nicely, however it is not suitable for certainty estimation. Because to calculate certainty of the mission, the robot applies difference shape of calculate expression to each mission. Therefore, we define the "evidence structure" for certainty estimation.

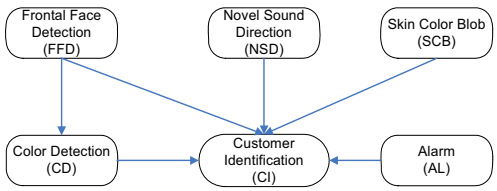


Figure 4. Evidence Structure For Customer Identification Mission

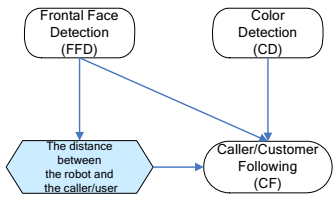


Figure 5. Evidence Structure For Caller/Customer Following Mission

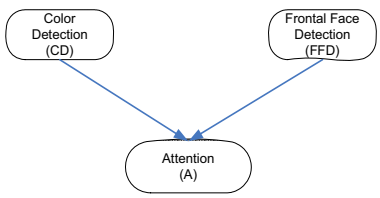


Figure 6. Evidence Structure For Attention Mission

Our analysis of current service robot's ability tell us that main objects of service robot are recognizing user and providing information to the user. Therefore, bring a current service robot platform into focus, we created four missions which are caller identification, customer identification, caller/customer following and attention. Consequently, evidence structure was made suitability for each individual mission. The robot selects adapted evidences for using this structure. The reason why we was not make one united structure but made individual structures for four missions is that if some missions are extended in the future, it is difficult to design architecture graph to extended missions. The evidence structure described by Fig. 3 through Fig. 6 is equivalent to a Bayesian net, except that we consider explicitly the conjunctions of evidences that becomes sufficient for proving the truth of another evidence and represent them with AND operations. This is to make it easier to define the joint conditional probabilities required for the computation of certainties based on the Bayesian probability theorem. The actual implementation of computing certainty update is based on the Bayesian net update procedure.

4.2 Certainty Estimation based on Bayesian Theorem

In this paper, we calculate the mission certainty based on Bayesian theorem.

$$
\begin{aligned}
&Mission\,Certainty\,(Mission) = \\
&P(Mission \mid Evidences) = \frac{1}{1 + \dfrac{P(Evidences \mid \overline{Mission})P(\overline{Mission})}{P(Evidences \mid Mission)P(Mission)}} = \frac{1}{1+\alpha} \\
&\therefore\ \alpha = \frac{P(Evidences \mid \overline{Mission})P(\overline{Mission})}{P(Evidences \mid Mission)P(Mission)}
\end{aligned} \tag{1}
$$

(1) shows that the formula of the mission certainty estimation. In here, α is calculated differently in each mission. Under assumption that each evidence is independent, from the evidence structures, we are able to calculate α. For example, if the caller identification mission is selected, α is calculated by formula (2).

$$
\alpha = \frac{p(FFD \mid \overline{CI})p(SCB \mid \overline{CI})p(NSD \mid \overline{CI})p(CHP \mid \overline{CI})p(\overline{CI})}{p(FFD \mid CI)p(SCB \mid CI)p(NSD \mid CI)p(CHP \mid CI)p(CI)} \tag{2}
$$

The rest α value of individual missions as follows:

- Customer identification

$$
\alpha = \frac{p(FFD \mid \overline{CI})p(SCB \mid \overline{CI})p(NSD \mid \overline{CI})p(CD \mid \overline{CI})p(AL \mid \overline{CI})p(\overline{CI})}{p(FFD \mid CI)p(SCB \mid CI)p(NSD \mid CI)p(CD \mid CI)p(AL \mid \overline{CI})p(CI)} \tag{3}
$$

- Caller/Customer Following

$$
\alpha = \frac{p(FFD \mid \overline{CF})p(CD \mid \overline{CF})p(\overline{CF})}{p(FFD \mid CF)P(CD \mid CF)p(CF)} \tag{4}
$$

- Attention

$$
\alpha = \frac{p(CD \mid \overline{A})p(FFD \mid \overline{A})p(\overline{A})}{p(CD \mid A)p(FFD \mid A)p(A)} \tag{5}
$$

4.3 Certainty Estimation with Consider Space-Time

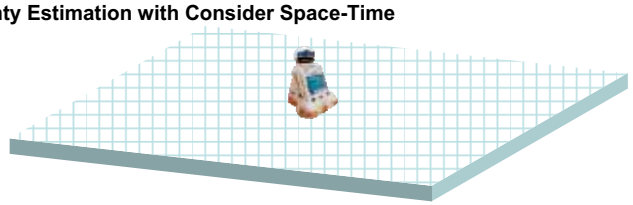


Figure 7. Interaction Space of the Robot for Certainty Representation

In this research, we implemented all perceptual processes with considering the two-dimensional interaction space of the robot. Fig 7 shows that interaction space of the robot. The interaction space is represented by 81(9*9) cells and each cell has around 50cm*50cm size. Since all processes have the information of two-dimensional space, each mission certainty is also represented by two-dimensional space and it is calculated for each cell. Therefore, the robot has spatial information. The spatial probability distribution is changed according to the robot behaviors and is estimated according to evidences continually.

Moreover, in order to provide time-related service, we implemented alarm process (AL). Using this process, the robot is able to provide service such as delivery information for the customer at specific time.

5. Evidence Collection Behaviors

The action should be selected to eliminate uncertainty of mission, not uncertainty of individual process. This means that the selected action has to improve the mission certainty best. Let B = {b1,b2, … , bn} is a set of proposed actions by a set of perceptual processes P= {p1, p2, … pn}, at time t. From the perceptual process, we can estimate the variation of certainty when the robot takes an action below.

b1 → ΔC(b1) = {Δc1(b1), Δc2(b1), … , Δck(b1), … ,Δcn(b1)}
b2 → ΔC(b2) = {Δc1(b2), Δc2(b2), … , Δck(b2), … ,Δ cn(b2)}
…
bk → ΔC(bk) = {Δc1(bk), Δ c2(bk), … ,Δck(bk), … , Δ cn(bk)}
…
bn → ΔC(bn) = {Δc1(bn),Δ c2(bn), … , Δck(bn), … , Δcn(bn)}

where Δck(bk) is expected certainty variation of pk when the action is selected. ΔC(bk) is a set of variation values. Now we can select an action using (6).

$$Selection\ of\ action = b_{\max}\{P(callerID \mid Evidences + \Delta C_{b1}),\ldots,P(callerID \mid Evidences + \Delta C_{bn})\} \quad (6)$$

The selected action will increase the mission certainty best.

6. Mission Management

Most of developed service robots recognize their mission by user's manual input. However, to provide advanced service, if there are several missions, the robot should be select mission naturally. Accordingly, we implemented the mission manager for advanced service of a robot. The mission manager should tell the mission with the minimum of perceptual processes.

The roles of mission manager are detailed below:

1. The manager should be monitoring enabled perceptual processes.
2. If any change of environment stimulus some perceptual process, the manager has to recognizes all the missions which are related to the process. The connection relation between missions and perceptual processes should be pre-defined.
3. Since enabled perceptual processes are very primitive, some missions will remain and be invoked among the subset of missions, or the others may be removed. To recognize which of them to be selected, additional perceptual processes should be enabled.
4. If there is one mission selected, the manager performs it, while the number of mission is bigger than one, they are took into queue based on the priority of missions. Note that, simultaneous and multiple mission will be considered later.
5. Performing a mission, the manager should check if the mission is on going, or success, or fail
6. With succeed/failure of the mission, the manager should change the state of robot naturally.

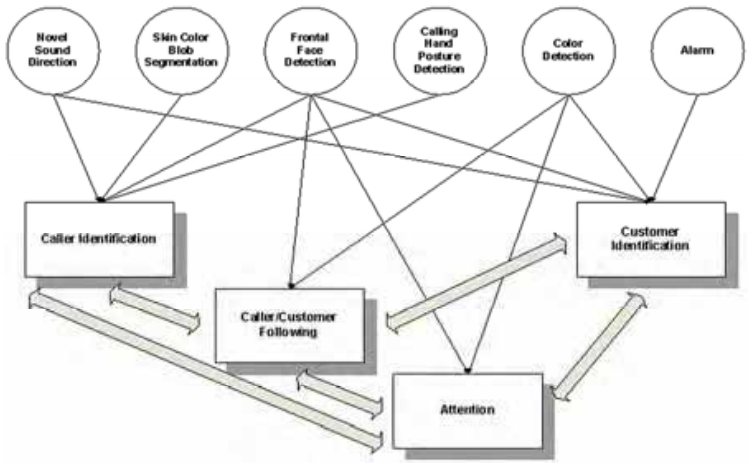


Figure 8. Mission Manager for Four Missions

Mission	Definition
Attention	Gazes into Caller/Customer
Caller Identification	Seeks for the caller and then identifies the caller
Customer Identification	Seeks for the customer and then identifies the customer
Caller/Customer Following	Follows the caller/customer

Table 2. List of missions and definition

7. Implementation

7.1 Hardware Specification

Figure 9. Robot Hardware

The approach outlined above has been implemented on the mobile robot iRobi. The specification of single-board-computer has Intel Pentium mobile processor 1.40GHz, 1GB RAM. And the Robot has three channel microphones for estimates the direction of sound source. Logitech Quickcam Pro 3000 camera as imaging sensor has approximately 60° horizontal-field-of-view (HFOV) and 320*240 square pixels.

7.2 Software Configuration

Overall architecture of the CRE system is presented in Fig. 10. As seen in the figure, the system is composed of server and client. In here, client means the robot and the robot and the server communicated by Common Robot Interface Framework (CRIF). It provides TCP/IP wireless connection so that CRE system could be adapted to another platform easily. Two multi threads in the server request image and sound continuously. A perceptual process is called when a thread get sensing information from robot. There procedures are operated asynchronously and concurrently.

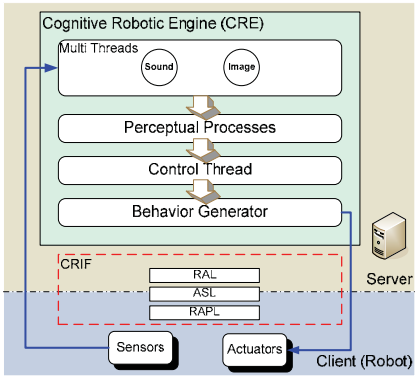


Figure 10. Overall Architecture of the System. (RAL: Robot API Layer, ASL: API Sync Layer, RAPL: Robot API Presentation Layer)

7.2.1 Sampling Time of Control based on Forgetting Curve

Among the several approaches for sampling time, we got the idea from psychology field (Brown, 1958, R. Peterson & J. Peterson, 1959, Atkison & Shiffrin 1968). Fig. 11 shows forgetting curve for human short-term memory. Based on that, the sampling time is determined as 600ms approximately.

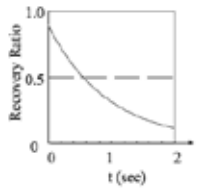


Figure 11. Forgetting curve of Brown Peterson paradigm

8. Experimentation

8.1 Experiment Condition

The experimental scenario is described in Fig. 12. Experimentation had proceeded in the around 6m*8m size tester bed without any obstacles and the caller is only one. Please see the figure with attention time and variance of the mission. Descriptions on abbreviation as below:

NSD: Novel Sound Detection, FFD: Frontal Face Detection, SCB: Skin Color Blob, CHP: Calling Hand Posture , CD: Color Detection, AL: Alarm.

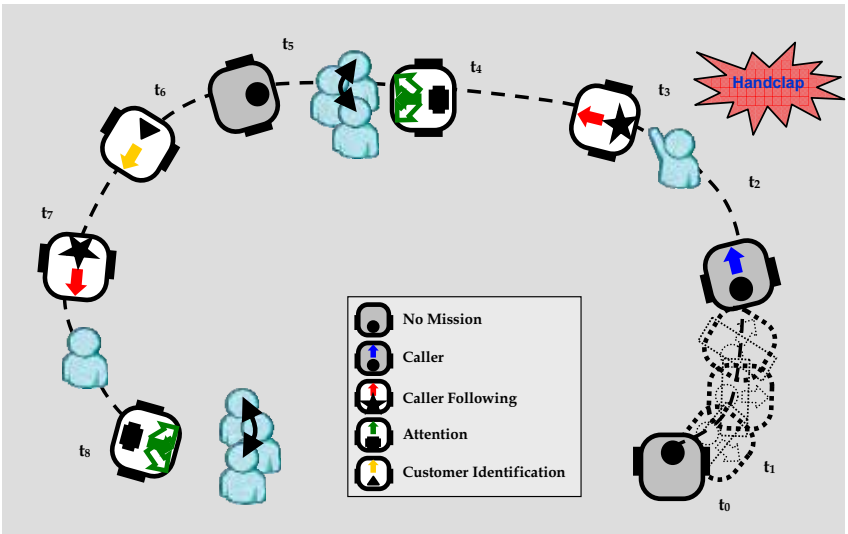


Figure 12. Experimentation of the multi-mission management and the certainty estimation of Cognitive Robotic Engine

8.2 Experiment Results

Initially, control part of CRE enables only NSD , FFD, AL processes.

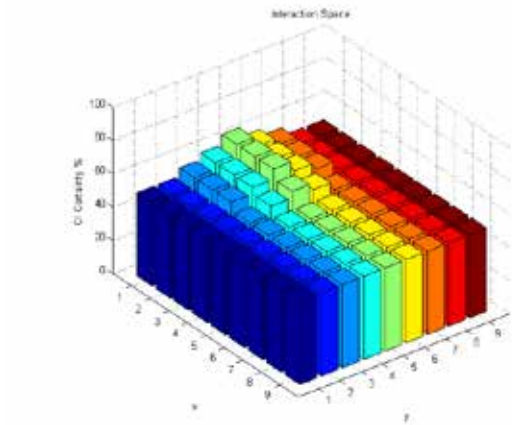


Figure 13. Certainty of the caller identification mission (t1)

First (t0), the caller called the robot behind the robot's back through the handclap. Then, the certainty of caller identification mission arisen as Fig. 13 by NSD process output, and the mission started (t1).
As the caller identification mission started, SCB and CHP processes activated to collect more evidences. Fig. 14 is certainty of the mission, just after turning to the caller, and the certainty increased when FFD and CHP processes detected caller's hand motion (Fig. 15).

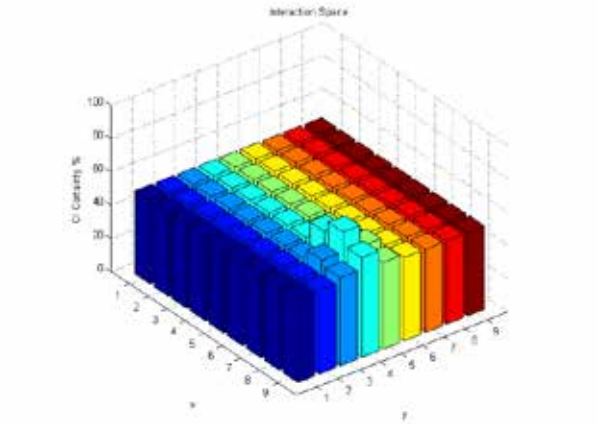


Figure 14. Certainty of the caller identification mission (t2, before calling hand posture detected)

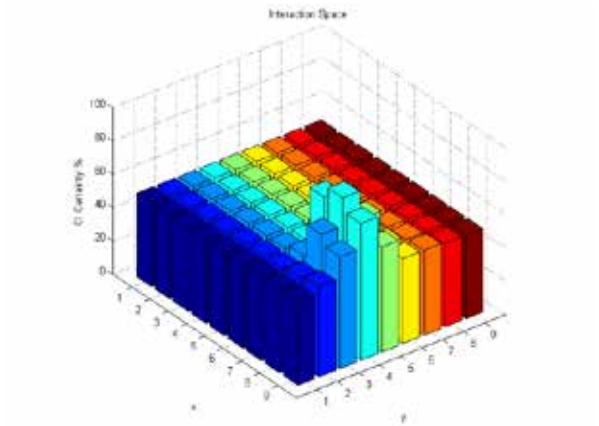


Figure 15. Certainty of the caller identification mission (t2, after calling hand posture detected))

At this moment (t2), the mission manager changed the mission to caller tracking. So, FD and CD processes activated, and started to move to the caller (t3). Fig. 16 shows the certainty of

caller tracking mission at t3. In Fig. 17, the certainty of frontal spaces of the robot is high enough to change the mission to attention (t4).

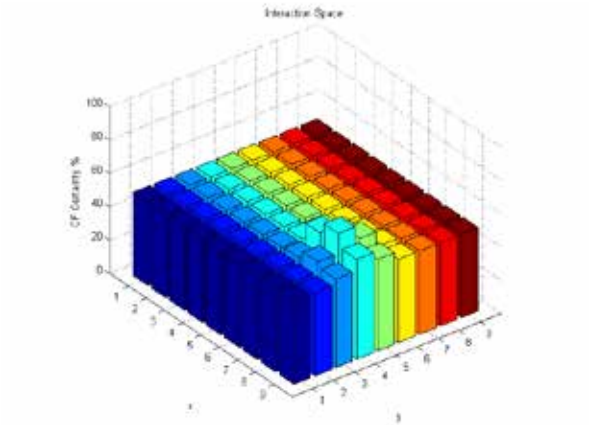


Figure 16. Certainty of the caller/customer tracking mission (t3)

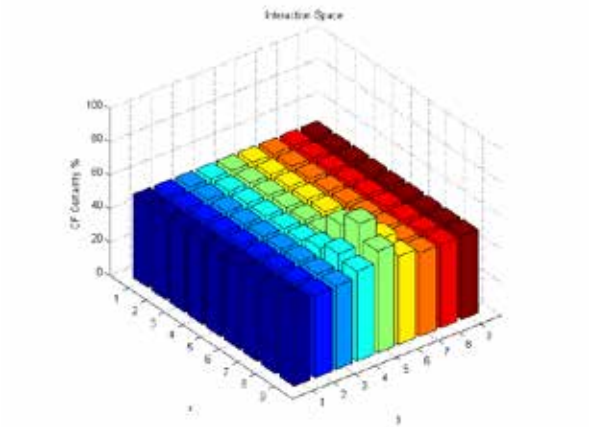


Figure 17. Certainty of the caller/customer tracking mission (t4)

Fig. 18 shows the certainty of attention mission. Generally, the service robot can convey information to the caller while doing attention mission. After a communication with the caller, mission manager of the robot dismissed attention mission like initial state. After for a while, the customer identification mission started by AL process, so the robot try to find customer who wears red shirt (reserved mission like timer). The certainty of customer identification mission is shown Fig.19 (t4). When the robot found the customer, the certainty changed like Fig. 20, then, attention mission started (t8).

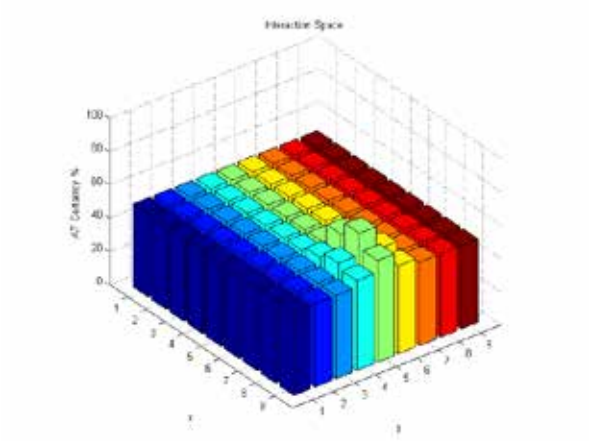


Figure 18. Certainty of the attention mission (t4)

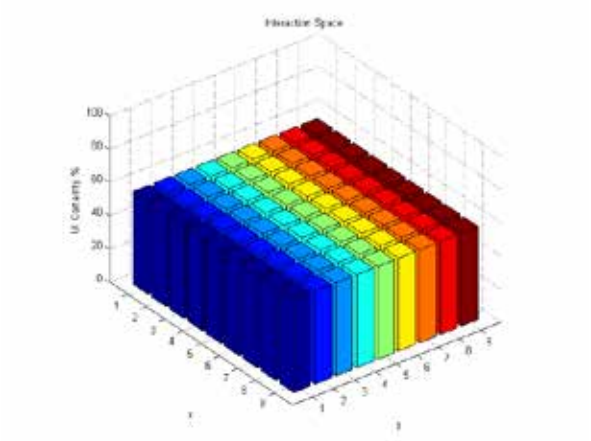


Figure 19. Certainty of the customer identification mission (t6)

We recorded the results several times of experimentation, the results shows that missions started, stopped and changed automatically based on variation of the certainty, and by defining the certainty of each mission in the interaction space, behavioral parameters can be easily obtained. Basic rules to choose behavior is that select one behavior among candidates suggested by perception processes to increase their certainties.

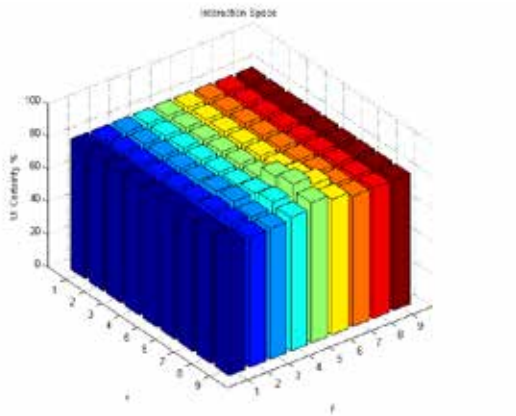


Figure 20. Certainty of the customer identification mission (t7)

9. Conclusion and Future work

In this paper, we described the robotic architecture for dependable perception and action for service robot in dynamic environment. This architecture is organized to accomplish perception mission in spite of the integration of imperfect perception processes, and updated for managing multi-missions. The next step, we are planning to research on automatic discrimination method of system dependability.

10. Acknowledgement

This work is supported by the Intelligent Robotics Program, one of the 21st Century Frontier R&D Programs funded by the Ministry of Commerce, Industry and Energy of Korea. This work is also supported in part by the Science and Technology Program of Gyeonggi Province as well as in part by the Sungkyunkwan University. And this work was also partly supported by Brain Korea 21 (BK21) project.

11. References

Ahn Hyunsang, Kim Dohyung, Lee Jaeyeon, Chi Suyoung, Kim Kyekyung, Kim Jinsul, Hahn Minsoo, Kim Hyunseok. (2006). A Robot Photographer with User Interactivity, *Proceeding of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 5637-5643, Beijing, China, October 2006.

Betkowska A, Shinoda K, Furui S. (2007). Robust speech recognition using factorial HMMs for home environments, *EURASIP Journal on advances in signal processing*, Vol. 2007.

Chen Bin and Kaneko Masahide. (2007). Behavior Selection of Mobile Robot Based on Integration of Multimodal Information, *Electrical engineering in Japan*, Vol. 158, No. 2, pp. 39-48.

Choi Jong-Suk, Kim Munsang and Kim Hyun-Don. (2006). Probabilistic Speaker Localization in Noisy Environments by Audio-Visual Integration, *Proceeding of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 4704-4709, Beijing, China, October 2006.

Fong T, Nourbakhah I, Dautenhahn K. (2003). A survey of socially interactive robots, *Robotics and Autonomous Systems*, Vol. 42, pp. 143-166.

J. Brown. (1958). Some Tests of the Decay Theory of Immediate Memory. *Quarterly Journal of Experimental Psychology*, Vol. 10, pp. 12-21.

J. Fritsh, M. Kleinehagenbrock, A. Haasch, S. Wrede, and G. Sagerer. (2005). A flexible infrastructure for the development of a robot companion with extensible HRI-capabilities, *Proceeding of the IEEE International Conference on Robotics and Automation*, pp. 3408- 3414, Barcelona, Spain, April 2005.

L. R. Peterson and M. J. Peterson. (1959). Short-term Retention of Individual Verbal Items. *Journal of Experimental Psychology*, Vol.58, No.3, pp. 193-198.

M. Danesh, F. Sheikholeslam, M. Keshmiri. (2006). Robust Robot Controller Design Using Joint Position and Velocity Dependant Uncertainty Bound. *Proceeding of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 3038-3043, Beijing, China, October 2006.

R. C. Arkin. (1998). *Behavior-Based Robotics*, MIT Press, Cambridge, MA.

R. C. Atkinson and R. M. Shiffrin. (1968). Human memory: A Proposed System and its Control Processes. *In K. W. Spence and J. T. Spence (Eds.), The Psychology of learning and motivation: Advances in research and theory*, Vol. 2, New York: Academic Press.

Rui Da Silva Neves, Eric Raufaste. (2001). Polymorphism of Human Judgment under Uncertainty. *Proceedings of the 6th European Conference on Symbolic and Quantitative Approaches to Reasoning with Uncertainty*, pp.647 - 658,Toulouse, France, September 2001.

Sakaue, F, Kobayashi, M, Migita, T, Satake, J. (2006). A Real-life Test of Face Recognition System for Dialogue Interface Robot in Ubiquitous Environments, *Proceeding of 18th International Conference on Pattern Recognition*, pp. 1155 - 1160, Hong Kong, China, August 2006.

S. Li, M. Kleinehagenbrock, J. Fritsch, B. Wrede, and G. Sagerer. (2004). BIRON, let me show you something: evaluating the interaction with a robot companion, *Proceeding of the IEEE International Conference on Systems, Man and Cybernetics*, vol.3, pp. 2827-2834, The Hague, Netherlands, October. 2004.

S. Li and B. Wrede. (2007). Why and how to model multi-modal interaction for a mobile robot companion. *Proceeding of the AAAI Spring Symposium on Interaction Challenges for Intelligent Assistants*, Stanford University, CA, USA, March 2007.

S. Liu, Q. Yu, W. Lin, S. X. Yang. (2006). Tracking Control of Mobile Robots Based on Improved RBF Neural Networks, *Proceeding of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 1879 - 1884, Beijing, China, October 2006.

Sukhan Lee, Hun-Sue Lee, Seung-Min Baek, Jongmoo Choi, Dong-Wook Shin, ByoungYoul Song, Young-Jo Cho. (2006a). Caller Identification Based on Cognitive Robotic Engine, *Proceeding of the IEEE International Workshop on Robot-Human Interactive Communication*, pp. 417 - 423, Hertfordshire, UK, September 2006.

Sukhan Lee, Hun-Sue Lee and Dong-Wook Shin. (2006b). Cognitive Robotic Engine for HRI, *Proceeding of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 2601 - 2607, Beijing, China, October 2006.

Wilson, T.D. (2000). Human information behaviour. *Journal of Informing Science*, Vol. 3 No.2, pp.49-56.

14

Contact Task by Force Feedback Teleoperation Under Communication Time Delay

Masahiro Nohmi[1] and Thomas Bock[2]
[1]*Kagawa University,* [2]*Technical University Munchen*
[1]*Japan,* [2]*Germany*

1. Introduction

Space robot systems are performing and expected to perform important missions, for example, large-scale structure on-orbit construction (as in the International Space Station or the Solar Power Satellite) and on-orbiting servicing tasks in Low Earth Orbit (LEO). It is difficult to develop an intelligent robot, which performs various tasks autonomously in complex environments. Current technology makes necessary to rely on human operator for providing overall task guidance and supervision, and for handling special situations. The benefits of teleoperation of a space robot have already been proved many times by Shuttle Remote Manipulator System, which is operated by astronauts inside the spacecraft to perform complex tasks, such as satellite handling and repairing (Ravindran, R. & Doetsch K. H., 1982).

Controlling space robots from ground is potentially much more effective than controlling them in space (Sheridan, T. B., 1993). There are many advantages: first, the total hourly cost of a ground operator is orders of magnitude lower than that of an astronaut in space; second, ground control stations can have greater computing resources available; third, ground teleoperation permits to reduce crew workload; and forth, ground control permits terrestrial scientists to perform remotely experiments, etc. Hence, ground control of space robots seems very attractive, but on the other side there are some important drawbacks or limitations that must be overcome: (i) communication time delay; (ii) a low communications bandwidth; (iii) lack of telepresence with difficult control in operation.

Under such a condition, special attention should be paid to contact forces and torques when performing contact task with a space-based manipulator teleoperated from ground. Therefore, the manipulator should be controlled autonomously with compliance feature. Note that some sort of compliance feature on the manipulator, either active or passive, is effective for contact task. Compliance is useful to cope with the error caused by an imperfect model. It can reduce execution time and overall forces applied upon the environment. The only problem is that it consists of an automatic remote feature and the operator can get confused if not fully aware of its behavior. Experimental researches for teleoperation have been well studied. The robot in the German space robot experiment, ROTEX, which was teleoperated both from within the space shuttle by an astronaut and from ground by an operator, was equipped with a force/torque sensor (Hirzinger, G. et al., 1993). In addition, ground-based experimental studies on teleoperation under time delay have been performed

as in (Hannaford, B., 1994), (Funda, J. et al., 1992), (Hirata, M. et al., 1994), also under time varying (Hirche, S. & Buss, M., 2004). The Engineering Test satellite VII, which was launched at the end of 1997 by the National Space Development Agency (Oda, M. et al., 1998), performs the most recent experiment for space teleoperation.
This paper describes experimental analysis of a new strategy for space teleoperation under communication time delay, which makes it possible for an operator to notice contact through force reflection of a hand controller. In experiment, contact force display and command input of a hand controller were focused. Organization of the paper is the following. Section 2 explains the concept, the algorithm, and virtual images of the proposed approach. Section 3 introduces our teleoperation system for experiment. Teleoperation experiment of vertical contact to target is described in section 4 and section 5, in order to examine effectiveness of force contact display and command input, respectively. Section 6 describes teleoperation experiment of tracking task.

2. Important Teleoperation by Force Reflection

2.1 Concept of the proposed approach

In teleoperation for a space-based manipulator from ground, a ground operator sends a command to the manipulator, which executes it. After duration of communication time delay, the operator receives telemetry data as a result of manipulator motion. Then, the current telemetry data is result of the command data sent before duration of the time delay, when sending the current command. In the proposed teleoperation approach, difference of the current command and the current telemetry is displayed to the operator by force reflection through a hand controller. The manipulator is moving or begins to move when an operator feels force reflection. On the other hand, when contact force is applied to the manipulator, it is added to force reflection of the time delay. In operation without contact, force reflection becomes to be zero when receiving telemetry data expressing that the manipulator finishes its motion. Under condition of contact, force reflection continues to be applied even if the manipulator stops its motion. Also, an operator feels change of force reflection when contact of the manipulator occurs, when a contact force applied to the manipulator is reduced, and when the manipulator is moving. Thus, the operator can know conditions of the manipulator. In order to apply the proposed approach, autonomous compliance control has to be used for the remote manipulator.

2.2 Force feedback algorithm

Figure 1 shows a data flow chart of the proposed approach. The following kinds of data are defined:

x_c : "command" operated by joystick;

x_r : "reference" point for compliance control ;

x_t : "telemetry" as a result of manipulator motion;

f : contact force applied to a remote manipulator;

f_a : contact force used for force feedback calculation;

$\boldsymbol{F}$: force reflection on a hand controller.

Two kinds of f_a can be obtained from force sensor $\boldsymbol{f}$ and compliance calculation K (x_t - x_r), respectively, they are examined in section IV. $\boldsymbol{M}$, $\boldsymbol{C}$, $\boldsymbol{K}$ denotes parameters for compliance control. k_t is control gains for calculation of force reflection of a hand controller. Teleoperation computer calculates command x_c from hand controller operation. x_c is sent to remote computer and compliance control is performed by reference x_t (= x_c). x_t, x_r, $\boldsymbol{f}$ are sent from remote computer to teleoperation computer as telemetry data, which is used for calculation of force reflection F of a hand controller.

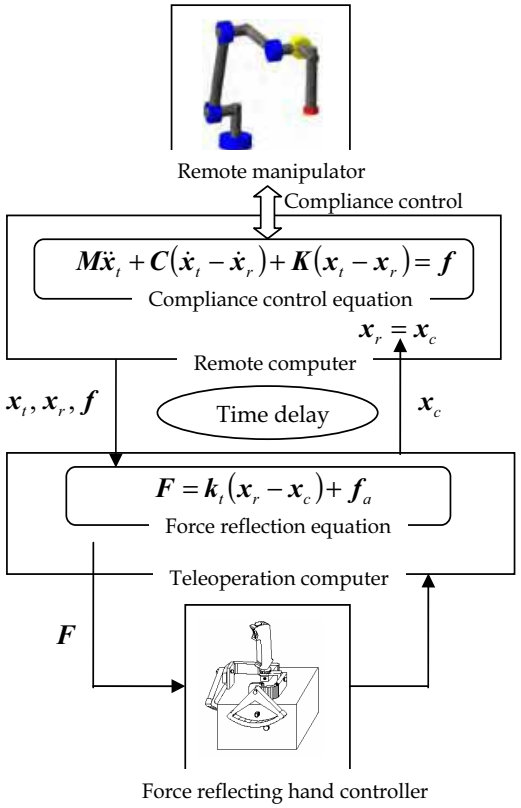


Figure 1. Data flow in force feedback teleoperation

2.2 Virtual feelings for operation

Figure 2 shows image of the proposed teleoperation approach without contact. A remote manipulator is operated as if the operator moves it through a virtual spring. The left figure shows command and telemetry manipulators. Sending command and receiving telemetry data configure them, respectively. The difference of command x_c and telemetry x_r of the

manipulator end tip positions are translated to extension of the virtual spring, which generates force reflection of communication time delay. As a result, the operator can recognize the time delay by extension of the virtual spring. The manipulator has executed the command when force reflection becomes to be zero, and then the operator feels no forces.
Figure 3 shows image when performing a contact task. The left side figure shows command x_c, telemetry x_t, and reference x_r manipulators. The reference manipulator denotes the command one sent before duration of the time delay. Both differences due to contact force and the time delay are translated to extension of the virtual spring, which generates force reflection. When the manipulator is moving, the operator feels that length of the virtual spring is changing, not constant. When the manipulator stops, and external force/torque is applied to the manipulator, the operator feels that the virtual spring is extended at constant length. Thus, the operator can know conditions of the manipulator.

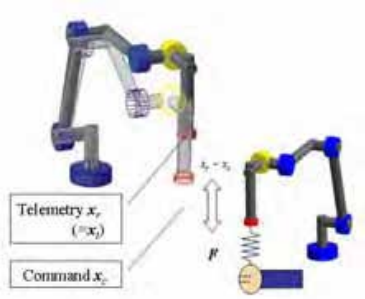


Figure 2. Virtual feeling of communication time delay without contact

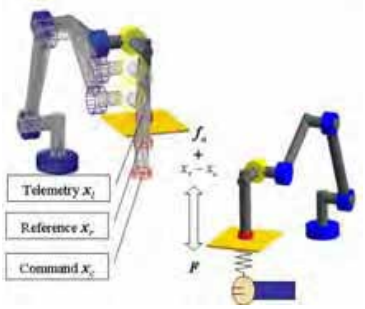


Figure 3. Virtual feeling of communication time delay with contact

3. Experimental System

Figure 4 shows the experimental teleoperation system. PA-10 (product of Mitsubishi Heavy Industry Ltd.), which is controlled on the MS-DOS, is used as a remote manipulator. Impulse Engine 2000 (product of Nissho Electronics Ltd.) is used as a hand controller. This is a joystick controlled on the Windows 2000 by an operation computer, and it has two

degrees of freedom, and also it can reflect forces to an operator. Figure 5 shows Impulse Engine 2000 in the left side and PA-10 in the right.
An operator inputs command through the joystick into the operation computer, and the command is buffered in the operation computer during duration of communication time delay, which is set for simulating space teleoperation from ground under the time delay. Then, the command is sent to the remote computer, and the manipulator executes it. As a result of manipulator motion, reference and telemetry data are sent from the remote computer to the operation computer. The joystick reflects force calculated on the operation computer based on information of telemetry data received from the remote computer, and the command input by the operator.

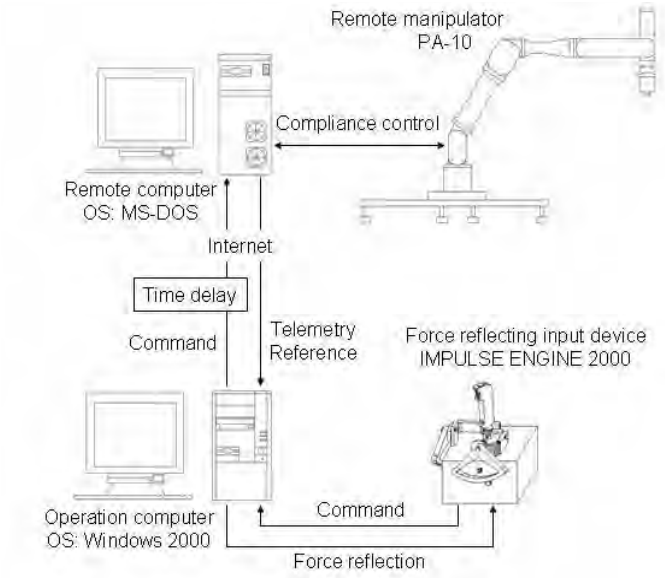


Figure 4. Experimental system for teleoperation

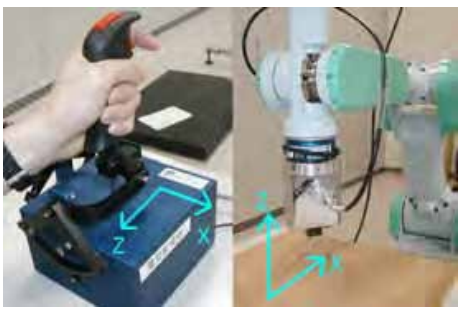


Figure 5. Experimental hardware

4. Force Telemetry

4.1 Experimental settings

The first experiment was performed to examine effectiveness of force reflection calculated as following cases:

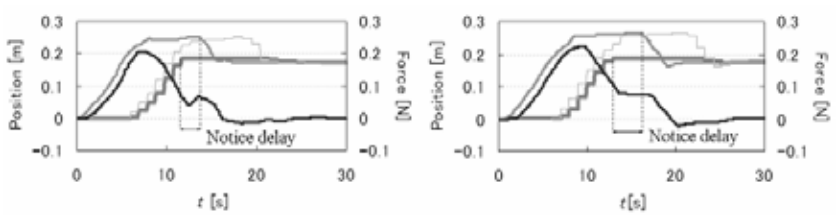
(A) $\boldsymbol{f_a} = \boldsymbol{f}$;

(B) $\boldsymbol{f}_a = k_c(\boldsymbol{x}_c - \boldsymbol{x}_t)$,

where k_c is control gain, and set as $k_c = k_t$. Force sensor value is used in case (A), and compliance calculation is employed in case (B). Experiment is vertical contact to target. Here, operation was performed based on "sequential manipulation," explained as follows. An operator terminates the first command which overshoots the target surface, before the manipulator begins to move. Then, the operator begins to send the second command to reduce contact force, after the manipulator stops its motion by the first command. Also it is terminated before the manipulator begins to move.

4.2 Experimental result

Figure 6 shows time history of an example data in the experiment. Here, command input by joystick was position. Reference $\boldsymbol{x}_r$ followed command $\boldsymbol{x}_c$ after the 5 seconds time delay. Telemetry $\boldsymbol{x}_t$ followed reference $\boldsymbol{x}_r$ with errors due to the compliance motion, and then it stopped at the contact point. The operator sends command to reduce contact force as soon as he notices contact. Then, "notice delay" shows that time delay for recognition of contact by the operator.



(A) Operation with force telemetry (B) Operation with position telemetry

Command Telemetry Reference Reflection

Figure 6. Time history of experimental result in cases (A) and (B)

Figure 7 shows averages and standard deviations of "notice delay" for three times trial by three operators. From the result in figures 8 and 9, it is noted that an operator notices contact more accurately in case of (A) f_a = telemetry force sensor. The reason is explained in figure 7 that change of the force reflection is sharper in case of (A) than that in case of (B) when the manipulator makes contact.

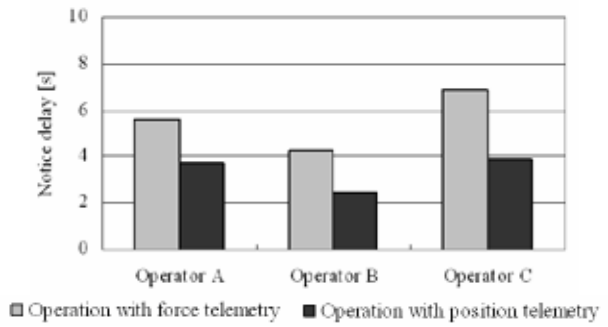


Figure 7. Average of notice delay in cases of (A) and (B)

5. Command Input

5.1 Experimental settings

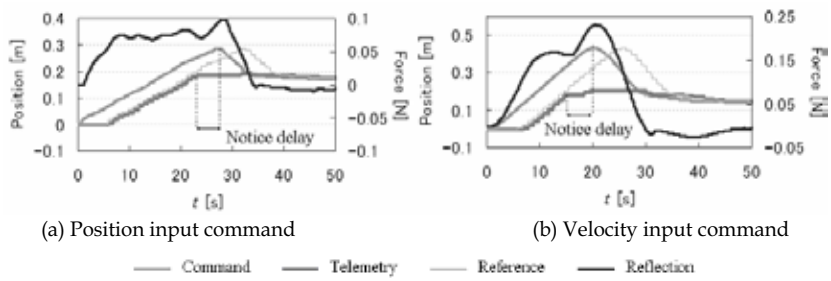
The second experiment was performed to examine difference of hand controller input:

(a) position command;
(b) velocity command.

Experiment is vertical contact to target. Here, operation was performed based on "continuous manipulation," explained as follows. The manipulator begins to move during sending the first command. An operator sends the second command to reduce contact force as soon as the contact is noticed.

5.2 Experimental result

Figure 8 shows time history of an example data in the experiment. Here, force reflection was based on position telemetry (B). In this experiment, the operator sends opposite command to reduce contact force during sending progressing command, as soon as he notices contact. Then, "notice delay" also shows that time delay for recognition of contact by the operator.
Figure 9 shows averages and standard deviations of "notice delay" for three times trial by each operator. From the result in figures 8 and 9, it is noted that an operator notices contact more accurately in case of (b) velocity input. The reason is explained in figure 8 that an operator can keep constant force reflection by velocity input before contact.



(a) Position input command (b) Velocity input command

Figure 8. Time history of experimental results in cases of (a) and (b)

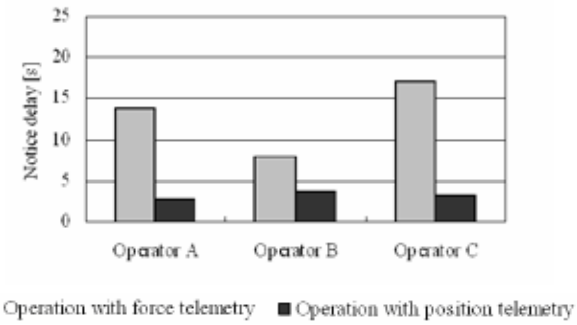


Figure 9. Average of notice delay in cases of (a) and (b)

6. Tracking Task

6.1 Experimental settings

The final experiment was performed to examine tracking task by the proposed teleoperation approach. Figure 10 shows image of the experimental task. Operation is tried as follows. First, the manipulator begins to move downward along the z axis (in figure 5). After the manipulator makes contact with slope, it is operated to track the slope. The tracking task was performed under the following condition:

(i) without force feedback by position command;
(ii) force telemetry feedback by position command;
(iii) position telemetry feedback by position command;
(iv) force telemetry by velocity command;
(v) position telemetry by velocity command.

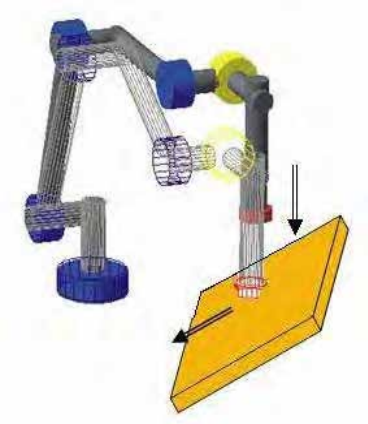


Figure 10. Tracking task

6.2 Experimental result

Figure 11 shows example data of tracking task in experiments (i) – (v). It is noted that smooth operation is possible by force feedback. Command input was adjusted many times, and then command line became discontinuous. Because force feedback based on force sensor is noisy and sensitivity, operation is smoother in cases (ii) and (iv) than that in cases (iii) and (v), respectively. On the other hand, contact point was recognized more accurately in case (ii) and (iv) when force sensor value is used for force feedback. It is also noted that delicate operation is possible by position command in cases (ii) and (iii), compared to operation by velocity command in cases (iv) and (v).

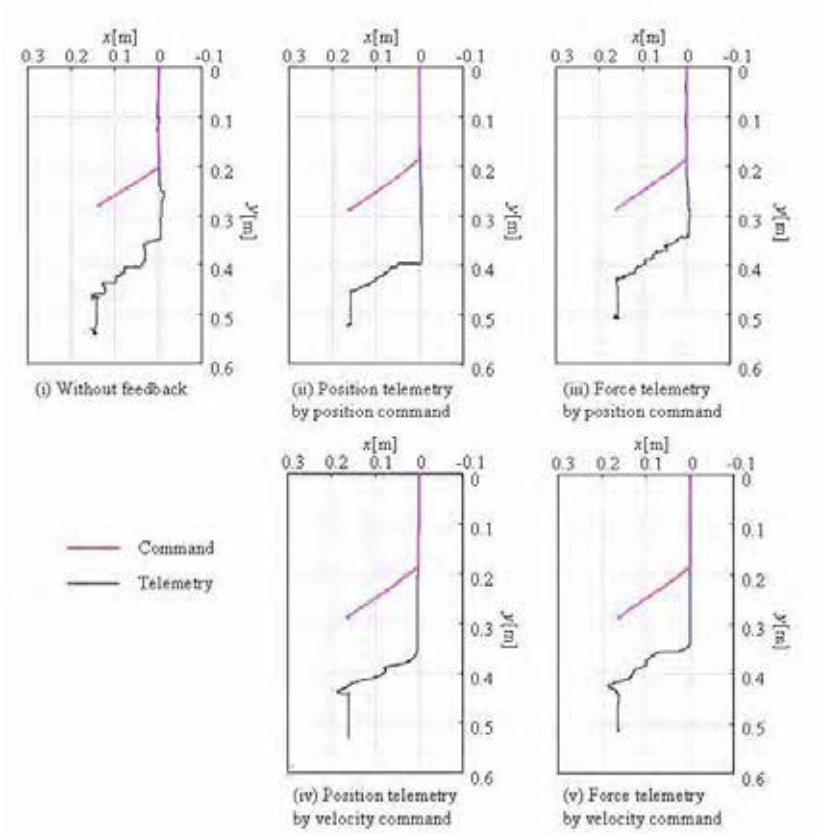


Figure 11. Experimental result of tracking task

7. Conclusion

This paper discusses our proposed strategy for space teleoperation under communication time delay, which makes it possible to know conditions of a remote manipulator through force reflection. In the proposed approach, the communication time delay and a contact force are displayed to the operator by the force reflection, and the remote manipulator can be operated as if the operator moves it through a virtual spring.

By experimenting example tasks, characteristics and effectiveness of the proposed approach have been clarified. From experiment of vertical contact with target, it is noted that an operator notices contact more accurately when;

(i) force reflection is calculated based on telemetry of force sensor;

(ii) command input by velocity.

Also, it is noted from tracking task that smooth operation is possible by force feedback. Also, operation based on position telemetry feedback is smoother than that based on force telemetry feedback, and delicate operation is possible by position command

8. References

Ravindran, R. & Doetsch K. H. (1982). Design aspects of the shuttle remote manipulator control, *AIAA 1982 Guidance and Control Conference*, pp. 456-465, AIAA-82-1581, San Diego, CA, August 1992

Sheridan, T. B. (1993). Space Teleoperation Through Time Delay: Review and Prognosis, *IEEE Transaction on Robotics and Automation*, Vol. 9, No. 5, October 1993, pp. 592-606, ISSN 1042-296X

Hirzinger, G. et al. (1993). Sensor-based space robotics --- ROTEX and its telerobotics features, *IEEE Transaction on Robotics and Automation*, Vol. 9, No. 5, 1993, pp. 649-663, ISSN 1042-296X

Hannaford, B. (1994). Ground Experiments Toward Space Teleoperation with Time Delay, *Teleoperation and Robotics in Space*, Ed. Skaar, S. B. and Ruoff, C. F., pp. 87-106, AIAA, ISBN 1563470950, Washington, DC

Funda, J. et al. (1992). Teleprogramming: Toward Delay Invariant Remote Manipulation, *Presence: Teleoperators and Virtual Environments*, Vol. 1, No. 1, MIT Press, Winter 1992, pp. 29-44, ISSN 1054-7460

Hirata, M. et al. (1994). Design of Teleproperation System with a Force-Reflecting Real-time Simulator, *3rd International Symposium on Artificial Intelligence, Robotics and Automation in Space*, pp. 125-135, Pasadena, CA, Octorber 1994

Hirche, S. & Buss, M. (2004). Telepresence Control in Packet Switched Communication Networks, *2004 IEEE International Conference on Control Applications*, pp. 236-241, Taipei, Taiwan, September 2004

Oda, M. et al. (1998). ETS-VII (Engineering Test Satellite #7) - a Rendezvous Docking and Space Robot Technology Experiment Satellite, *46th International Astronautical Congress*, IAF-95-U.2.01, Oslo, Norway, October 1998

15

What People Assume about Robots: Cross-Cultural Analysis between Japan, Korea, and the USA

Tatsuya Nomura[1,3], Tomohiro Suzuki[2], Takayuki Kanda[3], Jeonghye Han[4], Namin Shin[5], Jennifer Burke[6] and Kensuke Kato[7]

[1] Ryukoku University, [2] JSPS Research Fellow and Toyo University,
[3] ATR Intelligent Robotics and Communication Laboratories,
[4] Cheongju National University of Education, [5] Dongguk University,
[6] University of South Florida, [7] Kyushu University of Health and Welfare
[1, 2, 3,7]Japan, [4, 5]Korea, [6] USA

1. Introduction

It is known that the concept of "robots" itself is very old. However, it is only recently that they have appeared as commercialized products in daily life, even in Japan that is regarded as one of the most advanced nations in the development of robotics industries. Thus, it is predicted that the old imaginary concept and embodied objects in the daily-life context mutually interact, and as a result, novel psychological reactions toward robots are caused. Moreover, there may be differences in the above psychological reactions between nations, due to the degree of presence of robotics in the society, religious beliefs, images of robots transmitted through media, and so on. Thus, it is important to investigate in different cultures what people assume when they encounter the word "robot," from not only a psychological perspective but also an engineering one that focuses on such aspects as design and marketing of robotics for daily-life applications.

On cultural studies about computers, Mawhinney et al., (1993) reported some differences about computer utilization between the USA and South Africa. Gould et al., (2000) performed comparative analysis on WEB site design between Malaysian and US companies based on the cultural dimensions proposed by Hofstede (1991). On psychological impact of technology, Weil & Rosen (1995) showed based on social research for 3,392 university students from 23 countries, that there are some cultural differences on technophobia, in particular, anxiety and attitudes toward computers. Compared with computers, which have a rather fixed set of images and assumptions, images and assumptions of robots may widely vary from humanoids to vacuum cleaner and pet-type ones. Thus, cultural differences of assumptions about robots, that is, what people assume when they hear the word "robots," should be sufficiently investigated before discussing any differences on emotions and attitudes toward robots.

On psychological reactions toward robots, Shibata et al., (2002; 2003; 2004) reported international research results on people's subjective evaluations of a seal-type robot they

developed, called "Palo," in several countries including Japan, the U.K, Sweden, Italy, and Korea. Although their results revealed that nationality affected the evaluation factors, they were limited to a specific type of robots. Bartneck et al., (2007) reported some cultural differences on negative attitudes toward robots between several countries including the USA, Japan, the UK, and the Netherlands. However, this study did not take into account cultural differences of assumptions about robots. As mentioned above, cultural differences of assumptions about robots should be investigated before discussing those on attitudes toward robots, in the current situation where images of robots are not so fixed as those of computers. Nomura et al., (2006a; 2006b) reported some relationships between assumptions about, anxiety toward, and negative attitudes toward robots. However, these studies were limited to one culture, using Japanese data samples. Moreover, the questionnaire items used in the studies were not designed for cross-cultural studies.
This chapter reports about cross-cultural research aiming at a more detailed investigation of assumptions about robots based on comparisons between Japan, Korea, and the USA.

2. Method

2.1 Subjects

Data collection for the cross-cultural study was conducted from May to July, 2006. The participants were university students in Japan, Korea, and the USA. Table 1 shows the sample size and mean age of the participants.
In each country, sampling was performed in not only departments on natural science and engineering but also those on social sciences.

Country	#. Univ.	Male	Female	Total	Mean Age
Japan	1	200	111	313	18.68
Korea	3	159	158	317	23.54
USA	1	96	69	166	23.93

Table 1. Sample Size and Mean Age of Participants

2.2 Instrumentation

A questionnaire for measuring assumptions about robots was prepared based on discussion between researchers of engineering and psychology in Japan, Korea, and the USA, as follows. First, types of robots to be assumed were discussed. Considering the existing research result on assumptions about robots (Nomura et al., 2005), the current presence of robots in the nations, and length of the questionnaire, seven types of robots were selected. Table 2 shows these types of robots.
Then, questionnaire items measuring degrees of characteristics which each type of robot is assumed to have, and answer types were discussed. As a result, the items about autonomy, emotionality, roles to be played in the society, and images of each type of robot were prepared. On the items of autonomy and emotionality, degrees of the assumptions were measured by three levels of answers. Table 3 shows these items and their choices. On the items of roles and images, ten and seven subitems were prepared respectively, and each subitem had seven-graded scale answer to measure degrees of the assumptions. Table 4 shows these items.

1. Humanoid robots the size of toys or smaller
2. Humanoid robots between the sizes of human children and adults
3. Humanoid robots much taller than a person
4. Robots with appearances and sizes the same as animals familiar to humans, such as dogs, cats, rabbits, and mice
5. Machine-like robots for factories or workplaces
6. Non-humanoid robots bigger than a person, such as animal-, car-, or ship-shaped robots
7. Non-humanoid robots smaller than a person, such as animal-, car-, or ship-shaped robots

Table 2. Robot Types Dealt with in the Questionnaire (in order on the questionnaire)

Degree of autonomy to be assumed for the robot
1. Complete self decision-making and behavior
2. Self decision-making and behavior for easy tasks, and partially controlled by humans for difficult tasks
3. Completely controlled by humans, such as via remote controllers
Degree of emotional capacity that the robot is assumed to have
1. Emotional capacity equal to that of humans
2. Some capacity for emotion, but not as much as humans
3. No capacity for emotion at all

Table 3. Items Measuring Assumptions about Autonomy and Emotionality of Robots and their Choices (Common in all the Robot Types)

The questionnaire, the Robot Assumptions Questionnaire (RAQ), was originally made in Japanese, including the instructions. Then, the English version was made through formal back-translation.

Roles that the robot is assumed to play in the society (seven-graded scales from 1: Not likely at all to 7: Almost certainly)
1. Housework
2. Communication partners in the home
3. Physical tasks in the office
4. Intelligent tasks in the office, including communication
5. Tasks related to life-and-death situations in hospitals
6. Tasks related to nursing, social works, and education
7. Monotonous assembly line work in factories
8. Toys in the home or at amusement parks
9. Tasks hard for humans to do , or tasks in places hard for humans to go (such as burdensome tasks in space, the deep sea, or the battlefield)
10. Acts of hostility in the battlefield, such as causing blood shed
Images to be assumed for the robot (seven-graded scales from 1: Not likely at all to 7: Almost certainly)
1. Raise difficult ethical issues
2. Beneficial to society
3. A cause of anxiety in society, for example, as a cause of unemployment
4. Very interesting scientific and technological products
5. A technology requiring careful management
6. Friends of human beings
7. A blasphemous of nature

Table 4. Items Measuring Assumptions about Roles and Images of Robots (Common in all the Robot Types)

2.3 Procedures

Each colleague was sent the English version of the RAQ including the instructions to be read to the students. In Japan, the Japanese version of the questionnaire was administered to undergraduate classes in the departments of engineering and social sciences. In the USA, the English version was administered to both graduate and undergraduate classes in the schools of engineering and psychology. In Korea, back-translation from the English to the Korean was performed, and then the Korean version of the questionnaire was administered to classes in the departments of natural sciences, engineering, and social sciences. Participation was voluntary.

3. Results

3.1 Autonomy and Emotionality

Table 5 shows the numbers of respondents for assumed degrees and levels of autonomy and emotional capacity of each robot type. Then, to compare between the countries on the assumed degrees of autonomy and levels of emotional capacity of each robot type, correspondence analysis was performed for six cross tables shown in table 5.

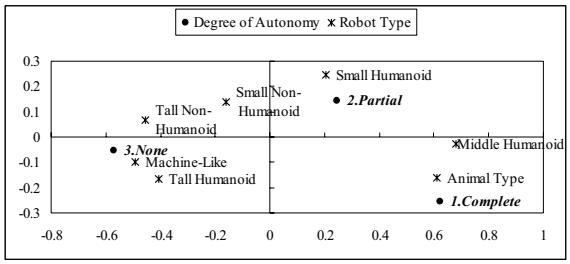
Autonomy

		RT1	RT2	RT3	RT4	RT5	RT6	RT7
Japan	1. Complete	44	100	38	106	26	14	27
	2. Partial	185	178	83	154	85	108	137
	3. None	77	20	176	35	184	170	126
Korea	1. Complete	17	23	20	39	14	6	6
	2. Partial	211	227	125	177	90	105	131
	3. None	78	55	157	83	192	181	154
USA	1. Complete	17	21	16	29	22	15	13
	2. Partial	101	107	73	86	69	61	69
	3. None	41	24	60	35	57	69	61

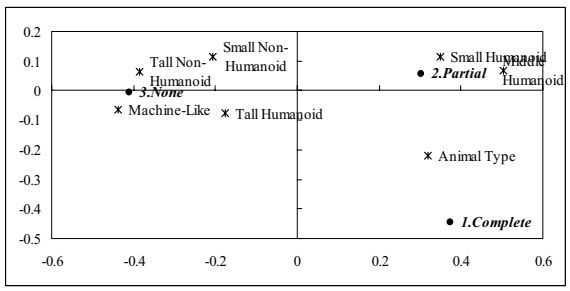
Emotionality

		RT1	RT2	RT3	RT4	RT5	RT6	RT7
Japan	1. Equal to Human	20	55	19	12	6	7	16
	2. Some	180	188	88	218	36	54	93
	3. None	102	52	189	62	252	228	180
Korea	1. Equal to Human	14	13	9	9	6	6	6
	2. Some	189	202	123	226	68	75	110
	3. None	101	90	166	63	221	210	175
USA	1. Equal to Human	6	16	7	7	5	5	5
	2. Some	41	62	47	85	11	19	40
	3. None	112	74	98	62	138	127	103

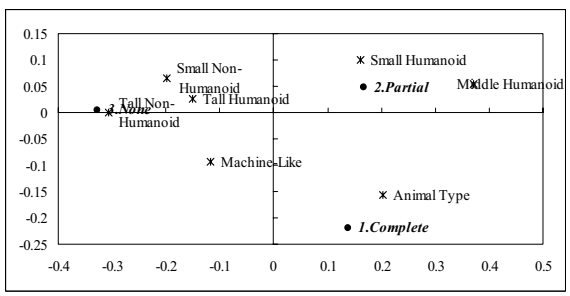
Table 5. The Numbers of Respondents for Assumed Degrees and Levels of Autonomy and Emotionality of Each Robot Type (RT1: Humanoid robots the size of toys or smaller, RT2: Humanoid robots between the sizes of human children and adults, RT3: Humanoid robots much taller than a person, RT4: Robots with appearances and sizes the same as animals familiar to humans, RT5: Machine-like robots for factories or workplaces, RT6: Non-humanoid robots bigger than a person, RT7: Non-humanoid robots smaller than a person)



Japan



Korea



USA

Figure 1. Results of Correspondence Analysis for Autonomy Item

Correspondence analysis allows us to visualize relationships between categories appearing in a cross table, on a 2-dimensional space. In this visualization, the categories familiar with each other are put at physically near positions. Our analysis with this method aims at clarifying what degree and level of autonomy and emotional capacity each robot is assumed to have in a specific country. On the other hand, we should note that the dimensional axes extracted from the data in a cross table are specific for the table data and are used to visualize the relative distances between categories, that is, they do not represent any absolute amount. Moreover, we should note that the axes are extracted to show the relative distances between categories arithmetically, and in general realistic meanings are hard to be assigned to these axes.

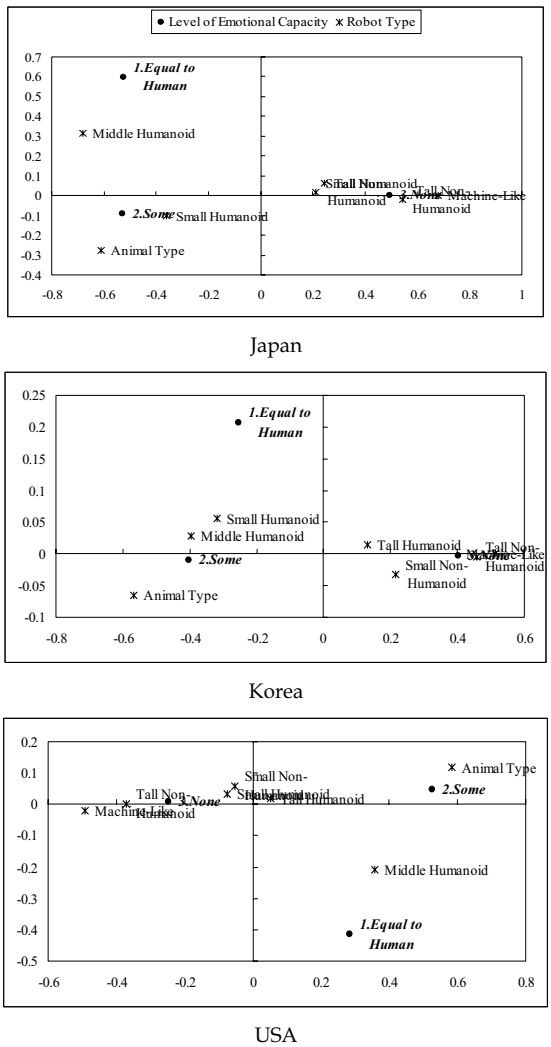


Figure 2. Results of Correspondence Analysis for Emotional Capacity Item

Fig. 1 shows the results of correspondence analysis for the cross tables on autonomy in the three countries. A common trend in all the countries was that the robot types except for "humanoid robots the size of toys or smaller," "humanoid robots between the sizes of human children and adults," and "robots with appearances and sizes the same as animals

familiar to humans" were positioned near "completely controlled by humans, such as via remote controllers." Moreover, there was another common trend in all the countries that "humanoid robots the size of toys or smaller" was positioned near "self decision-making and behavior for easy tasks, and partially controlled by humans for difficult tasks," and "robots with appearances and sizes the same as animals familiar to humans" was positioned near "complete self decision-making and behavior."

On the other hand, "humanoid robots between the sizes of human children and adults" was positioned between "self decision-making and behavior for easy tasks, and partially controlled by humans for difficult tasks" and "complete self decision-making and behavior" in the Japanese samples, although it was positioned near "self decision-making and behavior for easy tasks, and partially controlled by humans for difficult tasks" in the Korean and USA samples.

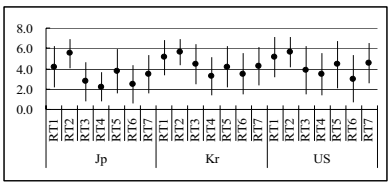
Fig. 2 shows the results of correspondence analysis for the cross tables on emotional capacity in the three countries. A common trend in Japan and Korea was that the robot types except for "humanoid robots the size of toys or smaller," "humanoid robots between the sizes of human children and adults," and "robots with appearances and sizes the same as animals familiar to humans" were positioned near "no capacity for emotion at all." Moreover, there was another common trend in these countries that "robots with appearances and sizes the same as animals familiar to humans" was positioned near "some capacity for emotion, but not as much as humans."

On the other hand, "humanoid robots between the sizes of human children and adults" was positioned between "emotional capacity equal to that of humans" and "some capacity for emotion, but not as much as humans" in the Japanese and USA samples, although it was positioned near "some capacity for emotion, but not as much as humans" in the Korean samples. Moreover, "humanoid robots the size of toys or smaller" was positioned near "some capacity for emotion, but not as much as humans" in the Japanese and Korean samples, although it was positioned near "no capacity for emotion at all" in the USA samples."

3.2 Roles and Images

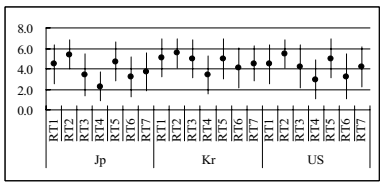
Next, to compare between the countries on the assumed degrees of roles played by and images of robots, two-way mixed ANOVAs with countries X robot type were performed for the scores of ten items of roles and seven items of images. The results revealed that there were statistically significant effects of countries in seven items of roles and five items of images, statistically significant effects of robot types in all the items of roles and images, and statistically significant interaction effects in almost all items of roles and images.

Fig. 3 shows the means and standard deviations of the role item scores related to the findings, and results of mixed ANOVAs with country and robot types, and posthoc analysis on country. As shown in the first, second, and third figures of Fig. 3, the Korean and USA students more strongly assumed housework and tasks in the office than the Japanese students. On the other hand, the posthoc analysis on each robot type revealed this difference did not appear in human-size humanoids. As shown in the fourth and fifth figures of Fig. 3, the Korean students more strongly assumed tasks related to life-and-death situations in hospitals than the Japanese and USA students. Moreover, the USA students did not assume tasks related to nursing, social works, and educations as much as the Korean and Japanese students. The posthoc analysis on each robot type revealed that this difference appeared in small-size humanoids and pet-type robots.



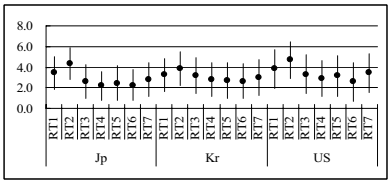
Role 1. Housework

Country:	$F(2, 585) = 41.858$***
Robot Type:	$F(6, 3510) = 186.615$***
Interaction:	$F(12, 3510) = 6.625$***
Posthoc:	USA, Kr > Jp



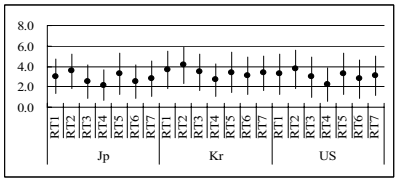
Role 3. Physical tasks in the office

Country:	$F(2, 583) = 22.553$***
Robot Type:	$F(6, 3498) = 177.456$***
Interaction:	$F(12, 3498) = 6.244$***
Posthoc:	Kr > USA > Jp



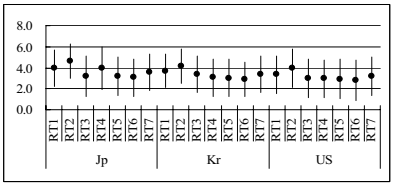
Role 4. Intelligent tasks in the office

Country:	$F(2, 580) = 11.475$***
Robot Type:	$F(6, 3480) = 116.845$***
Interaction:	$F(12, 3480) = 5.559$***
Posthoc:	USA, Kr > Jp



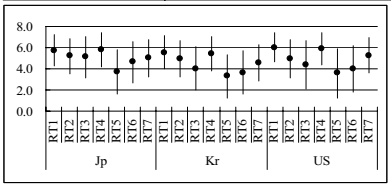
Role 5. Tasks related to life-and-death situations in hospitals

Country:	$F(2, 581) = 16.854$***
Robot Type:	$F(6, 3504) = 56.633$***
Interaction:	$F(12, 3504) = 2.003$*
Posthoc:	Kr > USA, Jp



Role 6. Tasks related to nursing, social works, and education

Country:	$F(2, 584) = 8.176$***
Robot Type:	$F(6, 3504) = 55{,}304$***
Interaction:	$F(12, 3504) = 3.472$***
Posthoc:	Jp, Kr > USA



Role 8. Toys in the home or amusement parks

Country:	$F(2, 582) = 12.552$***
Robot Type:	$F(6, 3492) = 152.0353$***
Interaction:	$F(12, 3492) = 5.061$***
Posthoc:	Jp, USA > Kr

(*$p < .05$, ** $p < .01$, *** $p < .001$)

Figure 3. Means and Standard Deviations of the 1st, 3rd, 4th, 5th, 6th, and 8th Role Item Scores, and Results of Mixed ANOVAs, and Posthoc Analysis on Country (RT1: Humanoid robots the size of toys or smaller, RT2: Humanoid robots between the sizes of human children and adults, RT3: Humanoid robots much taller than a person, RT4: Robots with appearances and sizes the same as animals familiar to humans, RT5: Machine-like robots for factories or workplaces, RT6: Non-humanoid robots bigger than a person, RT7: Non-humanoid robots smaller than a person)

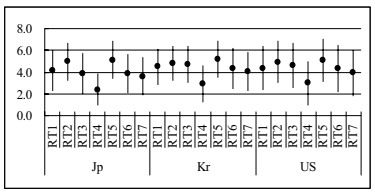


Image 3. A cause of anxiety in society

Country:	$F(2, 588) = 4.798$**
Robot Type:	$F(6, 3528) = 158.986$***
Interaction:	$F(12, 3528) = 5.697$***
Posthoc:	Kr, USA > Jp

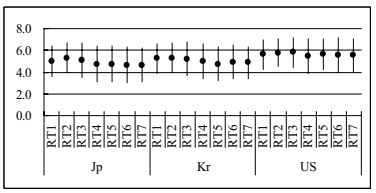


Image 4. Very interesting scientific and technological products

Country:	$F(2, 582) = 22.056$***
Robot Type:	$F(6, 3492) = 12.919$***
Interaction:	$F(12, 3492) = 2.623$**
Posthoc:	USA > Kr, Jp

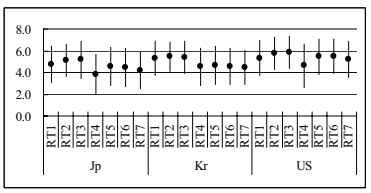


Image 5. A technology requiring careful management

Country:	$F(2, 581) = 21.804$***
Robot Type:	$F(6, 3486) = 60.640$***
Interaction:	$F(12, 3486) = 3.633$**
Posthoc:	USA > Kr > Jp

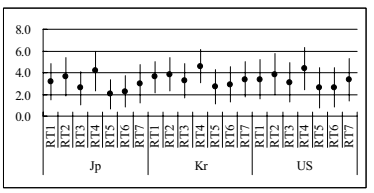


Image 6. Friends of human beings

Country:	$F(2, 587) = 7.999$***
Robot Type:	$F(6, 3522) = 159.658$***
Interaction:	$F(12; 3522) = 1.589$
Posthoc:	Kr, USA > Jp

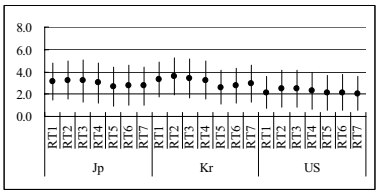


Image 7. A blasphemous of nature

Country:	$F(2, 587) = 19.054$***
Robot Type:	$F(6, 3522) = 26.291$***
Interaction:	$F(12, 3522) = 1.999$*
Posthoc:	Jp, Kr > USA

(*$p < .05$, ** $p < .01$, *** $p < .001$)

Figure 4. Means and Standard Deviations of the 3rd, 4th, 5th, 6th, and 7th Image Item Scores, and Results of Mixed ANOVAs, and Posthoc Analysis on Country

As shown in the sixth figure of Fig. 3, the Japanese and USA students more strongly assumed toys in the home or at amusement parks than the Korean students. The posthoc analysis on each robot type revealed that this difference also did not appear in human-size

humanoids. On the other hand, there was no difference between the countries for tasks hard for humans to do and tasks in space, the deep sea, and battle field (Role 9: country $F(2, 575) = 2.779$ (*n.s.*), robot type $F(6, 3450) = 169.792$ ($p < .001$), interaction $F(12, 3450) = 1.520$ (*n.s.*), Role 10: country $F(2, 582) = .436$ (*n.s.*), robot type $F(6, 3492) = 121.688$ ($p < .001$), interaction $F(12, 3492) = 2.199$ ($p < .01$)).

Fig. 4 shows the means and standard deviations of the image item scores related to the findings, and results of mixed ANOVAs with country and robot types, and posthoc analysis on country. As shown in the first and third figures of Fig. 4, the Korean students had more negative images of robots such as cause of anxiety in society, than the Japanese students. On the other hand, as shown in the fourth figure of Fig. 4, they also had more positive image such as friends of humans than the Japanese students.

As shown in the second, fourth, and fifth figures of Fig. 4, the USA students had more positive images such as friends of humans and interesting technology, and less negative images such as a blasphemous of nature, than the Japanese students. As shown in the first and third figures of Fig. 4, however, the USA students also more strongly assumed that robotics technology may cause anxiety in society and requires careful management, than the Japanese students.

4. Discussion

4.1 Findings

The results of the cross-cultural research imply several differences on robot assumptions between Japan, Korea, and the USA.

First, the results of section 3.1 show that the students in the three countries commonly did not assume autonomy and emotional capacity of the robots except for small humanoids, human-size humanoids, and pet-type robots. Moreover, they show that the Japanese students assumed higher autonomy of human-size humanoids than the Korean and USA students, and the Japanese and USA students assumed higher emotional capacity of human-size humanoids than the Korean students, although the USA students did not assume emotional capacity of small-size humanoids as well as the Japanese and Korean students. These facts imply that the Japanese students more strongly assume characteristics similar to humans in human-size humanoids than the Korean and USA students.

Second, the results in section 3.2 shows that the Korean and USA students more strongly assumed housework and tasks in the office than the Japanese students, although this difference did not appear in human-size humanoids. The Korean students more strongly assumed tasks related to life-and-death situations in hospitals than the Japanese and USA students. Moreover, the USA students did not assume tasks related to nursing, social works, and educations as much as the Korean and Japanese students, and this difference appeared in small-size humanoids and pet-type robots. In addition, the Japanese and USA students more strongly assumed toys in the home or at amusement parks than the Korean students, although this difference also did not appear in human-size humanoids. On the other hand, there was no difference between the countries for tasks that are hard for humans to do and tasks in space, the deep sea, and battlefield. These imply that there are more detailed cultural differences of robot assumptions related to daily-life fields.

Third, the Korean students had more negative images of robots such as cause of anxiety in society, than the Japanese students. On the other hand, they also had more positive images such as friends of humans than the Japanese students. The USA students had more positive

images such as friends of humans and interesting technology, and less negative images such as a blasphemous of nature, than the Japanese students, although the USA students also more strongly assumed that robotics technology may cause anxiety in society and requires careful management, than the Japanese students. These imply that the Korean and USA students have more ambivalent images of robots than the Japanese students, and the Japanese students do not have as either positive or negative images of robots as the Korean and USA students.

4.2 Engineering Implications

We believe that the investigation of cultural difference will greatly contribute to design of robots.

Our implications on autonomy, emotional capacity, and roles of robots suggest that cultural differences may not be as critical a factor in applications of robots to non-daily life fields such as hazardous locations; however, we should consider degrees of autonomy and emotional capacity of robots in their applications to daily-life fields such as home and schools, dependent on nations where they are applied. For example, even if the Japanese and Korean students may commonly expect robotics application to tasks related to nursing, social works, and education, the autonomy and emotional capacity of robots should be modified in each country since there may be a difference on assumed degree and level of these characteristics.

Moreover, our implications on images of robots are inconsistent with some discourses that the Japanese like robots more than the other cultures, and that people in the USA and European countries do not like robots, due to the difference of religious backgrounds or beliefs (Yamamoto, 1983). Thus, we should not straightforwardly adopt general discourses of cultural differences on robots when considering daily-life applications of robots.

4.3 Limitations

First, sampling of respondents in each country is biased due to the limited number of universities involved in the study. Moreover, we did not deal with differences between ages such as Nomura et al., (2007) found in the Japanese visitors of a robot exhibition. Thus, the above implications may not straightforwardly be generalized as the complete comparison between these countries. The future research should extend the range of sampling.

Second, we did not define "culture" in the research. Gould et. al., (2000) used the cultural dimensions proposed by Hofstede (1991) to characterize Malaysia and the USA, and then performed comparative analysis on WEB site design. "Culture" in our research means just geographical discrimination, and it was not investigated which cultural characteristics individual respondents were constrained with based on specific determinants such as ones presented in social science literatures. The future research should include demographic variables measuring individual cultural characteristics.

Third, we did not put any presupposition since it was a preliminary research on cross-cultural research on robots. Although our results found an inconsistent implication with general discourses about the differences between Japan and the Western nations, as mentioned in the previous section, it is not clear whether the implication can be sufficient disproof for the discourses. Kaplan (2004) focused on humanoid robots and argued the cultural differences between the Western and Eastern people including Japan. His arguments lie on the epistemological differences between these nations about relationships

of technological products with the nature. It should be sufficiently discussed what the difference between the Japan and the USA on reactions toward robot image item "a blasphemous of nature" in our research presents, based on theories on relationships between cultures and technologies, including Kaplan's arguments.

5. Conclusions

To investigate in different cultures what people assume when they encounter the word "robots," from not only a psychological perspective but also an engineering one including such aspects as design and marketing of robotics for daily-life applications, cross-cultural research was conducted using the Robot Assumptions Questionnaire, which was administered to university students in Japan, Korea, and the USA.
As a result, it was found that:

1. the Japanese students more strongly assume autonomy and emotional capacity of human-size humanoid robots than the Korean and USA students,
2. there are more detailed cultural differences of robot assumptions related to daily-life fields,
3. the Korean and USA students have more ambivalent images of robots than the Japanese students, and the Japanese students do not have as either positive or negative images of robots as the Korea and USA students.

Moreover, we provided some engineering implications on considering daily-life applications of robots, based on these cultural differences.
As future directions, we consider the extension of the sampling range such as different ages and other nations, and focus on a specific type of robot to clarify differences on assumptions about robots in more details.

6. Acknowledgments

The research was supported by the Japan Society for the Promotion of Science, Grants-in-Aid for Scientific Research No. 18500207 and 18680024. Moreover, we thank Dr. Kazuyoshi Tsutsumi, Dr. Yasuhiko Watanabe, Dr. Hideo Furukawa, and Dr. Koichiro Kuroda of Ryukoku University for their cooperation with administration of the cross-cultural research.

7. References

Bartneck, C.; Suzuki, T.; Kanda, T. & Nomura, T. (2007). The influence of people's culture and prior experiences with Aibo on their attitude towards robots. *AI & Society*, Vol.21, No.1-2, 217-230

Gould, E. W. ; Zakaria, N. & Mohd. Yusof, S. A. (2000). Applying culture to website design: A comparison of Malaysian and US websites, *Proceedings of 18th Annual ACM International Conference on Computer Documentation*, pp. 161-171, Massachusetts, USA, 2000

Hofstede, G. (1991). Cultures and Organizations: Software of the mind. McGraw-Hill, London

Kaplan, F. (2004). Who is afraid of the humanoid? : Investigating cultural differences in the acceptance of robots. *International Journal of Humanoid Robotics*, Vol.1, No.3, 465-480

Mawhinney, C. H.; Lederer, A. L. & Du Toit, W. J. D. (1993). A cross-cultural comparison of personal computer utilization by managers: United States vs. Republic of South Africa, *Proceedings of International Conference on Computer Personnel Research*, pp. 356-360, Missouri, USA, 1993

Nomura, T.; Kanda, T.; Suzuki, T. & Kato, K. (2005). People's assumptions about robots: Investigation of their relationships with attitudes and emotions toward robots, *Proceedings of 14th IEEE International Workshop on Robot and Human Interactive Communication (RO-MAN 2005)*, pp.125-130, Nashville, USA, August, 2005

Nomura, T.; Suzuki, T.; Kanda, T. & Kato, K. (2006a). Altered attitudes of people toward robots: Investigation through the Negative Attitudes toward Robots Scale, *Proceedings of AAAI-06 Workshop on Human Implications of Human-Robot Interaction*, pp.29-35, Boston, USA, July, 2006

Nomura, T.; Suzuki, T.; Kanda, T. & Kato, K. (2006b). Measurement of anxiety toward robots, *Proceedings of 15th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN 2006)*, pp.372-377, Hatfield, UK, September, 2006

Nomura, T.; Tasaki, T.; Kanda, T.; Shiomi, M.; Ishiguro, H. & Hagita, N. (2007). Questionnaire-Based Social Research on Opinions of Japanese Visitors for Communication Robots at an Exhibition, *AI & Society*, Vol.21, No.1-2, 167-183

Shibata, T.; Wada, K. & Tanie, K. (2002). Tabulation and analysis of questionnaire results of subjective evaluation of seal robot at Science Museum in London, *Proceedings of 11th IEEE International Workshop on Robot and Human Interactive Communication (RO-MAN 2002)*, pp.23-28, Berlin, Germany, September, 2002

Shibata, T.; Wada, K. & Tanie, K. (2003). Subjective evaluation of a seal robot at the national museum of science and technology in Stockholm, *Proceedings of 12th IEEE International Workshop on Robot and Human Interactive Communication (RO-MAN 2003)*, pp.397-407, San Francisco, USA, November, 2003

Shibata, T.; Wada, K. & Tanie, K. (2004). Subjective evaluation of a seal robot in Burunei, *Proceedings of 13th IEEE International Workshop on Robot and Human Interactive Communication (RO-MAN 2004)*, pp.135-140, Kurashiki, Japan, September, 2004

Weil, M. M. & Rosen, L. D. (1995). The psychological impact of technology from a global perspective: A study of technological sophistication and technophobia in university students from twenty-three countries. *Computers in Human Behavior*, Vol.11, No.1, 95-133

Yamamoto, S. (1983). Why the Japanese has no allergy to robots. *L'sprit d'aujourd'hui (Gendai no Esupuri)*, Vol.187, 136-143 (in Japanese)

16

Posture and movement estimation based on reduced information. Application to the context of FES-based control of lower-limbs

Nacim Ramdani, Christine Azevedo-Coste, David Guiraud,
Philippe Fraisse, Rodolphe Héliot and Gaël Pagès
LIRMM UMR 5506 CNRS Univ. Montpellier 2, INRIA DEMAR Project
France

1. Introduction

Complete paraplegia is a condition where both legs are paralyzed and usually results from a spinal cord injury which causes the interruption of motor and sensorial pathways from the higher levels of central nervous system to the peripheral system. One consequence of such a lesion is the inability for the patient to voluntary contract his/her lower limb muscles whereas upper extremities (trunk and arms) remain functional. In this context, movement restoration is possible by stimulating the contraction of muscles in an artificial way by using electrical impulses, a procedure which is known as Functional Electrical Stimulation (FES) (Guiraud, et al., 2006a; 2006b).

When attempting to control posture and locomotion through FES, an important issue is the enhancement of the interaction between the artificial FES system controlling the deficient body segments and the natural system represented by the patient voluntary actions through his valid limbs motion. In most FES-systems, voluntary movements of valid limbs are considered as perturbations. In the case of valid persons, the trunk movements strongly influence the postural equilibrium control whereas legs have an adaptive role to ensure an adequate support base for the centre of mass projection. Collaboration between trunk and legs sounds therefore necessary to ensure postural balance, and should be taken in account in a FES-based control system. Indeed, generated artificial lower body movements should act in a coordinated way with upper voluntary actions. The so-obtained synergy between voluntary and controlled movements will result in a more robust postural equilibrium, a both reduced patient's fatigue and electro-stimulation energy cost.

At the moment, in most FES systems, controls of valid and deficient limbs are independent. There is no global supervision of the whole body orientation and stabilization. Instead, it would be suitable to: 1) inform the FES controller about valid segments state in order for it to perform the necessary adaptations to create an optimal and safe configuration for the deficient segments and 2) give to the patient information about the lower or impaired body orientation and dynamics in order for him to behave adequately. The patient could therefore use his valid body limbs to somehow "*teleoperate*" the rest of his body (see Fig.1.). Involving valid segment movements in the control of the artificial system, and therefore voluntary action of the person is also a way to give the patient an active role in the control

of his/her movements which would have positive psychological effect. The FES-assistance system should adapt to patient behaviour and intentions expressed through his valid limbs motions, instead of imposing an arbitrary motion on the deficient limbs (Heliot et al., 2007).
The need for cooperation between healthy and deficient limbs led us to the idea that valid limbs should be observed in order to improve the artificial control as well as deficient limb states should be somehow fed back to the patient in order for him to be able to behave efficiently.

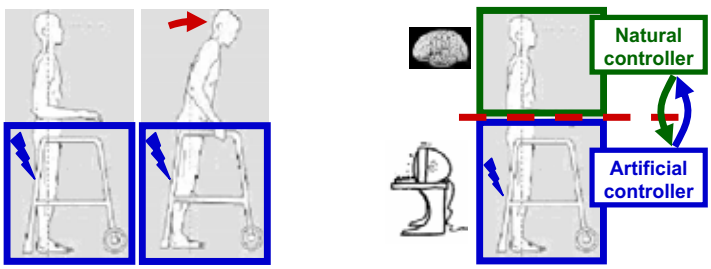


Figure 1. From no interaction to an efficient collaboration between artificial and natural controllers of patient deficient and valid limbs

These considerations led us to investigate the feasibility of characterizing and estimating patient posture and movement by observing the valid limbs by means of a reduced amount of information. Indeed, to be viable in everyday life context, the sensors involved have to be non obtrusive, easy and fast to position by the patient. On the contrary, laboratory-scale classical systems such as optoelectronic devices or force plates restrict the user to a constrained working volume and thus are not suitable.
In this chapter, we will develop two approaches for non-obtrusive observation:

1. The first one takes advantage of the available walker which is today still necessarily used by the patient. Hence, two six-degrees-of-freedom force sensors can be mounted onto the walker's handles in order to record upper limbs efforts. However, for safety reasons, the walker will be replaced in the sequel by parallel bars.
2. The second one disposes on patient's body, miniature sensors such as accelerometers. These sensors will be wireless and maybe implantable in the future.

We will illustrate the possible applications of these approaches for the estimation of posture while standing, for the detection of postural task transition intention and for the monitoring of movement's phases.
All the patients and subjects gave their informed consent prior to the experiments presented in this chapter.

2. Estimating posture during patient standing

In this section, we will show how one can reconstruct the posture of a patient, while standing, by using the measurement of the only forces exerted on the handles of parallel bars (Ramdani, et al., 2006 ; Pagès, et al., 2007).

2.1. The experimental procedure

2.1.1. Participants

Four spinal cord injured male subjects, with complete spinal lesions between T6 and T12, participated in the standing study program. The main selection criteria were the following: (1) participants show high motivation to the study, (2) post-injury standing experience, (3) appropriate contractions of the leg muscles in response to electrical stimulation, (4) sufficient upper body arm support strength to lift oneself up and maintain standing, (5) no cardiac or respiratory illness, (6) no previous stress fractures of upper and lower extremities, (7) no excessive body weight, (8) acceptable amount of spasticity and contracture in legs, (9) no psychological pathology.

2.1.2. Materials and Instrumentation:

For leg muscle stimulation during standing, an eight channel stimulator was used (see Fig.2). The self-adhesive surface electrodes were placed over the motor point areas of the quadriceps, the gluteus maximus, the tibialis anterior and the biceps femoris muscles of each leg. The stimulation device was driven directly in real time through a serial link by a PC. During active standing, patients were stimulated to predetermined FES constant currents, set up for each channel, in order to ensure safe standing. A video motion analysis system which included four infrared cameras was used to acquire kinematics data. The reaction forces measuring system, comprising two six-axis transducers, was attached to handles on adjustable supporting parallel bars. The six components of the handle reactions were measured and displayed throughout a real time implemented force sensor interface software. The handles height and separation were set to comfort for each patient.

2.1.3. Description of the protocol

In a first session, the subjects have been exposed to daily FES exercises, for up to 1 hour per day during 5 days, in order to strengthen their quadriceps, gluteal maximus/medius, biceps femoris and tibialis anterior muscles. In a second session, following a thorough explanation of the study procedure, the patients, under FES, were instructed to stand up from a chair, assisted by parallel bars, and stay in standing position and sit back down. The standing phase was as long as one minute. This training phase has been repeated several times in order for the participants to become familiar with the testing equipment. At session three, measurements were performed.

2.2. Modelling the human body and arm support

According to observations from human gait, most of joint movements during locomotion appear to take place in the sagittal plane. In our study, motion in the frontal plane during standing occurs at very low velocities. Moreover, stimulation on the different muscle groups of the lower limbs predominantly generates movement in the sagittal plane. For these reasons, the design of a two-dimensional model of the human body in the sagittal plane is sufficient for this study. During FES-standing, stimulation of the quadriceps and the hamstring locks the knee in extension, and therefore prevents knee movement. During stance, we consider that the distance between the thigh and the handle is constant, which allows us to assume that the ankle is immobilized. Hence, the lower limbs are here treated

as a single rigid link. The human body is thus regarded as a four bar linkage with a three degrees of freedom dynamic structure defined in the sagittal plane, as shown in Fig. 3.

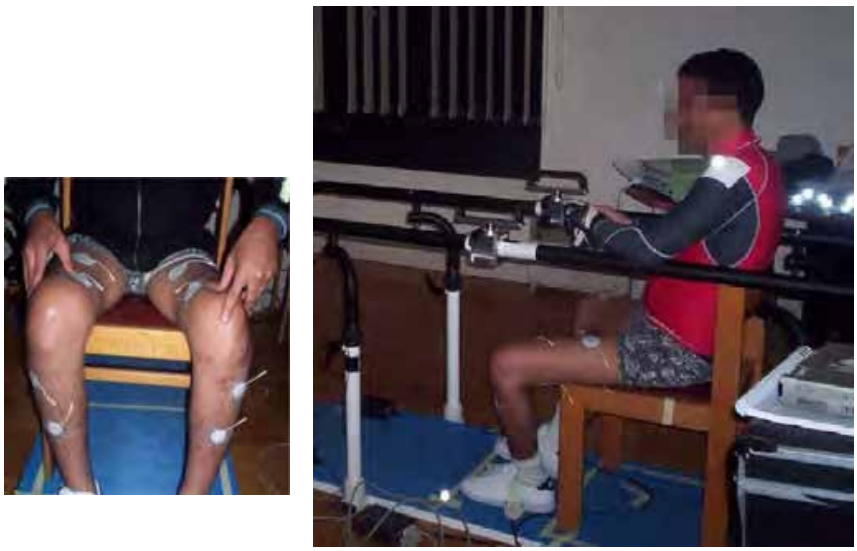


Figure 2. The experimental arrangement and placement of electrodes

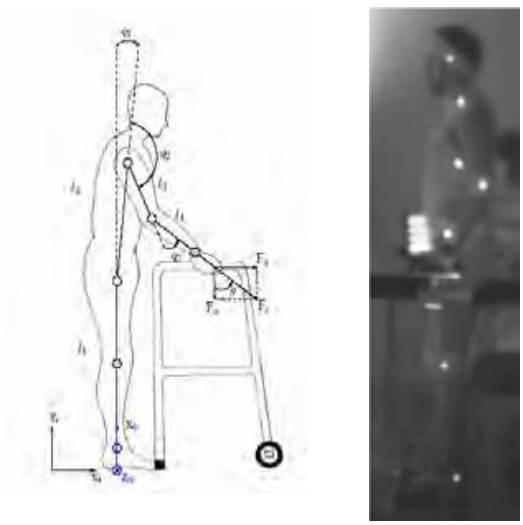


Figure 3. The four bar linkage human model (left). Actual captured image (right)

All links are assumed to be rigid bodies. We define $q = [q_1\ q_2\ q_3]^T$ as the joint angle vector, which is a function of time. It is expressed as a column vector with indices 1, 2 and 3 referring to the hip, the shoulder and the elbow joints respectively. The segments lengths are denoted by l_j. In Fig. 3, the variables q_1 and q_2 indicate positive angle directions while q_3 indicates a negative one, with respect to the zero position. Denote P_x and P_z the coordinates of the handle in the sagittal plane. The segmental model is given by (Khalil & Dombre, 2002)

$$P_x = l_2\sin(q_1) + l_3\sin(q_1+q_2) + l_4\sin(q_1+q_2+q_3) \quad (1)$$

and

$$P_z = l_1 + l_2\cos(q_1) + l_3\cos(q_1+q_2) + l_4\cos(q_1+q_2+q_3) \quad (2)$$

During FES-supported movements, paraplegic patients need their arms to maintain balance and sustain desired movement. Support is taken in charge by two handles, each equipped with the six axis force/torque sensor, mounted on the supporting frame.
Contact between the human hand and the handle creates a closed chain kinematics linkage. This interaction is described by the components of the resultant force vector F_c measured in the x and z directions. Under the assumption of working in the sagittal plane and considering that the orientation of the forearm is colinear to the resultant force F_c, which is true when the x-axis component of the resultant force satisfies $F_x \geq 0$ and the z-axis component satisfies $F_z < 0$ (see Fig.3), it is reasonable to write the following hypothesis :

$$q_1 + q_2 + q_3 - \pi \approx \arctan(F_x/F_z) \quad (3)$$

2.3. A Set membership identification of posture

Equations (1)-(3) can be re-written as

$$g(\mathbf{q}) = \mathbf{y} \quad (4)$$

where $\mathbf{y} = [P_x, P_z, \arctan(F_x/F_z)]^T$.
The patient's posture is given by the **q** vector, which can be obtained by solving (4). If the measured quantities **y** and anthropometric parameters l_j were known with no uncertainty, then the problem could be solved analytically through state-of-the-art tools by using inverse kinematics. Solving (4) when y is subject to uncertainty with classical techniques based on possibly weighted least squares optimisation for instance, derives reliable results only if the errors are stochastic and with known probability laws. In fact the measured data are subject to either stochastic or deterministic uncertainties and it is not easy to derive a reliable characterization of the probability distribution for these errors. Moreover, the model used may be based on some simplifying hypotheses for which a full probabilistic description might not be reliable. Consequently, it is more natural to assume all the uncertain quantities as unknown but bounded with known bounds and no further hypotheses about probability distributions. In such a bounded error context, the solution is no longer a point but is the set of all acceptable values of the q vector, which makes the model output $g(\mathbf{q})$ consistent with actual data **y** and prior error bounds.
Denote **E** a feasible domain for output error and $\mathbf{Y} = y + \mathbf{E}$ the feasible domain for model output. The set **S** to be estimated is the set of all feasible postures:

$$\mathbf{S} = \{\mathbf{q} \in \mathbf{Q} \mid g(\mathbf{q}) \in \mathbf{Y}\} \quad (5)$$

where the set **Q** is an initial search space for the q vector. Characterizing the set **S** is a set inversion problem which can be solved in a guaranteed way using a set inversion algorithm

based on space partitioning, interval analysis and constraint propagation techniques (see (Jaulin, et al., 2001) and the references therein). This algorithm explores all the search space without losing any solution. It makes it possible to derive a guaranteed enclosure of the solution set **S** as follows:

$$\mathbf{S}_{inner} \subseteq \mathbf{S} \subseteq \mathbf{S}_{outer} \tag{6}$$

The solution set **S** is enclosed between two approximation sets. The inner enclosure $\mathbf{S}_{inner}$ consists of the boxes that have been proved feasible. To prove that a box [**q**] is *feasible* it is sufficient to prove that g([**q**]) ⊆ **Y**. If, on the other hand, it can be proved that g([**q**])∩**Y**=∅, then the box [**q**] is *unfeasible*. Otherwise, no conclusion can be reached and the box [**q**] is said *undetermined*. It is then bisected and tested again until its size reaches a threshold to be tuned by the user. The outer enclosure $\mathbf{S}_{outer}$ is defined by $\mathbf{S}_{outer} = \mathbf{S}_{inner} \cup \Delta\mathbf{S}$ where Δ**S** is given by the union of all the *undetermined* boxes. The outer enclosure $\mathbf{S}_{outer}$ contains all the solutions, if they exist, without losing any of them. It contains also some elements that are not solution.

2.4. The estimated posture

Posture estimation was done during the standing phase. The subject's actual posture during that time interval were measured as :

$$q_1 \approx 0°,\ q_2 \approx 192°,\ q_3 \approx -36° \tag{7}$$

representing respectively the hip, shoulder and elbow joint angles. The body segment lengths were directly measured on the patient and are given by :

$$l_1 \approx 0.954 \text{ m},\ l_2 \approx 0.518 \text{ m},\ l_3 \approx 0.334 \text{ m},\ l_4 \approx 0.262 \text{ m} \tag{8}$$

The feasible domain for model output are taken as:

$$\begin{aligned} P_x &\in [-0.02, 0.02] \text{ m} \\ P_z &\in [0.895, 0.995] \text{ m} \\ \arctan(F_x/F_z)] &\in [-18.63, -15.63]° \end{aligned} \tag{9}$$

The prior search space **Q**, corresponding to the joints articular motion limit, is taken as:

$$[-11, 90]° \times [90, 210]° \times [-103, 0]°.$$

The projections of the computed inner and outer solution sets, $\mathbf{S}_{inner}$ and $\mathbf{S}_{outer}$ onto the $q_i \times q_j$ planes are given in Fig.4. Contrary to any optimization based techniques, there are no optimal solution, therefore any posture taken within the solution set is an acceptable one. Extreme postures taken from solution set are also plotted in Fig.5. These figures clearly show that the solution sets contain the actual posture (see also Table 1).

Joints	Projection of inner enclosure	Projection of outer enclosure
q_1	[-1.35 , 25.52]	[-4.14 , 28.79]
q_2	[192.5 , 213.66]	[190.34 , 215.32]
q_3	[-74.10 , -31.05]	[-77.81 , -28.28]

Table 1. Projection of solution posture

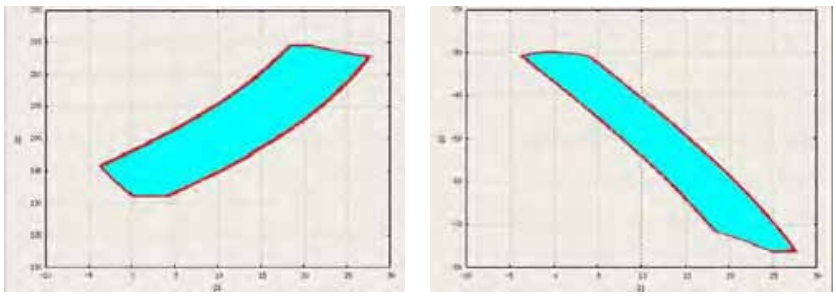


Figure 4. Projection of solution set onto $q_1 \times q_2$ (left) and $q_1 \times q_3$ (right)

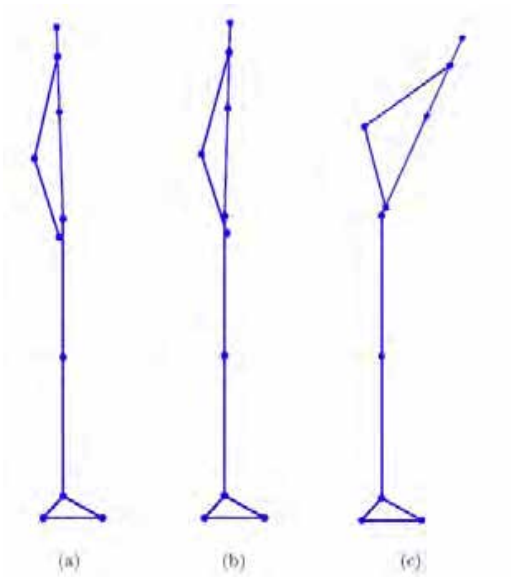


Figure 5. Postures taken from the solution sets:
(a) Patient leaning back, (b) Actual patient posture, (c) Patient leaning forward

Indeed, the experimental method introduced in this section is capabale of reconstructing the posture of a patient but with fairly large uncertainties. This reflects the fact that for a fixed position of the forearm taken within the feasible domain calculated by force measurements in the sagittal plane only, the hip, the shoulder and the elbow joints still have the possibility to reach other positions, while being consistent with the defined geometrical constraints. In order to further reduce the solution set, and hence have a more precise estimation of patient's posture, new constraints has to be introduced by using dynamic modelling and ground reaction forces measurements, for instance.

3. Observing valid limbs to detect patient intention

In this section, we propose an approach for the recognition of the "signature" of the postural task the subject intends to realize (sit-to-stand, object grasping, walking, stair climbing, gait initiation/termination...) through voluntary movement observation. This detection should occur as soon as possible after the subject has decided to initiate the task. It is particularly important to detect the transitions between activity modes as soon as possible after the patient has taken the decision to modify his functioning mode, in order to allow for optimal posture preparation and execution.
A good illustration for this is the transfer from sit to stand. In our FES context it is essential to optimize this task, muscle fatigue being a major issue. Minimizing efforts of rising up could improve the following activities of the patient. For this reason, classical techniques consisting of maximum stimulation of knee extensors throughout the rising process are not suitable and involve an over-use of arm support.
Two approaches are considered in the following to estimate patient attitude: the instrumentation of the walker and body-mounted micro-sensors.

3.1. Sit to stand dynamics analysis

Assuming that the body structure is rigid, continuous dynamics can be expressed under the Lagrangian form:

$$\mathbf{M}(\mathbf{q})\mathbf{q}''+\mathbf{N}(\mathbf{q},\mathbf{q}')\mathbf{q}'+\mathbf{G}(\mathbf{q})=\mathbf{\Gamma}+\mathbf{\Gamma}_{ext} \tag{10}$$

where: $\mathbf{q}$ stands for the parametrization vector of the whole configuration space of the biped considered as free in 3D, $\mathbf{\Gamma}$ is the joint actuation torque, $\mathbf{M}$ is the inertia matrix, $\mathbf{N}$ is the matrix of centrifugal, gyroscopic and Coriolis effects, $\mathbf{G}$ is the generalized gravity force vector. $\mathbf{\Gamma}_{ext}$ are torques generated by external forces such as ground contacts, interaction with a chair, a thrust, etc. They can be expressed as:

$$\mathbf{\Gamma}_{ext}=\mathbf{C}(\mathbf{q})^T\boldsymbol{\lambda}(\mathbf{q},\mathbf{q}') \tag{11}$$

$\mathbf{C}(\mathbf{q})$ is the Jacobian matrix of the points of the biped to which the external forces are applied and $\boldsymbol{\lambda}$ corresponds to the amplitudes of these forces. Biped dynamics are characterized by the existence of variable constraints resulting from interaction with the ground. Ground efforts correspond to a set of forces applied to the points of the biped in contact with the ground (Azevedo et al., 2007a).
Using this framework, we propose to express the sit-to-stand transfer as an optimization problem, where the posture configuration q minimizes a cost function over a time horizon h:

$$J=(H_{com}-H_{comd})^T(H_{com}-H_{comd}) \tag{12}$$

where $H_{com}(t)=[X_{com}(t); Y_{com}(t); X_{com}(t+1); Y_{com}(t+1);\ldots; X_{com}(t+h); Y_{com}(t+h)]^T$ is the sequence of centre of mass positions over the time horizon h, $H_{comd}=[X_{comd}; Y_{comd};\ldots; X_{comd}; Y_{comd}]^T$ is a vector made of the repetition of the *desired* position of the centre of mass (standing posture) over the time horizon h. The solution to this problem is illustrated in Figs.6 & Fig.7-a. The biped goes directly from seated posture to standing.

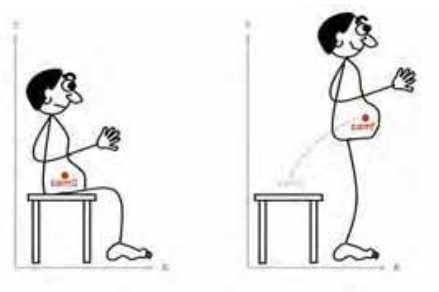
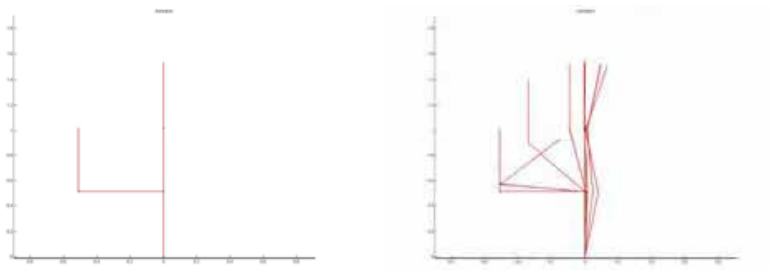


Figure 6. Illustration of the problem of sit to stand consisting in transferring the centre of mass projection from seat to feet



a- no constraint b- constrains on torques

Figure 7. Simulation of sit to stand transfer by solving an optimization problem minimizing distance between actual and desired center of mass position over a sliding time horizon

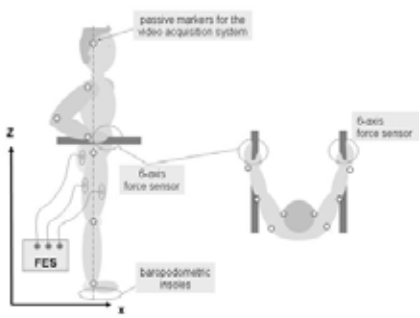


Figure 8. Description of the experimental protocol

If now some constraints are added to the problem in terms of limitation of joint torques, i.e.

$$\boldsymbol{\Gamma}_{min} \leq \boldsymbol{\Gamma} \leq \boldsymbol{\Gamma}_{max} \tag{13}$$

the result is that the system has to use its trunk inertia to achieve the movement (fig.7-b), upper body bends forward before legs initiate movement. This simulation results explain clearly the important need of coordination between upper and lower limbs to execute a transfer from seat to stand when available torques are limited, which is obviously the case for muscles. Without this coordination additional external efforts are needed, such as arm support.
Based on these preliminary considerations, we propose two approaches for the detection of sit-to-stand movement.

3.2. Walker instrumentation

We first investigate the possibility of considering body supportive forces as a potential feedback source for FES-assisted standing-up control (Azevedo et al., 2007b). The six-degrees of freedom force sensors were mounted onto handles fixed on parallel bars in order to record upper limbs efforts and insoles were fitted in the patient's shoes to record plantar pressure distribution (Fig.8). Eight volunteer complete paraplegic patients (T5-T12) were verticalized by means of adapted FES. The same training protocol as presented in the previous section was used. A video motion analysis system recorded the positions of passive markers placed on the body allowing us to measure kinematics. The results show that the transfer (phase 1) is mainly ensured by arm support in all our patients (Fig.9). We gave instruction to the patients to bend their trunk in preparation to the chair rising. An important observation when looking at trunk, knee and ankle angles is the low intra-variability between trials of one given patient (Fig.10). A main difference between valid subjects and patients is the onset of leg movement in regards to trunk bending (Fig.10). To be efficient, trunk bending forward should start before and last during knee and ankle movement. This was never the case in our trials on FES-assisted standing.
Minimizing arm support contribution is possible only if trunk inertia is used. This implies a good triggering of muscle contraction regarding limb movements. Trunk behaviour could be indirectly observed by analyzing efforts applied by arm support (Fig.11). Indeed, normal force decreases (pulling) while momentum around transversal axis increases. From these results we can say that proper threshold detection based on these signals could be used to initiate the leg stimulation and improve greatly the sit to stand. The same may be used for stand to sit as shown in Fig.11.

3.3.1. Body-mounted instrumentation

In parallel to the approach presented in the previous section, we have also worked on demonstrating the pertinence of observing the trunk using a movement sensor placed on the back of valid subjects (Azevedo & Héliot, 2005). Indeed, as seen before, the trunk normally initiates the sit-to-stand transfer. We have placed on the back of 10 valid subjects, at anatomical C7 level, an accelerometer. Trunk acceleration patterns present low intra and inter-variability as well as a high temporal reproducibility and are therefore a nice characteristic "signature" of the sit-to-stand transfer (see Fig.12).
It is possible to apply techniques such as abrupt changes theory (Basseville & Nikiforov, 1993) to detect the pattern of sit to stand "intention". This technique allows detecting robustly the transfer initiation with a good sensitivity. The algorithm is able to reject a "false" sit-to-stand movement involving trunk movements such as grasping an object placed in front of the person. Indeed, the acceleration pattern signs selectively the motion.

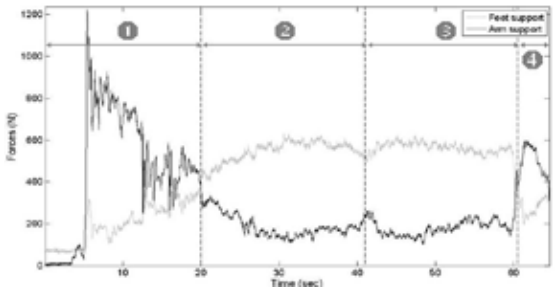


Figure 9. Patient #3, trial 6. Total feet and arm support. Phase labelled 1 corresponds to sit to stand phase, 2+3 corresponds to standing, and 4 to knee flexion

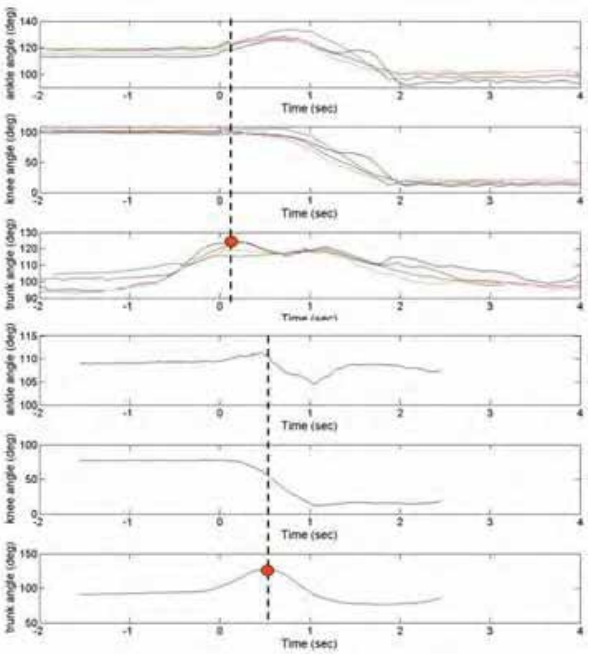


Figure 10. Posture coordination during sit to stand. Top**:** Patient #1, over 4 trials, Bottom**:** valid subject. The red dot indicates the maximum trunk bending

It is important to notice here that the detection of transfer has to occur as soon as possible before the legs should start moving to displace the body centre of mass from the seat to the feet (Fig.12). Around 600ms separate the instant when the trunk starts bending forward and the instant when the legs enter in extension movement. It is also necessary to recall here, that lower limb muscles start to contract before the legs move in order to prepare the

motion. These so-called anticipative postural adjustments should take place ideally together with trunk movement.
In order to apply these results on patient FES-assisted sit to stand it is necessary to train paraplegic patients in executing an optimal trunk movement in order to benefit from its inertia in the standing transfer. The detection algorithm should then be able to recognize patient intention to stand and trigger the proper stimulation sequences.

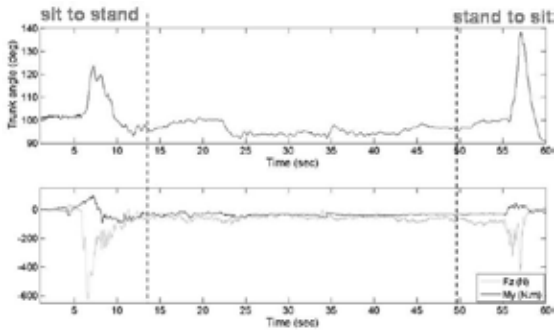


Figure 11. Correspondence between trunk angle and handle information. Patient #1, Trial 1. Top**:** angle, Bottom**:** right side vertical force and momentum around hip axis

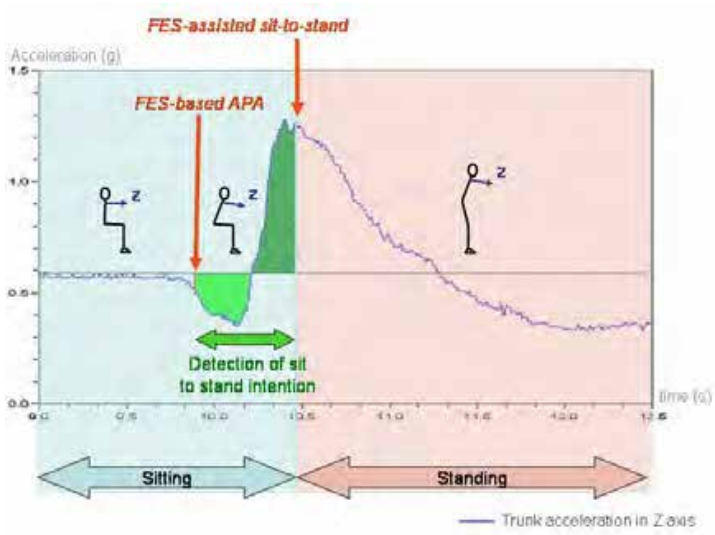


Figure 12. Principle of the detection of sit-to-stand based on the observation of trunk acceleration

The results presented in this section of the article clearly demonstrate the need for collaboration between lower and upper limbs in FES-assisted sit to stand. Triggering FES on arm support observation appears to be possible. A more anticipated timing of FES would be possible by detecting sit to stand trough trunk accelerations.

4. Classifying patient motor activities

In this section, we will address the issue of online classification of patient postures and motor activities, such as standing, sitting or walking for instance. Such a technique could be used to design a discrete-event based controller whereas the state estimation of the different joint angles could be used for a continuous controlling system. Both may be used in a hybrid controller where the best control strategy could be selected depending on the movement to be achieved.

Online classification can be performed by using neural networks and sensors such as accelerometers, when the purpose is only to detect phases of movements. It has been successfully applied with ageing persons in order to detect falling and to perform global activity monitoring (Fourty, et al., 2006 ; 2007) and thus could be used with disabled patients employing FES systems.

The classification algorithms described in the literature (Rumelhart & Mac Clelland, 1986) are generally implemented on desk computers. In biomedical engineering, and more specifically in the domain of ambulatory monitoring (Iwata, et al., 1990), classification is performed "off-line" from data collected on wearable systems. Our approach to the ambulatory monitoring of human activities is based on the design of wearable devices for automatic labeling. The aim of the procedure is to save time, reduce memory size and obtain relevant data. This constitutes a pattern recognition problem under specific constraints. Before describing the classification implementation, we briefly present the portable acquisition system.

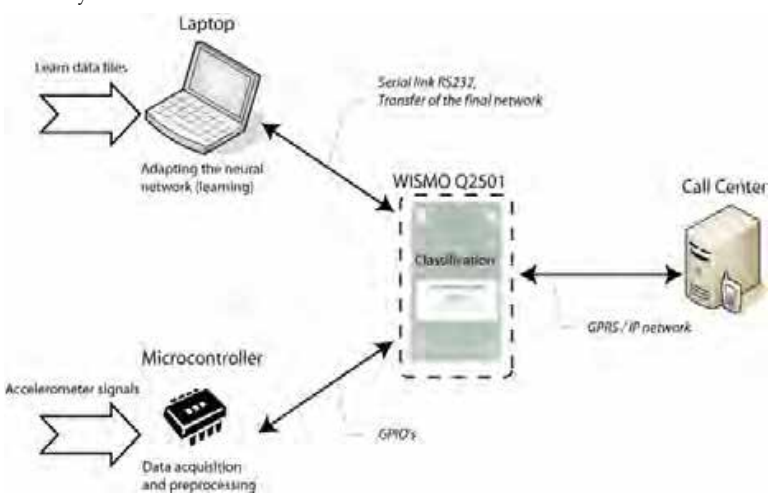


Figure 13. Hardware architecture

4.1. Materials

We use a microcontroller-based board with an ultra low power MSP430F169. One ADXL320 bi-axial accelerometer from Analog Device will sense activity. In order to quantify the patient's activity, some basic movements have to be recognised: steady-state movements (standing upright, sitting, walking, etc.) and transitional movements (sitting to standing, leaning to standing, etc.). This recognition in the system is done by the learning phase of an artificial neural network. Optimized learning can be done "off-line" on a classical computer, and only the pattern recognition algorithm and its associated memory have to be downloaded to the system. The Fig.13 describes the hardware system architecture.

4.2. A specific Artificial Neural Network (ANN)

To implement this classification we use hypersphere clustering with an incremental neural network. It is based on the evaluation of distances between the input vector and stored vectors in the memory. One n-dimension vector can be represented by a point in an n-dimensional space. Each component is a feature of the pattern to be recognized. Features can be raw data, or much more representative values given by feature-extraction procedures. A reference vector, the centroid, with its associated threshold, the radius, is labelled with a class. This defines a "prototype" which is represented by a hypersphere in an n-dimension space. A prototype is fired when an input point is situated within the hypersphere. Thus, fired prototypes participate in the final decision. This is the general functioning of hypersphere clustering-based methods.

4.2.1. Global structure and state dynamics

We are going to use some notations in this section:

- $I_0 = (I_{01},\ldots,I_{0n})$ and $I_1 = (I_{11},\ldots,I_{1n})$ are input vectors,
- I_{2j}, I_{3j} are output values of the second layer cell,
- R_j, $W_j = (W_{j1},\ldots,W_{jn})$ are radius (threshold) and coordinates of the centroid (reference vector) of the hypersphere (cell of the second layer)

Input layer (normalisation-saturation): The input layer performs a normalisation-saturation of the inputs in an eight bit resolution. An input value between Imin and Imax is transformed in the range of 0-255. Unexpected data below Imin or above Imax are set to 0 or 255, respectively. Each input comes from one real input datum, and each output is connected to all the cells in the hidden layer.

Hidden layer (prototypes): The hidden layer consists in prototype cells that compute distances between a normalised input vector and reference vectors (the centroids of the hyperspheres in the n-dimension space). Then, each cell makes a comparison between the computed distance and a threshold (the radius of the hypersphere) in order to obtain the following outputs: Output I2 is connected to a special cell that stores the minimum distance obtained, with the corresponding class. Output I3 is connected to only one output cell corresponding to the labelled class. The prototype is fired if the distance is less than the radius. The output also depends on the fact that the radius is set to the minimum, which means that during the learning phase it was reduced to the minimum value by examples from wrong classes. This situation occurs within an uncertain decision zone.

Norm: We will now consider the norm used to compute distances. In a continuous space the norms are strictly equivalent in a mathematical sense. But in a discrete space, things are different because parameters and data are integers. We can show that norm 1 is the best one

because it ensures the finest space clustering with the smallest step of number of points included in the hypersphere, with a unit increment or decrement of R. For a given radius the smallest number of points included within the hypersphere is also observed for norm 1. In terms of classification abilities, the radius can be tuned more precisely. Moreover, this norm requires only additions, subtractions, and comparisons. Norm 1 proves clearly to be the best, and was implemented on our algorithm.

Output layer (classes): Each cell of the output layer corresponds to one class and all the prototypes of the previous layer labelled with the same class are connected to it. Then, the operation carried out consists in a logical OR so that the output can be 0, 1, 2, or 3. Thus, the discriminent elements are only the types of prototype fired for each class :

0 : no prototype fired
1 : only reduced (R=Rmin) prototypes fired
2 : non-reduced prototypes fired
3 : both types of prototype fired

The most important characteristic of this algorithm is that it does not take into account statistical criteria (for instance the number of prototypes fired is not evaluated). This ensures the recognition of rare but well-defined events, a situation which frequently occurs in biomedical applications.

Fig.14 summarizes the global structure.

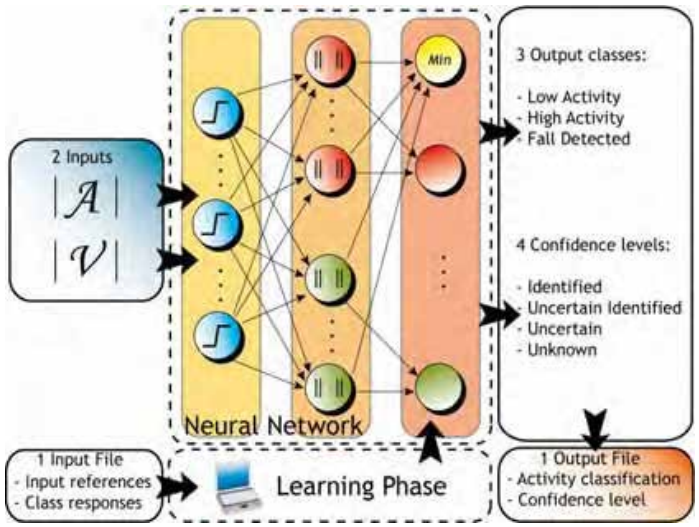


Figure 14. Overview of the ANN

4.2.2. Connection dynamics

Recognition process : The recognition phase is quite simple. The possible responses of the network are the following: unknown, uncertain, uncertain identified, and identified, depending on the criteria reported in Table 2. In all cases, a list of fired classes is given in decreasing order of confidence. We note that the nearest neighbour criterion is only used to

discriminate between uncertain classifications because this criterion is highly dependent on the way the network is learned (positions of the examples). Nonetheless, most of the time it becomes the best and simplest criterion when others have failed.

Learning rules : There are three main parameters in which a learning rule can be applied: creation/destruction of prototypes, displacement of the centroid, and adjustment of the radius. Papers on different algorithms using the creation/reducing radius (Nestor™ system), centroid position, and creation/position/radius have been published (Judge J., et al., 1996). Most of them do not affect more than one parameter at a time. Moreover, few algorithms can remove a prototype, and this could be useful when the set of examples includes errors. Some algorithms introduce an activation value for each prototype and use it in the recognition phase. We do not use this because of the statistical effect of this parameter. The learning phase becomes sensitive to the class representation in the learning set. Indeed, this explains why we developed our own algorithm, enabling creation and removal of prototypes, and simultaneous adjustment of the centroid position and the radius (increase or decrease) of the hypersphere (Table 3).

Four learning parameters appear: α and α' set the amplitude of the correction (0 means no correction, 1 excludes the sample from the prototype); and β and β' set the proportion between the centroid displacement (max when 1) and the radius adjustment (max when 0). Varying these parameters allows adjustment of the learning rule to the set of examples.

Identified	Only one class has obtained 2 or 3
Uncertain identified	Several classes have the same highest score but one class has the nearest neighbour (given by the "min " cell) Only one class has obtained 1 The nearest neighbour if no classes are fired (useful when a decision must always be taken)
Uncertain	Several classes have the same highest score but no nearest neighbour
Unidentified	No class fired and no nearest neighbour

Table 2. Recognition confidence level

Situation	Action
No prototype fired	Creation of a prototype where $W_i=I_{1i}$, R=Rmax or R=min distance of the centroid of prototypes of wrong classes Creation, if necessary, of a cell in the output layer, for the first occurrence of this class
At least one prototype fired	The nearest is approached and its radius is increased according to the formulae $R = R + \alpha(1-\beta)\lVert W - I_1 \rVert_1 \quad \alpha,\beta \in [0,1]$ $W_i = W_i + \alpha\beta(I_{1i} - W_i)$ the increase of R is limited to Rmax.
R=Rmin	If this occurs subsequently n times, the prototype is removed
R>Rmin	Radius is reduced and the centroid is displaced according to the formula $\Delta = \alpha'(R - \lVert W - I_1 \rVert_1 + 1)$ $R = R - (1-\beta')\Delta$ $W_i = W_i + \beta'\Delta\frac{(W_i - I_{1i})}{\lVert W - I_1 \rVert_1} \quad \alpha',\beta' \in [0,1]$ R can be reduced up to Rmin.

Table 3. Summary of learning rules

The advantages of such a neural network are the following:
- It can be easily implemented on a microcontroller.
- It is an incremental neural network, so that a new configuration can be learned without the need for learning again with the whole set of examples.
- The neural network algorithm avoids the consideration of statistics, which provides a learning phase less sensitive to the learning set, and the rare events can be well identified
- Unexplored spaces provide unknown responses, thus avoiding misclassification. The unlabelled data are stored, analysed "off-line", and then learned.

4.3. Validation of the prototype

The measurement of acceleration along two axes (horizontal and vertical) enabling fall detection is used to monitor gait activity of the patient. The use of the neural network method presented previously needs relevant input vector in order to provide relevant classification. To find out the best input the Neural Network should be provided with, we have assessed many different cases such as acceleration along x and y axes, average and standard deviation of acceleration or magnitude of acceleration and velocity. The result of this evaluation is that the best inputs to analyse the gait activity of the patient are the magnitude of acceleration and velocity. We have chosen three different output classes representing low and high activity respectively for instance walking and running, and fall detection.

Computation of velocity: The main difficulty encountered with the computation of the velocity is the offset signal stemming from accelerometers that disturbs deeply the result of the velocity. We have observed this offset signal on experimental measurements. It is not a constant value. To overcome this problem, we propose to compute the velocity by using centred accelerations, where the average is computed over a sliding window, which can be adapted according to accelerometer types.

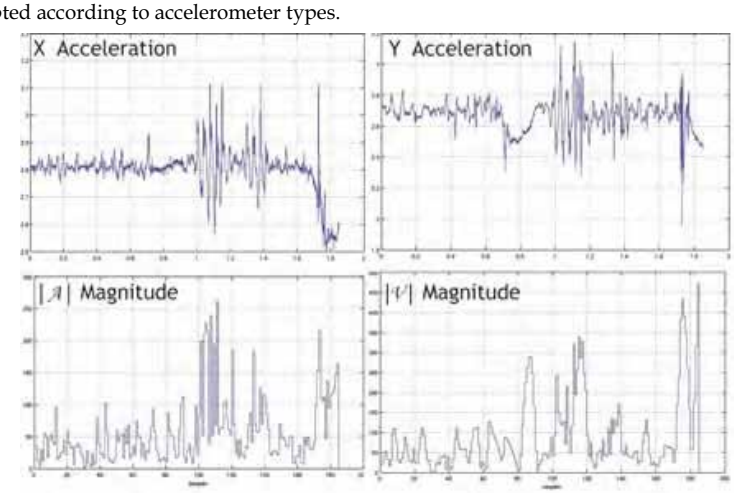


Figure 15. Sensor outputs and computed vectors

Learning Phase: We used some experimental data streams from elderly people simulating different activities such as walking, running and falling down. The reference file used is presented Fig.15. This file contains three different activities, which are:

- Low activity: t=0s, t=10s (walking, sitting)
- High activity: t>10s, t=16s (running)
- Fall detection: t>16s

Recognition phase: Fig.16 presents the recognition phase performing on the reference file by using the neural network. Each activity is well detected and recognised. The y axis of the Fig.16.b represents the classes such: 1→ low-activity, 2→ high-activity, 3→Fall.

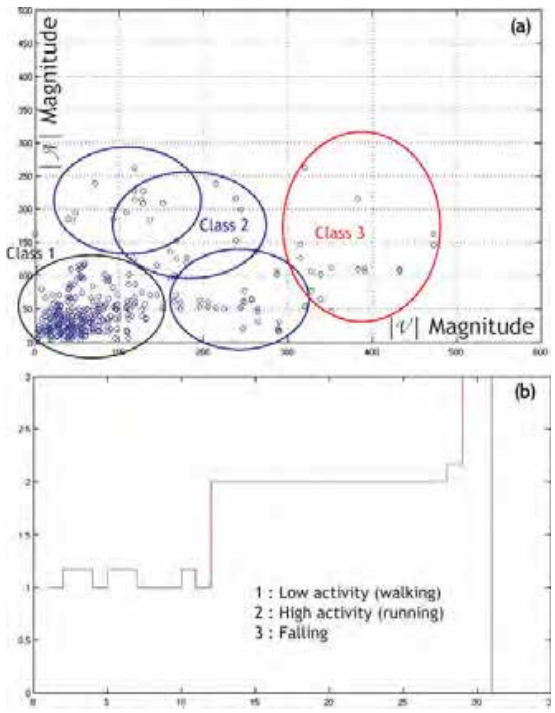


Figure 16. Recognition on learned file

Validation: To validate this development, we have performed the classification method on different patients keeping the previous learning phase as reference in order to estimate the robustness. Fig.17 shows the capacity to detect and discriminate the three phases even with a learning phase carried out on another patient. The results of the classification activities show the first period as an intermediate class between 1 and 2 (mean value is about 1,5). The second period is also higher than 2. These intermediate results (class 1,5 for instance) mean that there is an uncertainty between both of them (classes 1 and 2). Then, we compute an average value within this ambiguous period, which returns an intermediate class result. These results are

due to unknown activities detection which is obtained by using the nearest neighbour criterion. This criterion can propose alternatively class 1 or 2 as the nearest neighbour.

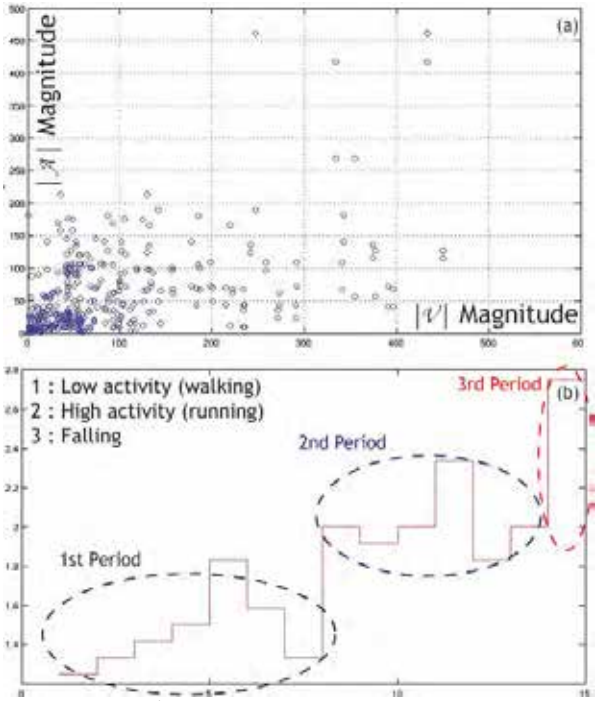


Figure 17. Recognition on another person

5. Conclusion

In this chapter we have introduced several approaches for the estimation, detection and classification of the posture or movement of disabled patients. They are founded on non-intrusive sensors and are meant for closed-loop control in the context of functional restoration via electrical stimulation.

While taking advantage of an available walker, we have investigated the potential of using only arm support measurements. Then, we found that we can reconstruct patients standing postures only with a fairly large uncertainty. However, we found that these measurements can be used for detecting patients trunk movement. When miniature sensors are attached onto the patient's body, then it is possible to efficiently detect transitions such as sit-to-stand or classify steady-state movements such as standing, sitting or walking. The two technologies could eventually be combined.

Finally, a synergy between artificial and voluntary movements can indeed be achieved by using these methods. For instance, in a FES-assisted sit-to-stand movement, the electrical stimulation should be triggered according to the patient's trunk movements.

6. References

Azevedo C. & Héliot, R. (2005). Rehabilitation of Functional Posture and Walking: towards a coordination of healthy and Impaired Limbs. *Journal of Automatic Control*. Vol. 15 (Suppl), pp. 11-15.

Azevedo C., Espiau, B., Amblard, B. & Assaiante, C. (2007a). Bipedal Locomotion: Towards Unified Concepts in Robotics and Neuroscience. *Biological Cybernetics* Vol. 96, No. 2, pp. 209-228.

Azevedo, C., Pages G., Maimoun L., Fattal C., D. Delignières, D. Guiraud (2007b). Description of postural coordination patterns during FES-assisted standing in complete paraplegia. *9th Vienna International Workshop on FES.*

Basseville, M. & Nikiforov, I.V. (1993). *Detection of Abrupt Changes - Theory and Application*. Prentice-Hall, Inc. Englewood Cliffs, N.J.

Fourty, N.; Guiraud, D.; Fraisse, P.; Perolle, G. ; Etxeberria, I. & Val, T. (2007). Evaluation of a neural network used for classifying motor activities of elderly and disabled people, *IEEE Transactions on Systems, Man and Cybernetics*, submitted.

Fourty, N.; Guiraud, D.; Fraisse, P. & Perolle, G. (2006). A specific neural network used on a portable system for classifying activities in ambulatory monitoring, in: *IEEE ICIT'06*, Mumbai, India.

Guiraud, D.; Stieglitz, T.; Koch, K. P. and Divoux, J. L. & Rabischong, P. (2006). An implantable neuroprosthesis for standing and walking in paraplegia: 5-year patient follow-up, Journal Neural Engineering, Vol. 3, pp.268-275.

Guiraud, D.; Stieglitz, T.; Taroni, G. & Divoux, J.L. (2006). Original electronic design to perform epimysial and neural stimulation in paraplegia, Journal Neural Engineering, Vol. 3, pp. 276-286.

Heliot R., Azevedo C., Espiau B. (2007) Functional Rehabilitation: Coordination of Artificial and Natural Controllers**.** ***ARS (Advanced Robotic Systems) Rehabilitation Robotics.***

Iwata A., Nagasaka Y., Suzumura N., (1990). Data compression of the ECG using neural network for digital holter monitoring, *IEEE Eng. In Med. end Bio.*, 9, 3, 53-57.

Jaulin, L.; Kieffer, M.; Didrit, M. & Walter, E. (2001). Applied Interval Analysis: with examples in parameter and state estimation, robust control and robotics, Springer-Verlag, London.

Judge J., Ounpuu, S., Davis R., (1996). Effect of age on the biomechanics and physiology of gait, In: *Clinical Geriatric Medicine. Gait and Balance Disorders*, S. Studenski (ed.), pp.659-678, Philadelphia: Saunders.

Khalil, W. & Dombre, E. (2002). *Modeling, Identification & Control of Robots*, Hermes Penton Science, London

Pagès, G.; Ramdani, N.; Fraisse, P.; Guiraud, D. (2007). Upper body posture estimation for standing function restoration, *Proceedings of IEEE International Conference on Robotics and Automation ICRA'07*, pp. 3742-3747, Roma.

Ramdani, N.; Pagès, G.; Fraisse, P.; Guiraud, D. (2006). Human upper body posture estimation from forces exerted on handles, *Proceedings of IEEE International Conference on Robotics and Biomimetics, ROBIO2006*, pp. 410-415, Kunming.

Rumelhart D., Mac Clelland J., (1986). *Parallel distributed processing*, MIT Press Cambridge, MA.

17

Intelligent Space as a Platform for Human Observation

Takeshi Sasaki and Hideki Hashimoto
Institute of Industrial Science, The University of Tokyo
Japan

1. Introduction

In the recent years, the needs for physical support such as release from household work, care for the elderly people and so on are rising. Therefore, researches on robots for daily life are being pursued actively. However, highly dynamic and complicated living environments make it difficult to operate mobile robots.

In order to solve this problem, it is important to design the environment for mobile robots as well as making mobile robots intelligent to adapt to it. But environmental design in the living space is limited because it should not have a big influence on human lives. We also have to consider how to deal with the dynamic environment. So, it is not enough just to apply a passive approach (e.g. elimination of difference in level on the floor, installation of markers on the wall and so on). Moreover, to provide appropriate service to the human according to the circumstances, mobile robots have to understand the request from human based on observation. The information extracted from observation of humans can also be used for the action of mobile robots because humans are expected to be producing intelligent reactions when confronted with various situations. However, it is not practical that a mobile robot keeps on observing humans while doing other tasks. In addition, owing to restrictions of the capability of mounted sensors and computers, it is difficult to observe humans using on-board sensors.

In order to realize this, we utilize "Intelligent Space (iSpace)" where many intelligent devices are distributed. (Lee & Hashimoto, 2002). Such an environment is referred as smart environment, smart space, intelligent environment and so on. The smart environments observe the space using distributed sensors, extract useful information from the obtained data and provide various services to users. This means their essential functions are "observation," "understanding" and "actuation."

The research field on smart environment has been expanding recently (Cook & Das, 2004) and, under the concept of ubiquitous computing, many researchers have developed smart environments for providing informative services to the users (e.g. support during meeting (Johanson, et al., 2002), health care (Nishida et al., 2000), support of the elderly (Mynatt et al., 2004), information display using a pan-tilt projector (Mori et al., 2004)). On the other hand, smart environments are also used for support of mobile robots. Kurabayashi et al. (Kurabayashi et al., 2002) evaluated an efficiency of multi-robot transportation task when the route to the goal is selected by individual mobile robots and by a smart environment (or

specific locations in the environment which can gather information). The formulation and simulation result showed that decision making by the intelligent environment achieved better performance. In (Mizoguchi et al., 1999), document delivery robot system was developed in an office room. When the user sends a request to the system, a delivery robot moves to receive the document either from the user or a handling robot which is located next to the printer. The mobile robot then takes it to the client. During the task, infrared sensors and cameras embedded in the space are used to localize and navigate mobile robots. Another intelligent environment was also developed to perform transportation of heavy items in a hospital (Sgorbissa & Zaccaria, 2004). In the research, distributed beacons are used for mobile robot localization. However, the authors of the paper consider that the environment design is a temporary solution to develop an intelligent robot. We think that the support of mobile robots' movement (e.g. localization or path planning) is just one application of intelligent environments. Smart environments and mobile robots have their own advantages, so it is desirable to realize services by their mutual cooperation.

As described above, although various smart spaces are proposed, few researches have focused on both support for mobile robots and human observation. Therefore, in this paper, we aim to develop a mobile robot navigation system which can support and navigate mobile robots based on the observation of human walking.

The rest of this paper is organized as follows. In section 2, we introduce the concept and present the configuration of iSpace. Section 3 describes a method for acquisition of human walking paths and extraction of information from obtained walking paths. In section 4, mobile robot navigation based on human observation is explained. Experimental results are shown in section 5. Finally, conclusion and future work are given in section 6.

2. Intelligent Space

2.1 Concept of Intelligent Space

Fig. 1 shows the concept of Intelligent Space (iSpace), which is a space with many distributed and networked sensors and actuators. In iSpace, not only sensor devices but sensor nodes are distributed in the space because it is necessary to reduce the network load in the large-scale network and it can be realized by processing the raw data in each sensor node before collecting information. We call the sensor node devices distributed in the space DINDs (Distributed Intelligent Network Device). A DIND consists of three basic components: sensors, processors and communication devices. The processors deal with the sensed data and extract useful information about objects (type of object, three dimensional position, etc.), users (identification, posture, activity, etc.) and the environment (geometrical shape, temperature, emergency, etc.). The network of DINDs can realize the observation and understanding of the events in the whole space. Based on the extracted and fused information, actuators such as displays or projectors embedded in the space provide informative services to users.

In iSpace, mobile robots are also used as actuators to provide physical services to the users and for them we use the name mobile agents. The mobile agent can utilize the intelligence of iSpace. By using distributed sensors and computers, the mobile agent can operate without restrictions due to the capability of on-board sensors and computers. Moreover, it can understand the request from people and offer appropriate service to them.

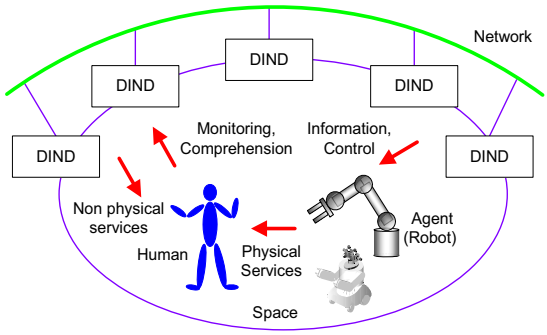


Figure 1. Concept of Intelligent Space

2.2 Configuration of Intelligent Space

Fig. 2 and 3 show a picture and configuration of the implemented iSpace. ISpace is currently implemented in a laboratory environment which has an area of about 5 meters × 5 meters. In this research, six CCD cameras and a 3D ultrasonic positioning system are used as sensors of DIND. The cameras are connected in pairs to computers with two video capture boards. As a result, each camera DIND can get the three dimensional position of objects by stereo vision. The 3D ultrasonic positioning system involves 96 ultrasonic receivers installed on the ceiling. This system can measure the three dimensional position of an ultrasonic transmitter to an accuracy of 20-80 millimeters using triangulation method. Moreover, a differential wheeled robot is used as mobile agent. For estimating the position and orientation of the robot, two ultrasonic transmitters are installed on the top of the mobile robot. The mobile robot is also equipped with a wireless network device to communicate with iSpace.

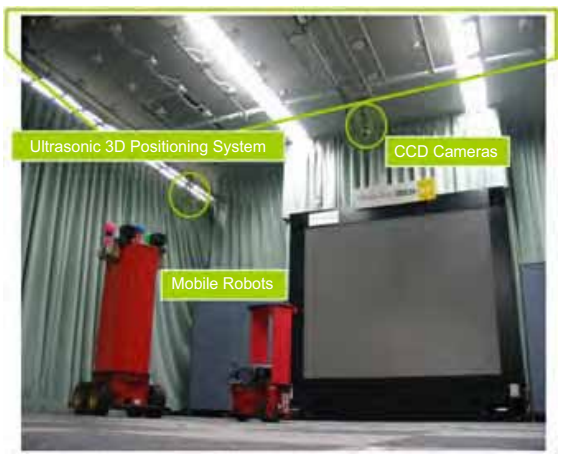


Figure 2. Sensors and mobile agents

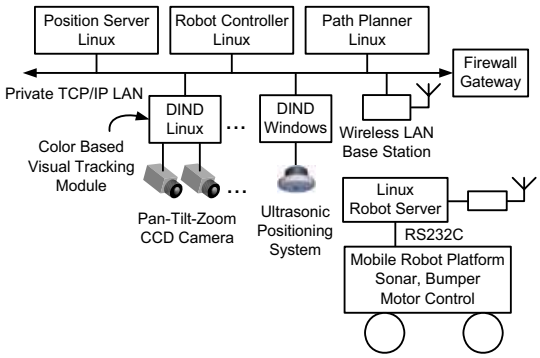


Figure 3. Implemented configuration

3. Observation of Human Walking for Mobile Robot Navigation

3.1 Proposed Method and Related Work

In this research, we focus on human observation for mobile robot navigation since moving though the space is one of the most basic functions for mobile robots. Related to this, some researchers have utilized human walking for mobile robot navigation and control.

In (Appenzeller, 1997), it was proposed that the area where human walks is also traversable for mobile robots and described a system that can generate topological maps for mobile robots by measuring the positions of people in iSpace. The same idea is found in (Tanaka et al., 2003) where mobile robot on-board sensors were used for observation. However, these maps are built based only on the positions of humans. This means that mobile robots can move along a safe path, but since the generated path doesn't reflect the human motion, robot's movement may interfere with humans'.

To the contrary, by avoiding the regions which have a high probability of the presence of people, a path which minimizes the expected travel time (sum of the travel time and the time needed for passing a person on the way) or probability of encountering people was generated in (Kruse & Wahl, 1998). However the authors utilized the observation result of the past and didn't take into account the current positions of people. As a result, mobile robots generate an inefficient path in case that no person exists between the start and the goal point. Furthermore, their motion may be unnatural for humans since the mobile robots move in the area where human doesn't walk. So, some researchers aim to navigate a mobile robot by predicting a future motion of currently tracked human from the history of the observed paths in the environment and changing the motion of the mobile robot only when it is needed (Bennewitz et al., 2005; Foka & Trahanias, 2002; Rennekamp et al., 2006; Vasquez et al., 2004). This approach is efficient because with the obtained path the mobile robots avoid unnecessary contact with people. However, these researches mainly focus on the prediction method and the initial path planning of the mobile robots in human-robot shared space isn't taken into account.

Therefore, in this paper, we consider the method for planning an efficient and natural path which is suitable for mobile robot navigation in a living environment based on observation of human walking.

When a person moves with purpose, the start and the goal point have meaning for the desired action and can be regarded as important points in the space. We also consider that paths frequently used by human are efficient and contain the "rules of the environment." So, we extract important points from the observation and average the human walking paths between two important points to get frequently used paths. The averaged paths are utilized as paths of the mobile robots. By using the important point based paths, mobile robots can choose to explore the parts of space that are meaningful to humans. Furthermore, since such a path is similar to the human chosen path, it is especially useful for robotic guidance applications. By comparing currently observed paths and the frequently used paths, we can combine a motion generation method based on the prediction of the human walking.

3.2 Acquisition of Human Walking Paths

We use vision sensors for tracking so that humans don't have to carry any special devices, e.g. tags for ultrasound system. In the tracking process, the position and field of view of all cameras are fixed. The intrinsic and extrinsic camera parameters are calculated beforehand using a camera calibration method (e.g. (Tsai, 1987; Zhang, 2000)).

In each DIND, human tracking based on background subtraction and color histogram is performed, and the three dimensional position is reconstructed by stereo vision. Then the position information of humans is sent to the position server. The position server also synchronizes the actions of DINDs.

In the position server, fusion of information is done in order to acquire global information about the whole space. Each position sent from DINDs (x_{send}, y_{send}, z_{send}) is compared with positions stored on the server (x_i, y_i, z_i), (i=1, 2, …, n). Let σ_x, σ_y, σ_z, a_x, a_y and a_z be positive constants. If the sent information satisfies

$$|x_{send} - x_i| < \sigma_x \text{ and } |y_{send} - y_i| < \sigma_y \text{ and } |z_{send} - z_i| < \sigma_z, \quad (1)$$

the position information is set to the sent information which has the minimum value of

$$a_x (x_{send} - x_i)^2 + a_y (y_{send} - y_i)^2 + a_z (z_{send} - z_i)^2. \quad (2)$$

In case no stored information satisfies (1), it is recognized as a new object's information. Then the position server creates a new ID and stores the information. The ID assigned to the new tracked human is sent back to the DIND. After that, if the DIND can continue to track the human, the DIND sends the ID as well as position information to the position server. In this case, the position server identifies the object based on ID and (1), and doesn't search all information. If more than one DIND can observe the same human, the mean value is used to determine the position of the human.

To avoid increasing the number of objects stored on the server as time passes, the information of a human who is not detected for a certain period of time (5 seconds in this research) is erased.

A human walking path is generated by projecting the time-series data of a human to the x-y (ground) plane. However human often stays in the same place. Therefore, the tracking system has to determine if the human is walking or not because human never completely stops in such a situation.

In order to do this, we define the absolute value of the velocity in the x-y plane v_{xy}, and x and y components of the mean position x_{mean}, y_{mean} in the past k steps:

$$v_{xy} = \frac{\sqrt{\left(x_{t_{now}} - x_{t_{now}-1}\right)^2 + \left(y_{t_{now}} - y_{t_{now}-1}\right)^2}}{\Delta t}, \tag{3}$$

$$\begin{aligned} x_{mean} &= \frac{1}{k} \sum_{t=t_{now}-k\Delta t}^{t_{now}} x_t, \\ y_{mean} &= \frac{1}{k} \sum_{t=t_{now}-k\Delta t}^{t_{now}} y_t \end{aligned}, \tag{4}$$

where Δt is the sampling rate, x_t and y_t is the position of human at time t in x and y components, respectively, and t_{now} is the current time. If v_{xy} is lower than a given threshold σ_v for k consecutive time steps, the system judges that the human is stationary at (x_{mean}, y_{mean}). Once the static condition is satisfied, the human is considered to stay there until he/she gets more than a certain distance σ_d away from (x_{mean}, y_{mean}).

3.3 Extraction of Frequently Used Paths

The extraction of frequently used paths from the obtained walking paths is done by three steps: 1) extraction of important points, 2) path clustering and 3) path averaging. The reason not to do path clustering directly but to extract important points at the beginning is that path clustering needs appropriate parameters to be set for every situation, which is more difficult than extraction of important points.

First, we explain the extraction of important points. In this research, we define important points as entry/exit points which are useful for mobile robots to move from one area to another, and stop points which are helpful when mobile robots approach humans to provide services. The entry/exit points are extracted based on the points where the tracking system finds new objects or loses objects. On the other hand, the stop points are extracted based on the points where the static condition (section 3.2) is satisfied. These candidates for entry/exit and stop points are grouped by hierarchical clustering and considered as important points if a cluster which consists of many points is formed. We use Euclidean distance in the x-y plane as measure of distance between the points. The clustering process is continued until the distance between clusters exceeds a certain value σ_c because it is hard to determine how many important points are in the environment.

In the next step, for all combinations of two important points, we consider paths which have these points for start and goal points. If there is more than one path that connects the two points path clustering is performed.

We use a hierarchical clustering method based on the LCSS (Longest Common Subsequence) similarity measure (S1 similarity function presented in (Vlachos et al., 2002)). There are several advantages in using this method. First, it can cope with trajectories which have different length, different sampling rates or different speeds. Second, it is robust to noise compared to Euclidean distance or DTW (Dynamic Time Warping) distance. Third, it can be calculated efficiently by using a dynamic programming algorithm.

Let A and B be two trajectories with n and m data points respectively, that is $A = ((a_{x,1}, a_{y,1}), \ldots, (a_{x,n}, a_{y,n}))$, $B = ((b_{x,1}, b_{y,1}), \ldots, (b_{x,m}, b_{y,m}))$. The LCSS models measure the similarity between A and B based on how many corresponding points are found in A and B. Similar to DTW method this model allows time stretching so the points which has close spatial position and the order in the path can be matched. The best match obtained under the

condition that the rearranging of the order of the points is prohibited is used for calculation of the similarity. This is formulated as follows. Let *Head*(*A*) and *Head*(*B*) be trajectories with *n-1* and *m-1* data points expressed as *Head*(*A*) = (($a_{x,1}$, $a_{y,1}$), …, ($a_{x,n-1}$, $a_{y,n-1}$)), *Head*(*B*)= (($b_{x,1}$, $b_{y,1}$), …, ($b_{x,m-1}$, $b_{y,m-1}$)). Given an integer δ (parameter of time stretching) and a real number ε (threshold for matching two values), $\mathrm{LCSS}_{\delta,\varepsilon}(A, B)$ is defined as:

$$\mathrm{LCSS}_{\delta,e}(A,B)=\begin{cases}0 \\ \quad (A \text{ or } B \text{ is empty}) \\ 1+\mathrm{LCSS}_{\delta,e}(Head(A),Head(B)) \\ \quad \left(|a_{x,n}-b_{x,m}|<\varepsilon \text{ and } |a_{y,n}-b_{y,m}|<\varepsilon \text{ and } |n-m|\le\delta\right) \\ \max\left(\mathrm{LCSS}_{\delta,e}(Head(A),B),\mathrm{LCSS}_{\delta,e}(A,Head(B))\right) \\ \quad (\text{otherwise})\end{cases}. \tag{5}$$

The ratio of the number of corresponding point to the number of points in the shorter path is defined as the similarity. As the similarity has a value of 0 (dissimilar) to 1 (similar), the distance between A and B is defined as *1-(similarity)*. So a distance function $D(\delta, \varepsilon, A, B)$ is expressed as follows:

$$D(\delta,\varepsilon,A,B)=1-\frac{\mathrm{LCSS}_{\delta,\varepsilon}(A,B)}{\min(n,m)}. \tag{6}$$

Using the distance function (6), clustering of paths can be performed.
Finally, clustered paths are averaged to extract frequently used paths in the environment. An averaged trajectory is derived from corresponding points between two trajectories, which can be obtained from the LCSS similarity measure. The middle point of corresponding points is used to acquire averaged paths.

4. Mobile Robot Navigation Based on Observation of Humans

4.1 Model of the Mobile Robot

We consider a two-wheeled mobile robot model shown in Fig. 4. Let O-^{w}x^{w}y be the coordinate system fixed to iSpace (world coordinate system) and C-^{R}x^{R}y be the coordinate system fixed to the mobile robot (robot coordinate system). The position and orientation of the mobile robot are denoted by (x, y, θ) in world coordinate system. The control inputs for the mobile robot are the translational velocity v and rotational velocity ω. Here, the kinematic model for the mobile robot is expressed as follows:

$$\begin{bmatrix}\dot{x}\\ \dot{y}\\ \dot{\theta}\end{bmatrix}=\begin{bmatrix}\cos\theta & 0\\ \sin\theta & 0\\ 0 & 1\end{bmatrix}\begin{bmatrix}v\\ \omega\end{bmatrix}. \tag{7}$$

In addition, two ultrasonic transmitters used with the ultrasonic positioning system are installed on the mobile robot. Their coordinates in the robot coordinate system are (L_1, 0), (-L_2, 0).

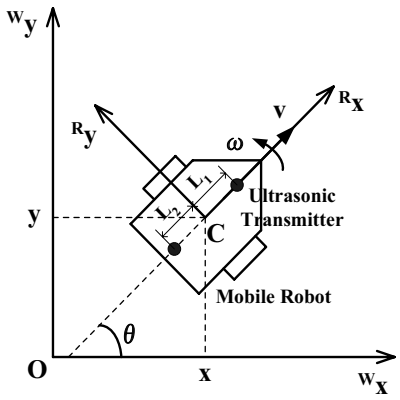


Figure 4. Model of a mobile robot

4.2 Navigation System

Fig. 5 shows the mobile robot navigation system. This system consists of the position server, the robot controller and the path planner. As shown in Fig. 3, each module is connected through the TCP/IP communication network. These modules are described below.

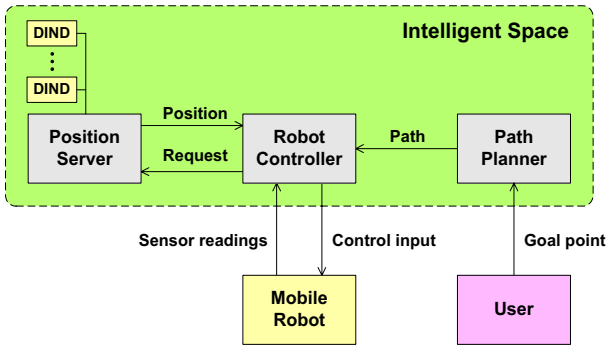


Figure 5. Navigation system

1. Position Server: The position server stores the position information of the mobile robot obtained by DINDs. Unlike in the case of human, ultrasonic transmitters can be installed on the mobile robot in advance. Therefore, the position of the mobile robot is measured by the 3D ultrasonic positioning system.
2. Robot Controller: The robot controller estimates the position and orientation of the mobile robot based on data from iSpace (3D ultrasonic positioning system) and mobile robot (wheel encoder). The dead reckoning method is frequently used to determine the position of the mobile robot. However, it has cumulative error because of slipping motion of wheels. On the other hand, localization using the 3D ultrasonic positioning

system shows high accuracy, but it suffers from errors, such as failure to receive the ultrasonic wave from the transmitter. So, those two measurement data are fused using EKF (Extended Kalman Filter) to minimize the position error.
In order to implement the EKF, the model of the system has to be developed. Discretizing (7), we obtain the following state equation:

$$\begin{bmatrix} x_{\mathrm{k}} \\ y_{\mathrm{k}} \\ \theta_{\mathrm{k}} \end{bmatrix} = \begin{bmatrix} x_{\mathrm{k}-1} + v\Delta t \cos\theta_{\mathrm{k}-1} \\ y_{\mathrm{k}-1} + v\Delta t \sin\theta_{\mathrm{k}-1} \\ \theta_{\mathrm{k}-1} + \omega\Delta t \end{bmatrix} + \mathbf{W}_{\mathrm{k}} w_{\mathrm{k}}, \tag{8}$$

where x_{k}, y_{k} and θ_{k} denote position and orientation of the mobile robot at time k, Δt is the sampling rate, v and ω are the translational velocity and the rotational velocity obtained from encoders, respectively. See (Welch & Bishop, 1995) for other symbols.
The observation equation is expressed as follows:

$$\begin{bmatrix} x_{\mathrm{zps}} \\ y_{\mathrm{zps}} \end{bmatrix} = \begin{bmatrix} x_{\mathrm{k}} + L\cos\theta_{\mathrm{k}} \\ y_{\mathrm{k}} + L\sin\theta_{\mathrm{k}} \end{bmatrix} + \mathbf{V}_{\mathrm{k}} v_{\mathrm{k}}, \tag{9}$$

where (x_{zps}, y_{zps}) is the position of the ultrasonic transmitter in world coordinate system, and L equals L_1 or $-L_2$ depending whether the signal is from the front or rear transmitter.
Linearizing the state equation, Jacobian matrix $\mathbf{A}_{\mathrm{k}}$ is obtained:

$$\mathbf{A}_{\mathrm{k}} = \begin{bmatrix} 1 & 0 & -v\Delta t \sin\theta_{\mathrm{k}-1} \\ 0 & 1 & v\Delta t \cos\theta_{\mathrm{k}-1} \\ 0 & 0 & 1 \end{bmatrix}. \tag{10}$$

We consider that the noise on the encoder is white noise with a normal distribution. Here, Jacobian matrix $\mathbf{W}_{\mathrm{k}}$ is expressed as follows:

$$\mathbf{W}_{\mathrm{k}} = \begin{bmatrix} -\Delta t \cos\theta_{\mathrm{k}-1} & 0 \\ -\Delta t \sin\theta_{\mathrm{k}-1} & 0 \\ 0 & -\Delta t \end{bmatrix}. \tag{11}$$

From the observation equation, Jacobian matrix $\mathbf{H}_{\mathrm{k}}$ is

$$\mathbf{H}_{\mathrm{k}} = \begin{bmatrix} 1 & 0 & -L\sin\theta_{\mathrm{k}} \\ 0 & 1 & L\cos\theta_{\mathrm{k}} \end{bmatrix}. \tag{12}$$

Jacobian matrix $\mathbf{V}_{\mathrm{k}}$ is determined as follows:

$$\mathbf{V}_{\mathrm{k}} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}. \tag{13}$$

In this research, we assume the process noise covariance $\mathbf{Q}$ and measurement noise covariance $\mathbf{R}$ are constant and use diagonal matrices. The values are tuned experimentally.

In the beginning of the experiment, using the ultrasonic positioning system measurement data the initialization process is done:

$$\begin{bmatrix} x_1 \\ y_1 \\ \theta_1 \end{bmatrix} = \begin{bmatrix} \dfrac{L_2 x_{zps1} + L_1 x_{zps2}}{L_1 + L_2} \\ \dfrac{L_2 y_{zps1} + L_1 y_{zps2}}{L_1 + L_2} \\ \text{atan2}\left(y_{zps1} - y_{zps2}, x_{zps1} - x_{zps2}\right) \end{bmatrix}, \tag{14}$$

where (x_{zps1}, y_{zps1}) and (x_{zps2}, y_{zps2}) are the positions of the front and rear transmitters, and atan2(·) denotes the four-quadrant inverse tangent function. After that, estimation is done using the EKF equations.

In addition, the robot controller controls the mobile robot along the paths generated by the path planner. In this research, we use the control law based on the dynamic feedback linearization. The dynamic compensator is given by (See (Oriolo et al., 2002) in detail):

$$\begin{aligned} \dot{\xi} &= u_1 \cos\theta + u_2 \sin\theta \\ v &= \xi \\ \omega &= \frac{u_2 \cos\theta - u_1 \sin\theta}{\xi} \end{aligned} \tag{15}$$

Given a desired smooth trajectory $(x_d(t), y_d(t))$, u_1 and u_2 are given as follows:

$$\begin{aligned} u_1 &= \ddot{x}_d + k_{p1}(x_d - x) + k_{d1}(\dot{x}_d - \dot{x}) \\ u_2 &= \ddot{y}_d + k_{p2}(y_d - y) + k_{d2}(\dot{y}_d - \dot{y}) \end{aligned}, \tag{16}$$

where k_{p1}, k_{p2}, k_{d1} and k_{d2} are positive constants.

3. Path Planner: The path planner generates the path which connects two important points. But the averaged path is not always suitable for mobile robots because it may consist of points aligned at irregular intervals or windingly. Therefore, the path planner extracts the significant points on the averaged path using the method shown in (Hwang et al., 2003) and interpolates them by cubic B-spline function.

5. Experiment

5.1 Experiment of Acquisition of Human Walking Paths

In the environment shown in Fig. 6, human walking paths are obtained. The observable area of each DIND on the ground plane is also shown in this figure. The arrangement of DIND is determined in order to make the observable region as large as possible.

Human walking paths obtained by the tracking system are shown in Fig. 7. We set the parameters in (1) and (2) to σ_x=σ_y=0.3m, σ_z=0.5m, a_x=a_y=1 and a_z=0.25. The objects that were observed outside of the experimental environment or vanished within 1 second since their appearance were ignored as noises. The parameters to determine the stop state are defined

by σ_v=0.3m/s, k=20 and σ_d=0.5m. This figure also shows some broken paths at the edges of the environment. These results were influenced by the observable region of DIND.

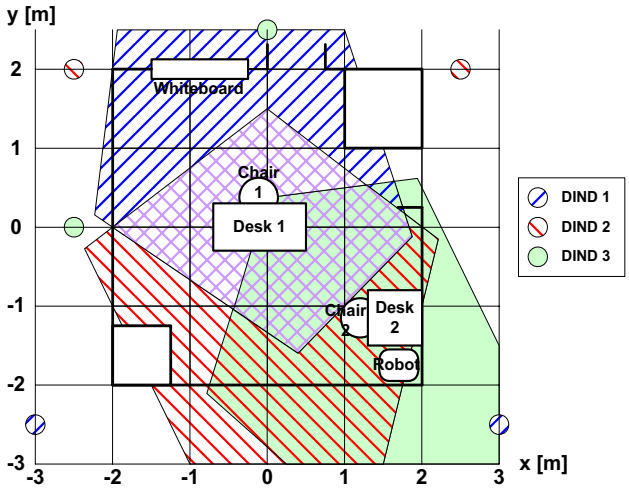


Figure 6. Experimental environment

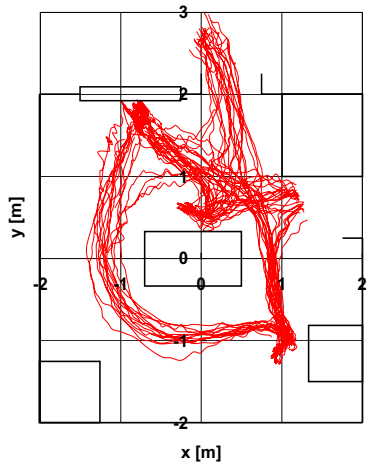


Figure 7. Obtained human walking paths

5.2 Experiment of Extraction of Frequently Used Paths

First of all, important points are extracted to obtain frequently used paths. Important points are defined by clusters including the points over 15% of the total at σ_c=0.5m in both cases of

entry/exit points and stop points. In addition, the distance between clusters is updated by using the centroid method. Fig. 8 and Fig. 9 show the results of clustering. In these figures, the clusters extracted as important points are indicated with ellipses.
In either case, the clusters that consisted of more than 30% of the total number of points were formed and important points were extracted. However, there were some points around Desk 1 that appeared as candidates for entry/exit points because of tracking interrupts. The results were caused by unobservable occlusions because the flow of people behind the Desk 1 was very intense. In order to solve the problem, in our future work entry/exit points will not be determined by extraction based on clustering but preset based on configuration of the space.

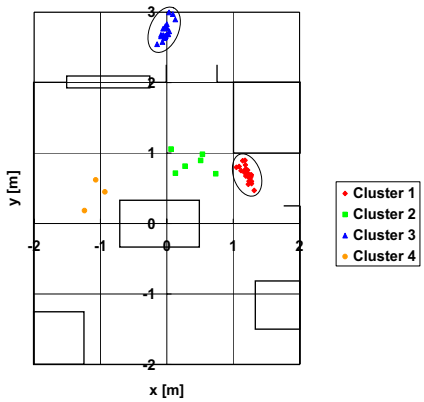


Figure 8. Extraction of entry/exit points

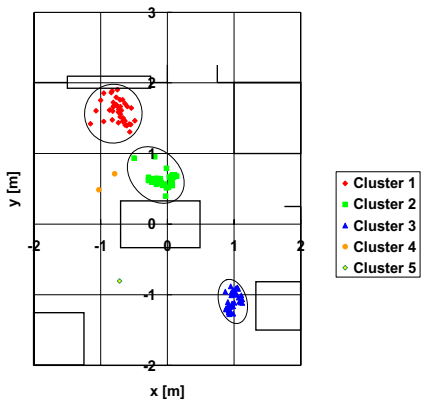


Figure 9. Extraction of stop points

After the extraction of important points, the walking paths between important points are clustered and averaged. Clustering parameters are defined by δ=10, ε=0.3 in every case and the process is continued until the minimum distance between clusters is over 0.4. Moreover, the distance between clusters is recalculated using the furthest neighbor method.
Fig. 10 (left) shows the result of averaging for the path from the important point around Desk 2, as shown in Cluster 3 of Fig. 9, to the entry/exit as shown in Cluster 3 in the upper part of Fig. 8. Five walking paths are obtained and the averaged path is positioned approximately in the center of them. On the other hand, Fig. 10 (right) shows the result of averaging for the path from the important point around the whiteboard as shown in Cluster 1 of Fig. 9 to the important point around Desk 2 as shown in Cluster 3 of Fig. 9, respectively. Nine walking paths are acquired and two paths are obtained by averaging. If the paths were calculated using the mean value, the obtained path would be located in the center of them, which means it would collide with Desk 1. The proposed method distinguishes the whole paths by using path clustering so that it is able to average clustered paths independently.

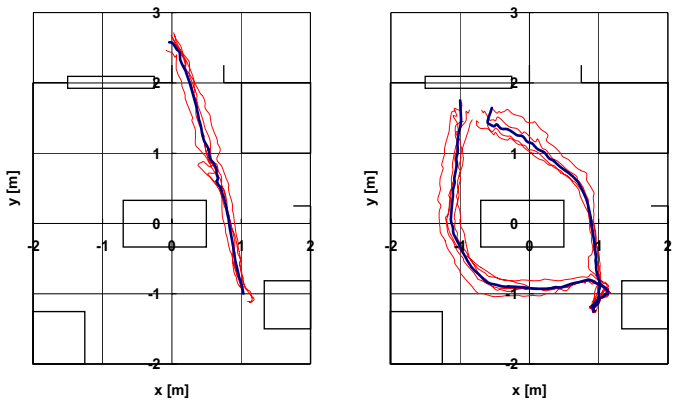


Figure 10. Examples of averaging

Fig. 11 shows all the averaged paths. We can observe that paths between important points were obtained. Moreover, between almost every pair of important points two similar paths were obtained. Pairs of paths were obtained because the direction in which the humans walked was taken into account. By including the direction information, some important details about the environment can be extracted, e.g. the robot should keep to the right side here, this street is one-way, etc.

5.3 Experiment of Mobile Robot Navigation

In this experiment, the mobile robot moves from the entry/exit as shown in Cluster 1 in the right part of Fig. 8 to the entry/exit as shown in Cluster 3 in the upper part of Fig. 8 through important point around whiteboard (Cluster 1 in Fig. 9) and important point around Desk 2 (Cluster 3 in Fig. 9). The experimental result is shown in Fig. 12. The mobile robot followed the paths generated by the path planner and reached the goal point successfully.

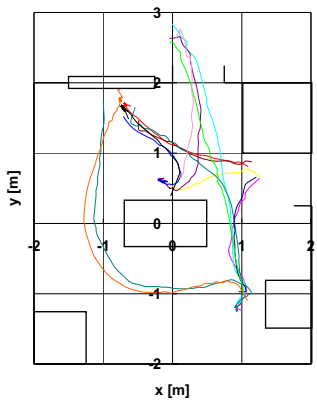


Figure 11. Averaged walking paths

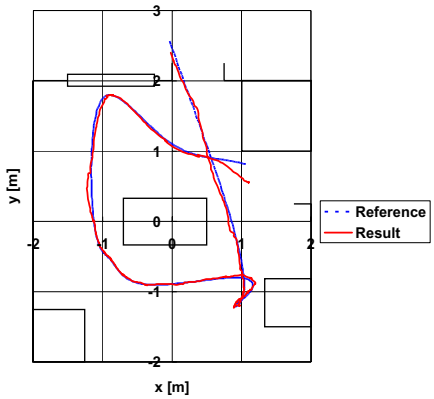


Figure 12. Result of the mobile robot navigation

6. Conclusion

In this paper, we propose that environmental design for mobile robots and human observation are important tasks to develop robots for daily life, and that utilization of many intelligent devices embedded in the environment can realize both of these tasks. As an illustration, we investigate a mobile robot navigation system which can localize the mobile robot correctly and navigate based on observation of human walking in order to operate in the human shared space with minimal disturbance to humans. The human walking paths are obtained from a distributed vision system and frequently used paths in the environment are extracted. The mobile robot navigation based on observation of human is also performed with the support of the system. The position and orientation of the mobile robot are estimated from wheel encoder and 3D ultrasonic positioning system measurement data

using extended Kalman filter. The system navigates the mobile robot along the frequently used paths by tracking control.
For future work, we will develop a path replanning and speed adjustment method based on the current positions of the people. Furthermore, we will apply iSpace to a larger area. Another research direction is to expand this framework into other applications such as human-robot communication, object manipulation and so on. In this paper, the mobile robot doesn't have any external sensors and it is fully controlled by the space. But an intelligent mobile robot can carry out observation and provide iSpace with additional information. This means it behaves as a mobile sensor as well as an actuator. Cooperation between mobile robots and iSpace should also be considered to get more detailed information about human and environment.

7. References

Appenzeller, G.; Lee, J-H. & Hashimoto, H. (1997). Building topological maps by looking at people: An example of cooperation between Intelligent Spaces and robots, *Proceedings of the 1997 IEEE/RSJ International Conference on Intelligent Robots and Systems,* Vol.3, pp.1326-1333, ISBN 0-7803-4119-8, Grenoble, France, Sep. 1997.

Bennewitz, M.; Burgard, W.; Cielniak, G. & Thrun, S. (2005). Learning motion patterns of people for compliant robot motion, *The International Journal of Robotics Research,* Vol.24, No.1, (Jan. 2005) pp.31-48, ISSN 0278-3649.

Cook, D. J. & Das, S. K. (2004). *Smart Environments: Technologies, Protocols, and Applications (Wiley Series on Parallel and Distributed Computing)*, Wiley-Interscience, ISBN 0-471-54448-7, USA.

Foka, A. & Trahanias, P. (2002). Predictive autonomous robot navigation, *Proceedings of the 2002 IEEE/RSJ International Conference on Intelligent Robots and Systems,* Vol.1, pp.490-495, ISBN 0-7803-7398-7, Lausanne, Switzerland, Sep.-Oct. 2002.

Hwang, J-H.; Arkin, R. C. & Kwon, D-S. (2003). Mobile robots at your fingertip: Bezier curve on-line trajectory generation for supervisory control, *Proceedings of the 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems,* Vol.2, pp.1444-1449, ISBN 0-7803-7860-1, Las Vegas, Nevada, USA., Oct. 2003.

Johanson, B.; Fox, A. & Winograd, T. (2002). The Interactive Workspaces project: experiences with ubiquitous computing rooms, *IEEE Pervasive Computing,* Vol.1, No.2, (Apr.-Jun. 2002) pp.67-74, ISSN 1536-1268.

Kruse, E. & Wahl, F.M. (1998). Camera-based monitoring system for mobile robot guidance, *Proceedings of the 1998 IEEE/RSJ International Conference on Intelligent Robots and Systems,* Vo.2, pp.1248-1253, ISBN 0-7803-4465-0, Victoria, BC, Canada, Oct. 1998.

Kurabayashi, D.; Kushima, T. & Asama, H. (2002). Performance of decision making: individuals and an environment, *Proceedings of the 2002 IEEE/RSJ International Conference on Intelligent Robots and Systems,* Vol.3, pp.2831-2836, ISBN 0-7803-7398-7, Lausanne, Switzerland, Sep.-Oct, 2002.

Lee, J-H. & Hashimoto, H. (2002). Intelligent Space - concept and contents, *Advanced Robotics,* Vol.16, No.3, (Apr. 2002) pp.265-280, ISSN 0169-1864.

Mizoguchi, F.; Ohwada, H.; Nishiyama, H. & Hiraishi, H. (1999). Smart office robot collaboration based on multi-agent programming, *Artificial Intelligence,* Vol.114, No.1-2, (Oct. 1999) pp.57-94, ISSN 0004-3702.

Mori, T.; Hayama, N.; Noguchi, H. & Sato, T. (2004). Informational support in distributed sensor environment sensing room, *Proceedings of 13th IEEE International Workshop on Robot and Human Interactive Communication,* pp.353-358, ISBN 0-7803-8570-5, Kurashiki, Japan, Sep. 2004.

Mynatt, E.D.; Melenhorst, A.-S.; Fisk, A.-D. & Rogers, W.A. (2004). Aware technologies for aging in place: understanding user needs and attitudes, *IEEE Pervasive Computing,* Vol.3, No.2, (Apr.-Jun. 2004) pp.36-41, ISSN 1536-1268.

Nishida, Y.; Hori, T.; Suehiro, T. & Hirai, S. (2000). Sensorized environment for self-communication based on observation of daily human behavior, *Proceedings of the 2000 IEEE/RSJ International Conference on Intelligent Robots and Systems,* Vol.2, pp.1364-1372, ISBN 0-7803-6348-5, Takamatsu, Japan, Nov. 2000.

Oriolo, G.; De Luca, A. & Vendittelli, M. (2002). WMR control via dynamic feedback linearization: design, implementation, and experimental validation, *IEEE Transaction on Control Systems Technology,* Vol.10, No.6, (Nov. 2002) pp.835-852, ISSN 1063-6536.

Rennekamp, T.; Homeier, K. & Kroger, T. (2006). Distributed sensing and prediction of obstacle motions for mobile robot motion planning, *Proceedings of the 2006 IEEE International Conference on Intelligent Robots and Systems,* pp.4833-4838, ISBN 1-4244-0259-X, Beijing, China, Oct. 2006.

Sgorbissa, A & Zaccaria, R. (2004). The artificial ecosystem: a distributed approach to service robotics, *Proceedings of the 2004 IEEE International Conference on Robotics and Automation,* Vol.4, pp. 3531- 3536, ISBN 0-7803-8232-3, New Orleans, LA, USA, Apr.-May 2004.

Tanaka, K.; Okada, N. & Kondo, E. (2003). Building a floor map by combining stereo vision and visual tracking of persons, *Proceedings of the 2003 IEEE International Symposium on Computational Intelligence in Robotics and Automation,* Vol.2, pp.641-646, ISBN 0-7803-7866-0, Kobe, Japan, Jul. 2003.

Tsai, R. Y. (1987). A versatile camera calibration technique for high-accuracy 3D machine vision metrology using off-the-shelf TV cameras and lenses, *IEEE Journal of Robotics and Automation,* Vol.3, No.4, (Aug. 1987) pp.323-344, ISSN 0882-4967.

Vasquez, D.; Large, F.; Fraichard, T. & Laugier, C. (2004). High-speed autonomous navigation with motion prediction for unknown moving obstacles, *Proceedings of the 2004 IEEE/RSJ International Conference on Intelligent Robots and Systems,* Vol.1, pp.82-87, ISBN 0-7803-8463-6, Sendai, Japan, Sep.-Oct. 2004.

Vlachos, M.; Kollios, G. & Gunopulos, D. (2002). Discovering similar multidimensional trajectories, *Proceedings of 18th International Conference on Data Engineering,* pp.673-684, ISBN 0-7695-1531-2, San Jose, CA, USA, Feb.-Mar. 2002.

Welch, G. & Bishop, G. (1995). A*n introduction to the Kalman filter, Technical Report* 95-041, Univ. of North Carolina at Chapel Hill, Dept. of Computer Science, 1995.

Zhang, Z. (2000). A flexible new technique for camera calibration, *IEEE Transaction on Pattern Analysis and Machine Intelligence,* Vol.22, No.11, (Nov. 2000) pp.1330-1334, ISSN 0162-8828.

18

Semiotics and Human-Robot Interaction

João Sequeira and M.Isabel Ribeiro
Institute Superior Técnico / Institute for Systems and Robotics
Portugal

1. Introduction

Creating artificial creatures capable of interacting with human beings, following standard social conventions and breaking them as humans do, is part of the technological expression of mankind. Around the 17th century renowned craftsman started producing mechanical automata with behavioral capabilities that imitated basic human skills, mainly related to locomotion and manipulation. A multitude of fictional robots were developed by science fiction authors since the early 20th century, most of them exhibiting behavioral capabilities far ahead of what science and technology would allow. Since then robots populate collective imagination and technological societies established as unconscious goal developing robots aiming at obtaining a human alter ego.

In addition to strict intelligence, a key feature of human beings, robotics also targets human like physical interaction properties such as locomotion. After a century of scientific research it seems clear that achieving intelligence in robotics requires mastering cognition, learning, reasoning, and physical interaction techniques. The Turing test to assess the intelligence of a generic machine can also be used to assess the intelligence of a robot. If human can be deceived by a robot, in a dialogue and also in a physical interaction with the environment, through locomotion or manipulation, then it passes the test. As a complementary argument, the way the interaction is performed might influence whether or not the robot qualifies as intelligent. For example, a robot can avoid obstacles in different manners, according to the environment conditions, and induce different perceptions in a human watching the motion. In the end, an intelligent robot must interact with an ordinary human being, probably not experienced in what concerns robotics, as if it were a human.

The robotics research community only recently started to pay attention to human robot interaction (HRI) as an independent research area. Besides the pure research interest, mass applications, both socially and economically relevant, are being envisaged for robots, namely as home companions, personnal assistants, security agents, office assistants, and generic workers. State of the art humanoid robotics is already capable of simple tasks in factory environments but the interaction abilities are still not up to pass a Turing test.

Generic HRI must assume that humans are inexperienced in what concerns robot motion and hence the interaction techniques robots should use must clone those used by humans among themselves. This suggests that models of human interactions be used to support the research and development of HRI models. In addition, these models might support human like schemes for interaction among robots themselves hence avoiding having to consider separate competences for each type of interaction.

The chapter reviews a number of topics directly related to HRI and describes a research effort to develop a HRI model inspired in semiotics concepts developed in linguistics to model interactions among humans. A set of experiments is presented to illustrate the ideas developed.

2. A brief overview of HRI related research

HRI evolved from the human-machine and human-computer interaction. Often it has been reduced to the design of interfaces aiming at optimizing specific performance indices. Nowadays, typical human strategies used to convey information, such as expressing emotions and specifying intentions through motion, are being also addressed in HRI research.

As an example, the design of keyboards is often subject to studies to optimize usability measures such as the time required by a human to type a benchmark sequence of keys, (Carroll, 2003). Interface design techniques have also rely on the study of maps of human thought obtained by cognitive psychology, (Raskin, 2000). Interfacing tools are always present in a robot control architecture though its synthesis does not aim directly at simplifying the interaction between robots and humans. Usability has been studied in (Ryu and Lee, 2006) in the context of map based interfaces. An agent based architecture for HRI based on an adaptive graphical interface is described in (Kawamura et al., 2003). The robot agent provides the human with the necessary information on the robot and environment. A commander agent maintains a model of the user that is used to decide the message forwarding policy from the human to the appropriate robot. A paradigm in which robots and humans cooperate through the abiliy to recognize emotions is described in (Rani and Sarkar, 2004). Universal user friendly human-computer interfaces were addressed in (Savidis, A. and Stephanidis, C., 2004). Physical indicators used in HRI analysis criteria can also be used in decision making, (Dautenhahn and Werry, 2002).

Robot control architectures have always been a key subject in robotics, fostering research work in multiple enabling areas, e.g., sensors, kinematics and dynamics modeling and control, and in specific functionalities, e.g., path planning and following, obstacle avoidance, world mapping, and localisation. During the 80's the concept of behavior gained wide visibility in robotics. The semantic content associated with the behavior concept seemed to indicate that robot missions could be easily specified almost as if using natural language (or, more generically, a natural interface). Despite multiple efforts to create a formal support for this concept[1] this has been an elusive concept in what concerns simplifying HRI. Human-computer interaction models have been used in (Scholtz, 2003) to define HRI models based on a set of roles, such as supervisor, operator and bystander. In a sense, these roles can be identified with the linguistic notion of behavior though they yield only weak guidelines for synthesis.

The research in robot control architectures is huge. Most of the general purpose architectures can be classified somewhere in the span of behavioral and functional models. The first tend to be specified in terms of models of global performance whereas the later use functional blocks to describe goal behaviors. For example, (Ogasawara, 1991) identifies five components in control architectures, namely, percepts, decomposition, strategies, arbitration

[1] A framework including a formal definition of behavior for generic dynamical systems can be found in (Willems, 1991).

and actions. Despite the semantic content identified with the names of the components, at the implementation level there are functional blocks such as map builders, obstacle avoidance and world modeling strategies, and user interfaces. An arbitration block controls the action to be executed. Other examples of single and multiple robot architectures can easily be found in the literature using concepts from artificial intelligence, biology, semiotics and economic trade markets (see for instance (Parker, 1998; Sequeira and M.I. Ribeiro, 2006a; Bias et al., 2005)).

HRI is also a key area in active surveillance systems. The development of interaction strategies that can be used both by robots and humans, reasonably independent of their relative skills, is likely to improve the performance of the systems. The vast majority of the existing commercial surveillance systems rely on three main components, namely (i) networks of fixed sensors covering the perimeter under surveillance, (ii) visual and keyboard interfaces as interaction tools, and (iii) human supervisors to handle contingency situations. When moving to robotics, a critical issue is that the devices must be able to interact with humans unskilled in what concerns robot specific issues, such as kinematics and dynamics.

First generation commercial surveillance systems rely mainly on networks of fixed sensors, e.g., CCTV systems and motion detectors, to acquire and send data directly to human experts. The development of computer vision led to smart cameras able to process images and extract specific features. Image processing techniques for detection and identification of human activities is an area with huge influence in the ability of robotic systems to interact and even socialize with humans. The surveys in (Valera and Velastin, 2005; Hu et al., 2004) identify key issues in image processing related to the surveillance problem, e.g., human identification using biometric data, the use of multiple cameras, and 2D/3D target modelling. An example of a network of portable video sensors is presented in (Kogut et al., 2003) to detect, track and classify moving targets, and gathering information used to control unmanned ground vehicles.

Robots have been employed in commercial surveillance systems mainly as mobile platforms to carry sensors. The PatrolBot (www.mobilerobots.com) is used in the surveillance of buildings like the Victoria Secret's headquarters at Columbus, USA, and in the United Nations building at Geneva, Switzerland. Mostitech (www.mostitech.com), Personal Robots (www.personalrobots.com), and Fujitsu (www.fujitsu.com) currently sell robots for domestic intruder detection (off the shelf video cameras, eventually with pan-tilt capabilities, constitute also simple robots that can be configured to detect intruders).

In military and police scenarios the robots are, in general, teleoperated to gather information on the enemy positions and in explosives ordinance disposal, (Nguyen and Bott, 2000; Everett, 2003). The Robowatch robot (www.robowatch.de) is supervised by a human through a graphic interface and allowed limited autonomy through information from ultrasound, radar, video and infrared sensors. These robots do not aim at cooperative operation with other robots or humans. Upon detecting unexpected events they just signal a human supervisor through the interface. The interfacing techniques are often developed to accomodate technical constraints, e.g., small number of sensors allowed due to power supply constraints, which might limit their usability. In difficult applications, such as bomb disposal, there are learning curves of several months (robot non-holonomy tends to be an issue in such applications).

In some examples with large number of robots, the interfacing aspects become much more relevant as having humans supervising of each of the individual robots may not be feasible. For example, the Centibots project, (Konolidge et al., 2004), aims at deploying a large number of robots in unexplored areas for world mapping, target searching and surveillance tasks. Distributed map building and fault tolerant communications are just two of the functionalities in each robot. The robots are organized hierarchically, in groups, each with a team leader, and are able to exchange data within the limited range of the communications system. There are four types of interaction allowed, null interaction, hypothesis generation, hypothesis testing and coordinated exploration. Basically, this corresponds to a negotiation strategy that controls the exchange of sensor data. Another example is given by the team of miniature robots with onboard cameras in (Rybski et al., 2000) for reconnaissance and surveillance tasks. The limited computational capabilities of these robots require that image processing and the decision processes are done offboard. The control of large groups of robots using loose interaction protocols in a surveillance task is discussed in (Khrisna and Hexmoor, 2003). Besides the common localisation, navigation and collision avoidance modules, the proposed architecture includes functionalities such as social notions, values, cooperative and shared reasoning and intruder detection.

Including social skills in robots is likely to facilitate their integration in human environments. A robot might be instructed to approach groups of people as a mean to demonstrate that it needs to communicate explicitly or just to acquire information on the group. This sort of social behavior matches typically human social behaviors; in some circunstances a single human tends to approach groups of people in order to foster interaction with the other humans in the group. Standard techniques can be used to design behaviors that convey information on the intentions of a robot to the outside environment (see for instance (Nicolescu and Mataric, 2001)). Still, current strategies to describe in a unified way the synthesis and detection/recognition from sensor data of such behaviours, both for humans and robots, do not yield user friendly interfacing.

3. Abstract concepts in HRI modeling

In general, robots and humans work at very different levels of abstraction. Developing new forms of representing human-robot interactions close to those used among humans, e.g., natural interfaces, is likely to yield robotic systems able to perform complex missions that currently can only be accomplished by humans.

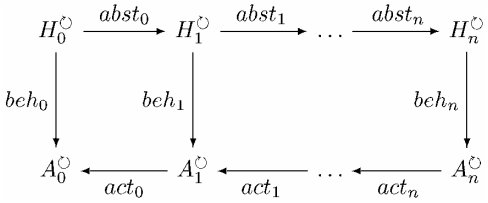
Most of the usual interactions among machines, and robots in particular, are supported on well defined protocols to wrap and transport data. This means that every intervenient knows exactly what to do when it receives data, how to transmit it and what to do with it. Interactions among humans follow different principles. Often, information is exchanged using loosely defined protocols and symbolic information is exchanged either conveying explicit data or wrapping a particular meaning that later it will be inferred by the receiver.

The difficulties in creating a model for linguistic interactions with the above characteristics are obviously immense. Despite the research efforts, dating back to the work of Chomsky, Harman and others, (Chomsky, 1968; Harman, 1969), current natural interfacing tools are still not powerful enough to mimic human natural language capabilities.

Mapping high level abstract concepts into low level concrete objects requires a roadmap, i.e., a set of organizing principles. Category theory (CT) provides a suitable framework to

represent the objects and relations among them. Other than providing deep formal results, CT clarifies these organizing principles.

Diagram 1 represents a model of a hierachy of abstractions (the level of abstraction increases towards the righthand side of the diagram). Each level follows a classical sense-think-act pipeline. The H_i objects represent the data perceived by the robot at each abstraction level. The $abst_i$ functors define the data processing between levels. The beh_i functors represent the decision processes on the perceived data. The A_i objects contain the information that directly affects the motion of the robot. The act_i functors stand for the processes that transform high level information into actuator controls. The circle arrows indicate endofunctors in each of the categories involved.



(1)

At the lowest level of abstraction, H_0 includes objects such as configuration spaces, the A_0 contains the control spaces, and the beh_0 account for low level control strategies, e.g., motor control feedback loops. Coordinate transformations are examples of endomaps in H_0. At the intermediate levels, the H_i can represent data, services, and functionalities such as path planning and world map building algorithms. At the highest level of abstraction, H_n stands for the objects used in natural interactions, that is, information units exchanged during a natural interactions such as those occurring when using natural language. The A_n stands for high level processing of such units. In an implementation perspective, the diagram suggests the use of concurrent beh_i maps, operating in different levels of abstraction and competing to deliver their outputs to actuators. In a sense, it generalizes the well known idea of subsumption architecture.

Abstractions defined through projection maps have been used in the context of dynamical systems to represent hierarchies of models that preserve good properties, e.g., controllability and observability; see for instance (Stankovic and Siljak, 2002) for examples with linear time invariant systems, (Asarin and Maler, 1994) for a definition on discrete event systems, or (Asarin and Dang, 2004) for nonlinear continuous systems. Simulation and bisimulation maps also express notions of equivalence (from an external perspective) between systems; see (van der Schaft, 2004) on linear time invariant systems.

Diagram 1, accounting for multiple decision levels through the beh_i maps, raises interesting robotics problems, namely (i) defining criteria for the number of abstraction levels, and (ii) optimally distributing the information processing through the abstraction levels, i.e., designing the $abst_i$ and act_i maps. The diagram can also be interpreted as relating different perspectives for modeling and controlling robots[2] and hence might represent a unifying structure for robot control architectures. In a sense, each of the abstraction levels represents

[2] This idea of different perspectives to system modeling is common in information systems architectures, see for instance the IEEE 1471 standard, (IEEE, 2000).

a different perspective from where to look to a problem and all the perspectives contribute to the global outcome.

4. A semiotics based HRI model

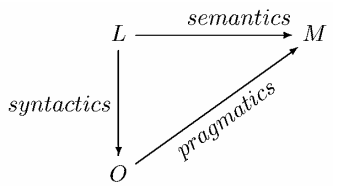
Semiotics is a branch of general philosophy addressing the linguistic interactions among humans. It has been used by several authors mainly to provide guidelines for control architectures synthesis (see for instance (Meystel and Albus, 2002)) and often related to the symbol grounding concept of artificial intelligence. Categorical approaches to semiotics have been proposed in human-computer interaction area. See for instance (Neumuller, 2000) for an application of hypertext theory to world wide web, (Codognet, 2002) for research on machine-machine and human-human interactions over electronic media, and (Malcolm and Goguen, 1998) for an algebraic formulation for semiotics and its use in interface design.

The underlying idea in this chapter is (i) to use the concept of *semiotic sign* as the basis information unit used by humans to interact among themselves, (ii) to establish principles for processing these semiotic signs, and (iii) to define adequate maps between sets of signs and spaces of control variables.

Interaction using natural language is probably the most complete example of interaction among heterogeneous agents using semiotic signs. Semiotic signs evolved mainly from the work of C. S. Pierce and F. Saussure (see for instance (Chandler, 2003; Bignell, 1997)). The concepts developed by Pierce and Saussure differ slightly, with Pierce's model being more flexible that Saussure's model. The signs and morphisms defined among them form *sign systems.*

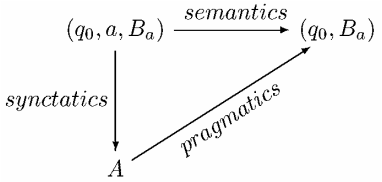
Following Pierce's work, signs can be of three categories, (Codognet, 2002; Malcolm and Goguen, 1998), (i) symbols, expressing arbitrary relationships, such as conventions, (ii) icons, such as images, (iii) indices, as indicators of facts or conditions. Signs defined in these three categories can represent any of the abstract entities, of arbitrary complexity, that are used by humans in linguistic interactions, (Bignell, 1997).

A generic semiotic sign, in any of the three classes above, encapsulates three atomic objects, named *meaning, object,* and *label,* and the relations between them. Under reasonable assumptions on the existence of identity maps, map composition, and composition association, signs can be modeled as a category. Diagram (2) illustrates the sign category, hereafter named SIGNS. A similar representation, known as the "semiotic triangle", is often used in the literature on semiotics (see for instance (Chandler, 2003)). For the sake of completeness it is worth to point that Saussure's model of a sign included only two objects, a signifier, defining, the form the sign takes, and a signified, defining, the concept it represents.



(2)

The *Labels,* (L), represent the vehicle through which the sign is used, *Meanings,* (M), stand for what the users understand when referring to the sign, and *Objects,* (O), stand for the real objects signs refer to. The morphisms are named *semantics,* standing for the map that extracts the meaning of an object, *syntactics,* standing for the map that constructs a sign from a set of syntactic rules, and *pragmatics,* standing for the maps that extract hidden meanings from signs, i.e., perform inference on the sign to extract the meaning.
In the mobile robotics context the objects in a concrete category resulting from SIGNS must be able to represent in a unified way (i) regular information exchanges, e.g., state data exchanged over regular media, and (ii) robot motion information. In addition, they should fit the capabilities of unskilled humans when interacting with the robots, much like a natural language.
A forgetful functor assigns to each object in SIGNS a set that is relevant for the above objectives. The co-domain category is denoted ACTIONS and is defined in Diagram (3),



(3)

where A represents the practical implementation of a semiotic sign, q_0 stands for an initial condition that marks the creation of the semiotic sign, e.g., the configuration of a robot, a stands for a process or algorithm that implements a functionality associated with the semiotic sign, e.g., a procedure to compute an uncertainty measure at q_0, B_a stands for a set in the domain space of a, e.g., a compact region in the workspace. A practical way to read Diagram (3) is to consider the objects in A as having an internal structure of the form (q_0,a,B_a) of which (q_0, B_a) is of particular interest to represent a meaning for some classes of problems.
The *syntactics* morphism is the constructor of the object. It implements the syntactic rules that create an A object. The constructors in object oriented programming languages are typical examples of such morphisms.
The *semantics* morphism is just a projection operator. In this case the semantics of interest is chosen as the projection onto $\{ q_0 \times B_a\}$.
The *pragmatics* morphism implements the maps used to reason over signs. For instance, the meaning of a sign can in general be obtained directly from the behavior of the sign as described by the object A (instead of having it extracted from the label)[3].
Diagram 3 already imposes some structure on the component objects of a semiotic sign. This structure is tailored to deal with robot motion and alternative structures are of course possible. For instance, the objects in ACTIONS can be extended to include additional components expressing events of interest.

[3] This form of inference is named *hidden meaning* in semiotics.

5. From abstract to concrete objects

Humans interact among each others using a mixture of loosely defined concepts and precise concepts. In a mixed human-robot system, data exchanges usually refer to robot configurations, uncertainty or confidence levels, events and specific functions to be associated with the numeric and symbolic data. This means that the objects in ACTIONS must be flexible enough to cope with all these possibilities. In the robot motion context, looseness can be identified with an amount of uncertainty when specifying a path, i.e., instead of specifying a precise path only a region where the path is to be contained is specified, together with a motion trend or intention of motion. To an unskilled observer, this region bounding the path conveys a notion of equivalence between all the paths contained therein and hence it embedds some semantic content.
The objects in ACTIONS span the above caracteristics. For mobile robots the following are two examples.

- The action map *a* represents a trajectory generation algorithm that is applied at some initial configuration q_0 and constrained to stay inside a region B_a, or
- The action map *a* stands for an event detection strategy, from state and/or sensor data, when the robot is at configuration *qo* with an uncertainty given by B_a.

Definition 1 illustrates an action that is able to represent motion trends and intentions of motion.

Definition 1 (ACTIONS) *Let k be a time index, q_0 the configuration of a robot where the action starts to be applied and $a(q_0)|_k$ the configuration at time k of a path generated by action a.*
A free action is defined by a triple (q_0,a,B_a) where B_a is a compact set and the q_0 the initial condition of the action, and verifies,

$$q_0 \in B_a, \tag{4}$$

$$a(q_0)|_0 = q_0, \tag{4b}$$

$$\exists_{\epsilon>\epsilon_{\min}} \ : \ \mathcal{B}(q_0,\epsilon) \subseteq B_a, \tag{4c}$$

with $\mathcal{B}(q_0,\epsilon)$ a ball of radius ϵ centered at q_0, and

$$\forall_{k\geq 0} \ \ a(q_0)|_k \in B_a \tag{4d}$$

Definition 1 establishes a loose form of equivalence between paths generated by *a*, starting in a neighborhood of q_0, and evolving in the bounding region B_a. This equivalence can be fully characterized through the definition of an equality operator in ACTIONS. The resulting notion is more general than the classical notion of simulation (and bisimulation)[4] as the relation between trajectories is weaker. Objects as in Definition 1 are rather general. These objects can be associated with spaces other than configuration spaces and workspaces. Also, it is possible to define an algebraic framework in ACTIONS with a set of free operators that express the motion in the space of actions, (Sequeira and M.I. Ribeiro, 2006b). The

[4] Recall that a simulation is a relation between spaces $\mathcal{X}_1$ and $\mathcal{X}_2$ such that trajectories in both spaces are similar independent of the disturbances in $\mathcal{X}_1$. A bisimulation extends the similarity also to disturbances in X2 (see (van der Schaft, 2004)).

interest of having such a framework is that it allows to determine conditions under which good properties such as controllability are preserved.
A free action verifying Definition 1 can be implemented as in the following proposition.
Proposition 1 (Action) *Let $a(q_0)$ be a free action. The paths generated by $a(q_0)$ are solutions of a system in the following form,*

$$\dot{q} \in F_a(q) \tag{5}$$

where F_a is a Lipschitzian set-valued map with closed convex values verifying,

$$F_a(q) \subseteq T_{B_a}(q) \tag{6}$$

where $T_{B_a}(q)$ is the contingent cone to B_a at q.
The demonstration of this proposition is just a restatement of Theorem 5.6 in (Smirnov, 2002) on the existence of invariant sets for the inclusion (5).
When $\{q\}$ is a configuration space of a robot, points in the interior of B_a have T_{B_a} equal to the whole space. When q is over the boundary of B_a the contingent cone is the tangent space to B_a at q. Therefore, when q is in the interior of B_a it is necessary to constrain $T_{B_a}(q)$ to obtain motion directions that can drive a robot (i) through a path in the interior or over the boundary of B_a, and (ii) towards a mission goal.
In general, bounding regions of interest are nonconvex. To comply with Proposition 1 triangulation procedures might be used to obtain a covering formed by convex elements (the simplicial complexes). Assuming that any bounding region can be described by the union of such simplicial complexes[5], the generation of paths by an action requires that (i) an additional adjustment of admissible motion directions such that the boundary of a simplicial complex can be crossed, and (ii) the detection of adequate events involved, e.g., approaching the boundary of a complex and crossing of the border between adjacent complexes. When in the interior of a simplicial complex, the path is generated by some law that verifies Proposition 1.
The transition between complexes is thus, in general, a nonsmooth process. Figure 1 shows 2D examples that strictly follow the conditions in Proposition 1 with very different behaviors.

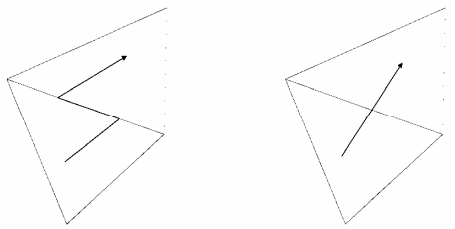


Figure 1. Examples for 2D simplicial complex crossing

[5] See for instance (Shewchuk, J. R., 1998) for conditions on the existence of constrained Delaunay triangulations.

In robotics it is important to ensure that transitions between complexes occur as smoothly as possible in addition to having the paths staying inside the overall bounding region. Proposition 2 states sufficient conditions for transitions avoiding moving over the boundary in adjacent complexes.

Proposition 2 (Crossing adjacent convex elements) *Let a path q be generated by motion directions verifying Proposition 1, and consider two adjacent (i.e., sharing part of their boundaries) simplicial complexes B_1 and B_2 assume that the desired crossing sequence is B_1 to B_2. Furthermore, let $\mathcal{B}(q,\epsilon)$ be a neighbourhood of q with radius e and define the points q_{ci} and sets C_i, i = 1,2 as,*

$$\begin{aligned} q_{c_i} &: \mathcal{B}(q_{c_i},\epsilon_i) \subset B_i, \quad for\ some\ \epsilon_i \\ C_i &= q_{c_i} + \lambda_i\left(\mathcal{B}_j - q_{c_i}\right), \quad \lambda_i \in [0,1] \end{aligned}$$

and let the set of admissible motion directions be defined by

$$T_{B_1 \cup B_2} = \begin{cases} T_{B_1}(q) & if \quad q \in B_1 \backslash C_1 \\ (\mathcal{B}_2 - q_{c_1}) & if \quad q \in B_1 \cap C_1 \\ -(\mathcal{B}_1 - q_{c_2})) & if \quad q \in B_2 \cap C_2 \\ T_{B_2}(q) & if \quad q \in B_2 \backslash C_2 \end{cases} \tag{7}$$

Then the path q crosses the boundary of B_1 and enters B_2 with minimal motion on the border $B_1 \cap B_2$.

The demonstration follows by showing that when *q* is over the border between B_1 and B_2 the motion directions given by $(B_1 \cap B_2) - q$ are not admissible.

Expression (7) determines three transitions. The first transition occurs when the robot moves from a point in the interior of B_1, but outside C_1, to a point inside $B_1 \cap C_1$. The admissible motion directions are then those that drive the robot along paths inside B_1 as if no transition would have to occur. At this event the admissible motions directions drive the robot towards the border between B_1 and B_2 because C_1 is a viability domain as $\mathcal{B}_2 - q_{c_1} \subset T_{C_1}$. The second transition occurs when the robot crosses the border between B_1 and B_2. At this point the admissible motion directions $-(\mathcal{B}_1 - q_{c_2}) \subset T_{C_2}$ and hence the path moves away from the border $B_1 \cap B_2$ towards the interior of B_2. At the third transition the path enters $B_2 \backslash C_2$ and the admissible motion directions yield paths inside B_2.

While $q \in B_1 \cap B_2$ there are no admissible motion directions either in $T_{B1}(q)$ or $T_{B2}(q)$ and hence the overlapping between the trajectory and the border is minimal.

Examples of actions can be easily created. Following the procedure outline above, a generic bounding region in a 2D euclidean space, with boundary defined by a polygonal line, it can be (i) covered with convex elements obtained through the Delaunay triangulation on the vertices of the polygonal line (the simplicial complexes), and (ii) stripped out of the elements which have at least one point outside the region. The resulting covering defines a topological map for which a visibility graph can be easily computed.

Figure 2 shows an example of an action with a polygonal bounding region, defined in a 2D euclidean space with the bounding region covered with convex elements obtained with Delaunay triangulation. The convex elements in green form the covering. The o marks inside each element stand for the corresponding center of mass, used to define the nodes of the visibility graph. The edges of elements that are not completely contained inside the

polygonal region are shown in blue. The red lines represent edges of the visibility graph of which the shortest path between the start and end positions are shown in blue.
Proposition 2 requires the computation of additional points, the q_{ci}. In this simple example it is enough to choose them as the centers of mass of the triangle elements. The neighbourhoods $B(q_{ci}, \varepsilon)$ can simply be chosen as the circles of maximal radius that can be inscribed in each triangle element. The second transition in (7) is not used in this example. The admissible motion directions are simply given by

$$\dot{q}(t) \in \mathcal{B} - q \tag{8}$$

where B stands for the neighborhood in the convex element where to cross, as described above.
The line in magenta represents the trajectory of a unicycle robot, starting with 0 rad orientation. The linear velocity is set to a constant value whereas the angular velocity is defined such as to project the velocity vector onto (8).

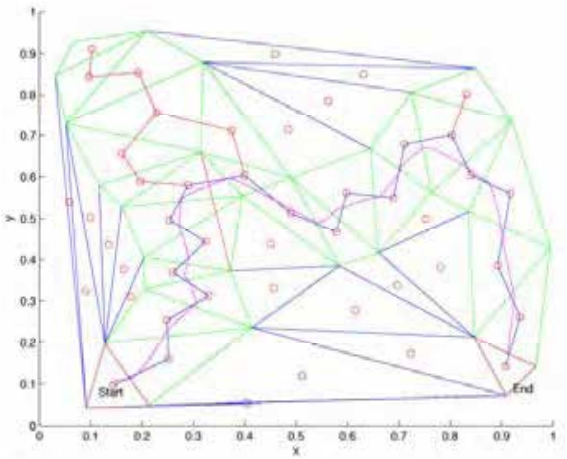


Figure 2. Robot moving inside a bounding region

6. Experiments

Conveying meanings through motion in the context of the framework described in this chapter requires sensing and actuation capabilities defined in that framework, i.e., that robots and humans have (i) adequate motion control, and (ii) the ability to extract a meaning from the motion being observed.
Motion control has been demonstrated in real experiments extensively described in the robotics literature. Most of that work is related to accurate path following. As aforementioned, in a wide range of situations this motion does not need to be completely specified, i.e., it is not necessary to specify an exact trajectory. Instead, defining a region where the robot is allowed to move and a goal region to reach might be enough. The tasks that demonstrate interactions involving robots controlled under the framework described in

this chapter are not different for classical robotics tasks, e.g., reaching a location in the workspace.
The extraction of meanings is primarily related to sensing. The experiments in this area assess if specific strategies yield bounding regions that can be easily perceived by humans and can also be used by robots for motion control.
Two kinds of experiments are addressed in this chapter, (i) using simulated robots, and (ii) using real robots. The former allow the assessment of the ideas previously defined under controlled conditions, namely the assessment of the performance of the robots independently of the noise/uncertainties introduced by the sensing and actuation devices. The later illustrate the real performance.

6.1 Sensing bounding regions

Following Diagram 3, a meaning conveyed by motion lies in some bounding region. The extraction of meanings from motion by robots or humans thus amounts to obtain a region bounding their trajectories. In general, this is an ill-posed problem. A possible solution is given by

$$\begin{aligned} q_0 &= \hat{q}(t-h) \\ B_a(t) &= \cup_t \mathcal{B}(\hat{q}(t), \epsilon) \end{aligned} \tag{9}$$

where $\hat{q}(t)$ is the estimated robot configuration at time t, $\mathcal{B}(\hat{q}(t), \epsilon)$, is a ball of radius ϵ and centered at $\hat{q}(t)$, and h is a time window that marks the initial configuration of the action. This solution bears some inspiration in typically human charateristics. For instance, when looking at people moving there is a short term memory of the space spanned in the image plane. Reasoning on this spanned space might help extrapolating some motion features.
In practical terms, different techniques to compute bounding regions can be used depending on the type of data available. When data is composed of sparse information, e.g., a set of points, clustering techniques can be applied. This might involve (i) computing a dissimilarity matrix for these points, (ii) computing a set of clusters of similar points, (iii) map each of the clusters into adequate objects, e.g., the convex hull, (iv) define the relation between these objects, and (iv) remove any objects that might interfere with the workspace.
Imaging sensors are commonly used to acquire information on the environment. Under fair lighting conditions, computing bounding regions from image data can be done using image subtraction and contour extraction techniques[6]. Figure 3 illustrates examples of bounding regions extracted from the motion of a robot, sampled from visual data at irregular rate. A basic procedure consisting in image subtraction, transformation to grayscale and edge detection is used to obtain a cluster of points that are next transformed in a single object using the convex hull. These objects are successively joined, following (9), with a small time window. The effect of this time window can be seen between frames 3 and 4, where the first object detected was removed from the bounding region.
The height of the moving agent clearly influences the region captured. However, if a calibrated camera is used it is possible to estimate this height. High level criteria and a priori knowledge on the environment can be used to crop it to a suitable bounding region. Lower abstraction levels in control architectures might supsump high level motion commands

[6] Multiple techniques to extract contours in an image are widely available (see for instance (Qiu, L. and Li, L., 1998; Fan, X. and Qi, C. and Liang, D. and Huang, H., 2005)).

computed after such bounding regions that might not be entirely adequate. A typical example would be having a low level obstacle avoidance strategy that overcomes a motion command computed after a bounding region obtained without accounting for obstacles.

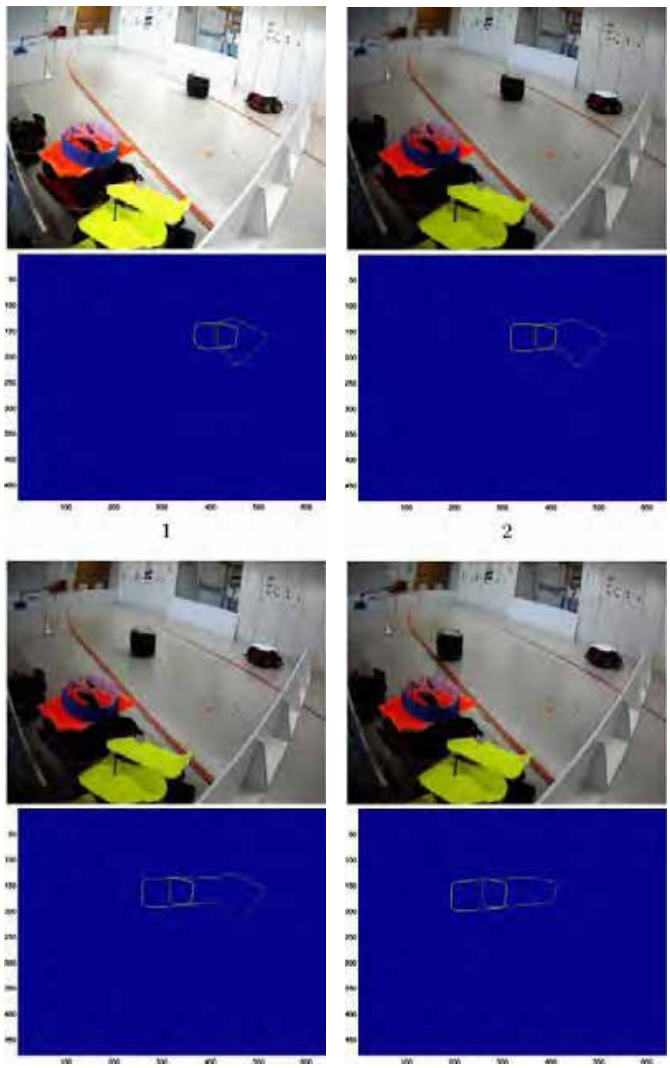


Figure 3. Bounding region extracted from the motion of a robot

6.2 Human interacting with a robot

Motion interactions between humans and robots often occur in feedback loops. These interactions are mainly due to commands that adjust the motion of the robot. While moving the robot spans a region in the free space that is left for the human to extract the respective meaning. A sort of error action is computed by the human which is used to define new goal actions to adjust the motion of the robot.

Map based graphical interfaces are common choices for an unskilled human to control a robot. Motion commands specifying that the robot is to reach a specific region in the workspace can be defined using the framework previously defined, forming a crude language. The interactions through these interfaces occur at sparse instants of time, meaning that the human is not constantly adjusting bounding regions. Therefore, for the purpose of illustrating the interaction under the framework described it suffices to demonstrate the motion when a human specifies a single bounding region.

Figure 4 illustrates the motion of a unicycle robot in six typical indoor missions. For the purpose of this experiment, the robot extracts its own position and orientation from the image obtained by a fixed, uncalibrated, camera mounted on the test scenario[7]. Position is computed after a rough procedure based on color segmentation. Orientation is obtained through the timed position difference. A first order low pass filtering is used to smooth the resulting information. It is worth to point that sophisticated techniques for estimating the configuration of a robot from this kind of data, namely those using a priori knowledge on the robot motion model, are widely available. Naturally, the accuracy of such estimates is higher than the one provided by the method outline above. However, observing human interactions in real life suggests that only sub-optimal estimation strategies are used and hence for the sake of comparison it is of interest to use also a non-optimal strategy. Furthermore, this technique limits the complexity of the experiment.

A Pioneer robot (shown in a bright red cover) is commanded to go to the location of a Scout target robot (held static), using a bounding region defined directly over the same image that is used to estimate the position and orientation. Snapshots 1, 2, 3 and 6 show the Pioneer robot starting in the lefthand side of the image whereas the target robot is placed on the righthand side. In snapshots 4 and 5 the region of the starting and goal locations are reversed.

The blue line shows the edges of the visibility graph that corresponds to the bounding region defined (the actual bounding region was omitted to avoid cumbersome graphics). The line in magenta represents the trajectory executed. All the computations are done in image plane coordinates. Snapshot 5 shows the effect of a low level obstacle avoidance strategy running onboard the Pioneer robot. Near the target the ultrasound sensors perceive the target as an obstacle and force the robot to take an evasive action. Once the obstacle is no longer perceived the robot moves again towards the target, this time reaching a close neighborhood without the obstacle avoidance having to interfere.

---

[7] Localisation strategies have been tackled by multiple researchers (see for instance, (Betke and Gurvits, 1997; Fox, D. and Thrun, S. and Burgard, W. and Dellaert, F., 2001)). Current state of the art techniques involve the use of accurate sensors e.g., inertial sensors, data fusion and map building techniques.

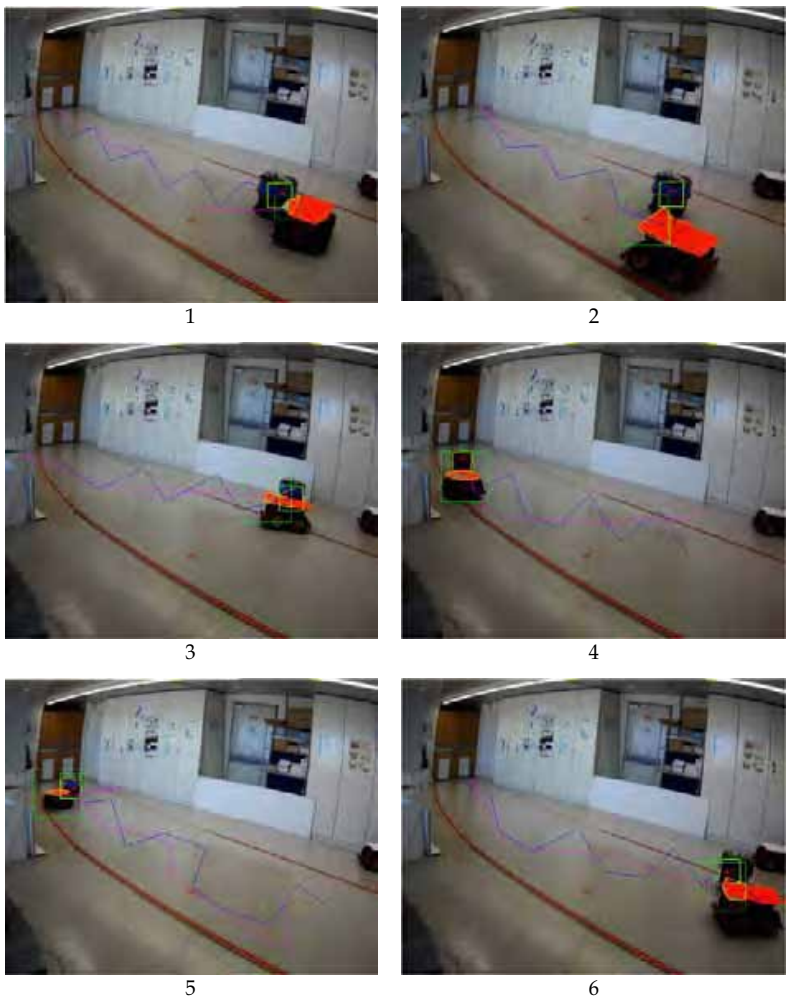


Figure 4. Robot intercepting a static target

6.3 Interacting robots

Within the framework described in this chapter, having two robots interacting using the actions framework is basically the same as having a human and a robot as in the previous section. The main difference is that the bounding regions are processed automatically by each of the robots.

In this experiment a Scout robot is used as a static target while two Pionner 3AT robots interact with each other aiming at reaching the target robot. The communication between the two Pioneer robots is based on images from a single camera. Both robots have access to the same image, from which they must infer the actions the teammates are executing.
Each of the Pioneer robots uses a bounding region for its own mission defined after criteria similar to those used by typical humans, i.e., chooses its bounding regions to complement the one chosen by the teammate.
A bounding region spanned by the target is arbitrarily identified by each of the chasers (and in general they do not coincide). Denoting by B_1 and B_2 the regions being used by the chasing robots in the absence of target and by B_{g_1} and B_{g_2} regions where the target robot was identified by each of them the bounding region of each of the chasers is simple $B_i' = B_i \cup B_{g_i}$. If shortest routes between each of the chasers and the target are required then it suffices to make the $B_i = q_i + \lambda (B_{g_i} - q_i),\ \lambda \in [0,1]$.
The two bounding regions, B_1' and B_2', overlap around the target region.
The inclusion of the B_{g_i} aims at creating enough space around the target such that the chaser robots can approach the target without activating their obstacle avoidance strategies. Figure 5 shows two simulations of this problem with unicycle robots. The target location is marked with a yellow $*$. In the lefthand image the target is static whereas in the righthand side one uniform random noise was added both to the target position and to the B_{g_i} areas. In both experiments the goal was to reach the target within a 0.1 distance.

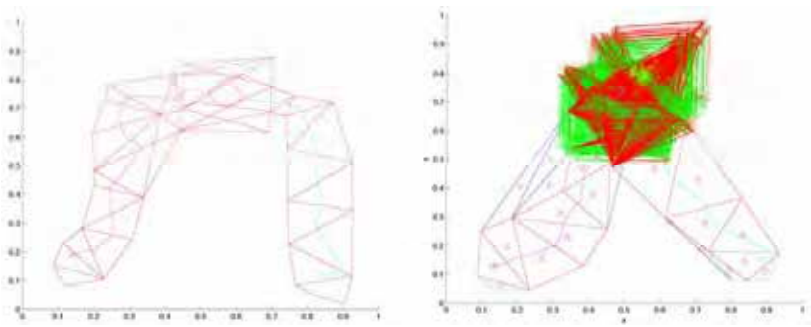


Figure 5. Intercepting an intruder

Figure 6 shows a sequence of snapshots obtained in three experiments with real robots and a static target. These snapshots were taken directly from the image data stream being used by the robots. The trajectories and bounding regions are shown superimposed.
It should be noted that the aspects related to robot dynamics might have a major influence in the results of these experiments. The framework presented can be easily adjusted to account for robot dynamics. However, a major motivation to develop this sort of framework is to be able to have robots with different functionalities, and often uncertain, dynamics interacting. Therefore, following the strategy outlined in Diagram 1, situations in which a robot violates the boundary of a bounding region due, for example, to dynamic constraints can be assumed to be taken care by lower levels of abstraction.

Figure 6. Intercepting an intruder

7. Conclusions

The social barriers that still constrain the use of robots in modern societies will tend to vanish with the sophistication increase of interaction strategies. Communication and interaction between people and robots occurring in a friendly manner and being accessible to everyone, independent of their skills in robotics issues, will certainly foster the breaking of barriers.

Socializing behaviors, such as following people, are relatively easy to obtain with current state of the art robotics. However, achieving human like interaction through motion (as

people do) requires the development of alternative models to synthesize behaviors for robots. The framework outlined in this chapter shows how models of human interactions from social sciences, can be merged with robot control techniques to yield a set of objects that simplifies the development of robotics applications.

The experiments presented demonstrate interactions involving humans and robots similar to those arising in classical approaches. Even though these similarities, for example measured through the visual quality of trajectories generated by the robots, the effort to develop these experiments was only a fraction of the effort that a classical approach would have cost. Furthermore, the results show that robots can operate and interact both among themselves and with people, with significant quality, in poorly modeled environments. The experiments were designed for minimal technological requirements, hence avoiding shadowing the performance of the framework described.

The discussion on how to make a concrete object out of an abstract concept, such as meaning, might lead to other alternative frameworks. The one described in this chapter priviledges the locomotion features that characterizes a robot, namely by using as support space the configuration space. Still, multiple extensions can be made out of the ideas developed. A virtual agent might require additional components in the objects in SIGNS or even alternative support spaces, for instance to simplify reasoning processes.

As a final comment, though this work aims at approaching robots to people, as referred in (Scholtz, 2003), robot designers should also strieve to enhance human skills through robot technology in addition to trying to substituting robot skills by human ones.

8. References

Asarin, A. and Dang, T. (2004). Abstraction by projection and application to multi-affine systems. In *Procs. of 7th Int. Workshop on Hybrid Systems: Control and Computation, HSCC 04,* volume 2993 of *Lecture Notes in Computer Science,* pages 32-47. Springer. Philadelphia, USA, March 25-27.

Asarin, E. and Maler, O. (1994). On some relations between dynamical systems and transition systems. In Abiteboul, S. and Shamir, E., editors, *Procs. of 21st Int. Colloquium on Automata, Languages and Programming, 1C ALP 94,* volume 820 of *Lecture Notes in Computer Science,* pages 59-72. Springer. Jerusalem, Israel, July 11-14.

Betke, M. and Gurvits, L. (1997). Mobile Robot Localization Using Landmarks. *IEEE Transactions on Robotics and Automation,* 13(2).

Bignell, J. (1997). *Media Semiotics: An Introduction.* Manchester University Press, 2nd edition.

Carroll, J., editor (2003). HCI Models, Theories, and Frameworks - Towards a Multidisciplinary Science. Morgan Kaufmann Publishers.

Chandler, D. (2003). *Semiotics, The basics.* Rutledge.

Chomsky, N. (1968). *Language and Mind.* Harcourt Brace Jovanovich Inc.

Codognet, P. (2002). The semiotics of the web. In *Leonardo,* volume 35(1). The MIT Press.

Dautenhahn, K. and Werry, I. (2002). A Quantitative Technique for Analysing Robot-Human Interactions. In *Procs. 2002 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems.* EPFL, Lausanne, Switzerland, October.

Bias, B., Zlot, R., Kalra, N., and A., S. (2005). Market-based multirobot coordination: A survey and analysis.

Everett, H. (2003). Robotic security systems. *IEEE Instrumentation & Measurement Magazine,* 6(4):30-34.

Fan, X. and Qi, C. and Liang, D. and Huang, H. (2005). Probabilistic Contour Extraction Using Hierarchical Shape Representation. In *Proceedings of the Tenth IEEE International Conference on Computer Vision (ICCV05).*

Fox, D. and Thrun, S. and Burgard, W. and Dellaert, F. (2001). Particle Filters for Mobile Robot Localization. In A. Doucet, N. de Freitas, and N. Gordon, editor, *Sequential Monte Carlo Methods in Practice,* pages 499-516. Springer Verlag.

Harman (1969). Linguistic competence and empiricism. In *Language and Philosophy.* New York University Press.

Hu, W., Tan, T., Wang, L., and Maybank, S. (2004). A survey on visual surveillance of object motion and behaviours. *IEEE Transactions on Systems, Man and Cybernetics, Part C,* 34(3):334-352.

IEEE (2000). IEEE Recommended Practice for Architecture Description of Software-Intensive Systems. E-ISBN 0-7381-2519-9, ISBN 0-7381-2518-0.

Kawamura, K., Nilas, P., Muguruma, K., Adams, J., and Zhou, C. (2003). An agent-based architecture for an adaptive human-robot interface. In *Procs. 36th Hawaii Int. Conf. on System Sciences.*

Khrisna, K. and H., H. (2003). Social control of a group of collaborating multi-robot multi-target tracking agents. In *Procs. 22nd Digital Avionics Systems Conference (DASC-22).*

Kogut, G., Blackburn, M., and Everett, H. (2003). Using video sensor networks to command and control unmanned ground vehicles. In *Procs Unmanned Systems in International Security 2003 (USIS 03).*

Konolidge, K., Fox, D., Ortiz, C., Agno, A., Eriksen, M., Limketkai, B., Ko, J., Morisset, B., Schulz, D., Stewart, B., and R., V. (2004). Centibots: Very large scale distributed robotic teams. In *Procs. Int. Symp. on Experimental Robotics (ISER-04).*

Malcolm, G. and Goguen, J. (1998). Signs and representations: Semiotics for user interface design. In *Procs. Workshop in Computing,* pages 163-172. Springer. Liverpool, UK.

Meystel, A. and Albus, J. (2002). *Intelligent Systems: Architecture, Design, and Control.* Wiley Series on Intelligent Systems. J. Wiley and Sons.

Neumiiller, M. (2000). Applying computer semiotics to hypertext theory and the world wide web. In Reich, S. and Anderson, K., editors, *Procs. of the 6th Int. Workshop on Open Hypertext Systems and Structural Computing,* Lecture Notes in Computer Science, pages 57-65. Springer-Verlag.

Nguyen, H. and Bott, J. (2000). Robotics for law enforcement: Applications beyond explosive ordnance disposal. Number 4232, pages 433-454.

Nicolescu, M. and Mataric, M. (2001). Learning and interacting in human-robot domains. *IEEE Transactions on Systems, Man, and Cybernetics, Part A: Systems and Humans,* 31(5):419-430.

Ogasawara, G. (1991). A distributed, decision-theoretic control system for a mobile robot. *ACM SIGART Bulletin,* 2(4):140-145.

Parker, L. (1998). ALLIANCE: An Architecture for Fault Tolerant Multirobot Cooperation. *IEEE Transactions on Robotics and Automation,* 14(2):220-240.

Qiu, L. and Li, L. (1998). Contour Extraction of Moving Objects. In *Procs. of 14th International Conference on Pattern Recognition (ICPR98).* Brisbane, Australia, August.

Rani, P. and Sarkar, N. (2004). Emotion-Sensitive Robots - A New Paradigm for Human-Robot Interaction. In *IEEE-RAS/RSJ International Conference on Hu-manoid Robots (Humanoids 2004).* Santa Monica, Los Angeles, CA, USA, November 10-12.

Raskin, J. (2000). The Human Interface - New Directions for Designing Interactive Systems. Addison-Wesley.

Rybski, P., Hougen, D., Stoeter, S., Gini, M., and Papanikolopoulos, N. (2000). Control of multiple small surveillance robots at AAAI 2000. In *Procs. AAAI 2000 Mobile Robot Competition and Exhibition Workshop.*

Ryu, H. and Lee, W. (2006). Where You Point is Where the Robot is. In *Procs. of 7th ACM SIGCHI New Zealand Chapter's Int. Conf. on Computer-Human Interaction: Design Centered HCI,* volume 158, pages 33-42. ACM International Conference Proceeding Series. Christchurch, New Zealand.

Savidis, A. and Stephanidis, C. (2004). Unified user interface design: designing universally accessible interactions. *Interacting with Computers,* 16(2):243—270.

Scholtz, J. (2003). Theory and evaluation of human robot interactions. In *Procs. of the 36th Hawaii Int. Conf. on System Sciences, HICSS'03.* Hawaii, January 5-8, USA,.

Sequeira, J. and M.I. Ribeiro, M. (2006a). Human-robot interaction and robot control. *Springer's Lecture Notes in Control and Information Sciences,* 335:375-390.

Sequeira, J. and M.I. Ribeiro, M. (2006b). A semiotic apporach to the control of semi-autonomous robots. *International Journal of Systems Science,* 37(6):361-376.

Shewchuk, J. R. (1998). A condition guaranteeing the existence of higher-dimensional constrained Delaunay triangulations. In *Procs. of 14th Annual Symposium on Computational Geometry,* pages 76—85. Minneapolis, Minnesota, USA.

Smirnov, G. (2002). Introduction to the Theory of Differential Inclusions, volume 41 of Graduate Studies in Mathematics. American Mathematical Society.

Stankovic, S. and Siljak, D. (2002). Model abstraction and inclusion principle: A comparison. *IEEE Transactions on Automatic Control,* 47(3):529-532.

Valera, M. and Velastin, S. (2005). Intelligent distributed surveillance systems: a review. *IEE Procs. On Vision, Image and Signal Processing,* 152(2):192-204.

van der Schaft, A. (2004). Equivalence of dynamical systems by bisimulation. *IEEE Transactions on Automatic Control,* 49(12).

Willems, J. (1991). Paradigms and puzzles in the theory of dynamical systems. *IEEE Transactions on Automatic Control,* 36(3).

19

Effect of Robot and Screen Agent Recommendations on Human Decision-Making

Kazuhiko Shinozawa[1] and Junji Yamato[2]
[1]*ATR Intelligent Robotics and Communication Laboratories,*
[2]*NTT Communication Science Laboratories*
Japan

1. Introduction

Two-dimensional (2D) character agents that have a human-like appearance are being developed. In the future, such agents will be able to interact with their users in a natural and friendly manner through speech recognition, synthesized speech, and action displays. In addition, robots or robotic companions that have a three-dimensional (3D) physical body are attracting attention as *communication partners*.

Such embodied social agents (ESAs) make interaction more meaningful than it is when interfaces do not appropriately display actions or speak (Beskow and McGlashan, 1997). It is known that people's attitudes towards computerized media are similar to the attitudes they have towards other people (Reeves and Nass, 1996). Even if people only read text or hear a voice from computers, they tend to assign some social existence to them. More social richness, defined as more complete human-like presentations, promises to make computers more attractive, productive, and easy to use. Some research has provided fruitful results and suggestions for presentation, i.e., graphical appearance (Massaro, 1998), non-verbal behavior (Cassell and Thórisson, 1999), and speech characteristics (Nass and Gong, 1999), and for personality (Nass and Isbister, 1998), emotion (Ball and Breese, 1998; Becheiraz and Thalmann, 1998), ethnicity (Takeuchi et al., 1998), and interpersonal communication strategy (Shinozawa et al., 2001) as well. Much of such research suggests that an ESA should be an effective interface for interactions with media.

The above research mainly focused on graphical on-screen agents and computers. On the other hand, robots having a physical body have attracted some attention as useful physical agents, and the above research results may apply to interaction with such robots. However, when we consider robots as ESAs, a new research topic, ``*dimensionality*'', appears. A robot has a 3D physical body while an on-screen agent has a 2D one. This leads to several questions: Does increasing dimension make a big difference or not? Does the physical 3D appearance affect us in a significant way during the interaction? When both a 2D agent like an on-screen agent and a physical 3D agent like a robot have a similar shape and use the same voice, what is the significance of the difference in *dimensionality*? Little research has focused on *dimensionality*, and we still have no solid answers. We live and work in 3D space. Everything has three dimensions and is located in 3D space. With a 3D body, pointing to some location makes it easy to understand what is being pointed at. When a

robot navigates a person, the combination of the robot's gestures and its body's direction has a strong relationship with high ratios of successful task completion (Ono et al., 2001). This suggests that the *dimensionality* produces a difference in the effect of interaction and that a 3D body makes interaction more meaningful.

How can we quantitatively measure the *dimensionality* or eye movement effects? Almost all research exploring the behavioral factor's effects of ESAs has been conducted by using questionnaire-based evaluations. For example, subjects are asked whether a robot is familiar or not. Getting a feeling of familiarity is important for a pet-like partner. However, it is not enough for a communication partner. Whenever we engage a communication partner, we listen to what the partner has to say, respond to it, and sometime change our thinking. If an ESA's behavior can influence human decision-making, it leads that he/she treats it as a communication partner. This would be one of evaluations for a communication partner. So, we mainly investigate an ESA's influence on human decision-making for evaluating above factors.

In this chapter, in an attempt to answer some the above questions, we discuss *dimensionality,* investigated by directly comparing results between an on-screen agent and a physical robot, and the role of a robot's eye movement in human-robot interaction. On both topics, the discussions are based on a quantitative evaluation of each factor's effect on human decision-making with a selection task. Section 2 describes the selection task with the direct comparison topic and presents the experimental results. Section 3 describes the effect of a tracking function in the 3D world case and presents the results. Section 4 discusses the effect of the dimensionality and eye movement with the experimental results. Then, Section 5 concludes with a short summary.

2. Difference in 3D and 2D agent's recommendation

Recommendation in an advertisement and assistance in a navigation task are two typical situations influencing human decision-making. An ESA acting as an assistant can easily influence users' decisions because users want to know appropriate information. Generally speaking, however, changing a user's mind is difficult in the advertisement situation. Advertising is an important application of ESAs, and we can also say that the recommendation includes helpful interaction-like assistance, because the initial recommendation does not always depend on what the user wants. So, we focus on the advertisement situation and measure the influence of ESAs on human decision-making.

2.1 Color-name selection task

The color-name selection task was introduced to quantitatively measure the influence of ESAs on human decision-making in an advertisement situation (Shinozawa et al., 2001). In a color-name selection task, a subject looks at a colored region and selects the color name from two options. The matching ratio of the recommended color names is measured. The ratio is treated as showing the degree of a recommendation's influence on human decision-making.

2.2 Recommendation situation

In the case of direct comparison between an on-screen agent and a physical robot, two situations are considered. In one, an ESA points to an object located in 3D space and in the other, it points to an object in 2D space during interaction. Accordingly, we prepared two scenes for interacting with an ESA.

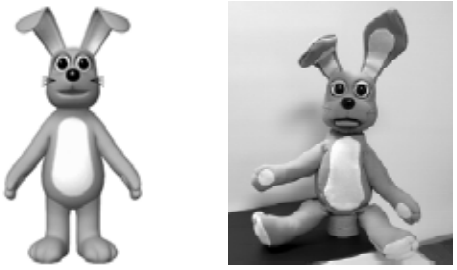
One scene is equivalent to the original one in the color-name selection task (Shinozawa et al., 2001). An ESA recommends a color name while it points to or looks at the color region and two color-name options on a CRT display when a subject should select one of them. In this case, objects used in the selection task are mainly located in 2D space. We therefore call this, the 2D world condition. The other scene is a new one that we call the 3D world condition. The color region that an ESA points to is in 3D space. Actually, we developed two new machines for displaying color regions in 3D space. One displays printed color plates according to external PC control. The other is a button box for displaying and selecting a color name. In the 3D world condition, a subject looks at printed color plates and selects color names using the button box (Fig. 1).

Figure 1. Display machines

The ESA recommended one of the two options under these two conditions, and we investigated the *dimensionality* by comparing the ESA's effect on user decision-making.

2.3 Robot and On-screen agent

Appearance is important for robots as well as for on-screen agents. Humans tend to recognize social roles, gender, or characters by analogy with appearance. Prior knowledge according to appearance has much influence on subjective evaluation (Shibata and Tanie, 2001).
In *dimensionality* issue, to avoid such influence, we made the appearance of robots and on-screen agents as equivalent to each other as possible [Fig. 2(a) and (b)]. The robot's height is 300 mm and on-screen agent's height is similar to the robot's one.



(a) On-screen agent (b) Robot

Figure 2. Appearance of ESAs

Similarly, voice plays an important role in molding the robot's or on-screen agent's character. Both the robot and on-screen agent use the same voice, which was made by

``Fluet'', a Japanese speech synthesizer developed by NTT (Mizuno and Nakajima, 1998). The robot was also developed by NTT.

2.4 Gestures

We prepared 27 gestures for both the robot and on-screen agent, which included pointing to a color region, nodding, blinking, and so on. We made the robot's motions similar to the on-screen agent's motions.

2.5 Color names

Before starting an experiment, subjects were told that this was a color-name selection task and that they should make a selection based on their own feeling and that there were no correct answers. Most of the color regions and options for color names in the experiment, such as vermilion or carmine, are unfamiliar to ordinary people. □

2.6 Speaking words

The robot and agent offered their personal opinions, for example, ``I think it's vermilion'', ``This shade is vermilion, isn't it?'', instead of making statements that would indicate it had definite knowledge about the displayed color. This was to avoid the effect of the subject's attributing any authority to the robot and agent.

2.7 Recommended color names

We carried out pretests without an ESA's recommendation and determined what color names the ESA should recommend and the orders of color name options. In both the 2D and 3D world condition, the same order of color names and the same recommended color name options were used.

2.8 Face direction when ESAs speak

Whenever we talk to someone, we look at that person's face or eyes. ESAs should behave in the same way. With an on-screen 2D agent, a well-known illusion associated with full-faced portraits occurs: from any viewing angle, it appears that the agent's gaze is always on the user (Bruce & Young, 1998). An on-screen agent's full-face animation can give a feeling that the agent talks to subjects. However, humans can easily detect that the eyes of a 3D face's are not looking at their face even if the difference from the correct direction is small. We therefore developed a subject's head tracking function by which a robot adjusts its head direction so that its head faces subject's head position.

2.9 Displayed Color region

In both world cases, the size of the displayed color region was about 270 mm x 160 mm, and the average distance from subjects to the color regions was about 600 mm. In the 2D world case, the colors displayed on the CRT were measured with the CRT color analyzer three times a day. The changes in these values were small (less than 10%) for the whole experiment. In the 3D world case, the colors displayed on printed plates were measured with a tristimulus colorimeter once a day. The changes in these values were also small (less than 7%) for the entire experiment.
Therefore, all of the subjects saw the same color in each world case.

2.10 Subjects

Six experiments were conducted to manage all combinations described above.

1. 2D world condition
   (a) No recommendation (Group **No2)**
   (b) Agent recommendation (Group **Ag2**)
   (c) Robot recommendation (Group **Ro2**)
2. 3D world condition
   (a) No recommendation (Group **No3**)
   (b) Agent recommendation (Group **Ag3**)
   (c) Robot recommendation (Group **Ro3**)

None of the subjects were experts on color names and all were recruited from the general public. Each subject participated in only one experiment; never more than one. Table 1 shows information about the subjects in each group.

	Group					
	No2	**Ag2**	**Ro2**	**No3**	**Ag3**	**Ro3**
Number	30	30	30	31	27	30
Mean age	23.87	27.60	23.30	25.40	25.00	26.29
Max. age	49	39	36	36	39	45
Min. age	19	19	18	18	20	18
Ratio of males	0.43	0.50	0.50	0.50	0.44	0.48

Table 1. Subjects in each group

2.11 No recommendation case

To investigate the influence of an ESA's recommendation on user decisions, we must know the mean of the matching ratios without recommendation. We therefore conducted no-recommendation experiments for the 2D and 3D world conditions. In these experiments, subjects did not see any on-screen agents or robots and chose a color name with no recommendation. In all recommendation conditions, the recommended color name options were fixed due to the pretest described above. The difference in matching ratios between the no-recommendation case and recommendation case shows the degree of the recommendation's influence on user decision-making. When matching ratios in the recommendation case are greater than in the no-recommendation case, the influence is considered to be positive.

2.12 Procedure

Upon arriving at the lab, subjects were told that the purpose of this experiment was to mainly investigate the relationship between color regions and color names, and that they should make a selection based on their own feeling because there were no correct answers. After this explanation, they signed an informed consent statement.

2.12.1 No-recommendation

In the no-recommendation condition, subjects did the task without the ESA's recommendation and there was no ESA near them.

2.12.2 Recommendation case

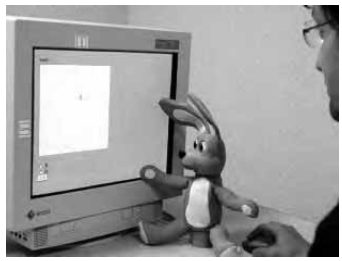
They encountered the robots or on-screen agents for the first time when they entered the experimental room. The ESA behaved like it was asleep until the subject pushed a button. Once the button had been pushed, the ESA behaved like it had been awakened and introduced itself, and the experiment started.

The experiment consisted of thirty questions, and each question had two possible responses. When presenting each question, the ESA made a statement endorsing one of the two possible responses. While the ESA was asking the question and presenting the two choices, these options appeared on the computer display in the 2D world condition and on the button box display in the 3D world condition. The subject in both conditions indicated his/her choice by clicking a radio button on the computer display and by pressing the corresponding button. The subject then pressed the ``OK'' button to send the selection to the computer.

If the choice matched the ESA's suggestion, the ESA nodded with approval while expressing a positive statement. If it did not match the suggestion, the ESA bowed and shook its head slowly while responding with a negative statement. This continued until all questions were answered. When the interaction finished, the experimenter gave the subject a questionnaire.

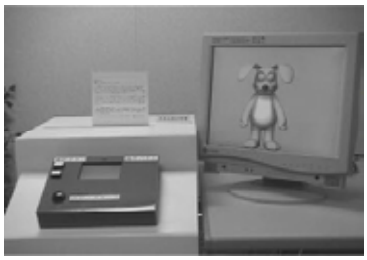
Figures 3(a) and (b) show scenes of the on-screen agent and robot experiment in the 2D world condition, respectively.

(a) Agent recommendation



(b) Robot recommendation

Figure 3. Scene in 2D world case



(a) Agent recommendation

(b) Robot recommendation

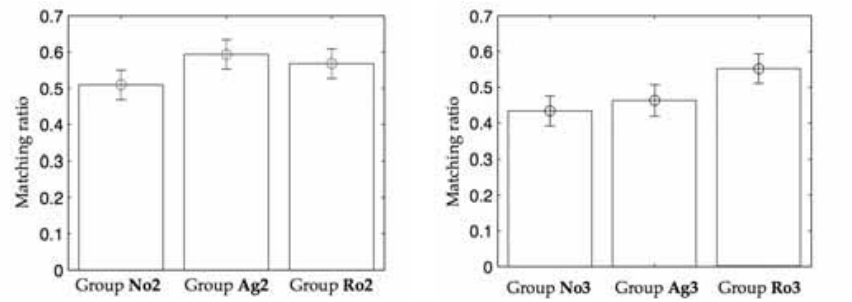
Figure 4. Scene in 3D world case

Almost all subjects finished one experiment in less than 20 minutes. The options that subjects selected were automatically recorded in a computer when subjects pushed the OK button. The scenes in one experiment were videotaped.

2.13 Results

We calculated the mean matching ratios of the color names that the agent or robot successfully recommended to each subject. The mean ratios in the groups were also calculated. In the no-recommendation case, subjects did not get any recommendation, but the same color name options as in the recommendation case were presented. In the recommendation case, one of the color names was recommended. In estimating the mean matching ratios in the no-recommendation case, the mean matching ratios of the color names that were recommended in the recommendation case were calculated.

2.13.1 2D world case



(a) 2D world case (b) 3D world case

Figure 5. Matching ratios

Figure 5(a) shows the mean matching ratios for each group in the 2D world case. Factorial analyses of variance (ANOVA) were conducted for each mean of matching ratios. We compared the mean between no-recommendation, on-screen agent's recommendation, and robot's recommendation. There was a significant difference only between Group **No2** and Group **Ag2** (ANOVA, F=3.457, p=0.036, Scheffé, p=0.043). The difference between Group **No2** and Group **Ro2** was not statistically significant.

2.13.2 3D world case

Figure 5(b) shows the mean matching ratios for each group in the 3D world case. Again, factorial analyses of variance (ANOVA) were conducted for each mean of matching ratios. We again compared the mean between no-recommendation, on-screen agent's recommendation and robot's recommendation. There were significant differences between Group **No3** and Group **Ro3** and between Group **Ag3** and Group **Ro3** (ANOVA, F=6.725, p=0.002, Scheffé, p=0.003, p=0.042).

Table 2 summarizes the experiment results. The circles mean that the difference from the no-recommendation case is statistically significant.

	2D world case	3D world case
Agent	○	·
Robot	·	○

Table 2. Effect of recommendation

3. Gaze effect in 3D agent and 3D world case

We used the robot with the subject's head tracking function. In the actual experimental situation, subjects sat at fixed place and the relative position from a robot to the place was also fixed. Whenever the robot turned to the place with a fixed angle, its eyes could roughly catch the subject's face or head. However, subjects' sitting posture was not always same and their head sometime slightly moved during the experiment. Even in such a situation, does a robot need a tracking function? Is precisely facing a human head important for a 3D agent?

We conducted additional experiments to confirm the importance of having the robot directly facing subjects and investigated whether the importance also holds in a robot with a different appearance. The experiments consisted of three groups with same color-name selection task as in the above experiments. One group used the same rabbit-like robot without a tracking function. The robot could not adjust its head direction to the subject's head movement and always made a same motion. For the other groups, a head robot made a recommendation instead of a rabbit-like robot. Figure 7 shows the head robot's appearance. The robot was developed by MIT AI Laboratory and modified by NTT Communication Science Laboratories. The robot has only a head and neck (no arms or legs), both with 30 degrees of freedom. The robot can produce various facial expressions. In addition, each eyeball has a camera that can pan and tilt. Having cameras inside the eyeballs can makes the robot's gaze direction clear to subjects and ensures the center of the robot's eye can be directed toward subjects with an appropriate vision system. To enable the tracking function, a skin-color region was detected using a detector developed by MIT AI Lab, and the eye direction turned to the center of that region. Table 3 summarizes the three conditions. Subjects' mean age was 24.2 years.

Figure 7. Head robot

	Group		
	Ro3-2	**Ro3-3**	**Ro3-4**
Tracking/Non-tracking	Non-Tracking	Non-tracking	Tracking
Number	30	14	14
Mean age	20.97	25.07	23.21
Max. age	25	32	35
Min. age	19	19	19
Ratio of males	0.5	0.50	0.50

Table 3. Groups

3.1. Result in matching ratio

The mean of the matching ratio to the robot's recommended options for each subject is shown in Fig. 8. Factorial ANOVA were conducted for each mean of matching ratios. We compared the mean between non-recommendation, without tracking, and with tracking for the same robot. There is a significant difference in the mean between non-recommendation and the robot with tracking for both robots. (In the rabbit robot case, ANOVA, $F = 6.292$, $p = 0.003$, Scheffé, $p = 0.003$ and in the head robot case, ANOVA, $F = 4.759$, $p = 0.012$, Scheffé, $p = 0.015$). The difference between the non-recommendation case and without tracking is not statistically significant ($p > 0.1$).

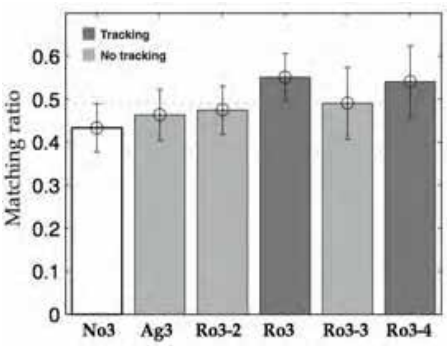


Figure 8. Matching ratios

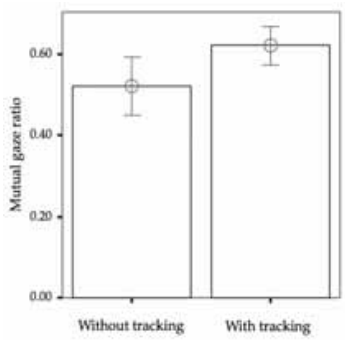


Figure 9. Mutual gaze ratios

3.2. Result in mutual gaze

We determined a gaze-period according to the directions of both the robot's gaze and subject's gaze. Subjects' gaze was classified into four categories: "color plate", "button box", "robot's face", and "other". The robot's gaze was classified into three categories: "color plate", "subject's face", and "other". This is because the robot could make its eyes turn toward the button box. In the head robot condition, we recorded the experiment scenes using the cameras equipped in head robot's eyes, which gave us an accurate recording from

the robot's viewpoint. All assignments were made on the basis of an experimenter's observations with those videos. We assumed that a combination of a robot's gaze to a subject's face and a subject's gaze to a robot's face achieved mutual gaze. We defined the mutual gaze period as the period for which the combination continues. Figure 9 shows the mean ratio of mutual gaze periods for the whole experiment. The ratio in the tracking condition is significantly larger than in non-tracking condition (ANOVA, F=6.468, p=0.018).

4. Discussion

Figure 5 shows that the *dimensionality* of the ESA causes differences in the recommendation's effect on user decision-making. The 3D body was not always superior to the 2D body for recommendation, and on-screen agents seem to have weak points, too.

Those differences cannot be explained only by the advantages or disadvantage of pointing. In the 2D world condition, the color region was presented on the computer display. The display was in 3D space, so the color region was presented in the 3D space. From this point of view, there should be no difference between the 2D and 3D world conditions and therefore no difference in the effect. There must therefore be some other reasons.

The results changed according to the combination of the location pointed to and the ESA's *dimensionality*. This seems to be evidence supporting the importance of consistency in the *dimensionality* between communication partners and the environment consisting of the pointing location and manipulated objects. In the 2D world condition, the color region was displayed on the CRT and color name options that should be selected were also on the CRT. In the 2D world and on-screen agent case, all were contained in the frame of the computer display. The frame might have emphasized the appropriate consistency and the on-screen agent might have had a strong influence through its recommendation. In the 3D world condition, the communication environment was in the 3D physical space. In the 3D world and robot case, the color region and color name options were contained in the physical 3D space, although they were physically separated and there was no frame.

In addition, the effect of robot's recommendation was much greater than that of the on-screen agent's. The robot's body might have had a strong influence for emphasizing appropriate consistency to 3D space without a visual frame. In addition, behavior in communication contains ambiguous meanings and depends on the situation and communication environment (Sperber and Wilson, 1993). So, humans tend to quickly recognize an environment where the communication partner exists for interpretation.

The results of our experiment seem to provide evidence that humans tend to quickly recognize communication environments even in interaction with an ESA, and also suggest that we must not forget the communication environment in designing ESA behavior.

4.1 Tracking function effect in 3D world case.

Figure 8 shows that the influence of a robot without a tracking function is still larger than both 2D agent's and no-recommendation case, although the difference is not significant. The result suggests that the *dimensionality* still works even if a robot has no tracking function, although the effect is reduced. And, 3D shape with predetermined motions is not sufficient for significantly producing the robot's *dimensionality* effect on human decision-making. In addition, those results also suggest that only a tracking function is not sufficient for

explaining all effects caused by the *dimensionality,* but that a tracking function must be important for robot's influencing human decision-making.
Much research has pointed out eye direction's importance in conversation. For example, our gaze is one way of encouraging someone to talk (Michael & Mark, 1976). Gaze fixation exerts a special pressure to communication. In addition, research based on questionnaires has confirmed that a robot's face direction makes a human notice its gaze (Imai et al, 2002) and suggested that its face direction is effective for signaling to whom the robot will talk in a multi-user situation.
As shown in Fig. 9, a tracking function increases robot's mutual gaze chances when subjects may have a feeling that the robot looks at them. In addition, robots with a tracking function influenced decision-making more significantly than robots without it. Thus, such a feeling of *being looked at* would be necessary for producing some changes on human decision. In other words, our results show that mutual gaze influences not only human feelings but also decision-making in cases of interaction between humans and robots. And, a feeling that a robot looks at us is fundamental and crucial to a robot's becoming a communication partner.

5. Conclusion

We experimentally confirmed through quantitative evaluation that the degree of recommendation effect firmly depends on the interaction environment. The results show that a three-dimensional body has some advantage when the interaction environment is a three-dimensional space, but has less advantage in two-dimensional space than a two-dimensional body does. This suggests that geometrical consistency between an ESA and the interaction environment plays an important role in communication.
In the 3D world case, we also experimentally confirmed that a tracking function for a robot can play the same important role that gaze has in humans; that is, it can increase the robot's influence on human decision-making through the interaction even if the tracking movement is small.
For a robot as a physical advertisement agent, tracking is an important function. The tracking function successfully makes the mutual gaze ratio greater during interaction; however, the ratio doesn't always have a relationship with the influence on decision-making.
Mutual gaze increases the chance when users may have feeling of being looked at, but it is not enough for insuring that a robot gives the feeling. To become a communication partner, a robot needs an effective method for emphasizing the feeling.

6. References

Ball, G. & Breese, J., (1998). Emotion and personality in a conversation character. In: *Proceedings of the Workshop on Embodied Conversation Characters*.

Becheiraz, P. & Thalmann, D., (1998). A behavioral animation system for autonomous actors personified by emotions. In: *Proceedings of the 1998 Workshop on Embodied Conversational Characters*.

Beskow, J. & McGlashan, S. (1997). Olga - conversational agent with gestures. In: P*roceedings of the IJCAI-97 Workshop on Animated Interface Agents: Making them Intelligent*.

Bruce, V. & Young, A. (1998). In the eye of the beholder: The science of face perception: Oxford University Press.

Cassell, J. & Thórisson, K. R. (1999). The power of a nod and a glance: Envelope vs. emotional feedback in animated conversational agents. In: *Applied Artificial Intelligence*. Vol. 13. pp. 519–538.

Imai, M., Kanda, T., Ono, T., Ishiguro H., & Mase, K. (2002). Robot mediated round table: Analysis of the effect of robot's gaze. In *Proceedings of 11th IEEE International Workshop on Robot and Human Communication* (RO-MAN2002), 2002, pp. 411–416.

Michael, A. & Mark, C. (1976) Gaze and Mutual Gaze. Cambridge University Press, UK.

Massaro, D. (1998). Perceiving talking faces: From speech perception to a behavioral principle. MIT Press.

Mizuno, O. & Nakajima, S. (1998). Synthetic speech/sound control language: Mscl. In: *3rd ESCA/COCOSDA Proceedings of International Workshop on Speech Synthesis*. pp. 21–26.

Nass, C. & Gong, L. (1999). Maximized modality or constrained consistency? In: *Proceedings of the AVSP '99 Conference*.

Nass, C. & Isbister, K. (1998). Personality in conversational characters: Building better digital interaction partners using knowledge about human personality preferences and perceptions. In: *Proceedings of the 1998 Workshop on Embodied Conversational Characters.*

Ono, T., Imai, M. & Ishiguro, H. (2001). A model of embodied communications with gestures between humans and robots. In: *Proceedings of Twenty-third Annual Meeting of the Cognitive Science Society*. pp. 732–737.

Reeves, B. & Nass, C. (1996). The Media Equation. Cambridge University Press.

Shibata, T. & Tanie, K. (2001). Physical and affective interaction between human and mental commit robot. In: *Proceedings of IEEE International Conference on Robotics and Automation*. pp. 2572–2577.

Shinozawa, K., Yamato, J., Naya, F. & Kogure, K. (2001). Quantitative evaluation of effect of embodied conversational agents on user decision. In: *Proceedings of HCI International* 2001. pp. 998–1002.

Shinozawa, K., Naya, F., & Kogure, K. & Yamato, J., Effect of robot's tracking users on human decision making. *Proceedings of 2004 IEEE/RSJ International Conference on Intelligent Robots and Systems* , pp. 1908-1913, 2004, Sendai, Japan

Shinozawa, K., Naya, F., Yamato, J., & Kogure, K., Differences in Effect of Robot and Screen Agent Recommendations on Human Decision-Making, *IJHCS*, Vol. 62/2, pp 267-279, 2005.

Sperber, D. & Wilson, D. (1993). RELEVANCE: Communication and cognition. Oxford: blackwell.

Takeuchi, Y., Katagiri, Y., Nass, C. & Fogg, B. (1998). Social response and cultural dependency in human-computer interaction. In: *Proceedings of PRICAI*. pp. 114–123.

Yamato, J., Shinozawa, K., Naya, F.: Effect of Shared-attention on Human-Robot Communication, In *Proceedings of ACM/CHI2004 Workshop Shaping Human-Robot Interaction*-Understainding the Social Aspects of Intelligent Robotic Products

Yamato, J., Brooks, R., Shinozawa, K., and Naya, F.: Human-Robot Dynamic Social Interaction, *NTT Technical Review*, Vol.1, No.6, Sep. 2003

20

Collective Motion of Multi-Robot System based on Simple Dynamics

Ken Sugawara[1], Yoshinori Hayakawa[2], Tsuyoshi Mizuguchi[3] and Masaki Sano[4]
[1]*Tohoku Gakuin University,* [2]*Tohoku University,* [3]*Osaka Prefecture University,* [4]*University of Tokyo*
Japan

1. Introduction

Multi-robot system is one of the most attractive systems in robotics, and many researchers have been investigating it from various viewpoints (Cao, 1997; Balch & Parker, 2002; Parker, 2003). Remarkable point of multi-robot system is that the robots are cooperatively able to complete a task that a single robot can hardly or cannot accomplish by itself. Especially, searching, transportation or conveyance, construction, and pattern formation are typical categories which are suitable for multi-robot system, and a lot of concrete tasks can be found in each category. Some approaches have been considered and proposed to accomplish them, and biology inspired robotics is one of the most effective method to develop useful multi-robot system. Actually, many researchers have been applying this approach to design multi-robot system (Bonabeau, 1999).

In this article, we treat collective motion of motile elements which was inspired by living things such as fishes, birds and small insects, assuming to apply to real robot system. It is considered that the collective motion of the robots can be utilized for some significant tasks such as traffic control, ground/ocean surveillance (Ogen, 2004), and so on.

This article is organized as follows. In section 2, we explain a fundamental kinetic model of collective behaviors based on the livings. Result of numerical simulation and analysis are shown in Section 3. In section 4, we show small scale of experiment based on the model.

2. Kinetic model for collective motions

Many animals form groups which we consider as cooperative systems of active elements. The collective motions of animals show extreme diversity of dynamics and patterns (Edelstein-Keshet, 1990; Partridge, 1982; Wilson 1975) For example, migrant fish, like the sardine, tend to school by aligning their headings and keeping a fixed mutual distance. Large birds such as cranes migrate in well-ordered formations with constant cluster velocity. Small birds such as sparrows fly in wandering, disordered aggregates. Insects, such as the mosquito, fly at random within spatially limited swarms. There seems a tendency that the smaller the size of animals, the more disorder in cluster motions, at least, for some flying or swimming animals. Many model equations claim to explain the collective motions of

animals (Niwa, 1994; Doustari & Sannomiya, 1992; Vicsek *et al.*, 1995). Most postulate that individuals are simply particles with the mutual interactions and motive force. In this simplification, the equations of motion become Newtonian equations for the particles, and the dependence on the characteristic scales of animals appears only through their mass. The resulting collective motions are mostly regular and ordered. Swarming, disordered aggregates and wandering, require external random perturbations.

To generalize these models, we introduce extended internal variables describing the particles, which we call motile elements (Shimoyama *et al.*, 1996). Basically, the motion of *i*-th element is described with a position vector $\vec{r}_i$ and a velocity vector $\vec{v}_i$ which are relative to fluid or air. Although the internal variables may have physical, physiological or ecological origins in each species, we additionally use a simple physical vector degree of freedom; the *heading unit vector* $\vec{n}_i$, parallel to the axis of the animal. Large birds often glide. In a glide, the heading, $\vec{n}_i$, and the velocity vector, $\vec{v}_i$, need not be parallel. Therefore, we assume that $\vec{n}_i$ and $\vec{v}_i$ relax to parallel with relaxation time τ. Including the heading dynamics, we propose a kinetic model of N interacting motile elements. For simplicity, we consider two-dimensions, but the model is easily extensible to three-dimensions. The state variables for the *i*-th element are the position vector $\vec{r}_i$, the velocity vector $\vec{v}_i$, and the heading unit vector $\vec{n}_i$, and these variables have the following dynamics:

$$m\frac{d\vec{v}_i}{dt} = -\gamma\vec{v}_i + a\vec{n}_i + \sum_{j\neq i}\alpha_{ij}\vec{f}_{ij} + \vec{g}_i \tag{1}$$

$$\tau\frac{d\theta_i}{dt} = \sin(\phi_i - \theta_i) + \sum_{j\neq i} J_{ij}\sin(\theta_j - \theta_i) \tag{2}$$

$$(i = 1,2,\cdots,N)$$

where θ_i is the angle between the unit vector $\vec{n}_i$ and a certain direction, say, the x axis ($\vec{n}_i = (\cos\theta_i, \sin\theta_i)$), and ϕ_i is the angle between the velocity vector $\vec{v}_i$ and the x axis($\vec{v}_i = |\vec{v}_i|(\cos\phi_i, \sin\phi_i)$) (Fig.1).

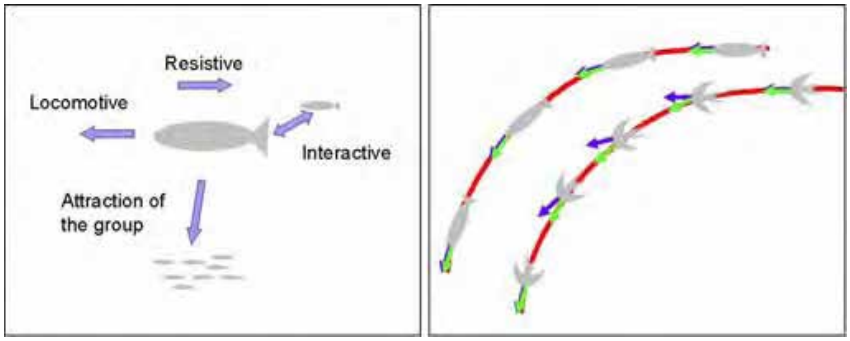


Figure 1. Schematic diagram for Newton's equation of motion for particles (left) , and the dynamics of the heading (right). The relaxation time τ of birds is much larger than that of fishes

In this model, every element is identical except for initial conditions. Eq.(1) is Newton's equation of motion for particles of mass m, is the resistive coefficient based on Stokes's law for an element moving in fluid. We assume that the motile force a acts in the heading direction $\vec{n}_i$. The term $\vec{f}_{ij}$ represents mutual attractive and repulsive forces between the i-th and j-th elements, and $\vec{g}_i$ is the force toward the center of group, which is taken as the gravitational center in our model. We use the analogy with the intermolecular forces as introduced by Breder (1954) based on the observations for fish schooling (see also (Aoki, 1980; Breder, 1976)). We assume that the interaction force is given by:

$$\vec{f}_{ij} = -c\left\{\left(\frac{\left|\vec{r}_j - \vec{r}_i\right|}{r_c}\right)^{-3} - \left(\frac{\left|\vec{r}_j - \vec{r}_i\right|}{r_c}\right)^{-2}\right\}\cdot\left(\frac{\vec{r}_j - \vec{r}_i}{r_c}\right)\cdot\exp(-\left|\vec{r}_j - \vec{r}_i\right|/r_c), \tag{3}$$

where c is a parameter that represents the magnitude of interactions and r_c the optimal distance between neighbors.

The interaction need not be isotropic. If the interaction is based on visual information, the interaction with elements in front of a given element is stronger than with those behind. Therefore, we introduce a direction sensitivity factor described by:

$$\alpha_{ij} = 1 + d\frac{\vec{n}_i \cdot (\vec{r}_j - \vec{r}_i)}{\left|\vec{r}_j - \vec{r}_i\right|}$$

$$(0 \le d \le 1) \tag{4}$$

or

$$\alpha_{ij} = 1 + d\cos\Phi, \tag{4'}$$

where Φ implies the angle formed by $\vec{n}$ and $\vec{r}_j - \vec{r}_i$. Here a new parameter d is introduced to control the anisotropy of sensitivity. When d=0, the interaction is isotropic.

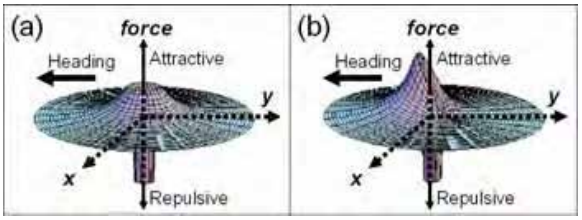


Figure 2. Schematic images of interaction force. (a) in case of d=0. (b) in case of d=1

Furthermore, we introduce a global attraction force $\vec{g}_i$ given by:

$$\vec{g}_i = c_g\frac{\vec{g} - \vec{r}_i}{N\left|\vec{g} - \vec{r}_i\right|} \tag{5}$$

where $\vec{g}_i$ is the center of group, *i.e.*,

$$\vec{g} = \sum_i \vec{r}_i / N \tag{6}$$

In the following discussions, we assume these two interaction forces have the same order of magnitude, *i.e.*, $c = c_g$.

The velocity vector $\vec{v}_i$ need not be in the heading direction $\vec{n}_i$, because of the inertial moment of the animal's body. We assume that the heading is parallel to the velocity for linear motion. We use equation (2) to relax the difference between the heading angle and the velocity direction angle. The relaxation time τ is related to the inertial moment and drag of the animal's body, and the time scale of maneuvering, such as the flapping period of wings or fins for birds or fishes, or tumbling period of flagella for bacteria. If the individuals are small (the inertial moment is small) and fast in flapping, τ is small (Fig.1 (right)).

To account for the tendency of animals to align their heads (Inoue, 1981; Hunter, 1966), we consider the interaction of $\vec{n}_i$ vectors. In the second term of the right hand side of Eq. (2), J_{ij} represents the tendency of individual i to align with individual j. We assume here that the interaction is a decreasing function of distance,

$$J_{ij} = k\left(\frac{\left|\vec{r}_j - \vec{r}_i\right|}{r_c}\right)^{-1} \tag{7}$$

However, in the most of the present work we take $k=0$ if not specified.

3. Numerical simulations and experiments

3.1 Characteristic of collective behavior

To investigate the qualitative properties of our model, we carried out numerical simulations for various control parameters and observed the collective motions. The equations of motion were solved with an explicit Euler method. Typical value of Δt to avoid numerical instability was less than 0.01. Initially, motile elements are placed at random by using Gaussian distributed random vectors in two dimensions within the standard deviation on the order of the inter-neighbour distance, r_c. The initial velocity of the elements is also given by Gaussian distributed random vectors with standard deviation of unity, and the heading vectors $\vec{n}$ are set in random directions.

We carried out numerical simulation for N ranging from 10 to 100, and we found several distinct collective behaviors which can be seen independently to $N < 100$. The trajectory of the center of the cluster are illustrated in Fig. 3. The trajectories are classified into four types.

1. **Marching:** When the anisotropy of mutual attraction is small, the elements form a regular triangular crystal moving at constant velocity. The formation is stable against disturbance and velocity fluctuations are very small. We call this motion a marching state.
2. **Oscillation:** Several group motions exhibit regular oscillations, including:
   (i) Wavy motion of the cluster along a linear trajectory.
   (ii) A cluster circling a center outside the cluster.
   (iii) A cluster circling a center inside the cluster.

The stability of oscillatory motion is weaker than that of marching. Oscillatory clusters often occur near the boundary between wandering, and the oscillation and marching may coexist for some parameters.

3. **Wandering:** For non-zero d, the center of the cluster can wander quite irregularly, while the lattice-like order inside the cluster persists. The mutual position of elements rearranges intermittently according to chaotic changes in the direction of motion. We call this behavior wandering. It occurs in flocks of birds, e.g., small non-migratory birds like the sparrow.
4. **Swarming:** Beyond the wandering regime, we found more irregular motion, where the regularity within the cluster fails, although the cluster persists. Compared to wandering, the velocity of elements has a wide distribution, and the mobility of the cluster is small, a behavior reminiscent of a cloud of mosquitoes.

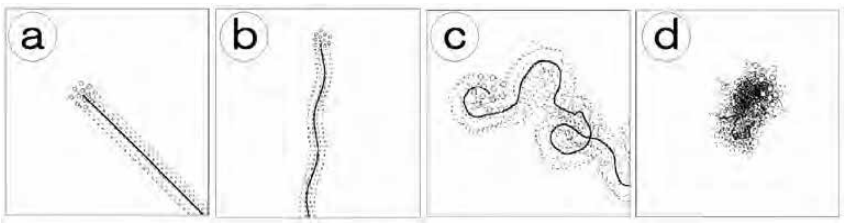


Figure 3. The trajectories of the elements (dotted lines) and the center of mass (solid line) obtained by numerical simulation. Each cluster consists of twelve motile elements (shown as white circle). Typical types of collective motions are shown as: (a) marching, (b) oscillatory (wavy), (c) wandering, and (d) swarming

Marching and oscillation form an ordered phase, while the others form a disordered phase. In the ordered phase, elements behave as a regularly moving cluster which is stable against perturbations by external force or small changes of kinetic parameters. This kind of stability would be required for the grouping animals, too, because the cluster of traveling birds or fishes should be structural stable. On the contrary, in disordered phase, the motion of clusters become unpredictable, which would be beneficial for small animals to escape from predators.

We refer to the transition between order and disorder as the marching/swarming transition. We have examined the parameter dependence of the behavior, fixing the number of elements, $N = 10$. In Fig. 4, we show characteristic behaviors in τ - γ and τ - a space. Fig. 4(a) shows that the transition between marching and wandering is well defined and the boundary occurs when $\gamma/\tau \sim 20$. Since γ is proportional to the relaxiation time in velocity, γ/τ gives the ratio of characteristic time for heading reoriantation and velocity relaxiation of individual elements. Fig. 4(b) shows that the transition line is approximately $a \sim \tau^{-1/2}$, which seems to be nontrivial.

From similar plots, We obtain $r_c \sim \tau$ and $c \sim \tau$ as transition lines. The former can be interpreted as the balancing of collision time between neighboring elements and heading relaxiation time, and the later the balancing of velocity and heading relaxations. These proportionalities suggest that we use dimensionless parameters. Futhermore, we expect that the proportionalities would held for larger N>10, while the factors, *i.e.*, the positions of

transitions, might change. However, more specific transition lines, such as wandering/ swarming, are difficult to draw without introducing proper order parameters.

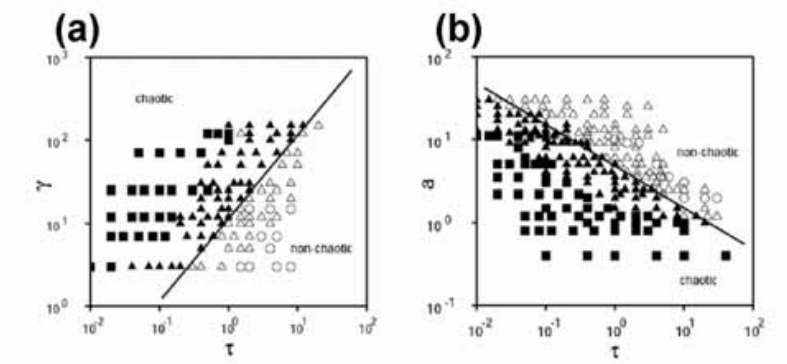


Figure 4. Parameter dependence of collective behavior. (a) γ versus τ. (b) *a* versus τ. Here, white circles indicate marching, white triangles oscillation, black triangles wandering, and black rectangles swarming

3.2 Dimensionless parameters

Next, we derive the dimensionless representation of our model and classify its behaviors. To reduce the model equation to a dimensionless form, we rescale each variable by a characteristic dimension: the characteristic length $L_0 \equiv r_c$ is comparable to the size of each individual, the steady state velocity $V_0 \equiv a/\gamma$ of elements, and the characteristic time $T_0 \equiv L_0 / V_0 = r_c \gamma / a$. Introducing the non-dimensional variables v', t', r' defined by $v = V_0 v'$, $t = T_0 t'$, $r = L_0 r'$, we obtain the following non-dimensional equations of motion for the *i*-th element (Shimoyama *et al.*, 1996):

$$\frac{d\vec{v}_i'}{dt'} = \frac{1}{R}(-\vec{v}_i' + \vec{n}_i - \frac{1}{Q}\sum_{j \neq i} \alpha_{ij} \vec{f}_{ij}), \tag{8}$$

$$\frac{1}{P}\frac{d\theta_i}{dt'} = \sin(\phi_i - \theta_i) + \sum_{j \neq i} J_{ij} \sin(\theta_j - \theta_i). \tag{9}$$

We have three independent dimensionless parameters *P,Q* and *R* defined by:

$$P \equiv \frac{r_c \gamma}{a\tau}, \quad Q \equiv \frac{a}{c}, \quad R \equiv \frac{ma}{\gamma^2 r_c}. \tag{10}$$

The physical interpretation of each parameter is: *P* is the ratio of the typical time scale for heading relaxation, τ and the "mean free time", $r_c\gamma/a$. *Q* is the ratio of the magnitude of the motive force and the interaction force with neighbors. *R* is the ratio of the inertial force and the viscous force, which is resembles a "Reynolds number" in fluid mechanics.

3.3 Phase diagram

We now review the numerical results, focusing on the marching/swarming transition, in the viscous regime where R < 0.05. Consider the dependence of the marching/swarming transition line in the phase diagram given in the previous section. At the transition line, we obtained $\gamma^* \sim \tau^*$, $a^* \sim \tau^{*1/2}$, $c^* \sim \tau^*$, and $r_c^* \sim \tau^*$, where * signifies the boundary between the states. Using a new dimensionless parameter defined by $G \equiv P/Q$, these relations simplify to

$$G = \frac{r_c^* \gamma^* c^*}{a^{*2} \tau^*} = const. \tag{11}$$

as shown in Fig. 5. All data from independent numerical simulations collapse onto the same representation, with the transition line at G^*=*const*.

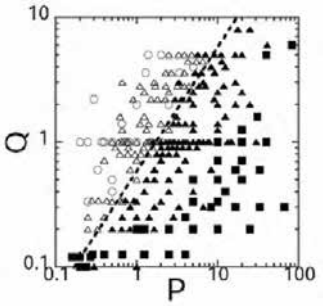


Figure 5. Phase diagram of collective motions in the viscous regime (R < 0.05) obtained by independent numerical simulations by changing parameters (*P versus Q*). In the diagram, white circles indicate marching, white triangles oscillation, black triangles wandering, and black rectangles swarming

3.4 Degree of disorder

To characterize the different collective motions quantitatively, we need suitable measures of disorder, *i.e.*, disorder parameters. In ordered motions (marching and oscillating), the trajectory of each element occupies a very limited region in velocity space. In chaotic motions, both temporal fluctuations of cluster velocity and velocity deviations of elements are large. Thus, we can define several disorder parameters. Letting the velocity of the cluster at a moment t be,

$$\vec{V}(t) = \frac{1}{N} \sum_i \vec{v}_i(t), \tag{12}$$

the fluctuation in velocity space can be evaluated by averaging the root mean square (*r.m.s.*) velocity deviation over time;

$$\left\langle (\Delta v)^2 \right\rangle \equiv \left\langle \frac{1}{N} \sum_i \left| \vec{v}_i(t) - \vec{V}(t) \right|^2 \right\rangle_t . \tag{13}$$

We can define a similar parameter, the fluctuation of $V(t)$ over time, by

$$\left\langle (\Delta V)^2 \right\rangle \equiv \left\langle \left(\vec{V}(t) - \left\langle \vec{V}(t) \right\rangle_t \right)^2 \right\rangle_t . \tag{14}$$

Both quantities are zero in ordered motions and non-zero in disordered motions. In the vicinity of the marching/swarming transition we calculated these quantities as a function of G as shown in Fig. 6. In the plot, both quantities are normalized by the average cluster velocity $<V^2>$, and the square root of the values is shown. Using these parameters, the order-disorder transition appears as a change in the disorder parameters. $<(\Delta V)^2>^{1/2}$ and $<(\Delta v)^2>^{1/2}$ are a feasible way to characterize ordered vs. disordered motions. Above the transition, the transition point, G^*, $<(\Delta v)^2>^{1/2}/<(\Delta V)^2>^{1/2}$ increases because the fluctuation of cluster motion approaches the cluster velocity. Fluctuations inside the cluster increase continuously as G increases. Swarming state corresponds to $<(\Delta V)^2>^{1/2}/<V^2>^{1/2} > 1$ and wandering and swarming states are continuous. It should be noted that the transition becomes sharper as N increases. The transition point G^* does not change. This suggests that the same transition can be seen for larger size of groupings.

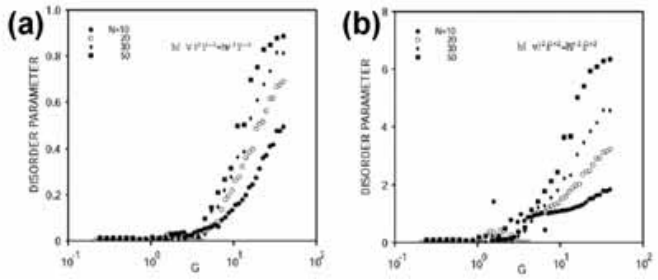


Figure 6. Characterization of the marching wandering transition in the viscous regime using disorder parameters. (a) $<(\Delta V)^2>^{1/2}/<V^2>^{1/2}$ and (b) $<(\Delta v)^2>^{1/2}/<(\Delta V)^2>^{1/2}$. The plots are made for several clusters of different size N from 10 to 50

3.5 Modification for formation control

Proposed model described above shows a variety of the group motions, however, it only shows a regular triangular crystal formation and its boundary is round when we focus on the formation of the group. In nature, we can observe other type of formations as well as spherical structure observed in small fish school or the swarms of small insects. Large migratory birds tend to form linear structure, which is considered to be hydrodynamically advantageous. In robotics, there are some researches which focus on the formation control of multi-robot (Balch & Arkin, 1998; Fredslund, 2002; Jadbabaie, 2003; Savkin, 2004), and most of them introduce a kind of geometrical formation rules.

Our interest is to express not only round-shaped structure but also other formations by modifying the above-mentioned model. In this section, modified model for formation control is explained, especially focusing on form of linear structure. Note that we just treat Newton's equation of motion for particles and do not treat geometrical rules.

The direction sensitivity is controlled by the parameter d in Eq.(4). From the simple analysis, we know that it is better to strengthen the direction sensitivity for linear formation. One of

the simplest way to strengthen the direction sensitivity is to use d^2 instead of d in Eq.(4). But for more drastic modification, we found it is more effective to replace r_c with $\alpha_{ij} \cdot r_c$ instead of strengthen d. Schematic images of interaction force are shown in Fig.7.

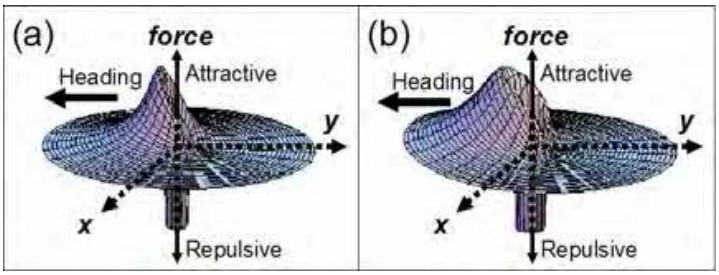


Figure 7. Images of interaction force in case of d=1, (a) r_c=*const.*, (b) r_c=a_{ij} x *const*

Fig.8 shows a typical behavior of the system based on the modified model. As you see, they organize a double line structure. This formation is stable and robust to perturbation.

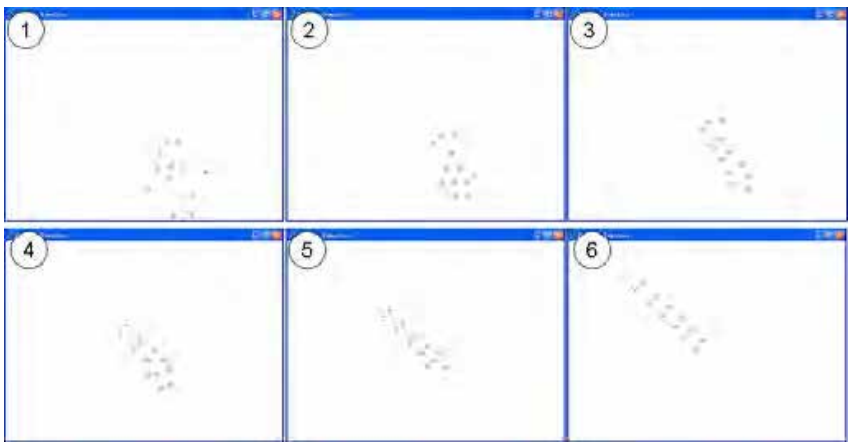


Figure 8. Self-organization of double line structure

We can also show that the angle between the forward direction and double line structure can be controled independently by modifying direction sensitivity.

$$\alpha_{ij} = 1 + d\cos(\Phi + \delta). \tag{15}$$

we can design the heading angle by δ. Fig.9 shows the process that the double line structure is organized, in which heading angle is controlled as $\delta = \pi/6$.

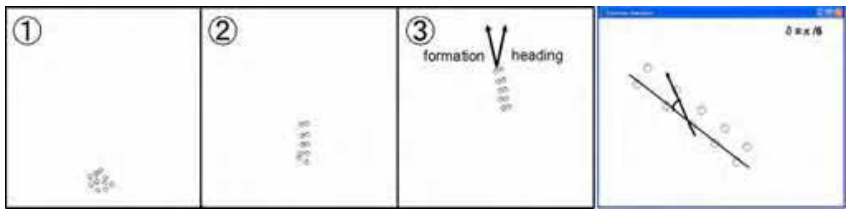


Figure 9. Self-organization of double line structure in case of $\delta=\pi/6$

4. Robot experiment

4.1 Collective behavior of the group

Performance of this system is also confirmed by the experiment of real robot system. Miniature mobile robots Khepera, which is one of the most popular robots for experiments, are used here. As the sensors on the robot are insufficient to measure the direction and the distance between the robots, positions and directions of the robot are measured by the overhead camera and each robot determines its behavior based on this information.

The model contains a degree of freedom for the heading. Khepera robot, however, has no freedom for heading. So we divide the movement of the robots into two phases: the phase to update the position, and the phase to update their directions. Fig. 10 shows the snapshot of the experiment and the trajectories of the robots in case of "marching", "Oscillatory", and "wandering."

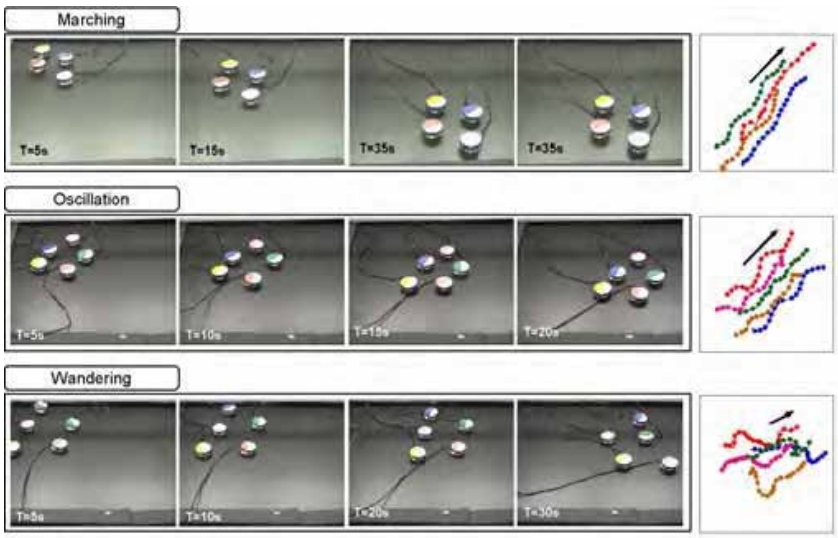


Figure 10. Snapshots of the experiment in case of "marching", "oscillatory" and "wandering" behaviour (pictures), and trajectories of the robots (plots)

4.2 Double line formation

The performance of the modified model is also confirmed by the robot experiment. The condition of the experiment is same as previous section. Fig.11 shows the snapshot of the experiment and the trajectories of the robots. We can see the robots organize double line formation.

Figure 11. Snapshots of the experiment in case of "double line formation"

5. Summary

In this article, we proposed a mathematical model which show several types of collective motions, and validated it. Firstly we constructed a model in which each element obeys the Newton equation with resistive and interactive force and has a degree of freedom of the heading vector which is parallel to the element axis, in addition to its position and velocity. Performance of the model was confirmed by numerical simulation, and we obtained several types of collective behavior, such as regular cluster motions, chaotic wandering and swarming of cluster without introducing random fluctuations. By introducing a set of dimensionless parameters, we formulated the collective motions and obtained the phase diagram and a new dimensionless parameter *G*. Lastly, we referred to the behaviour of extended model in which the anisotropy of the interaction force is modified, and showed the group organizes the double line formation.

6. References

Aoki, I. (1980). An Analysis of The Schooling Behavior of Fish: Internal Organization and Communication Process, *Bull. Res. Inst. Univ. Tokyo*, 12, pp.1-65.

Balch, T. & Arkin, R. C. (1998). Behavior-Based Formation Control for Multiagent Robot Teams, *IEEE Trans. on Robotics and Automation*, Vol.14, No.6, pp.926-939.

Balch, T. & Parker, L. E. (2002). *Robot Teams : From Diversity to Polymorphism*, A K Peters Ltd, ISBN:9781568811550

Bonabeau, E., Dorigo, M., & Theraulaz, G. (1999), *Swarm Intelligence: From Natural to Artificial Systems*, Oxford University Press, New York, ISBN:0-19-513159-2

Breder, C. M. (1954). Equations Descriptive of Fish Schools and Other Animal Aggregations, *Ecology*, 35, pp.361-370

Breder, C. M. (1976). Fish Schools as Operational Structures, *Fish. Bull.*, 74, pp.471-502.

Cao, Y. U., Fukunaga, A. S. and Kahng, A. B. (1997), "Cooperative Mobile Robotics: Antecedents and Directions, *Autonomous Robots*, 4, pp.7-27.

Doustari, M. A. & Sannomiya, N. (1992). A Simulation Study on Schooling Mechanism in Fish Behavior, *Trans. ISCIE*, 5, pp.521-523.

Edelstein-Keshet, L. (1990). Collective motion, In: *Lecture Notes in Biomathematics*, Alt, W. & Hoffmann, G., (Ed), pp.528-532.

Fredslund, J. & Mataric, M. J. (2002). A General, Local Algorithm for Robot Formations, *IEEE Trans. on Robotics and Automation*, Vol.18, No.5, pp.837-846.

Hunter, J. R.(1966). Procedure for Analysis of Schooling Behavior. *J. Fish. Res. Board Canada*, 23, pp.547-562.

Inoue, M. (1981). *Fish school; behavior* (in Japanese). Tokyo:Kaiyo-shuppan.

Jadbabaie, A., Lin, J. & Morse, A. S. (2003). Coordination of Groups of Mobile Autonomous Agents Using Nearest Neighbor Rules, *IEEE Trans. on Automatic Control*, Vol.48, No.6, pp.988-1001.

Niwa, H. (1994). Self-organizing Dynamic Model of Fish Schooling. *J. of Theor. Biol.*, 171, pp.123-136.

Ogren, P., Fiorelli, E. & Leonard, N. E. (2004). Cooperative Control of Mobile Sensor Networks: Adaptive Gradient Climbing in a Distributed Environment, *IEEE Transactions on Automatic Control*, Vol.49, No.8, 2004, pp.1292-1302.

Parker, L. E. (2003). Current Research in Multirobot Systems, *Artificial Life and Robotics*, vol. 7, pp.1-5.

Partridge, B. L. (1982). The Structure and Function of Fish Scholls. *Sci. Am.* 246, pp.90-99.

Savkin, A. (2004) Coordinated collective motion of groups of autonomous mobile robots: Analysis of vicsek's model, *IEEE Trans. on Automatic Control*, Vol.49, No.6, pp.981-983.

Shimoyama N., Sugawara K., Mizuguchi T., Hayakawa Y., Sano M. (1996). Collective Motion of a System of Motile Elements, *Phys. Rev. Lett.*, 79, pp.3870-3873.

Vicsek, T., Czirok, A., Ben-Jacob, E., Cohen I., & Shochet, O. (1995). Novel Type of Phase Transition in a System of Self-Driven Particles. *Phys. Rev. Lett.*, 75, pp.1226-1229.

Wilson, E. O. (1975). *Sociobiology*, Harvard.

21

Modeling and Control of Piezoelectric Actuators for Active Physiological Tremor Compensation

U-Xuan Tan[1], Win Tun Latt[1], Cheng Yap Shee[1], Cameron Riviere[2] and Wei Tech Ang[1]
[1]*Nanyang Technological University* [2]*Carnegie Mellon University*
[1]*Singapore,* [2]*United States*

1. Introduction

Humans have intrinsic limitations in manual positioning accuracy due to small involuntary movements that are inherent in normal hand motion. Among the several types of erroneous hand movements, physiological tremor is well studied and documented. Physiological tremor is roughly sinusoidal, in the frequency band of 8 – 12 Hz, and measures about 50 µm rms or more in each principal direction. Physiological hand tremor degrades the quality of many micromanipulation tasks and is intolerable in certain critical applications such as microsurgery and cell manipulation. In the human hand, humans are already in possession of a high dexterity manipulator with an unbeatable user interface. Hence, instead of replacing the human hand with a robotic manipulator, Riviere *et al.* [Riviere *et al.*, 2003] proposed a completely handheld ophthalmic microsurgical instrument, named Micron, that senses its own movement, distinguishes between desired and undesired motion, and deflects its tip to perform active compensation of the undesired component (Fig. 1).

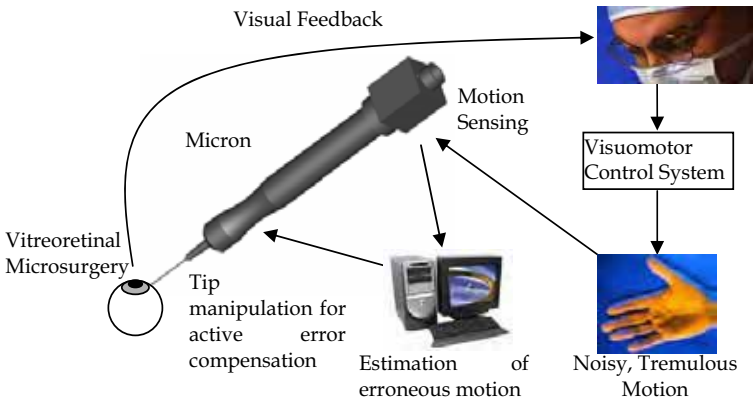


Figure 1. Overview of Micron

This active compensation approach presents several technical challenges in the control of the robotic mechanism that manipulates the intraocular shaft. The accuracy required can go down to a few microns for applications like microsurgery. To achieve that, the controller must be able to perform tracking control of the actuator to sub micron level. Physiological tremor is typically 8 – 12 Hz. Controlling the actuator to accurately track a motion of about 10 Hz is beyond the system bandwidth of many actuators. In order to actively compensate the tremor motion, real-time issue is another concern. Minimal phase difference is permitted as phase difference will result in larger tracking error. Most controllers, which introduce phase difference, are therefore not recommended. Thus, an open loop feedforward controller is proposed. To make things even more challenging, tremor is not rate-independent. The tremor frequency of a person modulates with type of motion and time.

Due to the high velocity and good resolution required, actuators involving smart materials like piezoelectric are proposed. However, their hysteretic behavior makes control difficult. In this chapter, the authors used the Prandtl-Ishlinskii (PI) hysteresis model to model the hysteretic behavior. The PI hysteresis model is a simple model. Its inverse can be obtained analytically, shortening the computational time and making it ideal for real-time application. Since the PI operator inherits the symmetry property of the backlash operator, a saturation operator is used to make it not symmetrical. The inverse model, also of the PI type, is used as the feedforward controller. A slight modification is also proposed to account for the one-sided characteristic of the actuator.

To accommodate human tremor's modulating frequency behavior, a rate-dependent hysteresis model is proposed. As the velocity or load increases, the slope of the hysteretic curve at the turning point tends to 0 and then negative, creating a singularity problem. This chapter also shows how the problem can be overcome by mapping the hysteresis through a transformation onto a singularity-free domain where the inversion can be obtained.

2. Piezoelectric Actuators

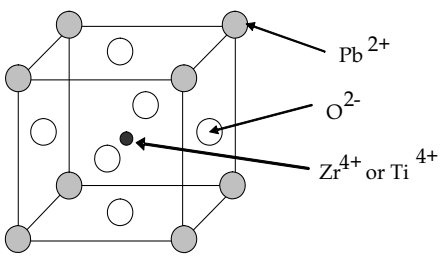


Figure 2. A Crystal Unit Cell of PZT Ceramics

Piezoelectric ceramic has been of increasing interest due to the developments in precision engineering and micro-positioning applications, especially in situations wherein precision, high frequency, and compactness are needed. Piezoelectric ceramic is also playing an increasing role in the medical industry as it is compatible with sensitive medical devices like MRI. Choi *et al*. [Choi *et al*., 2005] used piezoelectric actuators for their microsurgical instrument. One common example of piezoelectric ceramic is PZT ceramic. PZT is a solid

solution of $PbZrO_3$ and $PbTiO_3$ and the general formula is $Pb(Zr_yTi_{1-y})O_3$. PZT has the pervoskite ABO_3 structure (Fig. 2).
When a voltage is applied across the ceramic, the potential difference causes the atom at the centre (Zr or Ti) to displace (Fig. 3). A pole is thus induced and the net polarization in the PZT ceramic changes. This results in the deformation of the material. An opposite phenomenon occurs when the ceramic is loaded with a force. A change in polarization occurs and a voltage potential difference is induced. This explains why piezoelectric materials are commonly used both as actuators and sensors.

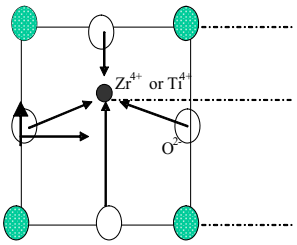


Figure 3. Polarization

A piezoelectric ceramic is an excellent choice because of its ability to output a large force, large operating bandwidth and fast response time. Unfortunately, effective employment of piezoelectric actuators in micro-scale dynamic trajectory-tracking applications is limited by two factors: (1) the intrinsic hysteretic behavior of piezoelectric material, and (2) structural vibration. The maximum hysteretic error is typically about 15%. To make matters worse, the hysteresis path changes according to rate (Fig. 4), as time is needed for the atoms to move and switching of the polarization to adjust and settle down. Landauer *et al.* [Landauer *et al.*, 1956] discussed about the dependence of the polarization, in barium titanate, on the rate at which the field is cycled.

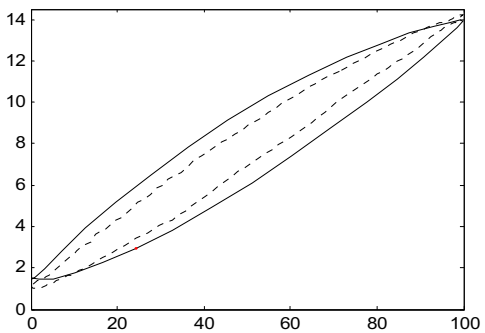


Figure 4. Hysteresis Path at different frequency

While research on rate-independent control of piezoelectric actuators has been extensive, there have been few attempts and little success at controlling the actuator at varying frequency. Hysteresis modeling or compensation can be generally classified into 5

categories: (1) Linear control with feedforward inverse hysteresis model; (2) Microscopic theories; (3) Electric Charge Control; (4) Phase Control; and (5) Closed-loop displacement control. The more recent methods comprise a hybrid of the methods.

Category (1) relates the underlying understanding of the material at microscopic level with respect to displacement. Landauer *et al.*, 1956 discussed the dependence of the polarization, in barium titanate, on the rate at which the field is cycled. Category (2) makes use of the knowledge that the hysteresis of the actuator's displacement to the applied voltage is about 15% while the displacement to induced charge is 2%. This motivated Furutani *et al.* [Furutani *et al.*, 1998] to combine induced charge feedback with inverse transfer function compensation. Category (3) includes Cruz-Hernandez & Hayward [Cruz-Hernandez & Hayward, 1998; 2001] proposing the idea of considering phase as a control approach to design a compensator to reduce hysteresis. Category (4) consists of many different approaches. Some proposed incorporating inverse hysteresis model with a controller while others proposed advance controllers like neural network [Hwang *et al.* 2003], fuzzy logic [Stepanenko *et al.* 1998], sliding mode [Abidi *et al.* 2004] and H_∞ control [Chen *et al.* 1999].

Category (5), a phenomenological approach, is about obtaining a mathematical representation of the hysteresis motion through observation. Phenomenological approach is more commonly used because the underlying physics of the relationship of the smart materials like piezo-actuator's hysteresis path with rate and load is not well understood. Thus, there are many different attempts to derive different mathematical models that best describe the complex hysteretic motion. The inverse model is then used as a feedforward controller to linearize the hysteresis response as shown in Fig. 5.

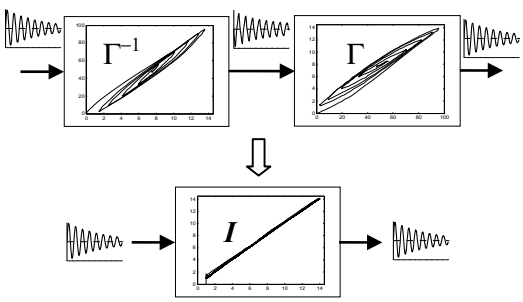


Figure 5. Linearization of Hysteresis Model using Inverse Feedforward Controller

A number of hysteresis mathematical models have been proposed over the years. Hu *et al.* [Hu *et al.*, 2002] and Hughes *et al.* [Hughes *et al.*, 1995] proposed using the Preisach model while Goldfarb *et al.* [Goldfarb *et al.*, 1996; 1997] and Choi *et al.* [Choi *et al.*, 1997] used Maxwell's model. Tao [Tao, 1995] used the hysteron model. The more recent papers are a variation from the classical models to avoid certain conditions.

Another model is the Prandtl-Ishlinskii model. [Kuhnen & Janocha, 2001; 2002] and [Janocha & Kuhnen, 2000] demonstrated that the classical Prandtl-Ishlinskii operator is less complex and its inverse can be computed analytically. Thus, it is more suitable for real-time applications because minimal mathematical computation time is required. Unfortunately, to use the model, the operating frequency must not be too high as the hysteresis non-linearity becomes more severe. Like most models, the classical Prandtl-Ishlinskii model is unable to

function as a feedforward controller when the largest displacement does not occur at the highest input signal (Fig. 6) as singularity occurs in the inverse.

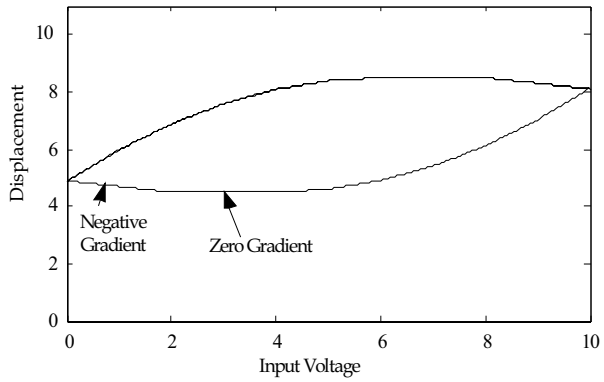


Figure 6. Ill-Conditioned Hysteresis

In this chapter, there are two main contributions: (1) In order to accommodate human tremor's modulating frequency behaviour, a rate-dependent feedforward controller is proposed; and (2) a solution to the inverse of the ill-conditioned hysteresis because as the velocity or load increases, the slope of the hysteretic curve at the turning point tends to 0 and then negative, creating a singularity problem. This is achieved by mapping the hysteresis through a transformation onto a singularity-free domain where the inversion can be obtained.

3. Hysteresis Modeling

3.1 Prandtl-Ishlinskii (PI) Operator

The elementary operator in the PI hysteresis model is a rate-independent backlash operator. It is commonly used in the modeling of backlash between gears with one degree of freedom. A backlash operator is defined by

$$y(t) = H_r[x, y_0](t) = \max\{x(t) - r, \min\{x(t) + r, y(t - T)\}\} \tag{1}$$

where x is the control input, y is the actuator response, r is the control input threshold value or the magnitude of the backlash, and T is the sampling period. The initial condition of (1) is normally initialised as

$$y(0) = \max\{x(0) - r, \min\{x(0) + r, y_0)\}\} \tag{2}$$

where $y_0 \in \Re$, and is usually but not necessarily initialized to 0. Multiplying the backlash operator H_r by a weight value w_h, the generalized backlash operator is

$$y(t) = w_h H_r[x, y_0](t) . \tag{3}$$

The weight w_h defines the gain of the backlash operator ($w_h = y/x$, hence w_h =1 represents a 45° slope) and may be viewed as the gear ratio in an analogy of mechanical play between gears, as shown in Fig.7.

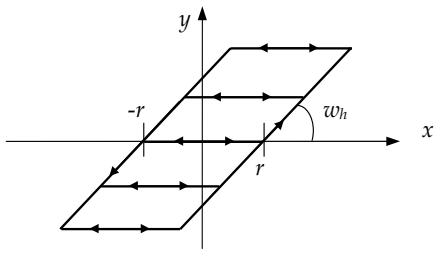


Figure 7. The rate-independent generalized backlash operator is characterized by the threshold or backlash magnitude, r, and the weight or backlash operator gain, w_h

Complex hysteretic nonlinearity can be modeled by a linearly weighted superposition of many backlash operators with different threshold and weight values,

$$y(t) = \vec{w}_h^T \vec{H}_r[x, \vec{y}_0](t)\,, \tag{4}$$

with weight vector $\vec{w}_h^T = [w_{h0} \ldots w_{hn}]$ and $\vec{H}_r[x, \vec{y}_0](t) = [H_{r0}[x, y_{00}](t) \ldots H_{rn}[x, y_{0n}](t)]^T$ with the threshold vector $\vec{r} = [r_0 \ldots r_n]^T$ where $0 = r_0 < \ldots < r_n$, and the initial state vector $\vec{y}_0 = [y_{00} \ldots y_{0n}]^T$. The control input threshold values $\vec{r}$ are usually, but not necessarily, chosen to be equal intervals. If the hysteretic actuator starts in its de-energized state, then $\vec{y}_0 = \vec{0}_{n\times 1}$.

Equation (4) is the PI hysteresis operator in its threshold discrete form. The hysteresis model formed by the PI operator is characterized by the initial loading curve (Fig. 8). It is a special branch traversed by equation (4) when driven by a monotonically increasing control input with its state initialized to zero (i.e. $y(0) = 0$). The initial loading curve is defined by the weight values $\vec{w}_h$ and threshold values $\vec{r}$,

$$\varphi(r) = \sum_{j=0}^{i} w_{hj}(r - r_j),\ r_i \le r < r_{i+1},\ i = 0, \ldots, n. \tag{5}$$

The slope of the piecewise linear curve at interval i is defined by W_{hi}, the sum of the weights up to i,

$$W_{hi} = \frac{d}{dr}\varphi(r) = \sum_{j=0}^{i} w_{hj}\,. \tag{6}$$

The subsequent trajectory of the PI operator beyond the initial loading curve with non-negative control input is shown as the dotted loop in Fig. 8. The hysteresis loop formed by the PI operator does not return to zero with the control input. This behaviour of the PI operator closely resembles the hysteresis of a piezoelectric actuator.

The backlash operators cause each of the piecewise linear segments to have a threshold width of $2r$ beyond the initial loading curve. As such, there is no need to define any

backlash operators beyond the midpoint of the control input range, i.e. $r_n \leq$ ½max{control input} [Ang 2003]. This also implies that the backlash operators have descending importance from the first to the last, since the first operator is always used and the subsequent operators are only used when the control inputs go beyond their respective threshold values, r_i's. Moreover, observations from the piezoelectric hysteretic curves suggest that more drastic changes in the slope occur after the turning points, i.e. in the region of the first few backlash operators. To strike a balance between model accuracy and complexity, the authors propose to importance-sample the threshold intervals $\vec{r}$, i.e., to have finer intervals for the first few backlash operators and increasing intervals for the subsequent ones.

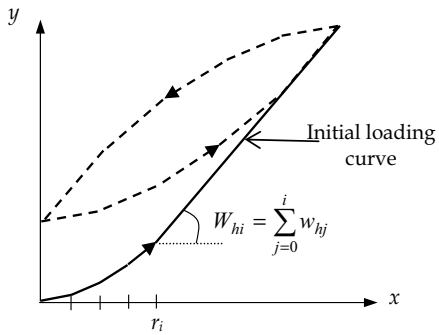


Figure 8. The PI hysteresis model with n = 4. The hysteresis model is characterized by the initial loading curve. The piecewise linear curve is defined by the equally spaced threshold values $\vec{r}$ and the sum of the weight values $\vec{w}_h$.

3.2 Modified Prandtl-Ishlinskii (PI) Operator

The PI operator inherits the symmetry property of the backlash operator about the center point of the loop formed by the operator. The fact that most real actuator hysteretic loops are not symmetric weakens the model accuracy of the PI operator. To overcome this overly restrictive property, a saturation operator is combined in series with the hysteresis operator. The general idea is to bend the hysteresis. A saturation operator is a weighted linear superposition of linear-stop or one-sided dead-zone operators. A dead-zone operator is a non-convex, asymmetrical, memory-free nonlinear operator (Fig. 9). A one-sided dead-zone operator and a saturation operator are given by

$$S_d[y](t) = \begin{cases} \max\{y(t) - d, 0\}, & d > 0 \\ y(t), & d = 0 \end{cases} \tag{7}$$

$$z(t) = \vec{w}_s^T \vec{S}_d[y](t), \tag{8}$$

where y is the output of the hysteresis operator, z is the actuator response, $\vec{w}_s^T = [w_{s0} \ldots w_{sm}]$ is the weight vector, $\vec{S}_d[y](t) = [S_{d0}[y](t) \ldots S_{dm}[y](t)]^T$ with the threshold vector $\vec{d} = (d_0 \ldots$

$d_m)^T$ where $0 = d_0 < d_1 < \ldots < d_m$. For convenience, intervals of $\vec{d}$ between d_0 and d_m need not be equal. Good selection of $\vec{d}$ depends on the shape of the hysteresis loop, and typically involves some trials and errors.
The modified PI operator is thus

$$z(t) = \Gamma[x](t) = \vec{w}_s^T \vec{S}_d \left[\vec{w}_h^T \vec{H}_r [x, \vec{y}_0] \right](t) . \tag{9}$$

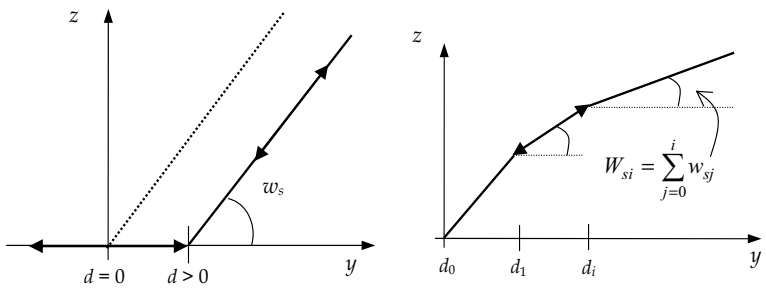


Figure 9. (a) The one-sided dead-zone operator is characterized by the threshold, d, and the gain, w_s. (b) The saturation operator with $m = 2$. The slope of the piecewise linear curve at interval i, W_{si} is defined by the sum of the weights up to i.

3.3 Parameter Identification

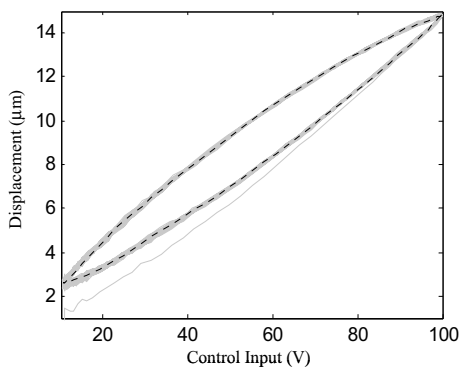


Figure 10. The lighter solid lines are the measured piezoelectric actuator response to a 10 Hz, 12.5 μm p-p sinusoidal control input. The dark dotted line is the identified modified PI hysteresis model with 10 backlash operators ($n = 9$) and 4 dead-zone operators ($m = 3$).

To find the hysteresis model parameters as shown in Fig. 10, we first have to measure experimentally the responses of the piezoelectric actuator to periodic control inputs. A good set of identification data is one that covers the entire operational actuation range of the piezoelectric actuator at the nominal operating frequency. Next decide the order of the PI operator (n) and the saturation operator (m), and set the threshold values $\vec{r}$ and $\vec{d}$ as described in the previous section. The weight parameters $\vec{w}_h$ and $\vec{w}_s$ are found by performing a least-squares fit of (9) to the measured actuator response, minimizing the error equation which is linearly dependent on the weights:

$$E[x,z]\left(\vec{w}_h,\vec{w}_s,t\right)=\left[\vec{w}_h^T\vec{H}_r[x(t),\vec{y}_0]\right](t)-\vec{w}'^T_s\,\vec{S}_{d'}z(t)\,. \tag{10}$$

Fig. 10 shows superposition of the identified modified PI hysteresis model on the measured piezoelectric actuator response, subjected to a sinusoidal control input.

3.4 Inverse Modified Prandtl-Ishlinskii (PI) Operator

The key idea of an inverse feedforward controller is to cascade the inverse hysteresis operator, Γ^{-1}, with the actual hysteresis which is represented by the hysteresis operator, Γ, to obtain an identity mapping between the desired actuator output $\hat{z}(t)$ and actuator response $z(t)$,

$$z(t)=\Gamma\left[\Gamma^{-1}[\hat{z}]\right](t)=I[\hat{z}](t)=\hat{z}(t) \tag{11}$$

The operation of the inverse feedforward controller is depicted in Fig.6.
The inverse of a PI operator is also of the PI type. The inverse PI operator is given by

$$\Gamma^{-1}[\hat{z}](t)=\vec{w}'^T_h\,\vec{H}_{r'}\left[\vec{w}'^T_s\,\vec{S}_{d'}[\hat{z}],\vec{y}'_0\right](t) \tag{12}$$

where the inverse modified PI parameters can be found by

$$w'_{h0}=\frac{1}{w_{h0}}\;;\;w'_{hi}=\frac{-w_{hi}}{(\sum_{j=0}^{i}w_{hj})(\sum_{j=0}^{i-1}w_{hj})}\,,\;i=1\ldots n;$$

$$r'_i=\sum_{j=0}^{i}w_{hj}(r_i-r_j),\;\;y'_{0i}=\sum_{j=0}^{i}w_{hj}y_{0i}+\sum_{j=i+1}^{n}w_{hj}y_{0j}\,,\;i=0\ldots n; \tag{13}$$

$$w'_{s0}=\frac{1}{w_{s0}}\;;\;w'_{si}=\frac{-w_{si}}{(\sum_{j=0}^{i}w_{sj})(\sum_{j=0}^{i-1}w_{sj})}\,,\;i=1\ldots m;$$

$$d'_i=\sum_{j=0}^{i}w_{sj}(d_i-d_j),\;i=0\ldots m; \tag{14}$$

4. Rate-Dependent Phenomena

Most, if not all, of the present mathematical models are defined rate-independent mathematically. This is too restrictive in real life. In this section, a rate-dependent hysteresis model is proposed.

4.1 Rate-dependent Hysteresis Slope

In this section, an extension to the modified PI operator is proposed in order to also model the rate-dependent characteristics of the piezoelectric hysteresis is proposed. One of the advantages of the PI hysteresis model is that it is purely phenomenological; there are no direct relationships between the modeling parameters and the physics of the hysteresis. While the rate dependence of hysteresis is evident from Fig. 4, the sensitivity of actuator saturation to the actuation rate is not apparent. Hence, assuming that saturation is not rate-dependent and hold the saturation weights, $\vec{w}_s$, as well as the threshold values, $\vec{r}$ and $\vec{d}$, constant a relationship between the hysteresis and the rate of actuation $\dot{x}(t)$ is constructed. The hysteresis slope (i.e., sum of the PI weights) at time *t* as a rate-dependent function is

$$W_{hi}(\dot{x}(t)) = \hat{W}_{hi} + f(\dot{x}(t)),\ i = 1 \ldots n.; \tag{15}$$

where

$$\dot{x}(t) = \frac{x(t) - x(t-T)}{T}, \dot{x}(0) = 0. \tag{16}$$

4.2 Rate-dependent Model Identification

The piezoelectric actuator, subjected to periodic constant-rate or sawtooth control inputs. Measurements were made over a frequency band whose equivalent rate values cover the entire operational range of the actuation rates. For example, in an application tracking sinusoids of up to 12.5 µm p-p in the band of 1 to 19 Hz, the operational range of the actuation rate is from 0 to 746 µm/s, which corresponds to the rate of 12.5 µm p-p sawtooth waveforms of up to about 60 Hz. PI parameter identification is then performed on each set of measured actuator responses. The sum of the hysteresis weights W_{hi}, $i = 0 \ldots n$, of each identification is then plotted against the actuation rate $\dot{x}(t)$ and shown in Fig. 11.

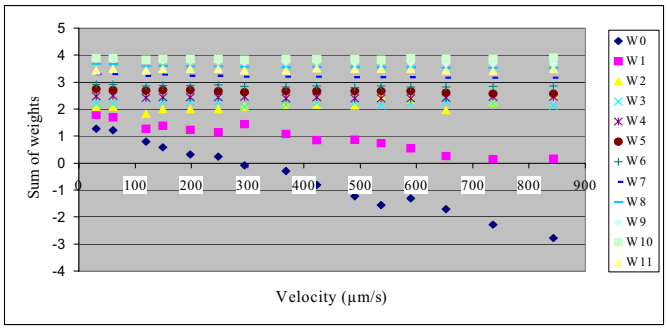


Figure 11. Plot of Sum of hysteresis weights against actuation rate

From Fig. 11, it can be seen that the hysteresis slope of the piezoelectric actuator can be modelled as linear to the velocity input with good approximation. Thus the rate-dependent hysteresis slope model would be:

$$W_{hi}(\dot{x}(t)) = \hat{W}_{hi} + c_i \dot{x}(t),\ i = 0 \ldots n \tag{17}$$

where c_i is the slope of the best fit line through the W_{hi}'s and the referenced slope, $\hat{W}_{hi}$, is the intercept of the best fit line with the vertical W_h axis or the slope at zero actuation. The individual rate-dependent hysteresis weight values can be calculated from

$$w_{hi}(\dot{x}(t)) = W_{hi}(\dot{x}(t)) - W_{h(i-1)}(\dot{x}(t)),\ i = 1 \ldots n;$$

$$w_{h0}(\dot{x}(t)) = W_{h0}(\dot{x}(t)) \,. \tag{18}$$

4.3 Rate-dependent Modified Prandtl-Ishlinskii Operator

The rate-dependent modified PI operator is defined by

$$z(t) = \Gamma[x, \dot{x}](t) \ = \vec{w}_s^T \vec{S}_d \left[\vec{w}_h^T(\dot{x}) \vec{H}_r[x, \vec{y}_0]\right](t) \tag{19}$$

The inverse rate-dependent modified PI operator is also of the PI type:

$$\Gamma^{-1}[\hat{z}](t) = \vec{w}'^T_h(\dot{x}) \vec{H}_{r'}\left[\vec{w}'^T_s \vec{S}_{d'}[\hat{z}], \vec{y}'_0\right](t) \,. \tag{20}$$

The inverse rate-dependent parameters can be found by (13), replacing $\vec{w}_h$ with the rate-dependent $\vec{w}_h(\dot{x})$,

$$w'_{h0}(\dot{x}(t)) = \frac{1}{w_{h0}(\dot{x}(t))};$$

$$w'_{hi}(\dot{x}(t)) = \frac{-w_{hi}(\dot{x}(t))}{W_{hi}(\dot{x}(t)) W_{h(i-1)}(\dot{x}(t))},\ i = 1 \ldots n;$$

$$r'_i = \sum_{j=0}^{i} w_{hj}(\dot{x}(t))(r_i - r_j),\ \ i = 0 \ldots n;$$

$$y'_{0i} = \sum_{j=0}^{i} w_{hj}(\dot{x}(t)) y_{0i} + \sum_{j=i+1}^{n} w_{hj}(\dot{x}(t)) y_{0j},\ i = 0 \ldots n. \tag{21}$$

5. Motion Tracking Experiments

Two motion tracking experiments were performed to demonstrate the rate-dependent feedforward controller. The first experiment compares the performance of the open loop feedforward controllers driven at fix frequencies. The rate-independent controller is based on the modified PI hysteresis model identified at the 10Hz at 12.5μm peak to peak sinusoid. The second experiment is tracking a multi-frequency (1, 10 and 19 Hz) nonstationary motion profile.

5.1 Experiment Setup

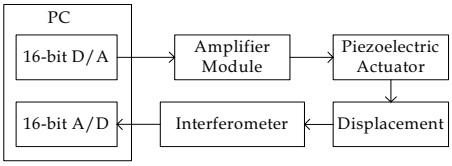


Figure 12. Experimental Architecture

As seen from Fig. 12, a 16-bit D/A card is used to give out the necessary voltage, which is then passed through the amplifier (the gain is approximately 10). Given the voltage, the actuator will move and the interferometer will detect the displacement and convert it to analog voltage. Using a 16-bit A/D card, the PC reads in the displacement.

5.2 Stationary Sinusoid Experiment

The first experiment compares the performance of the rate-independent and rate-dependent modified PI models based open-loop feedforward controllers in tracking 12.5 μm p-p stationary sinusoids at 1, 4, 7, 13, 16 and 19 Hz. The tracking rmse and maximum error of each controller at each frequency are summarized in Table 1 and plotted in Fig. 13.

	Without Model		Rate-independent		Rate-dependent	
Freq. (Hz)	rmse (μm)	max ε (μm)	rmse (μm)	max ε (μm)	rmse (μm)	max ε (μm)
1	1.13	2.11	0.25	0.63	0.21	0.57
4	1.12	2.07	0.19	0.67	0.16	0.46
7	1.23	2.24	0.18	0.52	0.16	0.50
10	1.19	2.26	0.14	0.46	0.17	0.47
13	1.21	2.31	0.19	0.53	0.17	0.55
16	1.30	2.49	0.27	0.59	0.17	0.53
19	1.37	2.61	0.34	0.70	0.18	0.59
Mean	1.22	2.30	0.23	0.59	0.18	0.52
$\pm\sigma$	± 0 09	± 0.19	± 0.07	± 0.8	± 0.02	±0.05

The rmse's and max errors are the mean results over a set of three 5-second (5000 data points) experiments.

Table 1. Measured Performance of the Rate-Independent and Rate-Dependent Inverse Feedforward Controllers

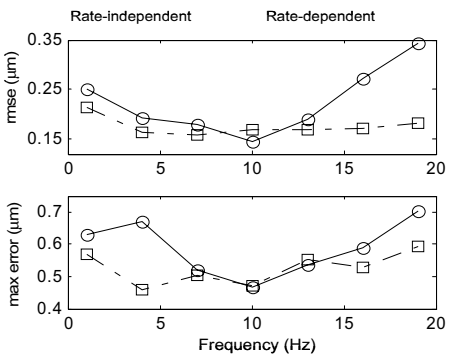
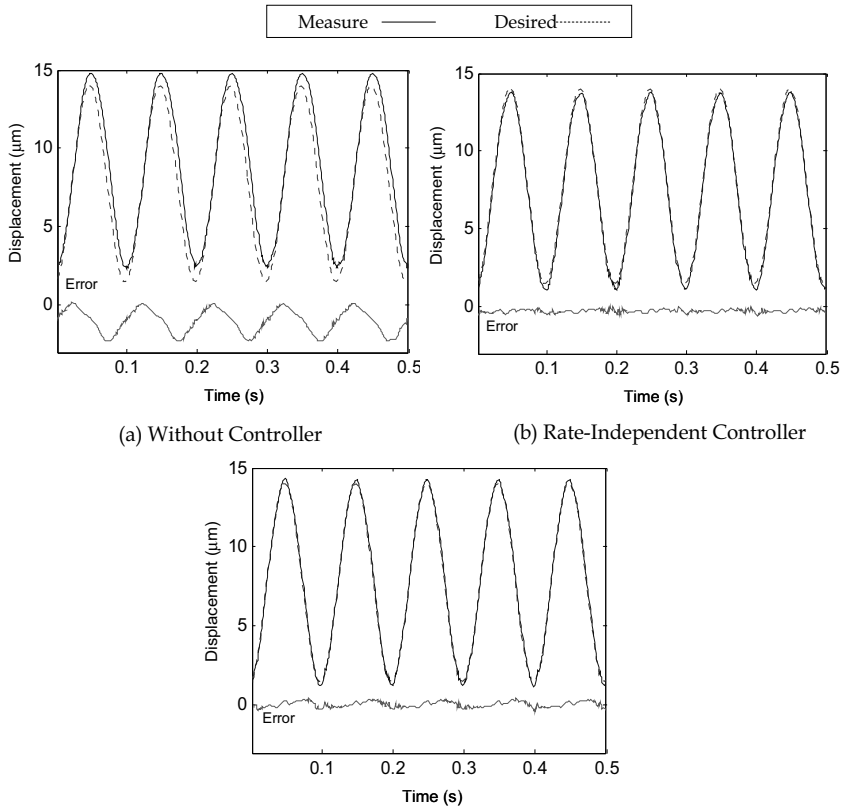


Figure 13: Experimental tracking results of different controllers for stationary 12.5 μm at 10 Hz

As shown in Fig. 13, at 19 Hz, the tracking rmse of the rate-independent controller is almost double that of the rate-dependent controller and will continue to worsen as the frequency increases. Fig. 14 shows the results of the different controllers. Fig. 14(a) plots the hysteretic response of the piezoelectric actuator with a proportional controller. Fig. 14(b) and Fig. 14(c) presents the tracking ability of the rate-independent and rate-dependent inverse feedforward controllers respectively. The rate-independent controller is based on the modified PI hysteresis model identified at the same 10Hz, 12.5 μm p-p sinusoid.
Both the rate-independent and rate-dependent controllers significantly reduced the tracking error due to the piezoelectric hysteretic behaviour. However, the tracking accuracy of the rate-independent controller deteriorates when the frequency deviates from 10 Hz. Meanwhile, the rate-dependent controller maintained a smaller rmse and maximum error.



(a) Without Controller

(b) Rate-Independent Controller

(c) Rate-Dependent Controller

Figure 14. Rmse and maximum errors of the rate-independent and rate-dependent controllers in tracking 12.5μm p-p stationary sinusoids at different frequencies

5.3 Multi-Frequency Nonstationary Experiment

The second experiment is an experiment to test the ability of the controllers to track a multi-frequency nonstationary motion profile. Both the feedforward controllers do improve the tracking capability. However, the rate-dependent controller did noticeably better. The result is shown in Fig. 15 and summarised in table 2.

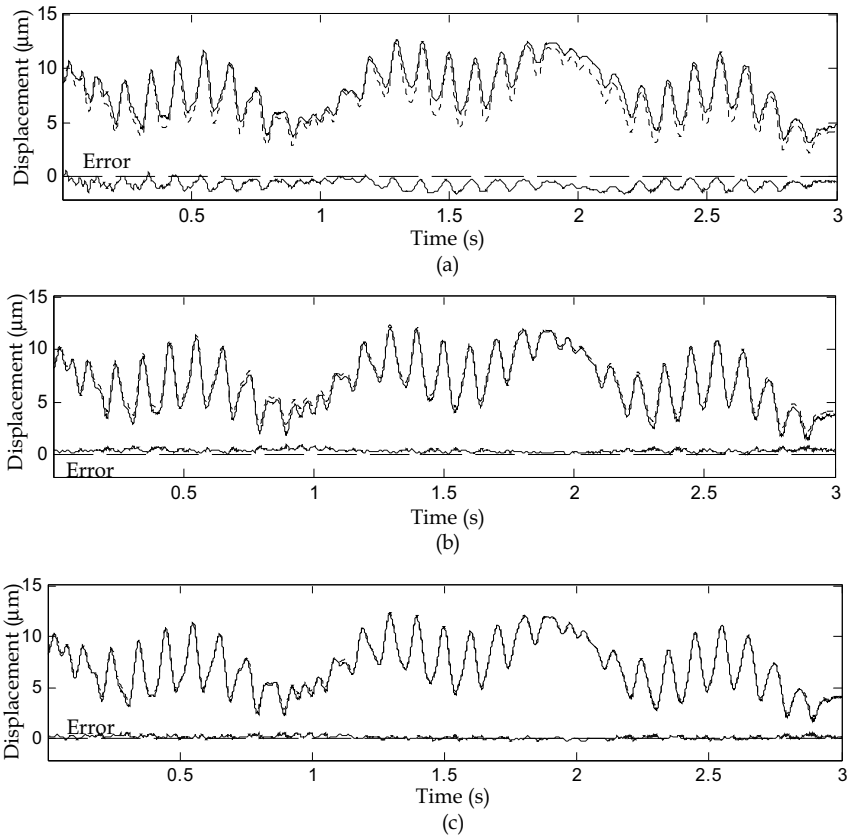


Figure 15. Experimental open-loop tracking results of a multi-frequency, nonstationary dynamic motion profile. The motion profile is made up of superimposed modulated 1, 10, and 19 Hz sinusoids with time-varying amplitudes. The rate-independent controller is based on the modified PI hysteresis model identified at the same 10Hz, 12.5 μm p-p sinusoid. Transient error is observed for the rate-independent controller in the first 2 seconds.(a) Without compensation. (b) Rate-independent controller. (c) Rate-dependent controller

	Without model	Rate-independent	Rate-dependent
rmse $\pm \sigma$ (μm)	1.02 ± 0.07	0.31 ± 0.03	0.15 ± 0.003
$\frac{\text{rmse}}{\text{p - p amplitude}}(\%)$	9.2	2.8	1.4
max error $\pm \sigma$ (μm)	1.91 ± 0.08	0.89 ± 0.04	0.59 ± 0.06
$\frac{\text{max error}}{\text{p - p amplitude}}(\%)$	17.3	8.0	5.3

The rmse and max errors are the mean results over a set of seven 5-second (5000 data points) experiments.

Table 2. Measured Performance of the Rate-Independent and Rate-Dependent Inverse Feedforward Controllers in Tracking Multi-Frequency (1, 10 and 19 Hz) Nonstationary Signals

The rate-dependent controller registers a tacking rmse less than half of that of the rate-independent controller. Maximum tracking errors for both controllers occur in the transient phase. This might explain why the improvement in maximum error with the rate-dependent controller is not as large as the improvement in rmse. One limitation of all PI-type hysteresis models is that singularity occurs when the first PI weight is 0 as seen in equation (13). Also, when the slope is negative, the inverse hysteresis loading curve violates the fundamental assumption that it should be monotonically increasing. Thus, the inverse model will be lost. In order to maintain a good tracking accuracy for high velocity by having small threshold intervals, a method to solve the singularity problem is proposed in the next section.

6. Using a different Domain to solve Singularity Problem

The PI operator, while being able to model the hysteresis behaviour of a piezoelectric actuator well, has one major inadequacy: the inverse of the operator does not exist when the slope of the hysteretic curve is not positive definite, i.e. singularity occurs when the PI weights ≤ 0. Such ill conditioned situations arise when the piezoelectric actuators are used to actuate heavy loads or when operating at high frequency. Another possible situation for ill condition is when small intervals between the threshold values are used. Presently, most people avoid this problem by having larger intervals between the threshold values. However, this is not solving the problem and resulted in higher error around the turning point.

This section presents how the authors managed to overcome this problem by mapping the hysteresis through a linear transformation onto another domain, where the inversion would be better behaved. The inverse weights are evaluated in this domain and are subsequently used to compute the inverse hysteresis model, which is to be used in the feedforward controller, before the inverse model is transformed back to the original domain. The singularity problem is first illustrated, followed by the solution to map the ill-conditioned hysteresis onto a singularity-free domain.

6.1 Illustration of Problem

As seen in Fig. 5, using the inverse as a feedforward controller linearizes the response. Unfortunately, to use the hysteresis model, the operating frequency must not be too high as the hysteresis non-linearity will become more severe and like most models, the classical

Prandtl-Ishlinskii model is also unable to function as a feedforward controller when the largest displacement does not occur at the highest input signal (Fig. 6). The inverse model equation (13) fails when the convex curve is encountered.

Most systems can be approximated as a spring mass damper system. When driven at high velocity, the actuator/mechanism has a high momentum at the turning point, especially if a rapid change is made. The large momentum tends to keep the system in motion and the large momentum results in the convex curve. Similar explanation is applicable for large loads.

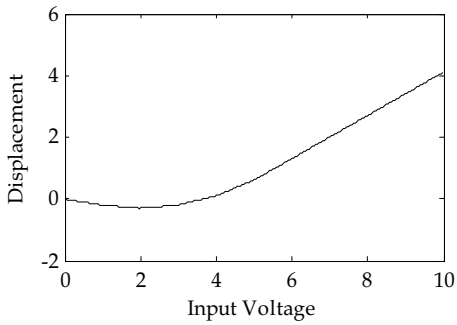


Figure 16. Loading Curve of Hysteresis example involving negative gradient

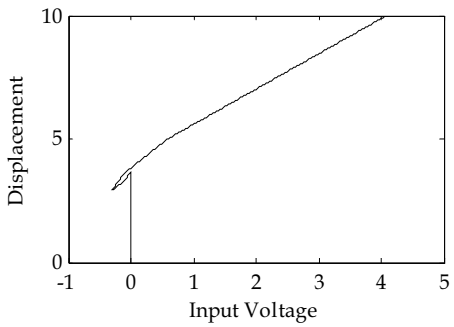


Figure 17. Inverse Loading Curve of example to illustrate failure of PI inverse operator when negative gradient is encountered

There is also an inevitable trade-off between modeling accuracy and inversion stability. The modeling of the hysteretic loop gets better with the number of backlash operators used in the modeling. However, as the piecewise continuous interval represented by each backlash operator shrinks, there is a greater chance for the reciprocal of the PI weights to be ill conditioned, especially at the hysteretic curve turning points. An example to show that the inverse model equation (13) fails when the convex curve is encountered is illustrated here. Given weights of $\vec{w}_h^T$ = [-0.2 0.1 0.2 0.2 0.2 0.2] and $\vec{r}$ = [0 1 2 3 4 5] for an application where the amplitude of the periodic input voltage is 10V. The loading curve is shown in Fig. 16.

Applying equation (13) to the get the inverse PI parameters, we obtain $\vec{w}_h'^T$ = [-5 -5 20 -6.6667 -1.333 -0.5714] and $\vec{r}'$ = [0 -0.2 -0.3 0.1 0.6]. Fig. 17 illustrates the inverse curve that will be obtained using equation (13). The two graphs are not a reflection of each other along the 45 degrees line. This simple proof clearly illustrates that equation (13) has failed as an inverse function when the condition of positive gradient is not met. Zero gradient is not demonstrated in this example as it is clear that the reciprocal of 0 is a singular point.

6.2 Obtaining Inverse Model in a Different Domain

6.2.1 Intuition of Proposed Method

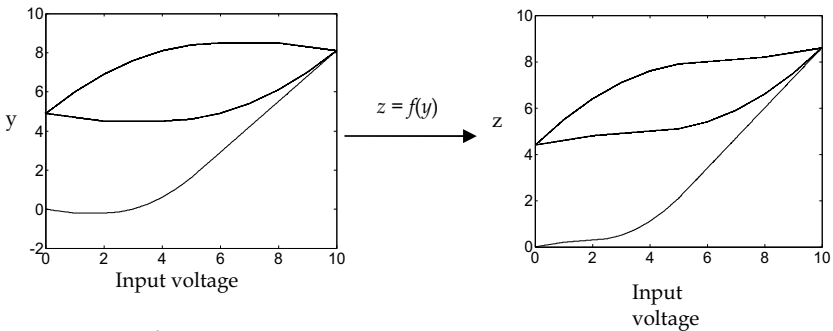


Figure 18. Transformation

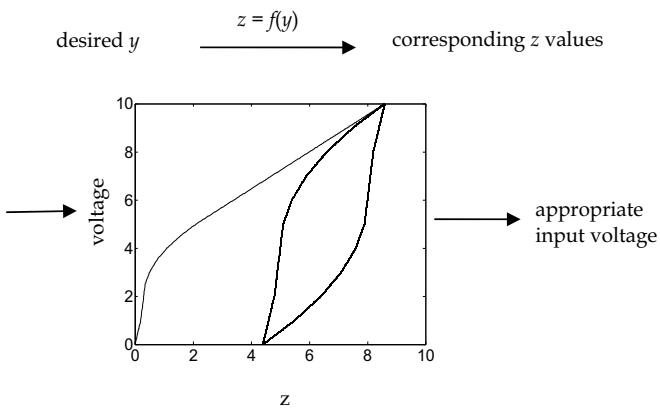


Figure 19. Method to Obtain Appropriate Input Voltage

With the singularity problem, the author came up with the idea to model the hysteresis in an alternative domain when the inverse of the PI model fails in a situation like the largest displacement not occurring at the highest input signal (Fig. 6). A transformation is used to map y (the original hysteretic displacement values) to z, which has no singular points as shown in Fig. 18. The inverse model can now be obtained in the new domain. The

appropriate input voltages can now be obtained using the inverse model found in the new domain as shown in Fig. 19.
The saturation operator is just another transformation and thus can be ignored for the time being. The inverse of the saturation operator can be applied after the inverse of the singularity-free model.
The desired value displacement *y* is first passed through the transformation function to obtain the corresponding new domain z value. This z value is then passed through the inverse model obtained in the new domain to get the appropriate input voltages.

6.2.2 Obtaining the Inverse Model in a different Domain

Although the inverse of the PI model fails, PI model can still describe the pneumonia path like figure 4. The PI parameters for the ill-conditioned hysteresis can be obtained as shown in section 2. Recall that equation (2) can describe the hysteresis.

$$y(t) = \vec{w}_h^T \vec{H}_r[x, \vec{y}_0](t) \tag{3}$$

Using least square method, $\vec{w}_h^T$ is be obtained. To illustrate the transformation, nine points (x_0 to x_8) are labelled in Fig. 20. Negative gradient for the loading curve occurs between x_0 and x_1 while 0 gradient is between x_1 and x_2. In the hysteresis loop, region between x_3 to x_4 and x_6 to x_7 has negative gradient while x_4 to x_5 and x_7 to x_8 contain gradient value 0.
The labelled points x_0 to x_8 can be calculated using:

$$\begin{aligned}
&x_0 = 0;\\
&\text{If } \sum_{j=1}^{i} w_{h_j} = 0 \text{ exists}, x_1 = r_i; x_2 = r_{i+1}\\
&\text{otherwise}, x_1 = x_2 = r_{\max} \quad \text{where } \sum_{j=1}^{\max} w_{h_j} > 0\\
&\qquad x_3 = 2r_{\max};\\
&\qquad x_4 = x_3 - 2x_1;\\
&\qquad x_5 = x_3 - 2x_2;\\
&\qquad x_6 = 0;\\
&\qquad x_7 = x_6 + 2x_1;\\
&\qquad x_8 = x_6 + 2x_2;
\end{aligned} \tag{22}$$

To obtain the points a_0 to a_8, x_0 to x_8 are substituted into equation (4).
The transformation function *f*(*x*) as shown in Fig. 19 is a function that changes the weights of the PI hysteresis model. The weights of the transformed hysteresis are obtained via:

$$w_{2h_i} = \begin{cases} -\sum_{j=1}^{i} w_{h_j} - \sum_{j=1}^{i-1} w_{2h_j} & , \sum_{j=1}^{i} w_{h_j} < 0 \\ c - \sum_{j=1}^{i-1} w_{2h_j} & , \sum_{j=1}^{i} w_{h_j} = 0 \\ \sum_{j=1}^{i} w_{h_j} - \sum_{j=1}^{i-1} w_{2h_j} & , \sum_{j=1}^{i} w_{h_j} > 0 \end{cases} \tag{23}$$

where c is a positive non-zero constant to force the transformed gradient to be positive non-zero number. Fig. 21 shows the relationship of z and y after passing through the transformation function. The constants a_i and b_i are the corresponding y and z values respectively to input voltage x_i.

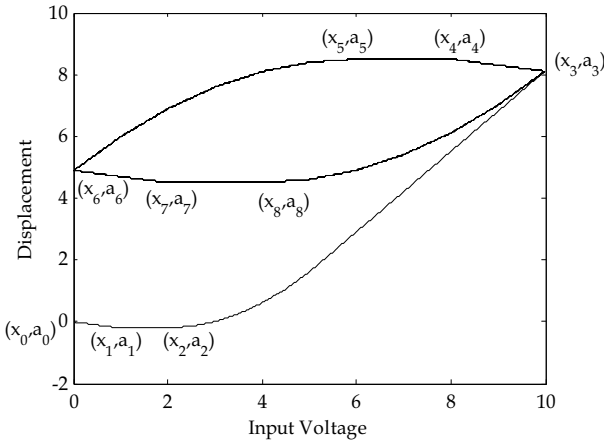


Figure 20: Graph of an ill-conditioned Hysteresis with points x_0 to x_8 labelled

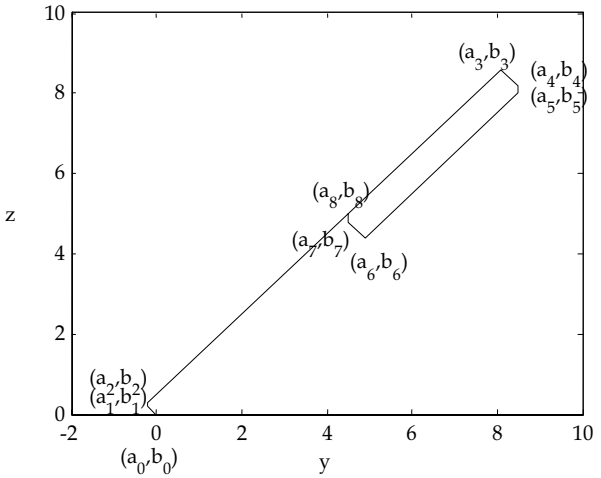


Figure 21: Relationship of z (new domain) with y (real displacement)

Because of the way the transformation function is formed, all the gradients of the line are 1, -1 or infinite. Points b_0 to b_8 are obtain using (24).

$$\begin{aligned} b_0 &= 0; \\ b_1 &= -a_1; \\ b_2 &= a_1 + c \times [x_2 - x_1]; \\ b_3 &= b_2 + [a_3 - a_2]; \\ b_4 &= b_3 - [a_4 - a_3]; \\ b_5 &= b_4 - 2c \times [x_2 - x_1]; \\ b_6 &= b_5 + [a_6 - a_5]; \\ b_7 &= b_6 - [a_7 - a_6]; \\ b_8 &= b_7 + 2c \times [x_2 - x_1]; \end{aligned} \tag{24}$$

With these points, the relationship between z to y is in table 3.

7. Simulation and Experimental Results of Transformation Method

7.1 Simulation

This section demonstrates how the transformation function is used to help the reader in applying the equations shown to their applications.

The actual ill conditioned hysteresis of the system is first obtained and modelled using Prandtl-Ishlinskii operator. The hysteresis curve is then mapped onto another domain using the transformation function illustrated in section 6. A well-conditioned hysteresis is obtained as seen in Fig.18. The inverse parameters of the well-conditioned hysteresis curve in the new domain are obtained using (13) and the inverse model is obtained.

After obtaining the inverse function in the new domain, the desired y values are passed through the transformation to obtain the desired z values using table 4, starting with A=0, B=0 and C=0. The desired z values are then passed through the inverse Prandtl-Ishlinskii model to obtain the required input x. An example is illustrated in Fig. 22, where the red graph is the hysteresis and blue graph is the inverse curve.

y value	Corresponding z value	Equation
a_0 to a_1	$z = -y$	(25)
a_1 to a_2,	$z = b_2$, any value between b_1 to b_2	(26)
a_2 to a_3,	$z = y + b_2 - a_2$	(27)
a_3 to a_4	$z = -y + b_3 + a_3$	(28)
a_4 to a_5	$z = b_5$, any value between b_4 to b_5	(29)
a_5 to a_6	$z = y + b_5 - a_5$	(30)
a_6 to a_7	$z = -y + b_6 + a_6$	(31)
a_7 to a_8	$z = b_8$, any value between b_7 to b_8	(32)

Table 3. Relation between the two domains

Condition (1)	*Condition (2)*	*Equation*	*Setting*
A′B′C′	$\|y(t)\| < \|a_1\|$ && $y(t) < y(t\text{-}1)$	(25)	
	otherwise	(26)	
A′B′C	$y(t) = a_2$	(26)	
	otherwise	(27)	B=1
A′BC	$y(t) \le a_3$	(27)	
	otherwise	(28)	A=1
ABC	$y(t) \le a_4$	(28)	
	otherwise	(29)	B=0
AB′C	$y(t) = a_5$	(29)	
	otherwise	(30)	
AB′C′	$y(t) \ge a_6$	(30)	
	otherwise	(31)	B=1
ABC′	$y(t) \ge a_7$	(31)	
	otherwise	(32)	A=0
A′BC′	$y(t) = a_8$	(32)	
	otherwise	(27)	

Table 4. Equations to obtain y values

The value of C is as follows:

$$C = \begin{cases} 0 & , y(t) < y(t-T) \\ 1 & , y(t) > y(t-T) \end{cases} \tag{33}$$

As shown in Fig. 22, the final inverse graph (blue) is a reflection of the hysteresis graph (red) along the line y = x. This clearly illustrates the ability of the transformation function to obtain the inverse of the hysteresis curve. The transformation function has no effect on well-conditioned hysteresis graphs as $x_2 = x_1 = x_0$.

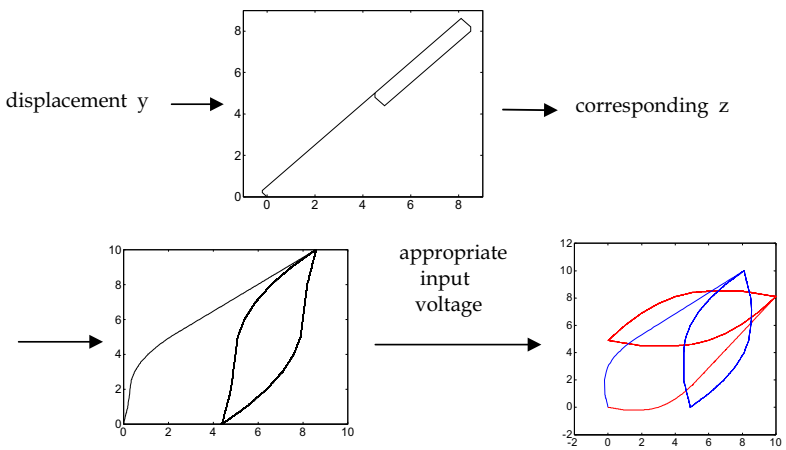


Figure 22. Simulation to illustrate that the inverse can be obtained with desired y as input

7.2 Experimental Results

The experiment setup is as described in section 5.1. Three types of experiments were carried out. The first set of experiments is 8 Hz triangular wave. Triangular wave is used because of its constant velocity. This is followed by varying amplitude linear motion with varying velocity to demonstrate the capability to model rate-dependent. The last experiment is a varying frequency with varying amplitude sinusoidal wave.

The same model is being used for both with and without mapping. The first experiment's desired displacement is a triangular wave with the velocity high enough for the first weight to go into the negative region.

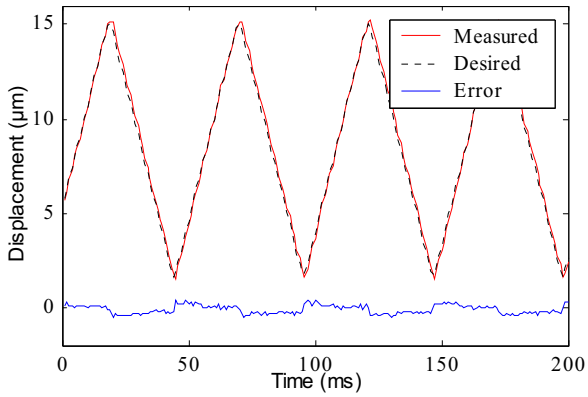


Figure 23. Triangular Wave Without Mapping

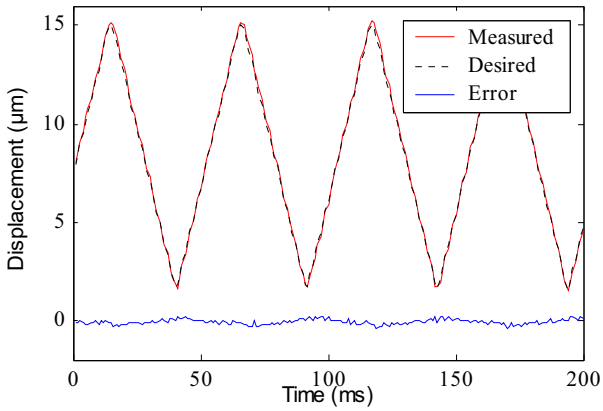


Figure 24. Triangular Wave With Mapping

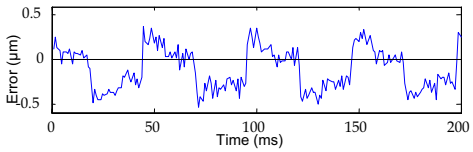


Figure 25. Exploded View of Error of Triangular Wave Without Mapping

	Triangular		Non-Periodic linear motion	
	Without Mapping	With Mapping	Without Mapping	With Mapping
rmse (μm)	0.2580	0.1544	0.2267	0.1328
rms error reduction	40.1%		41.4%	
Max. Error (μm)	0.5478	0.4222	0.4959	0.3473

Table 5. Experimental Results on Control of Piezoelectric Actuator

From Fig. 25, it can be clearly seen that the error has a general shape of a square wave. It has a general offset of overshoot when the desired displacement is increasing and an offset of undershoot when the displacement is decreasing. With mapping, this overshoot or undershoot are removed. Thus it can be clearly seen that the error in Fig. 23 (without mapping) is higher than the error in Fig. 24 (with mapping) and the rms error is greatly reduced by 40.1%. This proved that it is the singularity problem in the inverse expression that is creating the problem and not the hysteresis model.

Similar findings were obtained with non-periodic linear motion. Fig. 26 and Fig. 27 show the result of without and with mapping respectively. As seen from Fig. 26, like Fig. 23, there is also a constant overshoot or undershoot in the error depending on the direction of actuation. With mapping, the offsets are removed and the rms error is reduced by 41.4%. Table 5 is a summary of the results.

Fig. 28 is an experiment to show that the model is also valid for non-periodic varying sinusoidal waves and the rms error obtained is 0.14436μm. These figures demonstrate that the model is able to model non-periodic motion.

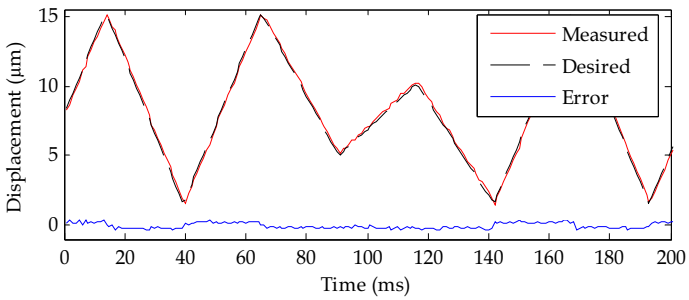


Figure 26. Non-Periodic Linear Motion Without Mapping

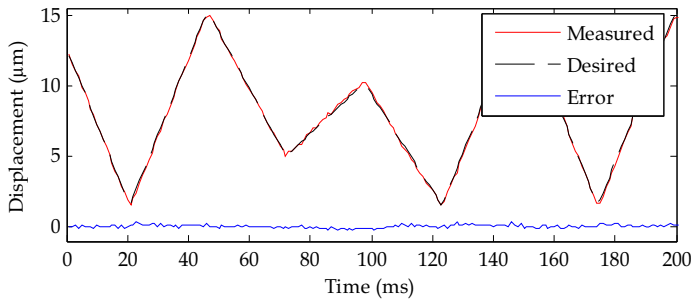


Figure 27. Non-Periodic Linear Motion With Mapping

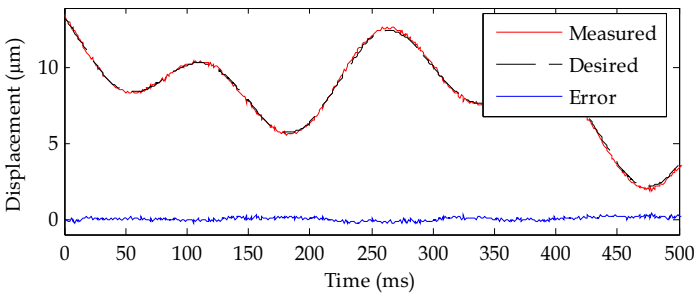


Figure 28. Superposition of a few different frequencies of sinusoidal waves

8. Conclusion

Human hand is the best manipulator available. However, its manual positioning accuracy is limited due to the involuntary tremor motion. Most of the current methods are non-active compensating methods like filtering of the tremor motion. Examples include the master and slave systems and the third hand. In this chapter, since humans are in possession of a high dexterity manipulator with an unbeatable user interface, instead of replacing the human hand with a robotic manipulator, active compensation of the physiological tremor is proposed.

Piezoelectric actuators are used to compensate the tremor motion. Two main contributions are made in this chapter, namely: (1) a rate-dependent feedforward controller; and (2) a solution to the inverse of an ill-conditioned hysteresis.

Physiological tremor is modulating frequency behaviour. Although the tremor is near sinusoidal at 8-12 Hz, tremor is non-periodic. Active compensation of active physiological tremor requires a zero phase rate-dependent controller. In this chapter, a rate-dependent Prandtl-Ishlinskii hysteresis model has been proposed. With this rate-dependent feedforward controller, piezoelectric actuators can now be used to compensate non-periodic

disturbance. With velocity as one of its input, the rate-dependent feedforward controller is now able to account for the non-periodic signals.
The feedforward controller is obtained through phenomenal modelling. Although the model obtained is specific to the hardware and setup, the method can be applied to other applications because underlying physics knowledge is not required. The feedforward controller is implemented in an open loop system. Some advantages of open loop control includes no stability problem faced by controllers and lower cost as sensors are not required for the feedback information.
Traditionally, people tried to control the conditions such that the ill-conditioned hysteresis situation is avoided. This is not solving the problem and will result in higher error. A method to overcome this problem has also been demonstrated. This is achieved by mapping the ill-conditioned hysteresis onto a different domain to obtain a well conditioned hysteresis. The inverse is then obtained in this new domain. The equations relating the two domains are also given in this chapter.

9. References

Abidi, K.; Sabanovic, A. & Yesilyurt, S. (2004). Sliding Mode Control Based Disturbance Compensation and External Force Estimation for a Piezoelectric Actuator, *IEEE Int. Workshop on Advance Motion Control*, pp. 529-534, Japan, March 2004.

Ang, W. T.; Garmon, F. A.; Khosla, P. K. & Riviere, C. N. (2003). Modeling Rate-dependent Hysteresis in Piezoelectric Actuators, *IEEE/RSJ Int. Conf. Intelligent Robots and Systems (IROS)*, pp. 1975-1980, Las Vegas, Nevada, Oct., 2003.

Chen, B. M.; Lee, T. H.; Hang, C. C.; Guo, Y. & Weerasooriya, S. (1999). An H_∞ Almost Disturbance Decoupling Robust Controller Design for a Piezoelectric Bimorph Actuator with Hysteresis. *IEEE Transactions on Control Systems Technology*, vol. 7 No. 2, (March 1999), pp. 160-174.

Choi, D. Y. & Riviere, C. N. (2005). Flexure-Based Manipulator for Active Handheld Microsurgical Instrument, *27th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBS)*, pp. 5085-5088, Shanghai, China, Sept. 2005.

Choi, G. S.; Kim, H. S. & Choi, G. H. (1997). A Study on Position Control of Piezoelectric Actuators, *IEEE Int. Symposium on Industrial Electronics*, pp. 851-855, Guimaraes, Portugal, 1997.

Cruz-Hernandez, J. M. & Hayward, V. (1998). Reduction of Major and Minor Hysteresis Loops in a Piezoelectric Actuator, *IEEE Conference on Design & Control*, pp. 4320-4325, Tampa, Florida USA, 1998.

Cruz-Hernandez, J. M. & Hayward, V. (2001). Phase Control Approach to Hystersis Reduction. *IEEE Transactions on Control Systems Technology*, vol. 9, No. 1, (Jan. 2001), pp. 17-26.

Furutani, K.; Urushibata, M. & Mohri, N. (1998) Improvement of Control Method for Piezoelectric Actuator by Combining Induced Charge Feedback with Inverse Transfer Function Compensation, *IEEE Int. Conf. On Robotics & Automation*, pp. 1504-1509, Leuven, Belgium, May 1998.

Goldfarb, M. & Celanovic, N. (1996). Behavioral Implications of Piezoelectric Stack Actuators for Control of Micromanipulation, *IEEE Int. Conf. on Robotics & Automation*, pp. 226-231, Minneapolis, Minnesota, USA, 1996.

Goldfarb, M. & Celanovic, N. (1997). Modeling Piezoelectric Stack Actuators for Control of Micromanipulation. *IEEE Control Systems Magazine*, vol. 17, No. 3, (June 1997), pp. 69-79.

Hu, H. & Mrad, R. B. (2002). On the Classical Preisach model for hysteresis in piezoceramic actuators. *Mechatron*, vol. 13, No. 2, (March 2002), pp. 85-94.

Hughes, D. & Wen. J. T. (1995). Preisach Modeling of Piezoceramic and Shape Memory Alloy Hysteresis, *4th IEEE Conf. on Control Applications*, pp. 1086-1091, New York, USA, Sep., 1995.

Hwang, C. L. & Jan, C. (2003). A Reinforcement Discrete Neuro-Adaptive Control for Unknown Piezoelectric Actuator Systems with Dominant Hysteresis. *IEEE Transactions on Neural Networks*, vol. 14, No. 1, (Jan 2003) pp. 66-78.

Janocha, H. & Kuhnen, K. (2000). Real-time Compensation of Hysteresis and Creep in Piezoelectric Actuators. *Sensors & Actuators A: Physical*, vol. 79, No. 2, (Feb. 2000), pp. 83-89.

Kuhnen, K. & Janocha, H. (2001). Inverse Feedforward Controller for Complex Hysteretic Nonlinearities in Smart-Material Systems. *Control and Intelligent Systems*, vol. 29, (2001), pp. 74-83.

Kuhnen, K. & Janocha, H. (2002). Complex hysteresis modeling of a broad class of hysteretic nonlinearities, *8th Int. Conf. on New Actuators*, Bremen, June 2002.

Landauer, R.; Young, D. R. & Drougard, M. E. (1956). Polarization reversal in the barium titanate hystersis loop. *Journal of Applied Physics*, vol. 27, No. 71, (1956) pp 752-758.

Riviere, C. N.; Ang, W. T. & Khosla, P. K. (2003). Toward Active Tremor Canceling in Handheld Microsurgical Instruments. *IEEE Transactions on Robotics and Automation*, vol. 19, No. 5, (Oct. 2003), pp. 793-800.

Stepanenko, Y. and Su, C. Y. (1998). Intelligent Control of Piezoelectric Actuators, *IEEE Conf. on Decision & Control*, pp. 4234-4239, Tampa, Florida USA, Dec. 1998.

Tao, G. (1995). Adaptive Control of Plants with Unknown Hystereses. *IEEE Transactions on Automatic Control*, vol. 40, No. 2, (Feb. 1995), pp. 200-212.

22

Automatic Speech Recognition of Human-Symbiotic Robot EMIEW

Masahito Togami, Yasunari Obuchi, and Akio Amano
Central Research Laboratory, Hitachi Ltd.
Japan

1. Introduction

Automatic Speech Recognition (ASR) is an essential function of robots which live in the human world. Many works for ASR have been done for a long time. As a result, computers can recognize human speech well under silent environments. However, accuracy of ASR is greatly degraded under noisy environments. Therefore, noise reduction techniques for ASR are strongly desired.

Many approaches based on spectral subtraction or Wiener filter have been studied. These approaches can reduce stationary noises such as fan-noise, but cannot reduce non-stationary noise such as human-speech.

In this chapter, we propose a novel noise reduction technique using a microphone-array. A microphone-array consists of more than one microphone. By using a microphone-array, robots can obtain information about sound sources' direction. When directions of noise sources and the desired source are different from each other, even if noises are non-stationary, noises can be reduced by spatial filtering with a microphone array. In this chapter, a new estimation method of direction of sources, MDSBF (modified delay and sum beam-former), is proposed. Then spatial filtering method using MDSBF named SB-MVBF (Sparseness Based Minimum Variance Beam-Former) is proposed.

The proposed noise reduction technique is implemented in a human-symbiotic prototype robot named EMIEW (Excellent Mobility and Interactive Existence as Workmate). It is shown that ASR technique with SB-MVBF is more accurate than ASR technique with the conventional method (MVBF) under noisy environments.

2. Human Symbiotic Robot EMIEW

Human symbiotic robot EMIEW (Excellent Mobility and Interactive Existence as Workmate) has been developed since 2004 by Hitachi Ltd (Hosoda et al., 2006).

EMIEW was designed as an assistant and a co-worker of human. Appearance of EMIEW is shown in Fig.1. When conventional robots live with human, one of major problems is lack of mobility. People can walk at about a few km/h, but robots before EMIEW cannot walk so rapidly. Maximum speed of him is about 6 km/h : the speed of a rapidly walking person. Therefore EMIEW can walk with human. Furthermore, it can avoid obstacles, so can move safely.

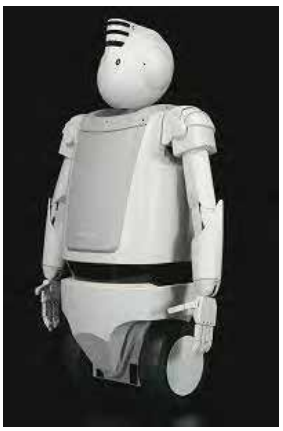


Figure 1. Appearance of EMIEW: body height is about 130cm. maximum speed is about 6 km/h. EMIEW has 8 microphones around his ears and neck. EMIEW is able to communicate with people even under noisy environment

For human symbiotic robots, in addition to mobility, communication capability is also important. It is desirable that robots can communicate with human in natural languages. EMIEW has speech synthesis function (Kitahara, 2006) and it can recognize human speech. In the future, EMIEW will work under noisy environments such as train-stations, airports, streets, and so on. Therefore, it is necessary that EMIEW can talk with humans under such environments.
We developed automatic speech recognition technology under noisy environments. We demonstrated this technology at EXPO 2005 AICHI JAPAN. Noise level of demonstration areas was from 70 db(A) to 80 db(A). It was verified that EMIEW can talk with guests at such environments.

3. Noise Reduction Technique for ASR

Automatic speech recognition (ASR) is a computational technology, which recognizes human speech which is recorded by microphones using pre-learned acoustic model.
Recognition performance of ASR is high for speeches which are recorded under noise-less and anechoic rooms. However, it is known that recognition performance of ASR is greatly degraded when human speech is convolved with noise or reverberation. Therefore, conventionally, noise reduction techniques have been studied (Boll, 1979) (Frost, 1972) (Aoki et al., 2001) (Hoshuyama et al., 1999). Microphone input signal is expressed as follows:

$$x(t) = s(t) + n(t) \tag{1}$$

, where t is the time-index, x(t) is the microphone input signal, s(t) is desired source signal, n(t) is the noise signal. Spectrum of speech signal is known to be stationary for a few dozen milliseconds. Therefore, many noise reduction approaches convert time domain expression to time-frequency domain expression by using short time Fourier transform as follows:

$$x(f,\tau) = s(f,\tau) + n(f,\tau) \tag{2}$$

,where f is the frequency index, τ is the frame index. When speech and noise is uncorrelated, power spectral of input signal is represented as follows:

$$E[|x(f,\tau)|^2] = E[|s(f,\tau)|^2] + E[|n(f,\tau)|^2]. \tag{3}$$

Spectral Subtraction (SS) (Boll, 1979) is the major noise reduction technique. SS subtracts time-averaged noise power spectral as follows:

$$\hat{s}(f,\tau) = \sqrt{|x(f,\tau)|^2 - \frac{1}{L}\sum_{\tau}|n(f,\tau)|^2}\ \frac{x(f,\tau)}{|x(f,\tau)|} \tag{4}$$

,where L is the number of averaged noise power spectral, $\hat{s}(f,\tau)$ is the output signal of SS. SS can reduce spectral-stationary noise such as fan-noise, but when noise is non-stationary (such as speech like noise), noise component cannot be reduced. To make things worse, in this case, the output signal of SS is greatly degraded by musical noise compared to the original speech. For this problem, noise reduction approaches using multi microphone elements (microphone array) have been widely studied. Direction of arrival (DOA) of sources can be estimated with a microphone array. One-channel noise reduction approaches such as SS cannot use DOA information. If DOA of noise and desired source are different from each other and DOA of desired source is given, even when noise is non-stationary, noise component can be reduced by spatially ``NULL'' beam-former such as MVBF (Minimum Variance Beam-Former) (Frost, 1972). However, when the given DOA of desired source is not accurate, the desired source is reduced or degraded. This problem is called signal cancellation problem.

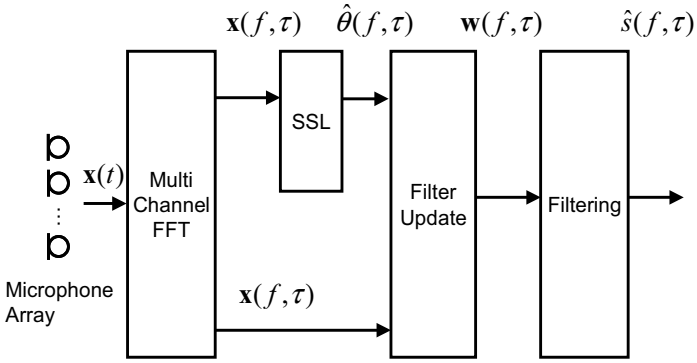


Figure 2. The noise-reduction block diagram of the proposed method at each frame

We will propose a novel noise reduction approach based on source's sparseness named SB-MVBF (Sparseness Based Minimum Variance Beam-Former). To solve signal cancellation problem, the spatial ``NULL'' beam-former is updated only when DOA of multi channel input signal is far from the given DOA of the desired source. The noise-reduction block diagram of the proposed method is shown in Fig. 2. Multi channel input signals of a

microphone array are transformed to frequency domain signals by FFT. DOA of input signal at each time-frequency point is estimated by Sound Source Localization (SSL). Filters for noise reduction are updated in ``Filter Update" block. Only when DOA of input signal is far from DOA of desired source, filters are updated in ``Filter Update" Block. Finally, noise is filtered by updated filters in ``Filtering" block.
In the following sections, we explain the proposed sound source localization method of each frequency component: MDSBF (Modified Delay and Sum Beam-Former) and adaptation method of noise reduction filter based on MDSBF: SB-MVBF (Sparseness Based Minimum Variance Beam-Former) and automatic speech recognition (ASR) based on SB-MVBF are shown.

3.1 Modified Delay and Sum Beam-Former (MDSBF)

Let M be the number of microphones, and $x_i(f,\tau)$ be the input signal of the i-th microphone at frame τ and frequency f. Sound source localization localizes direction of arrival of sources by the multi-channel input vector $\mathbf{X}(f,\tau)=[x_1(f,\tau),x_2(f,\tau),\ldots,x_M(f,\tau)]$. From now on, the suffix (f,τ) is omitted.

For simplicity, we assume that there is only one source at each time frequency point. In this case, the multi-channel input vector $\mathbf{X}$ is expressed as the following equation.

$$\mathbf{X}=\mathbf{a}(r,\theta)s \tag{5}$$

, where s is the source signal, r is distance between the source and the microphones, and θ is source's direction. The variable s is independent from the microphone index. The vector $\mathbf{a}(r,\theta)=[a_1,\ldots,a_M]$ is called steering vector. Each element is calculated as follows:

$$a_i=A_i e^{-2\pi f\rho} \tag{6}$$

, where A_i is the attenuation coefficient from the source position to the i-th microphone position, and ρ is time delay from the source position to the i-th microphone position. When microphones are sufficiently distant from the source position, A_i is independent from the microphone index, and it only depends on distance between the source and the microphones. Time-delay ρ is calculated as follows:

$$\rho=\frac{r}{c}+\lambda_i(\theta) \tag{7}$$

,where r is distance between the source and microphones, the term $\lambda_i(\theta)$ depends on source's direction and the microphone index, but it is independent from distance between the source and microphones.

Based on equation (5), we obtain the following inequality.

$$\frac{\left|\mathbf{a}(r,\theta)^*\mathbf{X}\right|}{\left|\mathbf{a}(r,\theta)\right|\left|\mathbf{a}(r,\theta)\right|}=\left|\mathbf{X}\right|\geq\frac{\left|\mathbf{a}(\tilde{r},\tilde{\theta})^*\mathbf{X}\right|}{\left|\mathbf{a}(\tilde{r},\tilde{\theta})\right|\left|\mathbf{a}(r,\theta)\right|}=\left|\mathbf{X}\right|\frac{\left|\mathbf{a}(\tilde{r},\tilde{\theta})^*\mathbf{a}(r,\theta)\right|}{\left|\mathbf{a}(\tilde{r},\tilde{\theta})\right|\left|\mathbf{a}(r,\theta)\right|} \tag{8}$$

Therefore, source's distance and direction are estimated as follows:

$$\hat{r},\hat{\theta} = \arg\max_{\tilde{r},\tilde{\theta}} \left| \mathbf{a}(\tilde{r},\tilde{\theta})^{*} \mathbf{X} \right| \tag{9}$$

,where l2-norm of $\mathbf{a}(\tilde{r},\tilde{\theta})$ is normalized to 1. By using equation (7), the vector $\mathbf{a}(\tilde{r},\tilde{\theta})$ is expressed as follows:

$$\mathbf{a}(\tilde{r},\tilde{\theta}) = e^{-2\pi j \frac{r}{c}} \hat{\mathbf{a}}(\tilde{\theta}) \tag{10}$$

,where $\hat{\mathbf{a}}(\theta) = [e^{-2\pi j \lambda_0(\theta)},...,e^{-2\pi j \lambda_1(\theta)},...,e^{-2\pi j \lambda_M(\theta)}]$. Therefore, equation (9) can be transformed as follows:

$$\hat{\theta} = \arg\max_{\tilde{\theta}} \left| \hat{\mathbf{a}}(\tilde{\theta})^{*} \mathbf{X} \right|. \tag{11}$$

Therefore, in this case, we can estimate only source's direction.

When there are more than one source at the same time frequency point, we cannot obtain sources' direction of arrival by Equation 11. However, it is known that speech is sparse signal in the time-frequency domain and few frequency components of multiple sources have big power at the same time point (Aoki et al., 2001). Therefore, it is considered to be the rare case that multiple sources are mixed in the same time-frequency point. Based on this sparseness assumption, we can estimate source's direction of arrival (DOA) at each time-frequency point by equation 11.

When sources are sparse, DOA of the input vector at each time-frequency point corresponds to the true source's angle, but this angle is variable with respect to each time-frequency point. Therefore, multiple sources' DOA is obtained by peak-searching in the histogram of the estimated DOA at all time-frequency points.

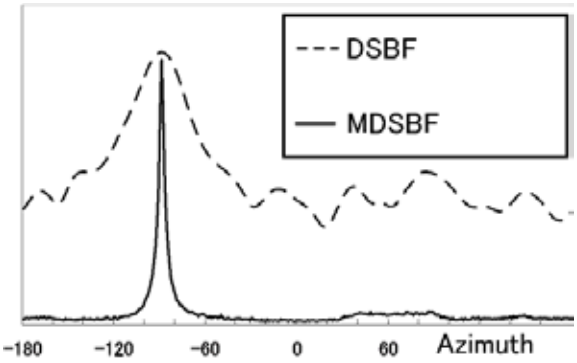


Figure 3. The DOA histogram at one source case: azimuth of the source is about -90 degree. Both DSBF and MDSBF succeeded to localize the source's DOA. However, the peak of DOA histogram by MDSBF is sharper than that by DSBF

Experimental results of DOA histogram at one source case (in Fig. 3) and two sources case (in Fig. 4) are shown. Reverberation time is about 300 milliseconds. Comparison to conventional delay and sum beam-former (DSBF) is shown. DOA histogram by DSBF has

only one peak in the two source case. However, DOA histogram by MDSBF has two peaks in the two source case.

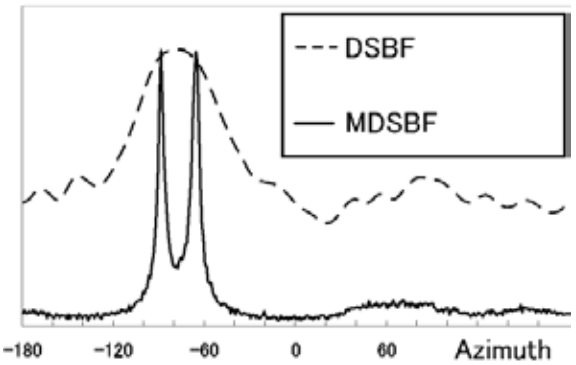


Figure 4. The DOA histogram at two source case: azimuth of the sources are about -90 degree and -70degree. MDSBF succeeded localization of sources' DOA. However, DSBF failed localize sources' DOA.

Success probability of DOA estimation by MDSBF was also checked. When there are only one source (ratio of one source (S1) 's power to the other source(S2) 's power is more than 30db), successful DOA estimation of MDSBF was 79%.

3.2 A Novel Adaptation Method : SB-MVBF

When steering vectors of desired source and noise are given, filtering process is simply expressed in Fig. 5.

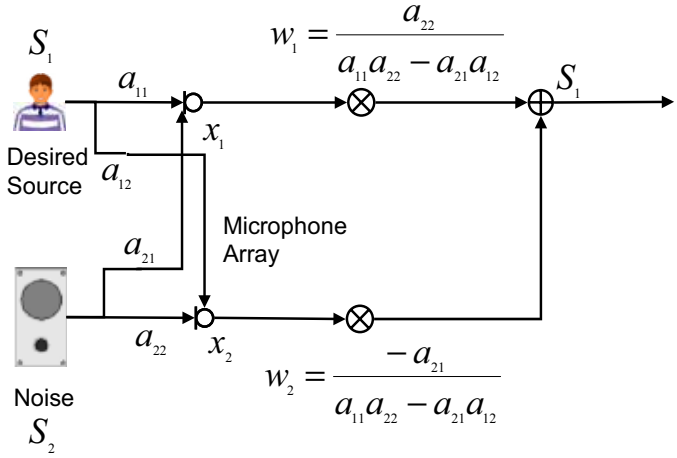


Figure 5. Filtering process under the condition that steering vectors are known

However, at least the steering vector of noise is not given and it is time-variable. Therefore, the steering vector of noise needs to be estimated at the beginning and updated when DOA of noise is changed. By Minimum Variance Beam-Former (MVBF) (Frost, 1972) , even when the steering vector of noise is unknown, noise reduction filter can be obtained as follows:

$$w = \frac{aR^{-1}}{aR^{-1}a^{*}} \tag{12}$$

,where R is the correlation matrix of the input vector x and is defined as follows:

$$R = E[xx^{*}]. \tag{13}$$

MVBF needs only desired steering vector a and correlation matrix of input vectors. The desired steering vector a can be calculated when DOA of desired source is given.

The filter w passes sources whose steering vector completely matches with given desired steering vector a, and reduce sources whose steering vector are different from a.

However, even when DOA of desired source is given based on prior knowledge such as ``desired speaker is in front of the robot'', actually the location of the speaker is different from given DOA. Additionally, given steering vector a is different from the actual steering vector because of reverberation.

Therefore, in this ``steering vector mismatch case", the filter made by MVBF cancels desired source (signal cancellation problem). This signal cancellation problem frequently occurs when the biggest component in correlation matrix R corresponds to desired signal. When the biggest component in correlation matrix R corresponds to noise signal, this signal cancellation problem does not occur. Therefore, correlation matrix R needs to be updated when desired source is absent. However, DOA of noise is time-variable. Therefore, correlation matrix R needs to be always updated.

To fill these requirements, time-variable coefficient to update correlation matrix R is proposed. Conventional MVBF updates correlation matrix R as follows:

$$\mathbf{R}_{\tau+1} = \beta\mathbf{R}_{\tau} + (1-\beta)\mathbf{x}_{\tau}\mathbf{x}_{\tau}^{*}. \tag{14}$$

Correlation matrix R is updated with the time-invariable coefficient. However, when the biggest source in the input vector is desired source, updating the correlation matrix R is unfavorable. Therefore, proposed SB-MVBF (Sparseness Based Minimum Variance Beam-Former) uses the time-variable coefficient to update correlation matrix R as follows:

$$\mathbf{R}_{\tau+1} = \alpha(\tau)\mathbf{R}_{\tau} + (1-\alpha(\tau))\mathbf{x}_{\tau}\mathbf{x}_{\tau}^{*} \tag{15}$$

,where correlation matrix R is updated with the time-variable coefficient $\alpha(\tau)$. This coefficient set to be 1 when desired source is likely to be inactive , and set to be 0 when desired source is likely to be active.

Estimation of desired source's status (active/inactive) is done by results of sound source localization. Proposed sound source localization MDSBF can accurately estimate DOA of sources at each time- frequency point. Therefore, when estimated DOA of one time-frequency point is far from DOA of desired source, it is likely that desired source is inactive in this time-frequency point.

SB-MVBF sets the time-variable coefficient to be 1 when estimated DOA by MDSBF is far from DOA of desired source and set it to be 0 when estimated DOA by MDSBF is close to DOA of desired source. An example of separated signal at an room (reverberation time is 300 ms) is shown in Fig. 6.

3.3 Evaluation of ASR Under Noisy Environment

SB-MVBF reduces noise. However, residual noise exists in noise reduced signal, performance of automatic speech recognition is degraded when the acoustic model of ASR is made by clean speeches. Therefore the acoustic model is adapted by speeches convolved with remained noise. In this experiment, the acoustic model is adapted to noise reduced signals by the proposed method.

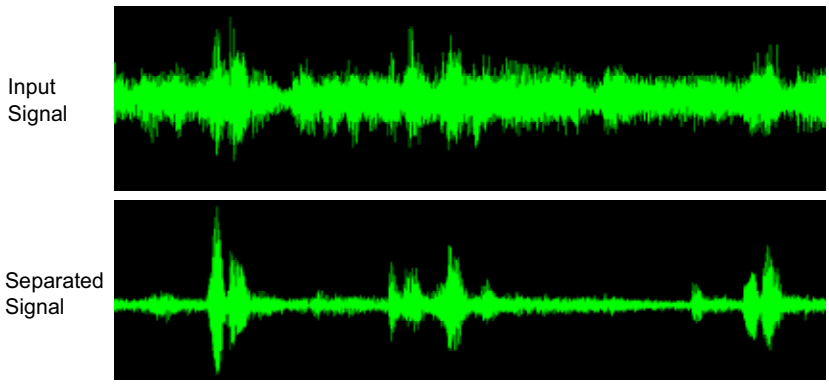


Figure 6. Input signal and separated signal: separated signal has less noisy than input signal

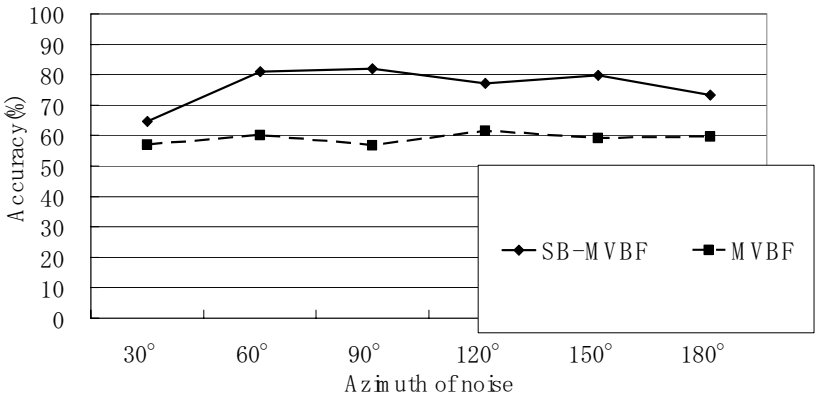


Figure 7. experimental results of ASR: Accuracy of ASR with Proposed method (SB-MVBF) is about from 10% to 20% higher than that with conventional MVBF

Acoustic features are 14-order LPC cepstrum ,14-order delta-LPC cepstrum and 1-oder delta power. Total dimension of features is 29. This experiment of ASR was done under 5db SNR (Signal to Noise Ratio) condition. The recognition vocabulary consists of 10-digits. The number of speakers is 80. The experimental result is shown in Fig. 7. In this experiment, desired source is in front of the microphone array (azimuth=0 degree). Direction of noise is variable from 30 degree to 180 degree.

3.4 Demonstration at EXPO 2005 AICHI JAPAN

Appearance of demonstration at EXPO 2005 AICHI JAPAN is shown in Fig. 8. EMIEW recognized guests' order under noisy environment (noise level= from 70 db(A) to 80 db(A)).

Figure 8. demonstration at EXPO 2005 AICHI JAPAN: EMIEW recognized guests' speech under noisy environment

5. Conclusion

We explained noise reduction technique and automatic speech recognition (ASR) under noisy environments. Human symbiotic robot EMIEW succeeded recognition under noisy environment at EXPO 2005 AICHI JAPAN.

For high accuracy of ASR under noisy environment, noise reduction technique is necessary. In this chapter, robust noise reduction technique with a microphone array was proposed. Proposed Modified Delay and Sum Beam-Former (MDSBF) can localize sources more accurately than conventional Delay and Sum Beam-Former (DSBF) . A novel adaptation method of Minimum Variance Beam-Former (MVBF) with time-variant coefficient (SB-MVBF) is proposed. Performance of ASR with proposed method was shown to be higher than conventional MVBF.

6. Acknowledgment

This work was done through entrustment by the New Energy and Industrial Technology Development Organization (NEDO).

7. References

Hosoda, Y.; Egawa, S. Tamamoto, J. Yamamoto, K. Nakamura, R. & Togami, M. (2006). Basic design of human-symbiotic robot EMIEW, *Proceedings of IROS 2006*, pp. 5079-5084

Kitahara, Y. (2006). Development of High Quality and Intelligent Speech Synthesis Technology. *Hitachi Review* , Vol.88, No. 06, pp. 60-65 (in Japanese)

Boll, S.F. (1979). Suppression of acoustic noise in speech using spectral subtraction, *IEEE Trans. ASSP*, Vol.27, No.2, pp. 113-120

Togami, M.; Sumiyoshi, T. & Amano, A. (2006). Sound source separation of overcomplete convolutive mixtures using generalized sparseness, *CD-ROM Proceedings of IWAENC2006*

Frost, III, O.L. (1972). An algorithm for linearly constrained adaptive array processing, *Proceedings IEEE* Vol.60, No.8, pp. 926-935.

Griffith, L.J. & Jim, C.W. (1982). An alternative approach to linearly constrained adaptive beamforming, *IEEE Trans. Anntenas Propagation*, Vol.30, i.1, pp. 27-34

Aoki, M. ; Okamoto, M., Aoki, S., Matsui, H., Sakurai, T. & Kaneda, Y. (2001). Sound source segregation based on estimating incident angle of each frequency component of input signals acquired by multiple microphones, *Acoust.Sci & Tech.* Vol.22, No.2, pp. 149-157

Hoshuyama, O. ; Sugiyama, A. & Hirano, A. (1999). A robust adaptive beamformer for microphone arrays with a blocking matrix using constrained adaptive filters, *IEEE Trans. Signal Processing* , Vol.47, No.10, pp.2677-2684

23

Mixed-initiative multirobot control in USAR

Jijun Wang and Michael Lewis
School of Information Sciences, University of Pittsburgh
USA

1. Introduction

In Urban Search and Rescue (USAR), human involvement is desirable because of the inherent uncertainty and dynamic features of the task. Under abnormal or unexpected conditions such as robot failure, collision with objects or resource conflicts human judgment may be needed to assist the system in solving problems. Due to the current limitations in sensor capabilities and pattern recognition people are also commonly required to provide services during normal operation. For instance, in the USAR practice (Casper & Murphy 2003), field study (Burke *et al.* 2004), and RoboCup competitions (Yanco *et al.* 2004), victim recognition remains primarily based on human inspection.

Human control of multiple robots has been suggested as a way to improve effectiveness in USAR. However, multiple robots substantially increase the complexity of the operator's task because attention must be continually shifted among robots. A previous study showed that when the mental demand overwhelmed the operator's cognitive resources, operators controlled reactively instead of planning and proactively controlling the robots leading to worse performance (Trouvain *et al.* 2003). One approach to increasing human capacity for control is to allow robots to cooperate reducing the need to control them independently. Because human involvement is still needed to identify victims and assist individual robots, automating coordination appears to be a promising avenue for reducing cognitive demands on the operator.

For a human/automation system, how and when the operator intervenes are the two issues that most determine overall effectiveness (Endsley 1996). How the human interacts with the system can be characterized by the level of autonomy (LOA), a classification based on the allocation of functions between human and robot. In general, the LOA can range from complete manual control to full autonomy (Sheridan 2002). Finding the optimal LOA is an important yet hard to solve problem because it depends jointly on the robotic system, task, working space and the end user. Recent studies (Squire *et al.* 2003; Envarli & Adams 2005; Parasuraman *et al.* 2005; Schurr *et al.* 2005) have compared different LOAs for a single operator interacting with cooperating robots. All of them, however, were based on simple tasks using low fidelity simulation thereby minimizing the impact of situation awareness (SA). In realistic USAR applications (Casper & Murphy 2003, Burke et al. 2004, Yanco et al. 2004) by contrast, maintaining sufficient SA is typically the operator's greatest problem.

The present study investigates human interaction with a cooperating team of robots performing a search and rescue task in a realistic disaster environment. This study uses USARSim (Wang *et al.* 2003), a high fidelity game engine-based robot simulator we

developed to study human-robot interaction (HRI) and multi-robot control. USARSim provides a physics based simulation of robot and environment that accurately reproduces mobility problems caused by uneven terrain (Wang *et al.* 2005), hazards such as rollover (Lewis *& Wang 2007*), and provides accurate sensor models for laser rangefinders (Carpin *et al.* 2005) and camera video (Carpin *et al.* 2006). This level of detail is essential to posing realistic control tasks likely to require intervention across levels of abstraction. We compared control of small robot teams in which cooperating robots exploring autonomously, were controlled independently by an operator, or through mixed initiative as a cooperating team. In our experiment mixed initiative teams found more victims and searched wider areas than either fully autonomous or manually controlled teams. Operators who switched attention between robots more frequently were found to perform better in both manual and mixed initiative conditions.

We discuss the related work in section 2. Then we introduce our simulator and multi-robot system in section 3. Section 4 describes the experiment followed by the results presented in section 5. Finally, we draw conclusion and discuss the future work in section 6.

2. Related Work

When a single operator controls multiple robots, in the simplest case the operator interacts with each independent robot as needed. Control performance at this task can be characterized by the average demand of each robot on human attention (Crandall *et al.* 2005) or the distribution of demands coming from multiple robots (Nickerson & Skiena 2005). Increasing robot autonomy allows robots to be neglected for longer periods of time making it possible for a single operator to control more robots. Researchers investigating the effects of levels of autonomy (teleoperation, safe mode, shared control, full autonomy, and dynamic control) on HRI (Marble *et al.* 2003; Marble *et al.* 2004) for single robots have found that mixed-initiative interaction led to better performance than either teleoperation or full autonomy. This result seems consistent with Fong's collaborative control (Fong *et al.* 2001) premise that because it is difficult to determine the most effective task allocation a priori, allowing adjustment during execution should improve performance.

The study of autonomy modes for multiple robot systems (MRS) has been more restrictive. Because of the need to share attention between robots, teleoperation has only been allowed for one robot out of a team (Nielsen *et al.* 2003) or as a selectable mode (Parasuraman *et al.* 2005). Some variant of waypoint control has been used in all MRS studies reviewed (Trouvain & Wolf 2002; Nielsen *et al.* 2003; Squire *et al.* 2003; Trouvain *et al.* 2003; Crandall *et al.* 2005; Parasuraman *et al.* 2005) with differences arising primarily in behaviour upon reaching a waypoint. A more fully autonomous mode has typically been included involving things such as search of a designated area (Nielsen *et al.* 2003), travel to a distant waypoint (Trouvain & Wolf 2002), or executing prescribed behaviours (Parasuraman *et al.* 2005). In studies in which robots did not cooperate and had varying levels of individual autonomy (Trouvain & Wolf 2002; Nielsen *et al.* 2003; Trouvain *et al.* 2003; Crandall *et al.* 2005) (team size 2-4) performance and workload were both higher at lower autonomy levels and lower at higher ones. So although increasing autonomy in these experiments reduced the cognitive load on the operator, the automation could not perform the replaced tasks as well. This effect would likely be reversed for larger teams such as those tested in Olsen & Wood's (Olsen & Wood 2004) fan-out study which found highest performance and lowest (per robot activity) imputed workload for the highest levels of autonomy.

For cooperative tasks and larger teams individual autonomy is unlikely to suffice. The round-robin control strategy used for controlling individual robots would force an operator to plan and predict actions needed for multiple joint activities and be highly susceptible to errors in prediction, synchronization or execution. A series of experiments using the Playbook interface and the RoboFlag simulation (Squire *et al.* 2003; Parasuraman *et al.* 2005) provide data on HRI with cooperating robot teams. These studies found that control through delegation (calling plays/plans) led to higher success rates and faster missions than individual control through waypoints and that as with single robots (Marble *et al.* 2003; Marble *et al.* 2004) allowing the operator to choose among control modes improved performance. Again, as in the single robot case, the improvement in performance from adjustable autonomy carried with it a penalty in reported workload. Another recent study (Schurr *et al.* 2005) investigating supervisory control of cooperating agents performing a fire fighting task found that human intervention actually degraded system performance. In this case, the complexity of the fire fighting plans and the interdependency of activities and resources appeared to be too difficult for the operator to follow. For cooperating teams and relatively complex tasks, therefore, the neglect-tolerance assumption (Olsen & Wood 2004; Crandall *et al.* 2005) that human control always improves performance may not hold. For these more complex MRS control regimes it will be necessary to account for the arguments of Woods et al. (Woods *et al.* 2004) and Kirlik's (Kirlik 1993) demonstration that higher levels of autonomy can act to increase workload to the point of eliminating any advantage by placing new demands on the operator to understand and predict automated behaviour. The cognitive effort involved in shifting attention between levels of automation and between robots reported by (Squire *et al.* 2003) seems a particularly salient problem for MRS.

Experiment	World	Robots	Task	Team
Nielsen et al. (2003)	2D simulator	3	Navigate/build map	independent
Crandall et al. (2005)	2D simulator	3	Navigate	independent
Trouvain & Wolf (2002)	2D simulator	2,4,8	Navigate	independent
Trouvain et al. (2003)	3D simulator	1,2,4	Navigate	independent
Parasuraman et al. (2005)	2D simulator	4,8	Capture the flag	cooperative
Squire et al. (2006)	2D simulator	4,6,8	Capture the flag	cooperative
Present Experiment	3D simulator	3	Search	cooperative

Table 1. Recent MRS Studies

Table 1 organizes details of recent MRS studies. All were conducted in simulation and most involve navigation rather than search, one of the most important tasks in USAR. This is significant because search using an onboard camera requires greater shifts between contexts than navigation which can more easily be performed from a single map display (Bruemmer *et al.* 2005; Nielsen & Goodrich 2006). Furthermore, previous studies have not addressed the issues of human interaction with cooperating robot teams within a realistically complex environment. Results from 2D simulation (Squire *et al.* 2003; Parasuraman *et al.* 2005), for example, are unlikely to incorporate tasks requiring low-level assistance to robots, while experiments with non-cooperating robots (Trouvain & Wolf 2002; Nielsen *et al.* 2003;

Trouvain *et al.* 2003; Crandall *et al.* 2005) miss the effects of this aspect of autonomy on performance and HRI.
This paper presents an experiment comparing search performance of teams of 3 robots controlled manually without automated cooperation, in a mixed-initiative mode interacting with a cooperating team or in a fully autonomous mode without a human operator. The virtual environment was a model of the Yellow Arena, one of the NIST Reference Test Arenas designed to provide standardized disaster environments to evaluate human robot performance in USAR domain (Jacoff *et al.* 2001). The distributed multiple agents framework, Machinetta (Scerri *et al.* 2004) is used to automate cooperation for the robotic control system in the present study.

3. Simulator and Multirobot System

3.1 Simulation of the Robots and Environment

Although many robotic simulators are available most of them have been built as ancillary tools for developing and testing control programs to be run on research robots. Simulators (Lee *et al.* 1994; Konolige & Myers 1998) built before 2000 typically have low fidelity dynamics for approximating the robot's interaction with its environment. More recent simulators including ÜberSim (Browning & Tryzelaar 2003) , a soccer simulator, Gazebo (Gerkey *et al.* 2003), and the commercial Webots (Cyberbotics Ltd. 2006) use the open source Open Dynamics Engine (ODE) physics engine to approximate physics and kinematics more precisely. ODE, however, is not integrated with a graphics library forcing developers to rely on low-level libraries such as OpenGL. This limits the complexity of environments that can practically be developed and effectively precludes use of many of the specialized rendering features of modern graphics processing units. Both high quality graphics and accurate physics are needed for HRI research because the operator's tasks depend strongly on remote perception (Woods *et al.* 2004), which requires accurate simulation of camera video, and interaction with automation, which requires accurate simulation of sensors, effectors and control logic.

Figure 1. Simulated P2DX robot

We built USARSim, a high fidelity simulation of USAR robots and environments to be a research tool for the study of HRI and multi-robot coordination. USARSim supports HRI by accurately rendering user interface elements (particularly camera video), accurately representing robot automation and behavior, and accurately representing the remote environment that links the operator's awareness with the robot's behaviors. It was built based on a multi-player game engine, UnrealEngine2, and so is well suited for simulating multiple robots. USARSim uses the Karma Physics engine to provide physics modeling, rigid-body dynamics with constraints and collision detection. It uses other game engine capabilities to simulate sensors including camera video, sonar, and laser range finder. More details about USARSim can be found at (Wang *et al.* 2003; Lewis *et al.* 2007).

a) Arena-1 b) Arena-2

Figure 2. Simulated testing arenas

In this study, we simulated three Activemedia P2-DX robots. Each robot was equipped with a pan-tilt camera with 45 degrees FOV and a front laser scanner with 180 degree FOV and resolution of 1 degree. Two similar NIST Reference Test Arenas, Yellow Arena, were built using the same elements with different layouts. In each arena, 14 victims were evenly distributed in the world. We added mirrors, blinds, curtains, semitransparent boards, and wire grid to add difficulty in situation perception. Bricks, pipes, a ramp, chairs, and other debris were put in the arena to challenge mobility and SA in robot control. Figure 1 shows a simulated P2DX robot and a corner in the virtual environment. Figure 2 illustrates the layout of the two testing environments.

3.2 Multi-robot Control System (MrCS)

The robotic control system used in this study is MrCS (Multi-robot Control System), a multi-robot communications and control infrastructure with accompanying user interface. MrCS provides facilities for starting and controlling robots in the simulation, displaying camera and laser output, and supporting inter-robot communication through Machinetta (Scerri *et al.* 2004). Machinetta is a distributed mutiagent system with state-of-the-art algorithms for plan instantiation, role allocation, information sharing, task deconfliction and adjustable autonomy (Scerri *et al.* 2004). The distributed control enables us to scale robot teams from small to large. In Machinetta, team members connect to each other through reusable software proxies. Through the proxy, humans, software agents, and different robots can work together to form a heterogeneous team. Basing team cooperation on reusable proxies allows us to quickly change size or coordination strategies without affecting other parts of the system.

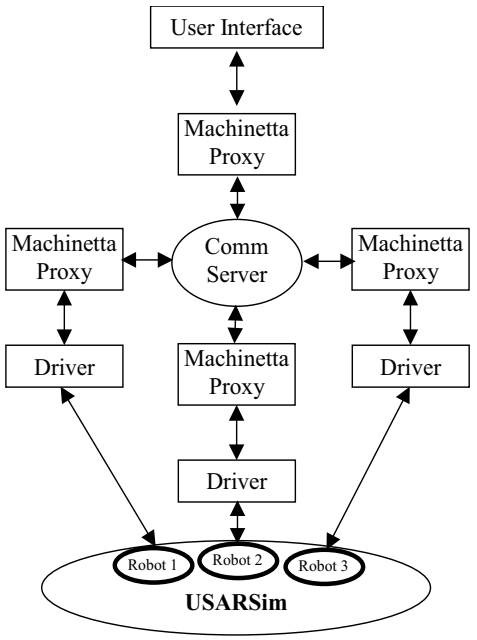


Figure 3. MrCS system architecture

Figure 3 shows the system architecture of MrCS. It provides Machinetta proxies for robots, and human operator (user interface). Each robot connects with Machinetta through a robot driver that provides low-level autonomy such as guarded motion, waypoint control (moving form one point to another while automatically avoiding obstacles) and middle-

level autonomy in path generation. The robot proxy communicates with proxies on other simulated robots to enable the robots to execute the cooperative plan they have generated. In the current study plans are quite simple and dictate moving toward the nearest frontier that does not conflict with search plans of another robot. The operator connects with Machinetta through a user interface agent. This agent collects the robot team's beliefs and visually represents them on the interface. It also transfers the operator's commands in the form of a Machinetta proxy's beliefs and passes them to the proxies network to allow human in the loop cooperation. The operator is able to intervene with the robot team on two levels. On the low level, the operator takes over an individual robot's autonomy to teleoperate it. On the intermediate level, the operator interacts with a robot via editing its exploration plan.

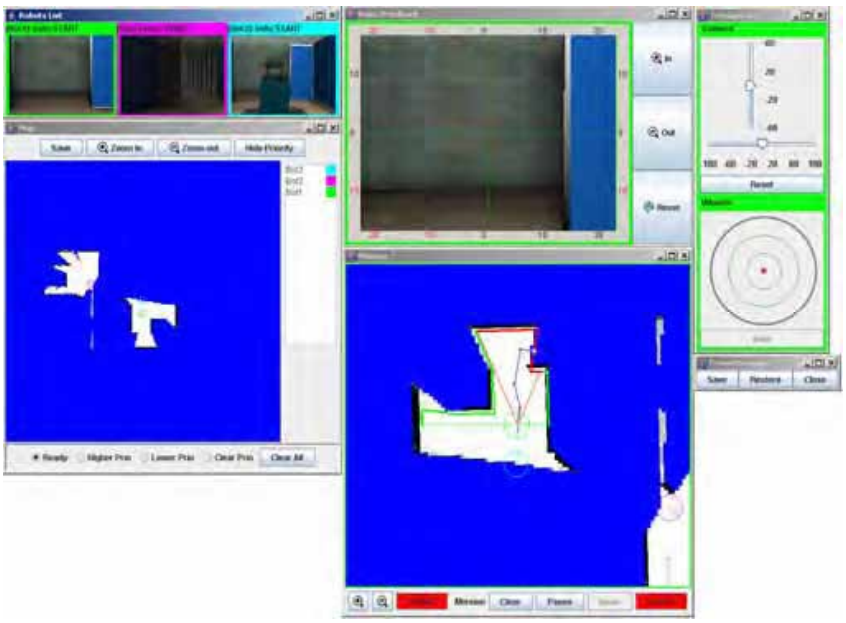


Figure 4. The graphic user interface

In the human robot team, the human always has the highest authority although the robot may alter its path slightly to avoid obstacles or dangerous poses. Robots are controlled one at a time with the selected robot providing a full range of data while the unselected ones provide camera views for monitoring. The interface allows the user to resize the components or change the layout. Figure 4 shows the interface configuration used in the present study. On the left side are the global information components: the Robots List (the upper panel) that shows each team member's execution state and the thumbnail of the individual's camera view; and the global Map (the bottom panel) that shows the explored

areas and each robot's position. From the Robot List, the operator can select any robot to be controlled. In the center are the individual robot control components. The upper component, Video Feedback, displays the video of the robot being controlled. It allows the user to pan/tilt and zoom the camera. The bottom component is the Mission panel that shows the controlled robot's local situation. The local map is camera up, always pointing in the camera's direction. It is overlaid with laser data in green and a cone showing the camera's FOV in red. With the Mission panel and the Video Feedback panel, we support SA at three ranges. The camera view and range data shown in the red FOV cone provide the operator the close range SA. It enables the operator to observe objects through the camera and identify their locations on the map. The green range data shows the open regions around the robot providing local information about where to go in the next step. In contrast, the background map provides the user long range information that helps her make a longer term plan. The mission panel displays the robot's current plan as well to help the user understand what the robot is intending to do. When a marked victim or another robot is within the local map the panel will represent them even if not sensed. Besides representing local information, the Mission panel allows the operator to control a robot by clearing, modifying, or creating waypoints and marking the environment by placing an icon on the map. On the right is the Teleoperation panel that teleoperates the robot or pans/tilts the camera. These components behave in the expected ways.

4. Experiment

4.1 Experimental Design

In the experiment, participants were asked to control 3 P2DX robots (Figure 1) simulated in USARSim to search for victims in a damaged building (Figure 2). The participant interacted with the robots through MrCS with fixed user interface shown in Figure 4. Once a victim was identified, the participant marked its location on the map.

We used a within subjects design with counterbalanced presentation to compare mixed initiative and manual conditions. Under mixed initiative, the robots analyzed their laser range data to find possible exploration paths. They cooperated with one another to choose execution paths that avoided duplicating efforts. While the robots autonomously explored the world, the operator was free to intervene with any individual robot by issuing new waypoints, teleoperating, or panning/tilting its camera. The robot returned back to auto mode once the operator's command was completed or stopped. While under manual control robots could not autonomously generate paths and there was no cooperation among robots. The operator controlled a robot by giving it a series of waypoints, directly teleoperating it, or panning/tilting its camera. As a control for the effects of autonomy on performance we conducted "full autonomy" testing as well. Because MrCS doesn't support victim recognition, based on our observation of the participants' victim identification behaviours, we defined detection to have occurred for victims that appeared on camera for at least 2 seconds and occupied at least 1/9 of the thumbnail view. Because of the high fidelity of the simulation, and the randomness of paths picked through the cooperation algorithms, robots explored different regions on every test. Additional variations in performance occurred due to mishaps such as a robot getting stuck in a corner or bumping into an obstacle causing its camera to point to the ceiling so no victims could be found. Sixteen trials were conducted in each area to collect data comparable to that obtained from human participants.

4.2 Procedure

The experiment started with collection of the participant's demographic data and computer experience. The participant then read standard instructions on how to control robots via MrCS. In the following 10 minutes training session, the participant practiced each control operation and tried to find at least one victim in the training arena under the guidance of the experimenter. Participants then began a twenty minutes session in Arena-1 followed by a short break and a twenty minutes session in Arena-2. At the conclusion of the experiment participants completed a questionnaire.

4.3 Participants

	Age		**Gender**		**Education**	
	19	20~35	Male	Female	Currently Undergraduate	Complete Undergraduate
Participants	2	12	5	9	10	4

	Computer Usage (hours/week)				**Game Playing (hours/week)**			
	<1	1-5	5-10	>10	<1	1-5	5-10	>10
Participants	0	2	7	5	6	7	1	0

	Mouse Usage for Game Playing		
	Frequently	Occasionally	Never
Participants	8	6	0

Table 2. Sample demographics and experiences

14 paid participants recruited from the University of Pittsburgh community took part in the experiment. None had prior experience with robot control although most were frequent computer users. The participants' demographic information and experience are summarized in Table 2.

5. Results

In this experiment, we studied the interaction between a single operator and a robot team in a realistic interactive environment where human and robots must work tightly together to accomplish a task. We first compared the impact of different levels of autonomy by evaluating the overall performance as revealed by the number of found victims, the explored areas, and the participants' self-assessments. For the small robot team with 3 robots, we expected similar results to those reported in (Trouvain & Wolf 2002; Nielsen *et al.* 2003; Trouvain *et al.* 2003; Crandall *et al.* 2005) that although autonomy would decrease workload, it would also decrease performance because of poorer situation awareness (SA). How a human distributes attention among the robots is an interesting problem especially when the human is deeply involved in the task by performing low level functions, such as identifying a victim, which requires balancing between monitoring and control. Therefore, in addition to overall performance measures, we examine: 1) the distribution of human

interactions among the robots and its relationship with the overall performance, and 2) the distribution of control behaviours, i.e. teleoperation, waypoint issuing and camera control, among the robots and between different autonomy levels, and their impacts in the overall human-robot performance. Trust is a special and important problem arising in human-automation interaction. When the robotic system can't work as the operator expected, it will influence how the operator control the robots and hereby impact the human-robot performance (Lee & See 2004; Parasuraman & Miller 2004). In addition, because of the complexity of the control interface, we anticipated that the ability to use the interface would impact the overall performance as well. At the end of this section, we report participants' self-assessments of trust and capability of using the user interface, as well as the relationship among the number of found victims and these two factors.

5.1 Overall Performance

All 14 participants found at least 5 of all possible 14 (36%) victims in each of the arenas. The median number of victims found was 7 and 8 for test arenas 1 and 2 respectively. Two-tailed t-tests found no difference between the arenas for either number of victims found or the percentage of the arena explored. Figure 5 shows the distribution of victims discovered as a function of area explored. These data indicate that participants exploring less than 90% of the area consistently discovered 5-8 victims while those covering greater than 90% discovered between half (7) and all (14) of the victims.

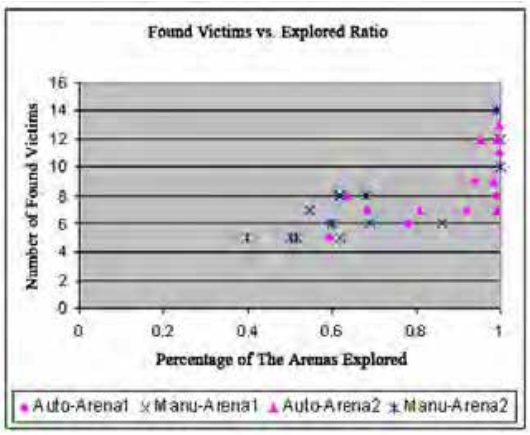


Figure 5. Victims as a function of area explored

Within participant comparisons found wider regions were explored in mixed-initiative mode, $t(13) = 3.50$, $p < .004$, as well as a marginal advantage for mixed-initiative mode, $t(13) = 1.85$, $p = .088$, in number of victims found. Comparing with "full autonomy", under mixed-initiative conditions two-tailed t-tests found no difference ($p = 0.58$) in the explored regions. However, under full autonomy mode, the robots explored significantly, $t(44) = 4.27$,

$p < .001$, more regions than under the manual control condition (left in Figure 6). Using two-tailed t-tests, we found that participants found more victims under mixed-initiative and manual control conditions than under full autonomy with $t(44) = 6.66$, $p < .001$, and $t(44) = 4.14$, $p < .001$ respectively (right in Figure 6). The median number of victims found under full autonomy was 5.

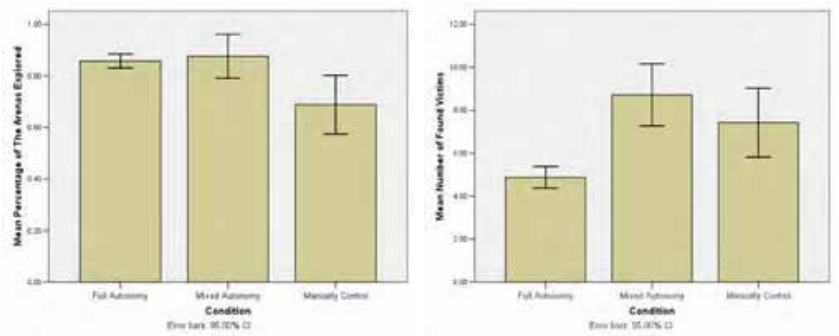


Figure 6. Regions explored by mode (left) and victims found by mode (right)

In the posttest survey, 8 of the 14 (58%) participants reported they were able to control the robots although they had problems in handling some components. All of the remaining participants thought they used the interface very well. Comparing the mixed-initiative with the manual control, most participants (79%) rated team autonomy as providing either significant or minor help. Only 1 of the 14 participants (7%) rated team autonomy as making no difference and 2 of the 14 participants (14%) judged team autonomy to make things worse.

5.2 Human Interactions

Participants intervened to control the robots by switching focus to an individual robot and then issuing commands. Measuring the distribution of attention among robots as the standard deviation of the total time spent with each robot, no difference ($p = .232$) was found between mixed initiative and manual control modes. However, we found that under mixed initiative, the same participant switched robots significantly more often than under manual mode ($p = .027$). The posttest survey showed that most participants switched robots using the Robots List component. Only 2 of the 14 participants (14%) reported switching robot control independent of this component.

Across participants the frequency of shifting control among robots explained a significant proportion of the variance in number of victims found for both mixed initiative, $R2 = .54$, $F(1, 11) = 12.98$, $p = .004$, and manual, $R2 = .37$, $F(1, 11) = 6.37$, $p < .03$, modes (Figure 7).

An individual robot control episode begins with a pre-observation phase in which the participant collects the robot's information and then makes a control decision, and ends with the post-observation phase in which the operator observes the robot's execution and decides to turn to the next robot. Using a two-tailed t-test, no difference was found in either total pre-observation time or total post-observation time between mixed-initiative and manually control conditions. The distribution of found victims among pre- and post-

observation times (Figure 8) suggests, however, that the proper combination can lead to higher performance.

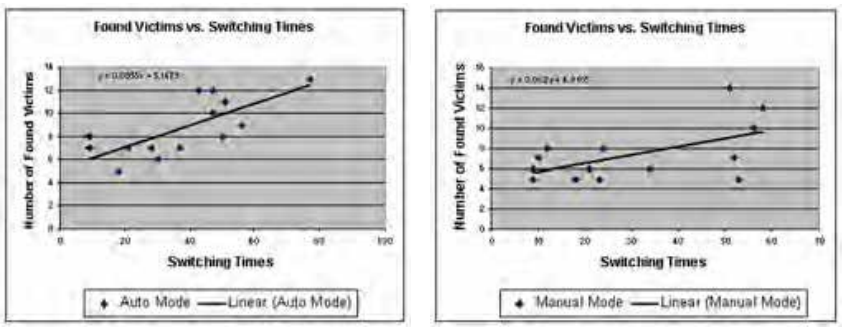


Figure 7. Victims vs. switches under mixed-autonomy (left) and manually control (right) mode

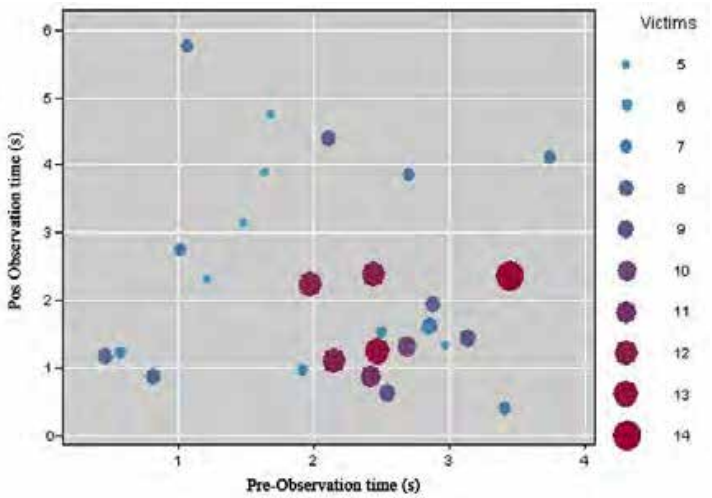


Figure 8. Pre and Post observation time vs. found

5.3 Forms of Control

Three interaction methods: waypoint control, teleoperation control, and camera control were available to the operator. Using waypoint control, the participant specifies a series of

waypoints while the robot is in pause state. Therefore, we use the times of waypoint specification to measure the amount of interaction. Under teleoperation, the participant manually and continuously drives the robot while monitoring its state. Time spent in teleoperation was measured as the duration of a series of active positional control actions that were not interrupted by pauses of greater than 30 sec. or any other form of control action. For camera control, times of camera operation were used because the operator controls the camera by issuing a desired pose, and monitoring the camera's movement.

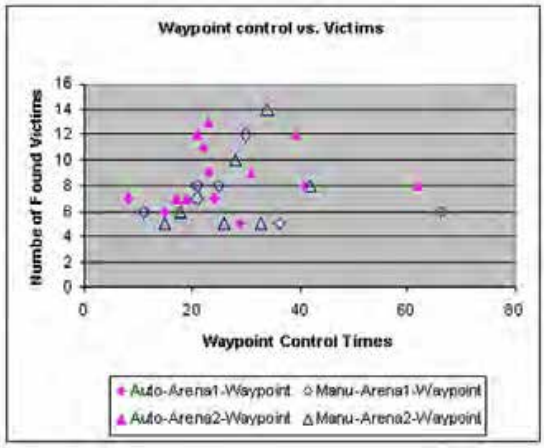


Figure 9. Victims found as a function of waypoint

While we did not find differences in overall waypoint control times between mixed-initiative and manual modes, mixed-initiative operators had shorter, $t(13) = 3.02$, $p < .01$, control times during any single control episode, the period during which an operator switches to a robot, controls it and then switches to another robot.

Figure 9 shows the relationship between victims found and total waypoint control times. In manual mode this distribution follows an inverted 'U' with too much or too little waypoint control leading to poor search performance. In mixed-initiative mode by contrast the distribution is skewed to be less sensitive to control times while holding a better search performance, i.e. more found victims (see section 5.1).

Overall teleoperation control times, $t(13) = 2.179$, $p < .05$ were reduced in the mixed-initiative mode as well, while teleoperation times within episodes only approached significance, $t(13) = 1.87$, $p = .08$. No differences in camera control times were found between mixed-initiative and manual control modes. It is notable that operators made very little use of teleoperation, .6% of mission time, and only infrequently chose to control their cameras.

5.4 Trust and Capability of Using Interface

In the posttest we collected participants' ratings of their level of trust in the system's automation and their ability to use the interface to control the robots. 43% of the

participants trusted the autonomy and only changed the robot's plans when they had spare time. 36% of the participants reported changing about half of the robot's plans while 21% of the participants showed less trust and changed the robot's plans more often. A one tail t-test, indicates that the total victims found by participants trusting the autonomy is larger than the number victims found by other participants ($p=0.05$). 42% of the participants reported being able to use the interface well or very well, while 58% of the participants reported having difficulty using the full range of features while maintaining control of the robots. A one tail t test shows that participants reporting using the interface well or very well found more victims ($p<0.001$). Participants trusting the autonomy reported significantly higher capability in using the user interface ($p=0.001$) and conversely participants reporting using the interface well also had greater trust in the autonomy ($p=0.032$).

6. Conclusion

In this experiment, the first of a series investigating control of cooperating teams of robots, cooperation was limited to deconfliction of plans so that robots did not re-explore the same regions or interfere with one another. The experiment found that even this limited degree of autonomous cooperation helped in the control of multiple robots. The results showed that cooperative autonomy among robots helped the operators explore more areas and find more victims. The fully autonomous control condition demonstrates that this improvement was not due solely to autonomous task performance as found in (Schurr *et al.* 2005) but rather resulted from mixed initiative cooperation with the robotic team. The superiority of mixed initiative control was far from a foregone conclusion since earlier studies with comparable numbers of individually autonomous robots (Trouvain & Wolf 2002; Nielsen *et al.* 2003; Trouvain *et al.* 2003; Crandall *et al.* 2005) found poorer performance for higher levels of autonomy at similar tasks. We believe that differences between navigation and search tasks may help explain these results. In navigation, moment to moment control must reside with either the robot or the human. When control is ceded to the robot the human's workload is reduced but task performance declines due to loss of human perceptual and decision making capabilities. Search by contrast can be partitioned into navigation and perceptual subtasks allowing the human and robot to share task responsibilities improving performance. This explanation suggests that increases in task complexity should widen the performance gap between cooperative and individually autonomous systems. We did not collect workload measures to check for the decreases found to accompany increased autonomy in earlier studies (Trouvain & Wolf 2002; Nielsen *et al.* 2003; Trouvain *et al.* 2003; Crandall *et al.* 2005), however, eleven of our fourteen subjects reported benefiting from robot cooperation.

Our most interesting finding involved the relation between performance and switching of attention among the robots. In both the manual and mixed initiative conditions participants divided their attention approximately equally among the robots but in the mixed initiative mode they switched among robots more rapidly. Psychologists (Meiran *et al.* 2000) have found task switching to impose cognitive costs and switching costs have previously been reported (Squire *et al.* 2003; Goodrich *et al.* 2005) for multi-robot control. Higher switching costs might be expected to degrade performance, however in this study; more rapid switching was associated with improved performance in both manual and mixed initiative conditions. We believe that the map component at the bottom of the display helped mitigate

losses in awareness when switching between robots and that more rapid sampling of the regions covered by moving robots gave more detailed information about areas being explored.

The frequency of this sampling among robots was strongly correlated with the number of victims found. This effect, however, cannot be attributed to a change from a control to a monitoring task because the time devoted to control was approximately equal in the two conditions. We believe instead that searching for victims in a building can be divided into a series of subtasks involving things such as moving a robot from one point to another, and/or turning a robot from one direction to another with or without panning or tilting the camera. To effectively finish the searching task, we must interact with these subtasks within their neglect time (Crandall *et al.* 2005) that is proportional to the speed of movement. When we control multiple robots and every robot is moving, there are many subtasks whose neglect time is usually short. Missing a subtask means we failed to observe a region that might contain a victim. So switching robot control more often gives us more opportunity to find and finish subtasks and therefore helps us find more victims. This focus on subtasks extends to our results for movement control which suggest there may be some optimal balance between monitoring and control. If this is the case it may be possible to improve an operator's performance through training or online monitoring and advice.

We believe the control episode observed in this experiment corresponds to a decomposed subtask of the team and the linear relationship between switches and found victims reveals the independent or weak relationship among the subtasks. For a multi-robot system, decomposing the team goal into independent or weakly related sub goals allowing the human to intervene into the sub goals is a potential way to improve and analyze human multi-robot performance. From the view of interface design, the interface should fit the sub goal decomposition (or sub goal template) and help the operator in attaining SA. Under mixed-initiative control condition, the number of found victims is less sensitive to waypoint specification than under manually control condition. The relation between found victims and waypoint specification can be generalized to the relationship between performance and human intervention. The potential of extending the present experiment to a generic HRI sensitivity evaluation methodology deserves a further study in the future. Moreover, the control episode can be used as a unit of human intervention, rather than the traditional counting of control actions or durations.

7. References

Browning B. & Tryzelaar E. (2003) UberSim: A Realistic Simulation Engine for Robot Soccer. In: *Proceedings of Autonomous Agents and Multi-Agent Systems, AAMAS'03*, Australia

Bruemmer D.J., Few D.A., Boring R.L., Marble J.L., Walton M.C. & Nielsen C.W. (2005) Shared Understanding for Collaborative Control. *IEEE Transactions On Systems, Man, And Cybernetics-Part A: Systems And Humans*, 35, 494-504

Burke J.L., Murphy R.R., Coovert M.D. & Riddle D.L. (2004) Moonlight in Miami: Field Study of Human-Robot Interaction in the Context of an Urban Search and Rescue Disaster Response Training Exercise. *Human Ccompter Interaction*, 19, 85-116

Carpin S., Stoyanov T., Nevatia Y., Lewis M. & Wang J. (2006) Quantitative assessments of USARSim accuracy. In: *Proceedings of PerMIS 2006*

Carpin S., Wang J., Lewis M., Birk A. & Jacoff A. (2005) High fidelity tools for rescue robotics: Results and perspectives. In: *Robocup 2005: Robot Soccer World Cup IX*, pp. 301-311

Casper J. & Murphy R.R. (2003) Human-robot interactions during the robot-assisted urban search and rescue response at the World Trade Center. *IEEE Transactions on Systems, Man and Cybernetics, Part B*, 33, 367- 385

Crandall J.W., Goodrich M.A., Olsen D.R. & Nielsen C.W. (2005) Validating human-robot interaction schemes in multitasking environments. *IEEE Transactions on Systems, Man, and Cybernetics, Part A*, 35, 438-449

Cyberbotics Ltd. (2006) Webots. URL http://www.cyberbotics.com/

Endsley M.R. (1996) Automation and situation awareness. In: *Automation and human performance: Theory and applications* (eds. Parasuraman R & Mouloua M), pp. 163-181. Erlbaum, Mahwah, NJ

Envarli I.C. & Adams J.A. (2005) Task Lists for Human-Multiple Robot Interaction. In: Proceedings of the 14th IEEE International Workshop on Robot and Human Interactive Communication

Fong T.W., Thorpe C. & Baur C. (2001) Collaboration, Dialogue, and Human-Robot Interaction. In: *Proceedings of the 10th International Symposium of Robotics Research*. Springer-Verlag, Lorne, Victoria, Australia

Gerkey B., Vaughan R. & Howard A. (2003) The Player/Stage Project: Tools for Multi-Robot and Distributed Sensor Systems. In: *Proceedings of the International Conference on Advanced Robotics (ICAR 2003)*, pp. 317-323, Coimbra, Portugal

Goodrich M., Quigley M. & Cosenzo. K. (2005) Switching and multi-robot teams. In: *Proceedings of the Third International Multi-Robot Systems Workshop*

Jacoff A., Messina E. & Evans J. (2001) Experiences in deploying test arenas for autonomous mobile robots. In: *Proceedings of the 2001 Performance Metrics for Intelligent Systems (PerMIS) Workshop*, Mexico City, Mexico

Kirlik A. (1993) Modeling strategic behavior in human automation interaction: Why an 'aid' can (and should) go unused. *Human Factors*, 35, 221–242

Konolige K. & Myers K. (1998) The Saphira Architecture for Autonomous Mobile Robots. In: *Artificial intelligence and mobile robots: case studies of successful robot systems* (eds. Kortenkamp D, Bonasso RP & Murphy R), pp. 211-242. MIT Press, Cambridge, MA

Lee J.D. & See K.A. (2004) Trust in Automation: Designing for Appropriate Reliance. *Human Factors*, 46, 50-80

Lee P., Ruspini D. & Khatib O. (1994) Dynamic simulation of interactive robotic environment. In: *Proceedings of International Conference on Robotics and Automation*, pp. 1147-1152

Lewis M., Wang J. & Hughes S. (2007) USARSim : Simulation for the Study of Human-Robot Interaction. *Journal of Cognitive Engineering and Decision Making*, 1, 98-120

Lewis, M. and Wang, J. (2007). Gravity referenced attitude display for mobile robots : Making sense of what we see, Transactions on Systems, Man and Cybernetics Part A, 37(1), 94-105.

Marble J.L., Bruemmer D.J. & Few D.A. (2003) Lessons learned from usability tests with a collaborative cognitive workspace for human-robot teams. In: *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics*, pp. 448-453

Marble J.L., Bruemmer D.J., Few D.A. & Dudenhoeffer D.D. (2004) Evaluation of supervisory vs. peer-peer interaction with human-robot teams. In: *Proceedings of the 37th Annual Hawaii International Conference on System Sciences*

Meiran N., Chorev Z. & Sapir A. (2000) Component processes in task switching. *Cognitive Psychology*, 41, 211-253

Nickerson J.V. & Skiena S.S. (2005) Attention and Communication: Decision Scenarios for Teleoperating Robots. In: *Proceedings of the 38th Annual Hawaii International Conference on System Sciences*

Nielsen C.N., Goodrich M.A. & Crandall J.W. (2003) Experiments in Human-Robot Teams. In: *Proceedings of the 2002 NRL Workshop on Multi-Robot Systems*

Nielsen C.W. & Goodrich M.A. (2006) Comparing the Usefulness of Video and Map Information in Navigation Tasks. In: *Proceedings of the 2006 Human-Robot Interaction Conference*, Salt Lake City, Utah

Olsen D.R. & Wood S.B. (2004) Fan-out: measuring human control of multiple robots. In: *Proceedings of the SIGCHI conference on Human factors in computing systems*, pp. 231-238. ACM Press, Vienna, Austria

Parasuraman R., Galster S., Squire P., Furukawa H. & Miller C. (2005) A Flexible Delegation-Type Interface Enhances System Performance in Human Supervision of Multiple Robots: Empirical Studies with RoboFlag. *IEEE Systems, Man and Cybernetics-Part A, Special Issue on Human-Robot Interactions*, 33, 481-493

Parasuraman R. & Miller C.A. (2004) Trust and etiquette in high-criticality automated systems. *Communications of the ACM*, 47, 51-55

Scerri P., Xu Y., Liao E., Lai G., Lewis M. & Sycara K. (2004) Coordinating large groups of wide area search munitions. In: *Recent Developments in Cooperative Control and Optimization* (eds. Grundel D, Murphey R & Pandalos P), pp. 451-480. Singapore: World Scientific

Schurr N., Marecki J., Tambe M., Scerri P., Kasinadhuni N. & Lewis J. (2005) The Future of Disaster Response: Humans Working with Multiagent Teams using DEFACTO. In: *Proceedings of AAAI Spring Symposium on AI Technologies for Homeland Security*

Sheridan T.B. (2002) *Humans and Automation: System Design and Research Issues*. Human Factors and Ergonomics Society and Wiley, Santa Monica, CA and New York.

Squire P., Trafton G. & Parasuraman R. (2003) Human control of multiple unmanned vehicles: effects of interface type on execution and task switching times. In: *Proceedings of the 2006 Human-Robot Interaction Conference*, pp. 26-32, Salt Lake City, Utah

Trouvain B., Schlick C. & Mevert M. (2003) Comparison of a map- vs. camera-based user interface in a multi-robot navigation task. In: *Proceedings of the 2003 International Conference on Robotics and Automation*, pp. 3224-3231

Trouvain B. & Wolf H.L. (2002) Evaluation of multi-robot control and monitoring performance. In: *Proceedings of the 2002 IEEE Int. Workshop on Robot and Human Interactive Communication*, pp. 111-116

Wang J., Lewis M. & Gennari J. (2003) A game engine based simulation of the NIST urban search and rescue arenas. In: *Proceedings of the 2003 Winter Simulation Conference*, pp. 1039-1045

Wang J., Lewis M., Hughes S., Koes M. & Carpin S. (2005) Validating USARsim for use in HRI Research. In: *Proceedings of the Human Factors and Ergonomics Society 49th Annual Meeting*, pp. 457-461

Woods D.D., Tittle J., Feil M. & Roesler A. (2004) Envisioning human-robot coordination in future operations. *IEEE Transactions on Systems, Man & Cybernetics*, 34, 210-218

Yanco H.A., Drury J.L. & Scholtz J. (2004) Beyond Usability Evaluation: Analysis of Human-Robot Interaction at a Major Robotics Competition. *Journal of Human-Computer Interaction*, 19, 117-149

24

Robotic Musicianship – Musical Interactions Between Humans and Machines

Gil Weinberg
Georgia Institute of Technology
USA

1. Introduction

The Robotic Musicianship project aims to facilitate meaningful musical interactions between humans and machines, leading to novel musical experiences and outcomes. The project combines computational modelling of music perception, interaction, and improvisation, with the capacity to produce acoustic responses in physical and visual manners. The motivation for this work is based on the hypothesis that real-time collaboration between human and robotic players can capitalize on the combination of their unique strengths to produce new and compelling music. Our goal is to combine human qualities such musical expression and emotions with robotic traits such as powerful processing, the ability to perform sophisticated mathematical transformations, robust long-term memory, and the capacity to play accurately without practice. A similar musical interaction can be achieved with software applications that do not involve mechanical operations. However, software-based interactive music systems are hampered by their inanimate nature, which does not provide players and audiences with physical and visual cues that are essential for creating expressive musical interactions. For example, motion size often corresponds to loudness and gesture location often relates to pitch. These cues provide visual feedback, help performers anticipate and coordinate their playing, and create an engaging musical experience by providing a visual connection to the generated sound. Software based interactive music systems are also limited by the electronic reproduction and amplification of sound through speakers, which cannot fully capture the richness of acoustic sound. Unlike these systems, the anthropomorphic musical robot we developed, named Haile, is designed to create acoustically rich interactions with humans. The acoustic richness is achieved due to the complexities of real life systems, as opposed to digital audio nuances that require intricate design and that are limited by the fidelity and orientation of speakers. In order to create intuitive as well as inspiring social collaboration with humans, Haile is designed to analyze music based on computational models of human perception and to generate algorithmic responses that are unlikely to be played by humans ("listen like a human, improvise like a machine"). It is designed to serve as a test-bed for novel forms of musical human-machine interactions, bringing perceptual aspects of computer music into the physical world both visually and acoustically. We believe that this approach can lead to new musical experiences, and to new music, which cannot be conceived by traditional means.

2. Related Work

Two main research areas inform our effort to develop robotic musicianship – *musical robotics*, which focuses on the construction of automated mechanical sound generators and *machine musicianship*, which centres on computer models of music theory, composition, perception, and interaction. Early work on musical robotics focused on mechanical keyboard instruments such as the Pianista by French inventor Fourneax (see a comprehensive historic review of musical robots in (Kapur 2005)). In recent years, the field has received commercial, artistic, and academic interest, expanding to anthropomorphic designs as well as robotic musical instruments, including chordophones, aerophones, membranophones and idiophones. Several approaches have been recently explored for robotic stringed instruments. GuitarBot (Singer et al. 2004) , for example, is a mechanical guitar operated by a set of DC servomotors driving a belt with multiple picks playing four strings. The pick position, controlled by a photosensor and a "clapper" solenoid, is used as a damper. Jordà's Electric Guitar Robot (Jordà 2002), on the other hand, has six strings, that can be plucked by twelve picks, driven by an electro-valve hammer-finger. Current approaches for mechanical guitars, however, are not designed to explore the full range of sonic variety through string techniques such as bouncing, bowing, strumming, scratching, or rubbing. Other attempts have been made to develop expressive wind instrument robots. The Anthropomorphic Flute Robot (Chida, Okuma et al. 2004), uses a complex mechanical imitation of human organs in an effort to accurately reproduce human flute playing. The elaborate apparatus includes robotic lungs, neck, lips, fingers, and tongue. Other examples for aerophone robotic instruments are Toyota's Robotic Trumpeter (Toyota 2007) and the Rae's Autosax (Rae 2005), which are programmed to follow deterministic rules. More varied work has been done on robotic percussionists, both for idiophone and membranophone instruments. The ModBots (Singer et al. 2004), for example, are miniature modular instruments designed to affix to virtually any structure. Each ModBot consists of only one electromechanical actuator (a rotary motor or a linear solenoid), which responds to varying degrees of supply voltage regulated by a microcontroller. A more elaborated mechanism by Singer is utilized in the TibetBots, which consist of six robotic arms that strike three Tibetan singing bowls. Here, an effort was made to capture a wider timbral variety by using two robotic arms (controlled by solenoids) for each bowl to produce a richer set of sounds. Another approach for broadening timbre and pitch versatility is taken by the Thelxiepeia (Baginsky 2004). The instrument consists of a mechanical drumstick and a motorized mechanism to rotate the drum circumference, which can lead to the production of a range of pitches. Other robotic instruments which influenced our work were developed by Trimpin (Trimpin 2000), Rae (Rae 2005), and Van Doressen (Dorssen 2006).

The second research area that informs our work is machine musicianship. Here, researchers design and develop computer systems that analyze, perform, and compose music based on theoretical foundations in fields such as music theory, computer music, cognition, artificial intelligence and human-computer-interaction (Rowe 1992). One of the earliest research directions in this field is the "Score Follower", in which the computer tracks a live soloist and synchronizes MIDI (Dannenberg 1984) (Vercoe 1984), and recently audio (Orio, Lemouton et al. 2003), accompaniment to the musical input. The classic score following approach focuses on matching predetermined musical events to real-time input. A more improvisatory approach is taken by systems such as Voyager (Lewis 2000) and Cypher (Rowe 1992). Here the software analyzes musical input in real time and generates musical

responses by manipulating a variety of parameters such as melody, harmony, rhythm, timbre, and orchestration. David Cope has taken a non-real time approach in his system for analyzing composers' styles based on MIDI renditions of their compositions (Cope 1996). Cope's algorithm learns the style of a given composer by modelling aspects such as expectation, memory, and musical intent. It can then generate new compositions with stylistic similarities to the originals. The "Continuator" system, on the other hand, takes a real-time approach for learning the improvisation style of musicians as they play polyphonic MIDI instruments (Pachet 2003). The application uses Hidden Markov Models to learn and analyze the input and continues the improvisation in the style of the human performer.

Particularly note-worthy research field in machine musicianship is computational modelling of music perception in which, researches develop cognitive and computational models of low- and high-level musical percepts. Lower level cognitive modelling address percepts from note onset detection to pitch and beat detection, using audio sources (Puckette 1998) (Scheirer 1998) (Foote and Uchihashi 2001) as well as MIDI (Winkler 2001). Higher-level rhythmic percepts include more subjective concepts such as rhythmic stability, melodic similarity and attraction. Desain and Honing's model of rhythmic stability is based on the relationship between pairs of adjacent note durations (Desain and Honing 2002); Tanguiane counts the number of coincident onset in an effort to model rhythmic similarity of different audio signals (Tanguiane 1993); Smith utilizes dynamic time warping techniques to retrieve similar melodies from a folk song database (Smith et al. 1998); and Lerdahl and Jackendoff calculate the melodic attraction between pitches in a given tonality based on a table of anchoring strengths (Lerdahl & Jackendoff 1983). Informed by such approaches for perceptual modelling, our robot is designed to respond with algorithmically generated musical outcomes using a novel approach for computational composition and improvisation. This aspect of the system is based on theoretical approaches for musical improvisation and interaction (Pressing 1994), (Johnson-Laird 2002), as well as practical efforts using methods such as genetic algorithms. GenJam, for example, is an interactive computer system that improvises over a set of jazz tunes using an initial phrase population that is generated stochastically (Biles 1994). GenJam's fitness function is based on human input, where in every generation the user determines which phrases remain in the population. Other systems use methods such as real-time fitness criteria (Moroni 2000) or human feedback for training a neural network-based fitness function (Tokui & Iba 2000). In the second phase of the robotic musicianship project we developed an improvisatory genetic algorithm that combines human aesthetics and perception with algorithmic "gene mixing" improvisation.

3. Research questions

A number of research questions guide our effort to create intuitive and inspiring musical human-robot interactions and to establish the concept of robotic musicianship:

- Can we effectively implement computational schemes that model how humans represent and process rhythmic, melodic, and harmonic structures in music? Can a robot use such models to infer high-level musical meaning from live musical input and respond in a musically intuitive manner?
- Can algorithmic models of musical improvisation create meaningful and inspiring musical responses? Can such algorithmic responses lead to novel socio-musical human-machine interaction and to music that cannot be created by humans?

- What is the role of physical, visual, and acoustic cues in multi-player musical interactions? Can a robot utilize physical properties to enrich musical interactions with humans?

Below we describe our efforts to address these research questions through physical and mechanical design (section 4), rhythmic and melodic applications (sections 5), user studies (section 6), and a number of directions for future work (section 7).

4. Physical and Mechanical Design

Figure 1. Haile's design

In order to support familiar and expressive interactions with human players, Haile's design is anthropomorphic, utilizing two percussive arms that can move to different locations and strike with varying velocities. The first prototype was designed to play a Native American Pow Wow drum – a unique multi player instrument that supports the collaborative nature of the project. For pitch-oriented applications, the robot was later adjusted to play a one-octave xylophone. In order to match the aesthetics of these musical instruments, we chose to construct the robot from wood. The wooden parts were made using a CnC wood cutting machine and constructed from several layers of plywood glued together. Metal joints were designed to allow shoulder and elbow movement as well as leg adjustability for different instrument heights. While attempting to create an organic look for the robot, it was also important that the technology was not completely hidden, so that co-players could see and understand the robot's operation. We therefore left the mechanical apparatuses uncovered and embedded a number of LEDs on Haile's body, providing an additional representation of the mechanical actions (See Figure 1).

Haile controls two robotic arms; the right arm is designed to play fast notes, while the left arm is designed to produce larger and more visible motions that produce louder sounds. Both arms can adjust the strikes sound in two manners: different pitches are achieved by striking the instruments in different locations, and volume is adjusted by hitting with varying velocities. To move to different vertical positions, each arm employs a linear slide, a

belt, a pulley system, and a potentiometer to provide feedback (see Figure 2). Unlike robotic drumming systems that allow hits at only a few discrete locations, Haile's arms move continuously over a distance of 10 inches (movement timing is 250 ms. from end to end). The right arm's striking mechanism is loosely based on a piano hammer action and consists of a solenoid driven device and a return spring (see Figure 3). The arm strikes at a maximum speed of 15 Hz, faster than the left arm's maximum speed of 11 Hz. However, the right arm cannot generate a wide dynamic range or provide easily noticeable visual cues, which limits Haile's expression and interaction potential. The left arm was designed to address these shortcomings, using larger visual movements, and a more powerful and sophisticated hitting mechanism. Whereas the striking component of the right arm is about the size of a finger and can only move 2.5 inches vertically (see figure 4), the entire left forearm takes part in the striking motion and can move up and down eight inches. A linear motor and an encoder located at the left elbow are used to provide sufficient force and control for the larger mass and motions (see Figure 5).

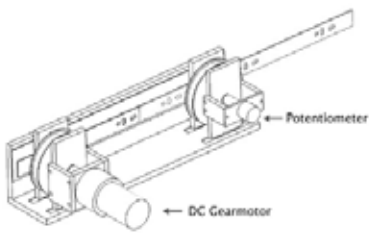


Figure 2. The right arm slider mechanisms

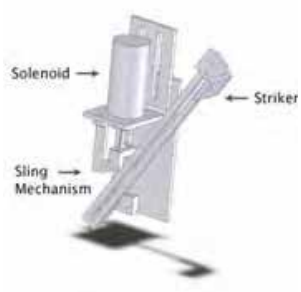


Figure 3. The right arm striking mechanism

Max/MSP, a graphical programming environment (Cycling74 2007), was used for high-level musical programming in an effort to make the project accessible to composers, performers, and students. The first right arm prototype incorporated the USB based Teleo System (MakingThings 2005) as the main interface between Max/MSP and Haile's sensors and motors. Low-level control of the solenoid-based right arm's position was computed within Max/MSP, which required a continuous feed of position updates to the computer. This consumed much of the communication bandwidth as well as processor time on the main computer. The final two-arm mechanism utilizes multiple onboard microprocessors for local low-level control as well as Ethernet communication with the main computer. The new

system, therefore, facilitates faster and more sophisticated control (2ms control loop) and requires only low bandwidth communications with the operating computer. Each arm is locally controlled by an 18F452 PIC microprocessor, both of which receive RS232 communications from a Modtronix Ethernet board (SBC68EC). The Ethernet board receives 3 byte packets from the computer, a control byte and two data bytes. The protocol utilizes an address bit in the control byte to send the information to the appropriate arm processor. The two data bytes typically contain position and velocity set points for each hit, but can also be used to update the control parameters.

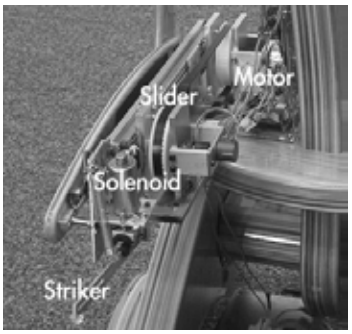


Figure 4. Haile's right arm design

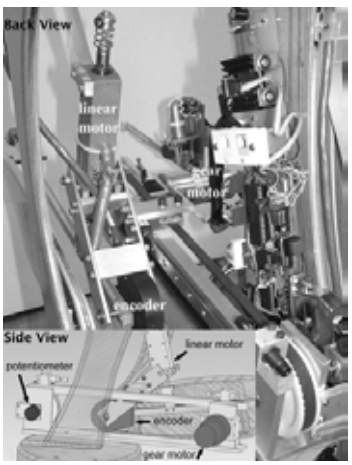


Figure 5. Haile's left arm design

Two onboard PIC microprocessors are responsible for controlling the arms' sliding and hitting mechanisms, ensuring that the impacts occur at the requested position and velocity. In order to allow enough time for the arms to move to the correct location and execute the strokes, a 300 ms. delay line is implemented between signal reception and impact. It has been shown that rhythmic errors of only 5 ms are detectable by average listeners (Coren, et al,

2003), therefore, it was important to ensure that this delay remained accurate and constant regardless of different hit velocities, allowing to easily compensate for it in the higher-level interaction application. Both arms store incoming hit commands in a First-In-First-Out queue, moving towards the location of a new note immediately after each hit. Due to its short vertical hitting range, the solenoid driven right arm allows for fairly consistent stroke time. We, therefore, implemented the 300 ms delay as a constant for this arm. The left arm, on the other hand, undergoes much larger movements, which requires complex feedback control to ensure that impact occurs at the right time, regardless of hits velocity. While waiting for incoming notes, the left arm remains about one inch above the surface of the instrument. When a new note is received, the arm is raised to a height proportional to the loudness of the hit. After a delay determined by the desired velocity and elevation, the arm descends towards the instrument under velocity control. After impact, the arm returns back to its standby position above the instrument. Extremely fast notes utilize a slightly different control mode that makes use of the bounce of the arm in preparation for the next hit. This mechanism allows the left arm to control a wide dynamic range and provides performers and viewers with anticipatory and real-time visual cues, enhancing expression and enriching the interaction representation.

5. Applications

5.1 Phase one – Rhythmic Interaction

The first phase of the project aimed at facilitating rhythmic collaboration between human drummers and Haile, addressing aspects such as rhythmic perception, improvisation, and interaction. In perception, we developed models for low- and high-level rhythmic percepts, from hit onset, amplitude, and pitch detection, through beat and density analysis, to rhythmic stability and similarity perception. For hit onset and amplitude detection we adjusted the Max/MSP bonk~ object (Puckette 1998) to address the unique character of the Pow Wow drum- a multi player Native American percussion instrument, which was chosen for the project due to its collaborative nature. Bonk~ provides effective onset attack detection but its frequency bands analysis is insufficient for accurate pitch detection due to the Pow Wow drum's low frequency and long reverberating sounds. Since bonk~ is hard-coded with a 256 point analysis window, the lowest frequency it can analyze is 172Hz – too high for the Pow Wow drum which has a natural frequency of about 60 Hz. Moreover, onset detection is complicated when high frequency hits are masked by the long decay of the previous low strikes (see Figure 6). To address these issues, we wrote a Max/MSP external object that used 2048 points FFT to determine both the magnitude of lower frequency bins and the change in those magnitudes between successive analysis frames. By taking into account the spectral changes in addition to the magnitudes, Haile could better determine whether energy in a particular frequency band came from a current hit or from previous ones (See Figure 6).

Other relatively low-level perceptual modules that were developed were beat detection, where domain detection was followed by autocorrelation of tempo and phase (Davies and Plumbley 2005) and density detection, where we looked at the number of note onsets per time unit to represent the density of the rhythmic structure. We also implement a number of higher-level rhythmic analysis modules for percepts such as rhythmic stability, based on (Desain and Honing 2002), and similarity based on (Tanguiane 1993). The stability model calculates the relationship between pairs of adjacent note durations, rated according to their perceptual expectancy based on three main criteria: perfect integer relationships are favoured,

ratios have inherent expectancies (i.e., 1:2 is favoured to 1:3 and 3:1 is favoured to 1:3), and durations of 0.6 seconds are preferred. The expectancy function can be computed as:

$$E_b(A,B) = \int_0^r (\text{round}(r) - r) \times \left| 2(r - \text{floor}(r) - 0.5) \right|^p \times \text{round}(r)^d \, dr$$

where A and B are the durations of the two neighbouring notes, $r = \max(A/B, B/A)$ represents the (near) integer relationship between note durations, p controls the shape of the peaks, and d is negative and affects the decay rate as the ratios increases. This function is symmetric around r=1 when the total duration is fixed (see Figure 7a). Generally, the expectancy function favours small near-integer ratios and becomes asymmetric when the total duration varies, exhibiting the bias toward the 600 ms. interval (see Figure 7b).

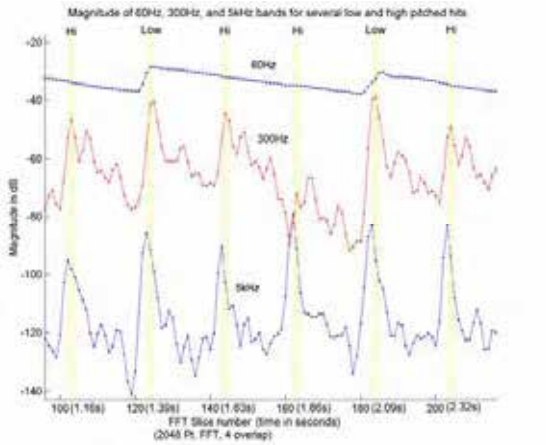
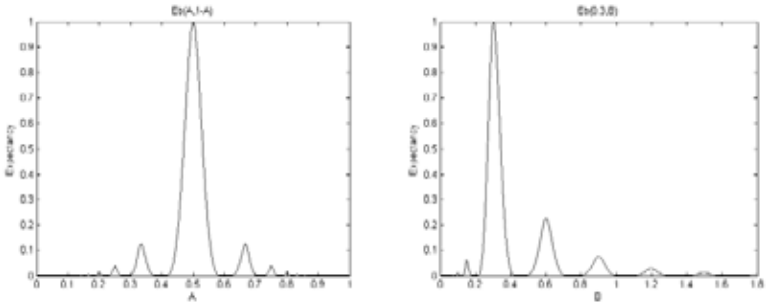


Figure 6. Magnitude plots from a 60Hz, 300Hz, and 5kHz frequency band over several low and high-pitched hits showing the relatively slow decay of the low-pitched hits



(a) (b)

Figure 7. Basic expectancy of intervals A and 1-A (a) and 0.3 and B (b), reproduced from (Desain and Honing 2002)

Our similarity rating is derived from Tanguiane' binary representation, where two rhythms are first quantized, and then given a score based on the number of note onset overlaps and near-overlaps. In order to support real-time interaction with human players, we developed two Max/MSP externals that analyzed and generated rhythms based on these stability and similarity models. These externals were embedded in a live interaction module that read measure-length rhythmic phrases and modified them based on desired stability and similarity parameters. Both parameters varied between 0 and 1 and were used together to select an appropriate rhythm from a database of pre-analyzed rhythms. A stability rating of 1 indicated the most stable rhythm in the database, 0.5 equated to the stability of the input rhythm, and 0 to the least stable rhythm. The similarity parameter determined the relative contribution of similarity and stability.

The main challenge in designing the rhythmic interaction with Haile was to implement our perceptual modules in a manner that would lead to an inspiring human-machine collaboration. The approach we took to address this challenge was based on a theory of interdependent group interaction in interconnected musical networks (Weinberg 2005). At the core of this theory is a categorization of collaborative musical interactions in networks of artificial and live musicians based on sequential and synchronous operations with centralized and decentralized control schemes. For example, in sequential decentralized interactions, players create their musical materials with no influence from a central system or other players and can then interact with the algorithmic response in a sequential manner (see Figure 8). In a synchronous centralized network topology, on the other hand, players modify and manipulate their peers' music in real-time, interacting through a computerized hub that performs analysis and executes generative functions (see Figure 9). More sophisticated schemes of interaction can be designed by combining centralized, decentralized, synchronous, and sequential interactions in different directions, and by embedding weighted gates of influence among participants (see Figure 10).

Figure 8. A model of sequential decentralized interaction. Musical actions are taken in succession without synchronous input from other participants, and with no central system to coordinate the interaction

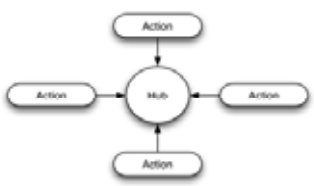


Figure 9. A model of synchronous centralized interaction. Human and machine players are taking musical actions simultaneously, and interact through a computerized hub that interpret and analyze the input data

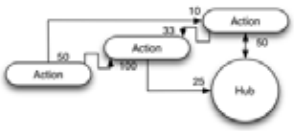


Figure 10. A combination of centralized, decentralized, synchronous and sequential musical actions in an asymmetric topology with weighted gates of influence. Based on (Weinberg 2005)

Based on these ideas, we developed six interaction modes for Haile: Imitation, Stochastic Transformation, Perceptual Transformation, Beat Detection, Simple Accompaniment, and Perceptual Accompaniment. These interaction modes utilize different perceptual modules and can be embedded in different combinations in interactive compositions and educational activities. In the first mode, Imitation, Haile merely repeats what it hears based on its low-level onset, pitch, and amplitude perception modules. Players can play a rhythm and after a couple of seconds of inactivity Haile imitates it in a sequential call-and-response manner. Haile uses one of its arms to play lower pitches close to the drumhead centre and the other arm to play higher pitches close to the rim. In the second mode, Stochastic Transformation, Haile improvises in a call-and-response manner based on players' input. Here, the robot stochastically divides, multiplies, or skips certain beats in the input rhythm, creating variations of users' rhythmic motifs while keeping their original feel. Different transformation coefficients can be adjusted manually or automated to control the level of similarity between users motifs and Haile's responses. In the Perceptual Transformation mode, Haile analyzes the stability level of users' rhythms, and responds by choosing and playing other rhythms that have similar levels of stability to the original input. In this mode Haile automatically responds after a specified phrase length. Imitation, Stochastic Transformation, and Perceptual Transformation are all sequential interaction modes that form decentralized call-and-response routines between human players and the robot. Beat Detection and Simple Accompaniment modes, on the other hand, allow synchronous interaction where humans play simultaneously with Haile. In Beat Detection mode, Haile tracks the tempo and beat of the input rhythm using complex domain detection function and autocorrelation, which leads to continuously refined assumptions of tempo and phase. A simpler, yet effective, synchronous interaction mode is Simple Accompaniment, where Haile plays pre-recorded MIDI files so that players can interact with it by entering their own rhythms or by modifying elements such as drumhead pressure to modulate and transform Haile's timbres in real-time. This synchronous centralized mode allows composers to feature their structured compositions in a manner that is not susceptible to algorithmic transformation or significant user input. The Simple Accompaniment mode is also useful for sections of synchronized unisons where human players and Haile play together. Perhaps the most advanced mode of interaction is the Perceptual Accompaniment mode, which combines synchronous, sequential, centralized and decentralized operations. Here, Haile plays simultaneously with human players while listening to and analyzing their input. It then creates local call-and-response interactions with different players, based on its perceptual analysis. In this mode we utilize the amplitude and density perceptual modules that are described above. While Haile plays short looped sequences (captured during the Imitation and Stochastic Transformation modes) it also listens to and analyzes the amplitude and density curves of human playing. It then modifies its looped sequence, based on the

amplitude and density coefficients of the human players. When the rhythmic input from the human players is dense, Haile plays sparsely, providing only the strong beats and allowing humans to perform denser solos. When humans play sparsely, on the other hand, Haile improvises using dense rhythms that are based on stochastic and perceptual transformations. Haile also responds in direct relationship to the amplitude of human players so that the louder humans play, the stronger Haile plays to accommodate the human dynamics, and vice versa (see a video excerpts of some of the interaction modes at http://www.cc.gatech.edu/~gilwein/Haile.htm.)

Figure 11. The composition Jam'aa, as performed at the RoboRave Festival in Odense, Denmark

As a creative outcome for these rhythmic applications, two compositions were written for the system, each utilized a different set of perceptual and interaction modules. The first composition, titled Pow, was written for one or two human players and a one-armed robotic percussionist. It served as test case for Haile's early mechanical, perceptual, and interaction modules. The second composition, titled Jam'aa ("gathering" in Arabic), builds on the unique communal nature of the Middle Eastern percussion ensemble, attempting to enrich its improvisational nature, call-and-response routines, and virtuosic solos with algorithmic transformation and human-robotic interactions. Here, the sonic variety of the piece was enriched by using two robotic arms and by including other percussive instruments such as darbukas (goblet shaped middle-eastern hand drum), djumbes, and tambourines. In Jam'aa Haile listens to audio input via directional microphones installed inside two darbuka drums played by humans. In some sections of the piece the left arm merely provides the beat while in other sections it participates in the algorithmic interaction. Jam'aa utilizes interaction modes that were not included in Pow, such as perceptual transformation and perceptual

accompaniment. We also developed a new response algorithm for Jam'aa titled "morphing", where Haile combines elements from two or more of the motifs played by humans, based on a number of integration functions. Jam'aa, was commissioned by Hamaabada Art Centre In Jerusalem, and later performed in invited and juried concerts in France, Germany, Denmark, and the United States (see a video excerpts from Jam'aa at - http://coa.gatech.edu/~gil/RoboraveShort.mov)

5.2 Phase Two – Melodic Interaction

As part of our effort to expand the exploration of robotic musicianship into pitch and melody, Haile was adapted to play a pitch-based mallet instrument. The one-octave xylophone we built for this purpose was designed to fit Haile's mechanical design – the left arm covered a range of 5 keys while the right arm, whose vertical range was extendable, covered a range of 7 keys. The different mechanisms driving each arm led to unique timbres, as notes played by the solenoid-driven arm sound different than those played by the linear-motor based arm. Since the robot could play only one octave, the algorithmic responses were filtered by pitch class.

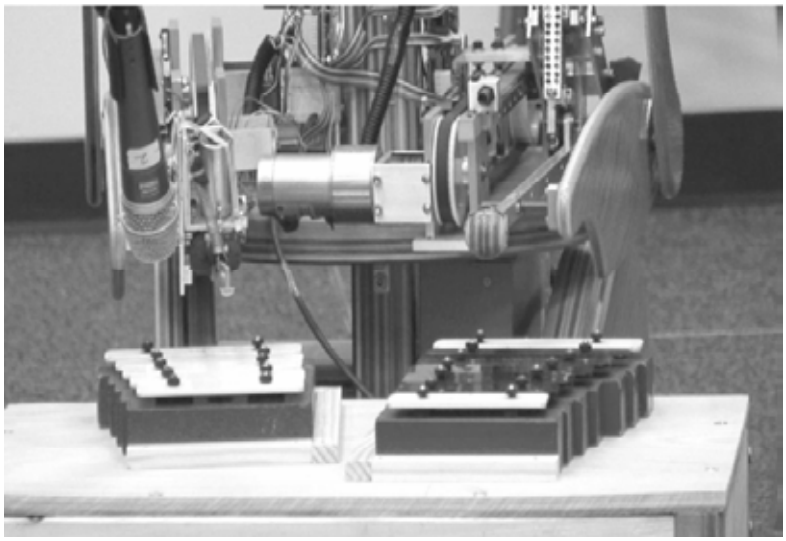


Figure 12. Haile's two robotic arms cover a range of one octave – from middle G to treble G

Following the guideline "listen like a human, improvise like a machine", we decided to implement a perceptual model of melodic similarity ("listen like a human") as the fit function of a genetic algorithm based improvisation engine ("improvise like a machine"). The algorithmic responses are based on the analyzed input as well as on internalized knowledge of contextually relevant material. The algorithm fragments MIDI and audio input to short phrases. It then attempts to find a "fit" response by evolving a pre-stored human-generated population of phrases using a variety of mutation and crossover functions over a variable number of generations. At each generation, the evolved phrases are

evaluated by a fitness function that measures similarity to the input phrase, and the least fit phrases in the database are replaced by members of the next generation. A unique aspect in this design is the reliance on a pre-recorded human-generated phrase set that evolves over a limited number of generations. This allows musical elements from the original phrases to mix with elements of the real-time input to create hybrid, and at times unpredictable, responses for each given input melody. By running the algorithm in real-time, the responses are generated in a musically appropriate timeframe.

Approximately forty melodic excerpts of variable lengths and styles were used as an initial population for the genetic algorithm (GA). The phrases were recorded by a jazz pianist who improvised in a similar musical context to that in which the robot was planed to perform. Having a distinctly "human" flavour, these phrases provided the GA with a rich initial pool of rhythmic and melodic "genes" from which to build its own melodies. This is notably different from traditional genetic algorithmic approaches in computer music, in which the starting population is generated stochastically. A similarity measure between the observed input and the melodic content of each generation of the GA was used as a fitness function. The goal was not to converge to an "ideal" response by maximizing the fitness metric (which could have led to an exact imitation of the input melody), rather to use it as a guide for the algorithmically generated melodies. By varying the number of generations and the type and frequency of mutations, certain characteristics of both the observed melody and some aspects of the base population could be preserved in the output.

Dynamic Time Warping (DTW) was used to calculate the similarity measure between the observed and generated melodies. A well-known technique originally used in speech recognition applications, DTW provides a method for analyzing similarity, either through time shifting or stretching, of two given segments whose internal timing may vary. While its use in pattern recognition and classification has largely been supplanted by newer techniques such as Hidden Markov Models, DTW was particularly well suited to the needs of this project, specifically the task of comparing two given melodies of potentially unequal lengths without referencing an underlying model. We used a method similar to the one proposed by (Smith, McNab et al. 1998), deviating from the time-frame-based model to represent melodies as a sequence of feature vectors, each corresponding to a note. Our dissimilarity measure assigns a cost to deletion and insertion of notes, as well as to the local distance between the features of corresponding pairs. The smallest distance over all possible temporal alignments was chosen, and the inverse (the "similarity" of the melodies) was used as the fitness value. The local distances were computed using a weighted sum of four differences: absolute pitch, pitch class, log-duration, and melodic attraction. The individual weights are configurable, each with a distinctive effect upon the musical quality of the output. For example, higher weights on the log-duration difference led to more precise rhythmic matching, while the pitch-based differences led to outputs that more closely mirrored the melodic contour of the input. Melodic attraction between pitches was calculated based on the Generative Theory of Tonal Music model (Lerdahl and Jackendoff 1983). The relative balance between the local distances and the temporal deviation cost had a pronounced effect upon the output – a lower cost for note insertion/deletion led to a highly variant output. Through substantial experimentation, we arrived at a handful of effective configurations.

The computational demands of a real-time context required significant optimization of the DTW, despite the relatively small length of the melodies (typically between two and thirty

notes). For searching through possible time alignments, we implemented a standard path constraint in which consecutive insertions or deletions were not allowed. This cut computation time by approximately one half but prohibited comparison of melodies whose lengths differ by more than a factor of two. These situations were treated as special cases and were assigned an appropriately low fitness value. Additionally, since the computation time was proportional to the length of the melody squared, a decision was made to break longer input melodies into smaller segments to increase the efficiency and remove the possibility of an audible time lag.

With each generation, a configurable percentage of the phrase population was chosen for mating. This "parent" selection was made stochastically according to a probability distribution calculated from each phrase's fitness value, so that more fit phrases were more likely to breed. The mating functions ranged from simple mathematical operations to more sophisticated musical functions. For instance, a single crossover function was implemented by randomly defining a common dividing point on two parent phrases and concatenating the first section from one parent with the second section from the other to create the child phrase. This mating function, while common in genetic algorithms, did not use structural information of the data and often led to non-musical intermediate populations of phrases. We also implemented musical mating functions that were designed to lead to musically relevant outcomes without requiring that the population converge to a maximized fitness value. An example of such a function is the pitch-rhythm crossover, in which the pitches of one parent are imposed on the rhythm of the other parent. Because the parent phrases were often of different lengths, the new melody followed the pitch contour of the first parent, and its pitches were linearly interpolated to fit the rhythm of the second parent.

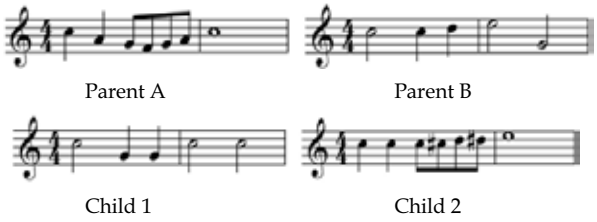


Figure 13. Mating of two prototypical phrases using the pitch-rhythm crossover function. Child 1 has the pitch contour of parent A and rhythm pattern of parent B while Child 2 has the rhythm of parent A and the pitch contour of parent B

Additionally, an adjustable percentage of each generation was mutated according to a set of functions that ranged in musical complexity. For instance, the simple random mutation function added or subtracted random numbers of semitones to the pitches within a phrase and random lengths of time to the durations of the notes. While this mutation seemed to add a necessary amount of randomness that allowed a population to converge toward the reference melody over many generations, it degraded the musicality of the intermediate populations. Other functions were implemented that would stochastically mutate a melodic phrase in a musical fashion, so that the outcome was recognizably derivative of the original. The density mutation function, for example, altered the density of a phrase by adding or removing notes, so that the resulting phrase followed the original pitch contour with a different number of notes. Other simple musical mutations included inversion, retrograde,

and transposition operations. In the end, we had seven mutation functions and two crossover functions available, any combination of which was allowed through a configurable interface.

In order for Haile to improvise in a live setting, we developed a number of human-machine interaction schemes driven by capturing, analyzing, transforming, and generating musical material in real-time. Much like a human musician, Haile was programmed to "decide" when to play, for how long, when to stop, and what notes to play in any given musical context. Our goal was to expand on the simple call-and-response format, creating autonomous behaviour in which the robot interacts with humans by responding, interrupting, ignoring, or introducing new material that aims to be surprising and inspiring. The system received and analyzed both MIDI and audio information. Input from a digital piano was collected using MIDI while the MSP object pitch~ was used for pitch detection of melodic audio from acoustic instruments. In an effort to establish Haile's listening abilities in live performance settings, simple interaction schemes were developed that do not use the genetic algorithm. One such scheme was direct repetition of human input, in which Haile duplicated any note that was received from MIDI input. In another interaction scheme, the robot recorded and played back complete phrases of musical material. A simple chord sequence caused Haile to start listening to the human performer, and a repetition of that chord caused it to play back the recorded melody. Rather than repeating the melody exactly as played, Haile utilized a mechanism that stochastically added notes to the melody, similarly to the density mutation function described above.

The main interaction scheme used with the genetic algorithm was an adaptive call-and-response mechanism. The mean and variance of the inter-onset times in the input was used to calculate an appropriate delay time; then if no input was detected over this period, Haile generated and played a response phrase. In other interaction schemes, developed in an effort to enrich the simple call-and-response interaction, Haile was programmed to introduce musical material from a database of previous genetically modified phrases, interrupt human musicians with responses while they are playing, ignore human input, and imitate melodies to create canons. In the initial phase of the project, a human operator was responsible for some real-time playback decisions such as determining the interaction scheme used by Haile. In addition, the human operator of the system triggered events, choosing among the available playback modes, decided between MIDI and audio input at any given time, and selected the different types of mutation functions for the genetic algorithm. In order to facilitate a more autonomous interaction, an algorithm was then developed that choses between these higher-level playback decisions based on the evolving context of the music, thus allowing Haile to react to musicians in a performance setting without the need for human control. Haile's autonomous module involved switching between four different playback modes: call-and-response (described above), independent playback, canon mode, and solo mode. During independent playback mode, Haile introduced a previously generated melody from the genetic algorithm after waiting a certain period of time. Canon mode employed a similar delay, but here Haile repeated the input from a human musician. If no input was detected for a certain length of time, Haile entered solo mode, where it continued to play genetically generated melodies until a human player interrupted the robotic solo. Independently of its playback mode, Haile decided between inputs (MIDI or audio) and changed the various parameters of the genetic algorithm (mutation and crossover types, number of generations, amount of mutation, etc.) over time. The human performers did not know who Haile was listening to or exactly how Haile will

respond. We feel this represents a workable model of the structure of interactions that can be seen in human-to-human musical improvisation.
Two compositions were written for the system and performed in two separate concerts. In the first piece, titled "Svobod," a piano and a saxophone player freely improvised with a semi-autonomous robot (see video excerpts - http://www.coa.gatech.edu/~gil/Svobod.mov). The second piece, titled "iltur for Haile," involved a more defined and tonal musical structure utilizing genetically driven and non-genetically driven interaction schemes, as the robot performed autonomously with a jazz quartet see video excerpts – http://www.coa.gatech.edu/~gil/iltur4haile.mov).

Figure 14. Human players interact with Haile as it improvises based on input from saxophone and piano in "iltur for Haile"

6. Preliminary Evaluation

In order to evaluate our approaches in design, mechanics, perception, and interaction we conducted a user study where subjects were asked to interact with Haile, to participate in a perceptual experiment, and to fill a questioner regarding their experience. The study addressed only the first phase of the project, and included only rhythmic applications (user studies for the second phase of the project will be conducted in the future). The 14 undergraduate students who participated in the study were enrolled in the percussive ensemble class at Georgia Tech in Spring 2006 and had at least 8 years of experience each in playing percussion instruments. This level of experience was required to support the musical interaction with Haile as well as to support a meaningful discussion about subjects' experience. Each subject spent about 20 minutes experimenting with four different

interaction modes – imitation, stochastic transformation, perceptual accompaniment, and perceptual transformation. Subjects were then asked to compare their notion of rhythmic stability with Haile's algorithmic implementation. As part of the perceptual experiment on stability, subjects were asked to improvise a one-measure rhythmic phrase while Haile provided a 4/4 beat at 90 BPM. Subjects were then randomly presented with three transformations of their phrase: a less stable version, a version with similar stability, and a more stable version. The transformed measures were generated by the Max/MSP stability external (see section 5.1) using stability ratings of 0.1, 0.5, and 0.9 for less, similar, and more stability, respectively. All phrases, including the original, were played twice. Students were then asked to indicate which phrase, in their opinion, was less stable, similar, or more stable in comparison to the original input. Stability was explained as representing the "predictability of" or "ease of tapping one's foot along with" a particular rhythm. The goal of this experiment was not to reach a definite well-controlled conclusion regarding the rhythmic stability model we used, but rather to obtain a preliminary notion about the correlation between our algorithmic implementation and a number of human subjects' perception in an interactive setting. The next section of the user study involved a written survey where subjects were asked to answer questions describing their impression of Haile's physical design, mechanical operation, the different perceptual and interaction modes, as well as a number of general questions about human-robot interaction and "robotic musicianship". The survey included 39 questions such as: "What aspects of the design and mechanical operation make Haile compelling to play with?" "What design aspects are problematic and require improvements"? "What musical aspects were captured by Haile in a satisfactory manner"? "What aspects were not captured well?" "Did Haile's response make Replica with Musical sense?" "Did the responses encourage you to play differently than usual and in what ways"? "Did the interaction with Haile encourage you to come up with new musical ideas?" "Do you think that new musical experiences, and new music, can evolve from musical human-robot interaction"?

Most subjects addressed Haile's physical design in positive terms, using descriptors such as "unique", "artistic", "stylized", "organic", and "functional". Other opinions included "the design offered a feeling of comfort", "the design was pleasing and inviting, and "if Haile was not anthropomorphic it would not have been as encouraging to play with". When asked about caveats in the design several subjects mentioned "too many visible electronics" "exposed cabling" and suggested that future designs should be "less cluttered." Another critique was that "the design did not appear to be versatile for use with other varieties of drums." Regarding Haile's mechanical operation, subjects provided positive comments regarding the steadiness and accuracy of the left hand and the speed and "smoothness" of the right hand. The main mechanical caveats mentioned were Haile's limited timbre and volume control as well as the lack of larger and more visual movements. Only one respondent complained about the mechanical noise Haile produces. In the perceptual rhythmic stability study, half of the respondents (7/14) correctly identified the three transformations (in comparison, a random response would choose 2.3/14 correctly on average). The majority of confusions were between similar and more stable transformations and between similar and less stable transformations. Only 3 responses out of the total 42 decisions confused a more stable version for a less stable version, implying that larger differences in algorithmic stability ratings made differentiation easier. Only one subject labelled all three generated rhythms incorrectly. Subjects' response to the four interaction

modes was varied. In Imitation Mode respondents mentioned Haile's "accuracy," and "good timing and speed" as positive traits and its lack of volume control as a caveat. Responses to the question "How well did Haile imitate your playing?" ranged from "pretty well" to "amazingly well." Some differences between the interaction modes became apparent. For example, in Stochastic Transformation Mode (STM), about 85% of the subjects provided a clear positive response to the question "Was Haile responsive to your playing?" Only about 40% gave such a clear positive response to this question in Perceptual Accompaniment Mode (PAM). Respondents refer to the delay between user input and robotic response in PAM as the main cause for the "less responsive feel." To the question "Did Haile's responses encourage you to play differently than usual?" 50% of the subjects provided a positive response in STM while only 30% gave a positive response to this question in PAM. When asked to describe how different than usual their playing was in STM, subjects focused on two contradicting motivations: Some mentioned that they played simpler rhythms than usual so Haile could transform them easily and in an identifiable manner. Others made an effort to play complex rhythms to challenge and test Haile's abilities. These behaviours were less apparent in PAM. While only 40% (across all interaction modes) provided a positive answer to the question "Did Haile's responses encourage you to come up with new musical ideas"?, more than 90% percent of participants answered positively to the question "Do you think that new musical experiences and new music, can evolve from human-machine musical interactions?", strengthening their answers with terms such as "definitely," "certainly," and "without a question".

Based on the experiment and survey, we feel that our preliminary attempt at Robotic Musicianship provided promising results. The most encouraging survey outcome, in our opinion, was that subjects felt that the human-machine collaboration established with Haile did, on occasions, lead to novel musical experiences and new musical ideas that would not have been conceived by other means. It is clear, though, that further work in mechanics, perception, and interaction design is required to create a robot that can truly demonstrate "musicianship." Nearly all subjects addressed Haile's design in positive terms, strengthening our assumption that the wood and the organic look function well in a drum circle context. Our decision to complement the organic look with exposed electronics was criticized by some subjects, although we feel that this hybrid design conveys the robotic functionalities and reflects the electroacoustic nature of the project. Mechanically, most subjects were impressed with the speed and smoothness of Haile's operation. Only one subject complained about the noise produced by the robot, which suggests that most players were able to either mask the noise out or to accept it as an inherent and acceptable aspect of human-robot interaction. Several subjects, however, indicated that Haile's motion did not provide satisfactory visual cues and could not produce adequate variety of loudness and timbre. The control mechanism for the left arm was developed subsequently to the user study and is currently providing a wider dynamic range. The user study and survey also provided encouraging results in regards to Haile's perceptual and interaction modules. The high percentage of positive responses about the Imitation Mode indicates that our low-level onset and pitch detection algorithms were effective. In general, a large majority of the respondents indicated that Haile was responsive to their playing. Perceptual Accompaniment Mode (PAM), however, was an exception to this rule, as subjects felt Haile was not responding to their actions with acceptable timing. PAM was also unique in the high percentage of subjects who reported that they did not play differently in comparison to

playing with humans. We explain this results by the synchronous accompaniment nature of PAM, which is more familiar to most percussion students. Most subjects, on the other hand, felt compelled to play differently than usual in sequential call-and-response modes such as Stochastic Transformation Mode (STM). Here subjects changed their usual drumming behaviours either by simplifying their rhythms to better follow Haile's responses or by playing complex rhythms in an effort to challenge the robot's perceptual and mechanical abilities. We believe that these behaviours were caused by the novelty effect as players attempted to explore Haile's physical and cognitive boundaries. We assume that subjects may develop more complex interaction behaviours if given longer play times. Given the high level of variance in the notion of rhythmic stability in human perception we feel that our rhythm stability experiment performed better than expected. Some caveats in our method may have also hindered the results. For example, misalignment of subject drumming with the metronome during recording led to misaligned transformations, which may have been unjustifiably perceived as unstable. Also, since the transformed rhythms were generated based on subjects' input, the relative difference between the output stabilities in some cases became minimal and difficult to identify. For example, when a subject's original phrase was extremely stable the algorithm would not be able to produce an identifiably "more stable" phrase. Asking subjects to play a unified mid-stability rhythm as input could have solved this problem, although we were specifically interested in evaluating Haile's perception in a live improvisatory context. As indicated above, the most encouraging results were that 40% of subjects stated that the interaction with Haile encouraged them to come up with new musical ideas and more than 90% claimed that they believe that new musical experiences, and new music, can evolve from such human-machine interaction. This may indicate that although the potential for creating novel musical experiences between humans and robots was not fully realized in our current implementation, the experience led a large majority of the subjects to feel optimistic about the prospect of achieving such novel musical experiences in the future.

7. Future Work

The preliminary user feedback stressed the importance of large visual motions for enabling humans to synchronize with and anticipate the robot's actions. In an effort to address this need, particularly for pitched instruments, we plan to develop a new robotic marimba player that will use several mallets with large and visible striking motions, both horizontally and vertically. Inspired by common human playing techniques, the robot will consist of four arms, each with three degrees of freedom, and a span of one octave (see Figure 15). The robotic arms will be arranged in pairs with overlapping workspaces to allow various combinations of chords to be played. Four arms were chosen because marimba players typically hold four mallets (2 in each hand). However, due to the layout of the bars and necessary grips, human players must rotate their wrists in difficult angles to play certain chords, thus limiting their ability to quickly transition between such chords. Due to its four independent arms, the robot will only limited by the speed at which each mallet can move, although a certain amount of coordination between neighbouring arms will be required to avoid collisions in the shared workspace. The independent operation of each arm will enable the robot to play sophisticated note combinations faster and more accurately than humans. In an effort to extend the perception, improvisation and interaction capabilities of the robot we plan to develop new algorithmic models for melodic tension, attraction, and

similarity as well as new models for sequential and synchronous musical human-robot interaction. We are also working on a number of interactive improvisational algorithms based on fractals and cellular automata and intend to conduct user studies that will lead to workshops and concerts with the new robot.

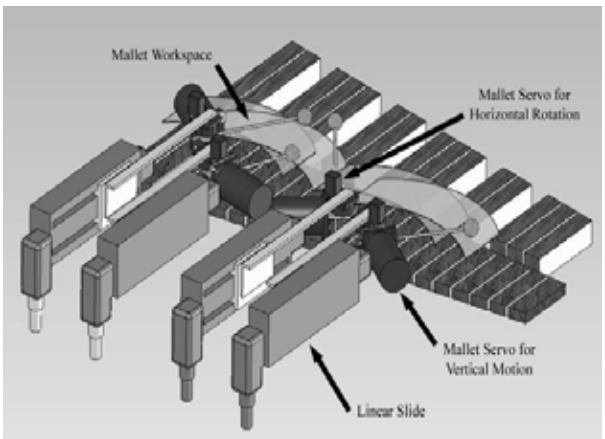


Figure 15. The robotic marimba player will have of four arms, each with three degrees of freedom. A servomotor mounted at the end of each arm will rotate the mallets in a vertical plane to strike the bars. An additional servomotor and a linear slide assembly will act together to position the mallets over an octave range

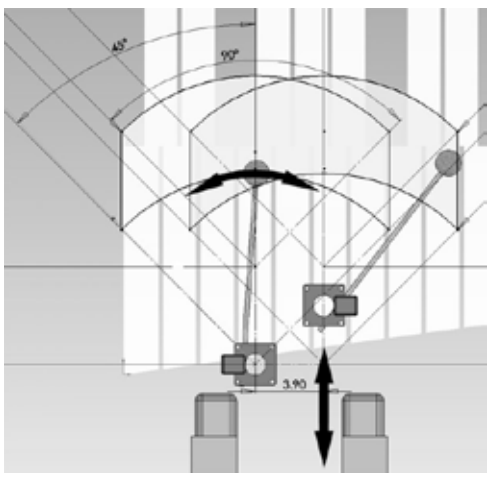


Figure 16. The pivot point of each mallet will be able to move 14 cm towards and away from the marimba to play upper and lower keys, and the mallets will rotate laterally by 90 degrees to reach a one-octave span

8. Acknowledgments

I would like to thank Scott Driscoll, who took a significant role in designing and developing Haile and its rhythmical applications in Phase 1, as well Mark Godfrey, Alex Rae, and John Rhoads, who helped in developing the melodic application in Phase 2. I would also like to acknowledge Georgia Tech's College of Architecture, Music Department and the GVU centre for their support.

9. References

Baginsky, N. A. (Accessed 2007). The Three Sirens: A Self Learning Robotic Rock Band. *http://www.the-three-sirens.info/binfo.html*

Biles, J. A. (1999). Life with GenJam: interacting with a musical IGA. *Processions of IEEE Systems, Man, and Cybernetics Conference*, pp. 652-656, ISBN: 0-7803-5731-0, Tokyo, Japan, October 1999

Brooks, A. G.; Gray J.; Hoffman, G.; Lockerd, A.; Lee, H.; & Breazeal C. (2004). Robot's play: interactive games with sociable machines. *Computer Entertainment*, Vol. 2, No. 3, 10-20

Chida, K.; Okuma, I.; Isoda, S.; Saisu, Y.; Wakamatsu, K.; Nishikawa, K.; Solis, J.; Takanobu, H.; Takanishi, A. (2004). Development of a new Anthropomorphic Flutist Robot WF-4. *Proceedings of IEEE International Conference on Robotics and Automation*. pp. 152-157, ISBN: 0-7803-8232-3, Barcelona, Spain, April 2004

Cope, D. (1996). *Experiments in Music Intelligence,* A-R Editions, ISBN 0-89579-256-7 Madison WI

Coren, S.; Ward, L.M.; & Enns. G. 2003. Sensation and Perception. John Wiley & Sons Inc, ISBN: 0-4712-7255-8, Hoboken, New Jersey

Cycling74 (Accessed 2007). Max/MSP Web site, *http://www.cycling74.com/products/maxms*.

Dannenberg, R. (1984). An On-line Algorithm for Real-Time Accompaniment. P*roceedings of International Computer Music Conference,* pp. 193-198, Paris, France

Davies, M.E.P.; Plumbley, M.D. Beat tracking with a two state model. *Proceedings of the 2005 IEEE International Conference on Acoustics, Speech, and Signal Processing*, pp. 241–244, Philadelphia, Penn., USA, 2005.

Desain, P.; & H. J. Honing (2002). Rhythmic stability as explanation of category size. *Proceedings of the International Conference on Music Perception and Cognition* Sydney, CD-Rom, Australia, July 2007

Dorssen, M. V. (Accessed 2007). The Cell website. http://www.cell.org.au/

Foote, J. & S. Uchihashi (2001). The Beat Spectrum: a new approach to rhythmic analysis. *Proceedings of IEEE International Conference on Multimedia and Expo*, pp. 881-884, August 2001, Tokyo, Japan

Johnson-Laird, P. N. (2002). How Jazz Musicians Improvise. *Music Perception,* Vol. 19, No. 3, Spring 2002 415-442

Jordà, S. (2002). Afasia: The Ultimate Homeric One-man multimedia- band. *Proceedings of the International Conference on New Interfaces for Musical Expression*, pp.1-6, Dublin, Ireland, May 2002

Kapur, A. (2005). A History of Robotic Musical Instruments. *Proceedings of the International Computer Music Conference*, pp. 21-28, Barcelona, Spain, September 2005

Lerdahl, F. and R. Jackendoff (1983). *A Generative Theory of Tonal Music*. MIT Press. ISBN 0-2626-2107-X Cambridge, MA

Lewis, G. (2000). Too Many Notes: Computers, Complexity and Culture in Voyager. *Leonardo Music Journal,* Vol. 10, 33-39

MakingThings (Accessed 2007). Teleo web site, *http://www.makingthings.com/teleo/*

Moroni, A. (2000). Vox Populi: An Interactive Evolutionary System for Algorithmic Music Composition. *Leonardo Music Journal* Vol. 10, 49-54.

Orio, N.; S. Lemouton & Schwarz. D. (2003). Score Following: State of the Art and New Developments. *Proceedings of International Conference on New Interfaces for Musical Expression,* pp. 36-41, Montreal, Canada, May 2003

Pachet, F. (2003). The continuator: Musical interaction with style. *Journal of New Music Research*, Volume 32, Issue 3, 333 - 341

Pressing, J. (1994). *Compositions for Improvisers: An Australian Perspective*, Trobe University Press, ISBN: 1-8632-4415-8, Melbourne, Australia

Puckette, M.; Apel, T. & Zicarelli, D (1998). Real-time Audio Analysis Tools for Pd and MSP. *Proceedings of the International Computer Music Conference*, pp. 109–112, Ann Arbor, Michigan, October 1998

Rae, G. W. (2005). Robotic Instruments web site http://logosfoundation.org/instrum_gwr/automatons.html

Rowe, R. (1992). *Interactive Music Systems: Machine Listening and Composing*. MIT Press, ISBN: 0262181495, Cambridge, MA

Scheirer, E. (1998). Tempo and beat analysis of acoustic musical signals, *Journal of the Acoustical Society of America*, Vol. 103, No. 1, January 1998, 588-601

Singer, E.; Feddersen J.; Redmon, C. & Bowen B. (2004). LEMUR's Musical Robots. *Proceedings of International Conference on New Interfaces for Musical Expression*, pp. 181-184, Hamamatsu, Japan, June 2004

Smith, L., R. McNab, et al. (1998). Sequence-based melodic comparison: A dynamic-programming approach. *Computing in Musicology,* Vol. 11, 101-128

Tanguiane, A. (1993). *Artificial Perception and Music Recognition*. Springer-Verlag, ISBN: 3-5405-7394-1, Berlin, Germany

Tokui, N. & H. Iba (2000). Music Composition with Interactive Evolutionary Computation. *Proceedings of the International Conference on Generative Art,* CD-Rom, Milan, Ital

Toyota (2004). The Trumpet Robot website, http://www.toyota.co.jp/en/special/robot

Trimpin (2000). *SoundSculptures Exhibition*: Five Examples. Munich, Germany, MGM MediaGruppe Munchen.

Vercoe, B. (1984). The Synthetic Performer in the Context of Live Performance, *Proceedings of the International Computer Music Conference*, pp. 199-200, Paris, France.

Weinberg, G. (2005). Interconnected Musical Networks - Toward a Theoretical Framework. *Computer Music Journal,* Vol. 29, No. 2, pp. 23-39

Winkler, T. (2001). *Composing Interactive Music: Techniques and Ideas Using Max*, MIT Press, ISBN: 0262731398, Cambridge, MA

25

Possibilities of force based interaction with robot manipulators

Alexander Winkler and Jozef Suchý
Chemnitz University of Technology
Germany

1. Introduction

One way of interaction between a human and a robot manipulator is the interaction via forces and torques. We will call it also force guidance. For this purpose the human acts on the robot arm or on the robot end-effector. From the interaction forces and torques than a suitable motion of the robot is generated.
This kind of human robot interaction may be useful e.g. for the comfortable teach-in process. Commonly, positions and orientations of the robot tool are taught by the operator using the manual control pendant. With the keys on this device he or she moves the robot in joint or in task space. To improve the usability of the robot, some manual control pendants are additionally equipped with a more intuitive teach in device. It is called 6D mouse or space mouse (Hirzinger & Heindl, 1986). For further optimization of the teach-in process another way to move the robot would be force guidance. It will be shown that it is possible, with some differences, both in joint or in task space.
Force based human robot interaction can be seen as a special kind of active robot force control (Zeng & Hemami, 1997). To perform this, the robot has to be equipped with a force/torque sensor (Gorinevsky et al., 1997). Usually this sensor is mounted in the robot wrist and it measures forces and torques in all Cartesian directions. The cost of such a 6D F/T sensor can exceed 10% of the price of a low payload six axes articulated robot. For that reason it should be searched for an alternative possibility of force/torque measurement. One idea is to estimate the interaction forces and torques from the joint motor currents. For this purpose an algorithm is presented and verified with experiments.
Besides the kinematics of the robot motion during human robot interaction also its dynamics is important. For its representation the so called target or desired impedance behaviour will be defined as the relationship between interaction forces/torques and the velocity components of the robot motion. The simplest desired impedance behaviour is the behaviour of the mass damper system. Moreover, there are some more variants and additional features, e.g. the intuitive collision avoidance which will be described in this article.
Apart from the desired impedance behaviour selected and parameterized by the operator the dynamics of the robot system has been respected. It depends on the access level of motion generation. Commonly, the robot motion is controlled by the trajectory generator. However, some robot systems permit the direct access to the position or velocity control loops which is favourable in all kinds of robot force control.

One special application of force based human robot interaction is robot teleoperation. Sometimes it is necessary to perform it with force feedback. A very simple structure of such a teleoperation system will be proposed. It consists of a slave robot controlled by a force guided master robot.

2. Kinematics in force based human robot interaction

The basic structure of a robot system controlled by an operator via forces and torques is shown in Fig. 1.

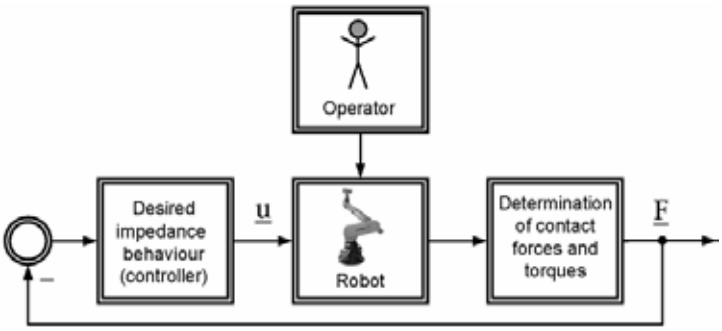


Figure 1. Basic structure of force based human robot interaction

The contact forces and torques $\underline{F}$ caused by the operator are determined e.g. by F/T sensor. From the measured values the desired impedance behaviour computes a suitable motion of the robot represented by the vector of the actuating variables $\underline{u}$. This structure may be also understood as zero force/torque control. Apart from the desired impedance behaviour which determines decisively the dynamics of the robot motion, the kind of space where the motion is performed specifies the robot behaviour during force guidance. Generally, it can be distinguished between robot guidance in task space or in joint space.

2.1 Force guidance in task space

First, it will be assumed that the forces and torques act only on the end-effector of the manipulator and the values are measured by a 6D F/T sensor mounted in the robot wrist. The measured values already free of gravitational forces/torques and transformed into tool coordinate frame are represented by vector $\underline{F}_T$:

$$\underline{F}_T = \begin{bmatrix} F_{T_x} & F_{T_y} & F_{T_z} & M_{T_x} & M_{T_y} & M_{T_z} \end{bmatrix}^T \tag{1}$$

Using the current rotational matrix $\underline{R}$ of the robot

$$\underline{R} = \begin{bmatrix} n_x & s_x & a_x \\ n_y & s_y & a_y \\ n_z & s_z & a_z \end{bmatrix} \tag{2}$$

which describes the orientation of the tool coordinate frame with respect to the base coordinate frame (McKerrow, 1995), the force and torque values from (1) can be transformed into the orientation of the base coordinate frame:

$$\underline{F} = \begin{bmatrix} F_x \\ F_y \\ F_z \\ M_x \\ M_y \\ M_z \end{bmatrix} = \begin{bmatrix} n_x & s_x & a_x & 0 & 0 & 0 \\ n_y & s_y & a_y & 0 & 0 & 0 \\ n_z & s_z & a_z & 0 & 0 & 0 \\ 0 & 0 & 0 & n_x & s_x & a_x \\ 0 & 0 & 0 & n_y & s_y & a_y \\ 0 & 0 & 0 & n_z & s_z & a_z \end{bmatrix} \cdot \begin{bmatrix} F_{T_x} \\ F_{T_y} \\ F_{T_z} \\ M_{T_x} \\ M_{T_y} \\ M_{T_z} \end{bmatrix} \tag{3}$$

This transformation does not take into account the additional torque components generated by F_{Tx}, F_{Ty} and F_{Tz}. It just expresses the already existing vectors of forces and torques in base coordinate frame. Now, the desired impedance behaviour $\underline{I}$ is introduced as the relationship between robot velocities and interaction forces/torques. It is used to compute a suitable motion of the robot end-effector. In general the motion is described by its velocity vector $\underline{v}$ including the linear and angular velocity components:

$$\underline{v} = \begin{bmatrix} \dot{p}_x & \dot{p}_y & \dot{p}_z & \omega_x & \omega_y & \omega_z \end{bmatrix}^T = \underline{I}(\underline{F}, \quad t) \tag{4}$$

Usually the desired impedance behaviour is described by differential equations, the velocity values are time dependent. To command the robot motion, from the current desired velocities in task space the corresponding joint velocities have to be calculated. For this purpose the inverse of the Jacobian matrix $\underline{J}^{-1}$ may be used. Another possibility is to calculate the desired location of the end-effector in Cartesian coordinates via integration of the velocity values. If the inverse kinematics of the robot is known from the Cartesian location the desired joint coordinates can be computed and sent to the joint position control loops.
The properties of force guidance in task space are analysed and compared with the joint space approach in section 2.3.

2.2 Force guidance in joint space

Another possibility to guide the manipulator with forces and torques acting on its end-effector is the joint space approach to force guidance (Winkler & Suchý, 2006b). For this purpose it is necessary to compute from the interaction forces/torques free of gravity influences and orientate in base coordinate frame (see Eq. 3) the equivalent joint torques and/or forces represented by vector $\underline{\tau}$, where m is the number of joints of the robot. To achieve this, the transposed Jacobian matrix $\underline{J}^T$ is used (Sciavicco & Siciliano, 2004):

$$\underline{\tau} = \begin{bmatrix} \tau_1 & \tau_2 & \dots & \dots & \dots & \tau_m \end{bmatrix}^T = \underline{J}^T \cdot \underline{F} \tag{5}$$

Eq. (5) is only valid if $\underline{J}$ is the geometric Jacobian matrix. In the case that $\underline{J}$ is calculated by way of differentiation of the end-effector location $\underline{P}$ with respect to the joint positions $\underline{q}$, the torque values in vector $\underline{F}$ have been brought into the orientation of the orientation representation chosen in $\underline{P}$. Here is an example: The location of the end-effector is given by the position coordinates p_x, p_y, p_z and its orientation is represented by the zy'z'' Euler angles ϕ, θ, ψ:

$$\underline{P} = \begin{bmatrix} p_x & p_y & p_z & \phi & \theta & \psi \end{bmatrix}^T \tag{6}$$

The vector $\underline{F}$ has to be transformed into vector $\underline{F}_{zy'z''}$, (Sciavicco & Siciliano, 2004) by:

$$\underline{F}_{zy'z''} = \begin{bmatrix} F_x \\ F_y \\ F_z \\ M_z \\ M_{y'} \\ M_{z''} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & -\sin(\phi) & \cos(\phi) & 0 \\ 0 & 0 & 0 & \cos(\phi)\sin(\theta) & \sin(\phi)\sin(\theta) & \cos(\theta) \end{bmatrix} \cdot \underline{F} \tag{7}$$

The force components F_x, F_y, F_z keep unchanged. Of course the torque M_z is directed around the z axis of the original coordinate frame. $M_{y'}$ is directed around the new y′ axis arising by rotation around z. Eventual $M_{z''}$ is directed around the new z′′ axis arising by rotation around z and y′. Now it is possible to compute the joint torques and/or forces by an analytical Jacobian matrix J_A:

$$\underline{\tau} = \underline{J}_A^T \cdot \underline{F}_{zy'z''} = \left[\frac{d\underline{P}}{d\underline{q}} \right]^T \cdot \underline{F}_{zy'z''} \tag{8}$$

In this case the desired impedance behaviour is given in joint space. It defines the relationship between joint torques and/or forces caused by the operator and the desired joint velocities:

$$\dot{\underline{q}} = \begin{bmatrix} \dot{q}_1 & \dot{q}_2 & \dots & \dots & \dots & \dot{q}_m \end{bmatrix}^T = \underline{I}(\underline{\tau}, \ t) \tag{9}$$

If the robot controller provides the access to the joint velocity control loops the desired impedance behaviour may be connected directly to their inputs. Otherwise, the velocities have to be integrated to get the desired joint positions which can be sent to the joint position control loops or to the trajectory generation module.

2.3 Comparison of both approaches to force guidance

First, for comparison of the task space approach to force guidance with the joint space approach a manipulator with very simple kinematics was chosen. It will be called Planar Two Link Manipulator here. It consists of only two links of lengths L_1 and L_2 equipped with rotational joints. The joint angles are denoted as q_1 and q_2. As the manipulator has only two degrees of freedom vector $\underline{P}$ from Eq. (6) can be reduced and written as follows (Stadler, 1995):

$$\underline{P} = \begin{bmatrix} p_x & p_y \end{bmatrix}^T = \begin{bmatrix} L_1 \cos(q_1) + L_2 \cos(q_1 + q_2) & L_1 \sin(q_1) + L_2 \sin(q_1 + q_2) \end{bmatrix}^T \tag{10}$$

The end-effector orientation has not been taken into consideration because it can not be given independently of the position coordinates. For the experiment it will be assumed an external force acts on the end-effector described by vector $\underline{F}=[F_x \ F_y]^T=[-1 \ 1]^T$. The desired impedance behaviour of the task space approach is given by Eq. (11) as the linear relationship between velocity and force components:

$$\begin{bmatrix} \dot{p}_x \\ \dot{p}_y \end{bmatrix} = \begin{bmatrix} c_1 & 0 \\ 0 & c_2 \end{bmatrix} \cdot \underline{F} \tag{11}$$

It will be assumed there is no time lag in the transfer behaviour because the dynamics of the system is not important when analysing the kinematics of the robot motion. In the same case for the desired impedance behaviour in joint space it follows:

$$\begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \end{bmatrix} = \begin{bmatrix} c_3 & 0 \\ 0 & c_4 \end{bmatrix} \cdot \left(\underline{J}^T \cdot \underline{F} \right) \tag{12}$$

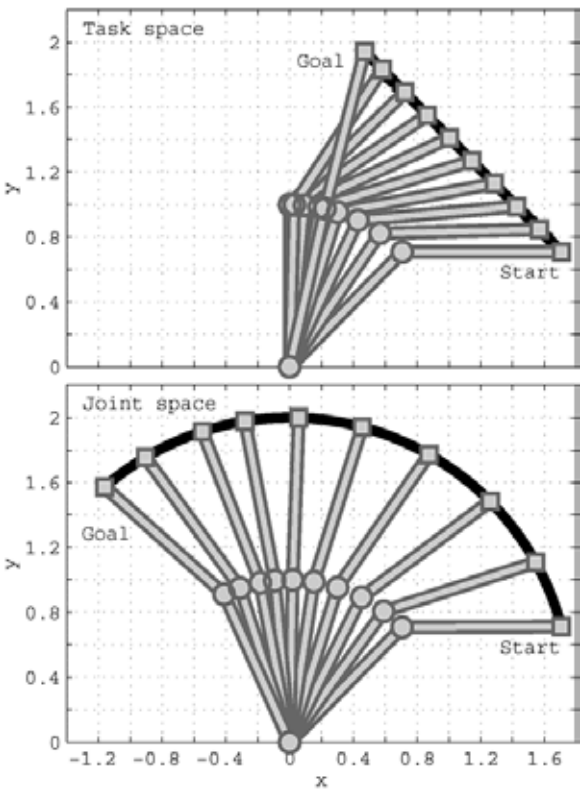


Figure 2. Paths of the force guided Planar Two Link Manipulator

Fig. 2 shows the two paths of the Planar Two Link Manipulator end-effector as the reaction to the external force $\underline{F}$. In the first case (force guidance in task space) robot stops at the singular position where joint angle q_2 becomes equal to 0. From this position it can never be moved away using this algorithm. The inverse kinematics is ambiguous and the inverse Jacobian matrix is not defined. The path resulting from the joint space approach is different.

The kinematical structure of the robot is taken into consideration by this algorithm. This feature may be an advantage in some applications because the operator associates the expected behaviour with the robot kinematics.
It can be also seen that the singularity is crossed. The singularities may also be understood as the boundaries between different configurations of the robot. The configuration defines how the manipulator reaches a location given in Cartesian coordinates. In the case of the Planar Two Link Manipulator there are two possibilities to reach the position described by vector $\underline{P}$. They are distinguished by the sign of the joint angle q_2. One preferred kinematical structure of industrial robots is the kinematics of the six axes articulated robots. It has the property to be able to reach a Cartesian location with 8 variants selectable by 3 configuration parameters (Craig, 2005). During force based human robot interaction in task space the posture of the manipulator is restricted to unique configuration. The change of robot arm configuration is only possible while performing the joint space approach to force guidance. The teach-in application, e.g., may require a particular posture due to restriction caused by the environment.

	Task Space	Joint Space
Dependence of robot behaviour on kinematics	No	Yes
Admissibility of singularities	No	Yes
Workspace	Restricted	Full
Separation of position and orientation changes of the end-effector	Possible	Not possible
Applicability to redundant manipulators	Restricted	Yes
Applicability to parallel manipulators	Yes	Not expedient

Table 1. Properties of both approaches to force guidance

Besides the six axes articulated robots there are some other popular robot kinematics. SCARA robots have one singular position similar to the Planar Two Link Manipulator. Guiding a SCARA robot in joint space it is easy possible to cross its singularity. One special case is the Cartesian robot. Its behaviour during force guidance in task space is identically equal to the joint space approach. Thinking of redundant manipulators the problem of configuration management becomes extremely important. Generally speaking infinite many solutions of the inverse kinematics problem may exist. In this case Cartesian motions are only possible with some constraints laid on certain joints. With force guidance in joint space this task would be easier realizable.
Different from serial manipulators are parallel robots with respect to kinematics. It is relatively easy to compute their inverse kinematics. In general the direct kinematics can be solved only by successive approximation. So it is proposed to use the Cartesian interaction forces and torques to guide the parallel robot directly in task space.
As a conclusion of this section some properties of both algorithms to force guidance are compared in Table 1.

3. Dynamics in force based human robot interaction

3.1 Desired impedance behaviour

In the previous section the kinematics of the robot motion during force based human robot interaction was detailed discussed. For comparison of the task space with the joint space approach the desired impedance behaviour, introduced as the relationship between interaction forces/torques and desired velocities, was simplified to direct proportional dependency, see (11) or (12). In this case all acceleration and deceleration processes have to be performed infinitely fast which is not feasible for the real robot manipulator. Applied to a mechanical system it would mean that the mass of the system is zero. Hence, the desired impedance behaviour has to be extended. One suitable choice is the behaviour of a simple mass damper system. It is described by two parameters mass m and damping d. Applied to joint space the particular desired joint velocity can be calculated from the interaction joint torque or force τ by:

$$\tau = m \cdot \ddot{q}_D + d \cdot \dot{q}_D \tag{13}$$

The desired joint velocity can be integrated to get the desired joint position q_d which may be connected to the joint position control loop.

It is possible to extend the desired impedance behaviour with some additional features. One very important feature is the dead zone nonlinearity within the interaction force branch to prevent unintentional drifts of the manipulator arm. The undesirable robot motions during force guidance, e.g., are the result of force/torque measurement inaccuracies or imprecise gravity compensation of the tool. For several applications, except the force based teach-in process, it may be convenient to insert the spring into the desired impedance behaviour. This will result in spring mass damper behaviour which will bring the robot end-effector back to starting position. It is described by Eq. (14) where k represents the spring constant:

$$\tau = m \cdot \ddot{q}_D + d \cdot \dot{q}_D + k \cdot q_D \tag{14}$$

Very important for the operator security is to bound the desired joint velocity of the desired impedance behaviour. However, in the robot controller also the current joint velocities and the current Cartesian end-effector velocities have to be supervised and limited according to valid standards (Deutsches Institut für Normung, 1993). For the direct human robot interaction in the automatic mode without deadman button it may be necessary to use a two-channel safety controller to supervise the robot controller (Som, 2004). Besides velocity limitation of the joint its position should also be bounded. This can be realized by a limiting nonlinearity in the desired velocity branch together with an anti-wind-up feature. Apart from this hard limit stop a comfortable alternative is the soft position limit stop. For this purpose the damping d of the desired impedance behaviour may be increased in the neighbourhood of the joint position boundary or an additionally spring behaviour can be activated. The just introduced feature allows the operator to feel the joint boundaries intuitive during force based human robot interaction.

3.2 Intuitive collision avoidance

Besides the implementation of the virtual joint boundaries already mentioned while thinking of the desired impedance behaviour, a lot of features are imaginable to make the force based human robot interaction more comfortable. One of these features is the intuitive collision avoidance. In contrast to the joint boundaries it is realized in task space. The aim

which should be achieved is that the operator guiding the robot will be detained to bring the end-effector in contact with an obstacle located within the robot workspace. Such functionality may be important in particular when obstacles are present after the teach-in process is finished. It is possible to use, e.g., the CAD data of the robot work cell to generate the intuitive collision avoidance.

There are a lot of possibilities to implement virtual restrictions of the robot workspace during force guidance. One very interesting approach are force potential fields, (Choset et al., 2005). Virtual forces act on the robot end-effector against the operator near obstacles, so that the operator feels these obstacles. Nevertheless, if the end-effector is located in a close distance to the obstacle, which could be somewhat dangerous, as a result of the virtual force the robot will be moved automatically away from it.

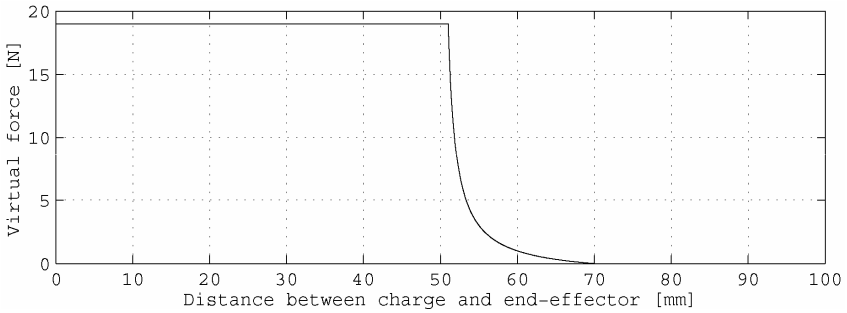


Figure 3. Proposal of the virtual force in the neighbourhood of one charge

A very convenient approach to generate the force potential field is the idea of using virtual force charges. These charges may be seen as electric charges which generate an electrostatic field in their neighbourhood. Between two charges the electrostatic force acts. Its value is reciprocally proportional to the square of the charges distance. However, for the realization of a virtual force potential field this kind of dependency is not obligatory. One possible proposal of the force potential function F_v of single virtual charge acting on the robot end-effector is shown in Fig. 3.

From the virtual force value F_V the force vector $\underline{F}_V$ orientated from charge position $\underline{e}=[e_x\ e_y\ e_z]^T$ to the end-effector position $\underline{p}=[p_x\ p_y\ p_z]^T$ can be computed as follows:

$$\underline{F}_V = F_V\left(\left\|\underline{p}-\underline{e}\right\|\right)\cdot\frac{\underline{p}-\underline{e}}{\left\|\underline{p}-\underline{e}\right\|} \tag{15}$$

For the generation of a virtual force field surrounding an obstacle several charges are necessary. It is favourable to place these on the object surface. The number of charges be denoted with n. Now for the calculation of the virtual force acting on the robot end-effector the principle of superposition can be used:

$$\underline{F}_V = \sum_{i=1}^{n}\left(F_V\left(\left\|\underline{p}-\underline{e}_i\right\|\right)\cdot\frac{\underline{p}-\underline{e}_i}{\left\|\underline{p}-\underline{e}_i\right\|}\right) \tag{16}$$

Fig. 4 shows an example of realization the intuitive collision avoidance system for its validation. Robot STÄUBLI RX90B is located close to an obstacle. On its surface certain number of charges has been placed, some of them are visible in the figure. In the robot controller the algorithms of force guidance in task and in joint space are implemented. For this goal, in the robot wrist a 6D F/T sensor is mounted. Additional to the operator the virtual force $\underline{F}_V$ caused by the charges acts on the end-effector and tries to move the robot away from the obstacle. $\underline{F}_V$ is computed within every interpolation cycle taking the current end-effector position $\underline{p}$ and the positions of the charges $\underline{e}_i$ into consideration.
After starting the program in the robot controller the robot is guided by the operator around the object via force/torque interaction. The virtual force field emitted by the obstacle is being felt by the human operator. The smaller the distance between end-effector and object the higher the force acting against the operator. As a result of the experiment, the end-effector path during force based human robot interaction and the corresponding virtual force vectors are shown in Fig. 5. It was possible to move the robot intuitively around the obstacle without danger of any collision.
This approach which is based on the virtual force field generated by fictive charges may be understood as an active collision avoidance system. Robot is accelerated by the force field in direction away from the obstacle. Another possibility is the passive approach. For this purpose near the object the damping parameters of the desired impedance behaviour have to be increased, e.g. as follows:

$$d = d_{const} + d_V(\underline{p}) \tag{17}$$

Consequently, the operator has to increase its force applied to the end-effector to generate an adequate motion of the robot and the force guidance is hindered. Also the combination of the active and the passive approach seems to be possible for intuitive collision avoidance.

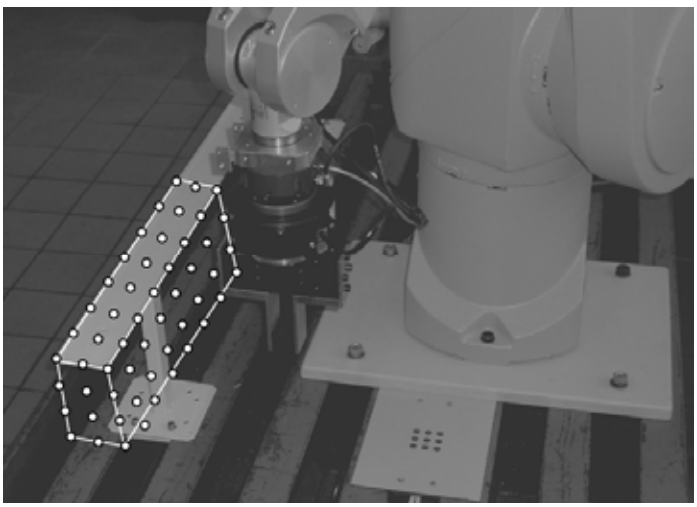


Figure 4. Experimental setup for the validation of the intuitive collision avoidance

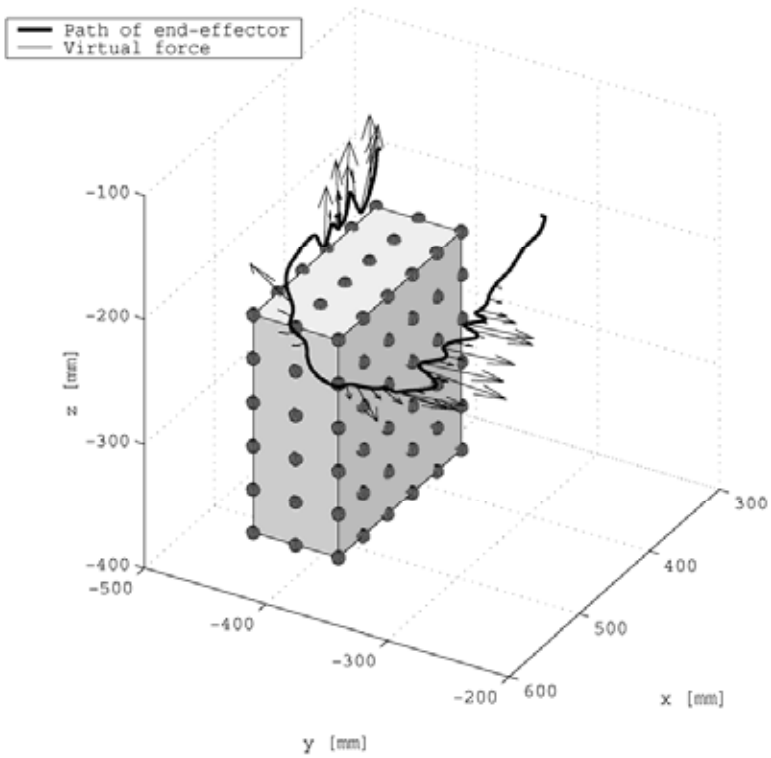


Figure 5. Virtual forces acting on the end-effector during its motion around the obstacle

3.3 Dynamics of the controlled robot manipulator

The task of the manipulator and its controller is tracking the path given by the desired impedance behaviour. Therefore, it is necessary to take also the dynamics of the robot together with its controller into consideration. Generally the dynamics of the robot arm can be described by the following equation (Angeles, 2003):

$$\underline{\tau}_M = \underline{M}(\underline{q})\ddot{\underline{q}} + \underline{C}(\underline{q},\dot{\underline{q}}) + \underline{G}(\underline{q}) + \underline{V}(\underline{q},\dot{\underline{q}}) \tag{18}$$

Matrix $\underline{M}$ is the inertia matrix, vectors $\underline{C}$, $\underline{G}$ and $\underline{V}$ represent the Coriolis/centrifugal, gravitational and frictional forces and/or torques, respectively. The elements in vector $\underline{\tau}_M$ are the torques/forces provided by the drives to the particular joints.

Most commonly used robots are equipped with AC servo motors connected via gears to the links. The motors are controlled by joint power amplifiers including usually power converters, current and velocity controllers. The desired joint velocities are sent to the joint power amplifiers, e.g., by analogue voltage signals or by digital bus interface. The closed

loop controllers of the joint power amplifiers are adjusted by the robot manufacturer. In some cases it is possible to modify the controller parameters or to even replace the complete power amplifier. This would be justifiable in research laboratories but not in industrial applications. Thus the dynamics of the manipulator is significantly affected by the joint power amplifiers. Because it is not possible to manipulate the motor currents on an industrial robot system Eq. (18) need not been regarded any longer here.

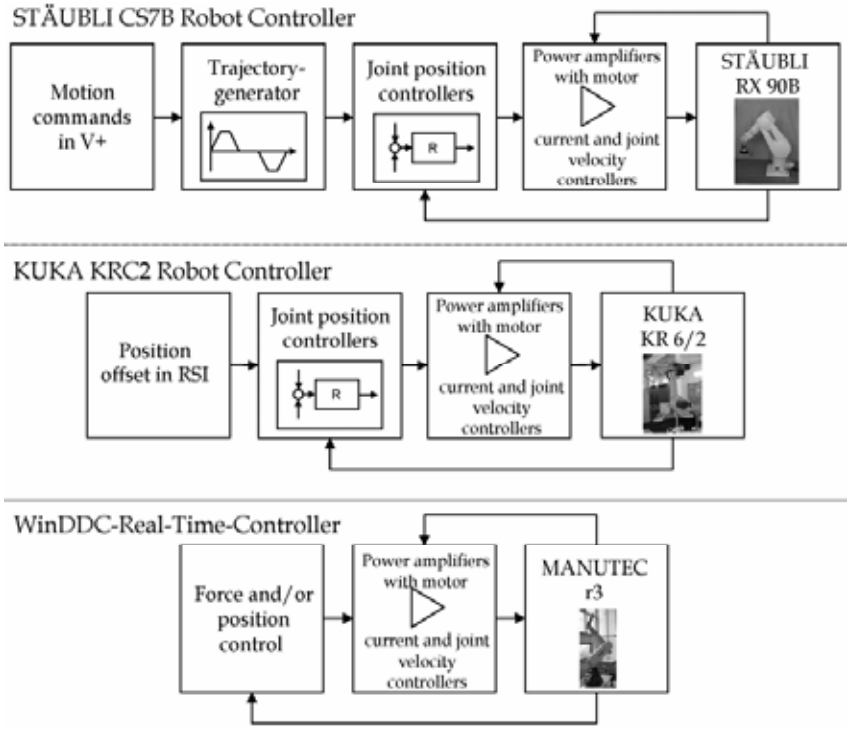


Figure 6. Robot systems for dynamic comparison

Generally, the user of an industrial robot system can program the motion of the end-effector with commands for joint, linear and circular interpolation. The path of the end-effector and the time series of joint positions and velocities are computed by the so called trajectory generator every interpolation cycle. It is a part of the robot controller. The trajectory generator is assigned to the joint position control loops which are part of the robot controller, too. Robot motion generated in this way is a motion without jerk. That means the time courses of desired joint positions, velocities and accelerations are continuously differentiable. In spite of setting the acceleration and speed parameters to 100% the trajectory generator results in a large time delay between desired and current positions.

Some industrial robot controllers provide the possibility of direct access to the position control loops. Thereby a desired position offset may be added to the current commanded

variable. It can be performed in joint or in task space. This kind of motion control is suitable in particular for sensor guided motion of an industrial robot, e.g. for robot force control.
For the realization of improved control algorithms it could be necessary to command the desired joint velocities to the joint power amplifiers. For this purpose the original robot controller may be replaced by a self developed robot controller. It will be increase the freedom in robot programming, e.g. the closed loop joint position controllers can be designed individually.

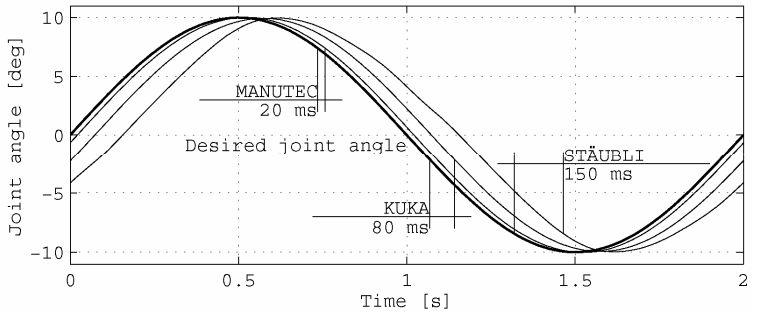


Figure 7. Dynamical behaviour of robots with different possibilities of motion generation

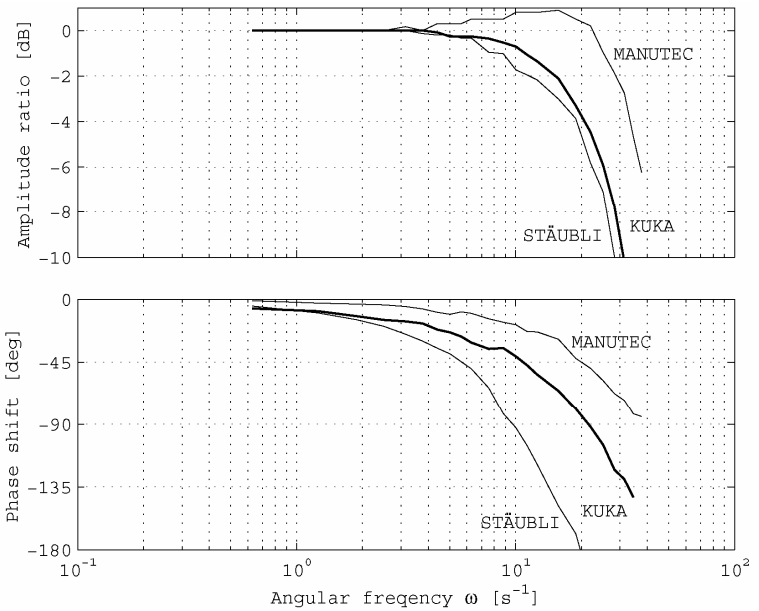


Figure 8. Frequency responses of joint No. 1

Thus, three variants to influence the robot motion have been presented. They are different with respect to their implementation level. It goes from the trajectory generator over the joint position controller to the velocity controller. The possibility to command the desired motor current was not taken into consideration. The effects will be demonstrated using three robot systems. Every of them consist of one low payload robot equipped with different robot controller. The detailed configuration of these different robot systems are shown in Fig. 6. Realization of sensor guided motions with the STÄUBLI robot controller is only possible on the level of the trajectory generator without additional software options. KUKA enables the access to the joint position controllers. For this purpose it is necessary to install the so called Robot Sensor Interface (RSI). Using RSI different controller structures can be programmed. They consist of RSI objects for signal processing. Afterward this controller structure is executed in real time in the interpolation cycle. The WinDDC-Real-Time-Controller is an universal controller with analogue/digital inputs and outputs based on a digital signal processor (DSP). It can be programmed by a special programming language called WinDDC. The analogue output signals to command the motions of the MANUTEC r3 robot represent the desired joint velocities for the joint power amplifiers. The WinDDC-Real-Time-Controller has also digital inputs for incremental position encoders measuring the current joint angles.
First, one very simple experiment will show the differences in robot dynamics caused mainly by robot controllers. The desired joint angle of joint No. 1 is commanded by the sinusoidal signal with the amplitude of 10deg and the period time of 2s. The current joint angle values were recorded for every of the three robot systems. The lag between desired and current joint angle time curves is shown in Fig. 7 and may be analyzed.
It can be seen that the maximum phase shift occurs with the STÄUBLI robot system where the desired position was sent to the trajectory generator via motion commands for joint interpolation. To get this sinusoidal series the motion was divided into a lot of small parts and the mode of the continuous motion was used. Having access to the desired values of the joint position control loops may reduce the phase shift. An example is the KUKA robot controller programmed by RSI. The minimum phase shift can be reached if it is possible to design the control loops individually which can be seen e.g. with the WinDDC-Real-Time-Controller connected to the robot MANUTEC r3. However, for this purpose only a simple proportional controller with additional feed forward control of the desired joint velocity was implemented. To get more information about the dynamics of the different robot systems the frequency responses were recorded. Their amplitude and phase responses are shown in Fig. 8. It can be seen that the system dynamics depends decisively on the level of motion generation. If the level is low the bandwidth will be high. It has to be taken into consideration while specifying the parameters of the desired impedance behaviour. Its bandwidth should be lower then the bandwidth of the robot to guarantee following the desired impedance behaviour.

4. Sensorless force based human robot interaction

The algorithms and features of force based human robot interaction presented in the previous section require that the robot is equipped with a 6D F/T sensor. The cost of such a sensor is of the order of some thousand Euros at present. Therefore, it would be favourable to find a way to do without F/T sensor. One possibility is to estimate the interaction forces and torques from the motor currents.

4.1 Estimation of interaction forces and torques

Most commonly used robots are driven by AC servo motors. Fig. 9 shows the signal flow diagram of single joint which is position controlled in cascade control mode.

The position controller is a proportional controller whereas velocity and motor current are controlled by proportional plus integral controllers. The electrical time constant of the motor is very small in comparison to the mechanical response time. It will be assumed that there are no losses due to magnetisation and internal friction. Hence, there is a linear dependency between motor current i and joint driving torque τ_M determined by motor constant k_M and gear ratio k_G ($k_{MG}=k_M k_G$). Besides the driving torque acting in the joint, robot is also influenced by Coriolis and centrifugal torque C, gravitational torque G, frictional torque V and the interaction torque τ caused by the operator or the environment. The joint acceleration torque τ_A is given according to (19):

$$\tau_A = k_{MG} \cdot i - C - G - V - \tau \tag{19}$$

Assuming the response time of the motor current control loop is very small and its static control error goes to zero due to the proportional plus integral controller structure, for the calculation of the interaction torque τ the desired value of the motor current i_D can be used instead of its current value i. The joint torque τ_A accelerates the particular robot link taking its inertia M into consideration. For operator security the velocity and acceleration values of all joints have to be small. As a result the joint torque and the Coriolis/centrifugal torque may be neglected. The interaction torque can then be approximately calculated as follows:

$$\tau = k_{MG} \cdot i_D - G - V \tag{20}$$

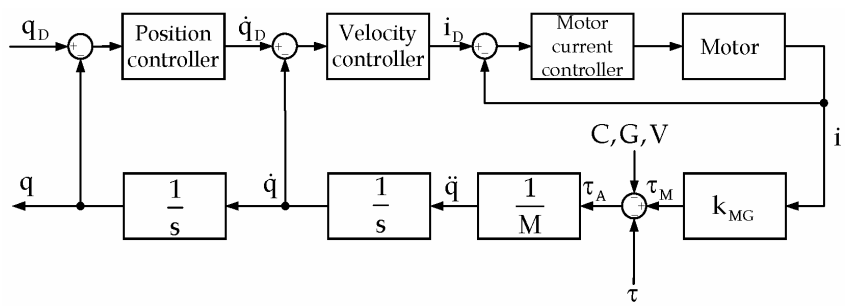


Figure 9. Signal flow diagram of one joint

To perform this calculation it is necessary to learn the vector of the gravitational torques of the robot. It is a function of the joint angle vector $\underline{q}$ and can be calculated from the dynamic model of the robot. The dynamic models of some manipulator arms are published, see e.g. (Türk & Otter, 1987) for the model of robot MANUTEC r3. The calculation of the frictional torques is difficult. Friction is speed dependent consisting of Coulomb friction, stiction and speed proportional friction. Furthermore, the particular friction components are dependent

e.g. on position, temperature and aging of the robot. Especially at low velocity motions where Coulomb friction and stiction are dominant it is very difficult to calculate feasible values and at robot standstill it is nearly impossible because of the PI-structure of the velocity controller the portion of Coulomb friction and stiction can not be separated from the motor current. An example of robot joint friction is shown in Fig. 10. Here the robot MANUTEC r3 was used. It was equipped with the WinDDC-Real-Time-Controller mentioned earlier in section 3 (Winkler & Suchý, 2005). The desired values of motor currents are provided by the joint power amplifiers via analogue voltages which are connected to the analogue inputs of the robot controller for signal processing according to Eq. (20).

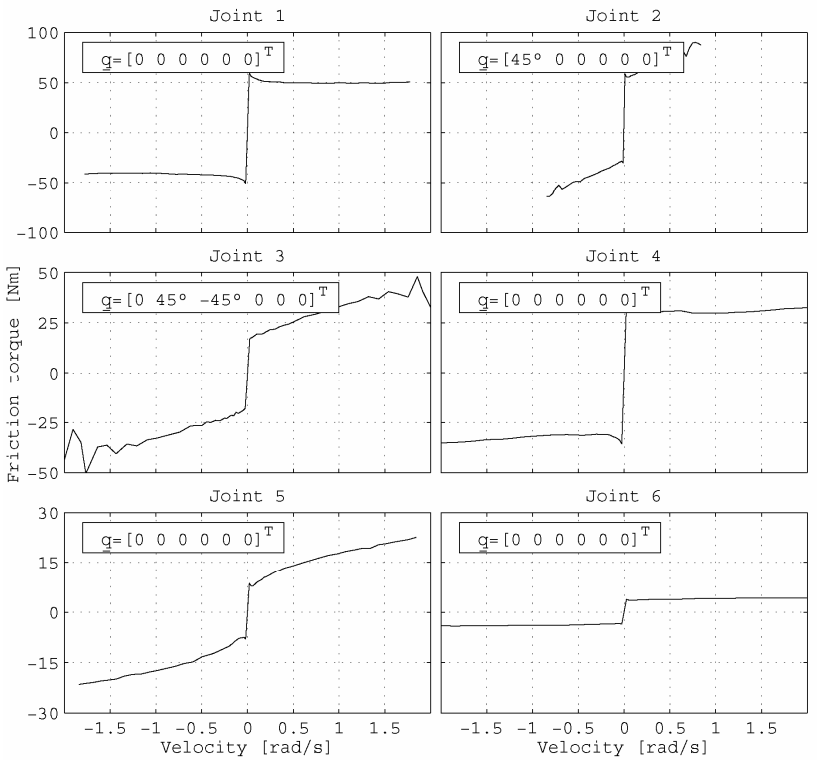


Figure 10. Joint friction torques determined for a robot of type MANUTEC r3

Fig. 10 shows clearly that during low velocity motion generated future by force based human robot interaction stiction and Coulomb friction are prevailing. Therefore, the proposed model for friction estimation is simplified to the form in Fig. 11. It consists of the dead zone nonlinearity in the neighbourhood of robot standstill described by velocity parameter q'_A to prevent oscillations. In the range between q'_A and q'_B the friction increases to its maximum value V_c. When computing friction torques for the estimation of the

interaction torques it is absolutely necessary to avoid the case that the value of any estimated friction torque is higher than its real value. Otherwise it may be happen that the force guided robot becomes unstable during human robot interaction which could be a very dangerous situation to the operator.
Verification of the proposed algorithm to estimate the interaction torques of an industrial robot from its motor currents was performed on the MANUTEC r3 robot equipped with the open WinDDC-Real-Time-Controller. For comparison of the estimated values with the real interaction forces/torques 6D-F/T sensor was additional mounted into robot wrist. For that reason the operator should touch the robot only on its end-effector behind the sensor. Pushing and polling the robot on its links would result in faulty measurements. However, it is also possible to guide the robot by means of motor currents when the operator acts on arbitrary location of the manipulator arm. The comparison is performed in joint space. From the interaction forces and torques measured by F/T sensor the corresponding joint torques can be computed using the transposed Jacobian matrix, see e.g. (5). Different postures of the robot arm were chosen to validate the estimation of the gravitational torques from the dynamic model. They are described by the joint angle vectors $\underline{q}_A=[0\ 0\ 90°\ 0\ 0\ 0]^T$, $\underline{q}_B=[-30°\ 45°\ 30°\ 0\ 0\ 0]^T$ and $\underline{q}_C=[0\ 0\ 90°\ 0\ -90°\ 0]^T$. The operator acted in a random way with forces and torques on the robot end-effector for a certain time period and the time series of actual and estimated interaction joint torques were recorded. They are compared in Fig. 12.

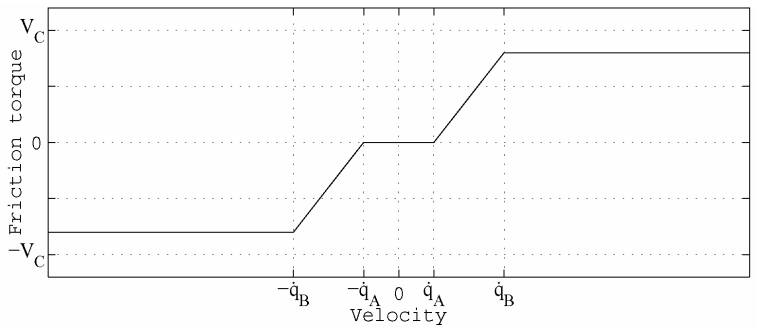


Figure 11. Proposal of friction torque model at low joint velocities

Because the signals which represent the desired motor currents sent from the velocity control loops to the motor current control loops are noisy, they have to be low pass filtered before signal processing. For this purpose a first order system was used which results in a time lag between the plot of the estimated time series and the measured time series. Furthermore, some more differences can be seen. They may result from inaccuracies of the dynamic model of the robot used for the calculation of the gravitational torques. Another source of error comes from the frictional torques. It was already mentioned that during robot standstill the proportional plus integral joint velocity controller outputs a control signal within the range of stiction. This motor current therefore is not able to generate a motion and may be misunderstood as interaction torque. This effect is shown in Fig. 12, e.g. in joint No. 1 when the robot is located at posture A.

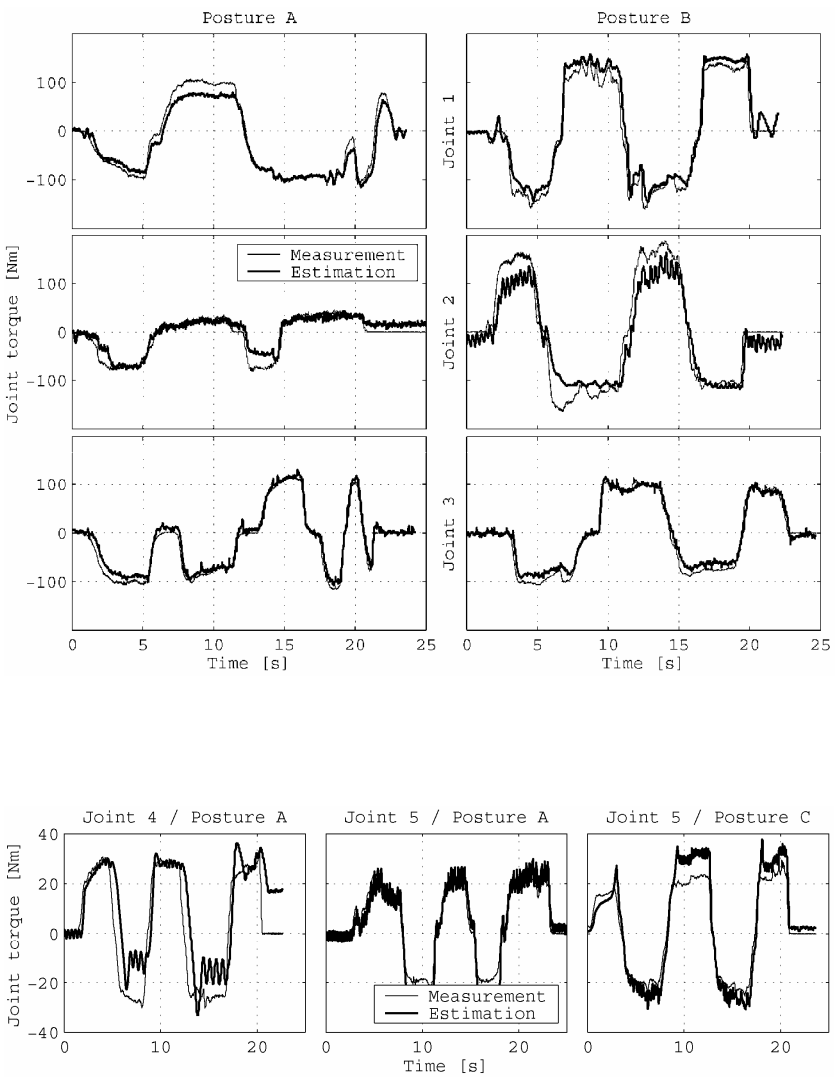


Figure 12. Comparison between interaction torques estimated from motor currents and values measured by F/T sensor at different postures of the manipulator arm

In spite of some differences the time series of measured and estimated interaction joint torques match quite well. It may thus be possible to use the desired values of the motor currents for robot force control or force based human robot interaction.

4.2 Human robot interaction based on motor currents

After estimating the interaction torques from the motor currents that values can be used for force guidance without F/T sensor. Because the forces and torques caused be the operator and estimated by the algorithm developed in the previous section are present in joint space, it will be convenient to perform the force guidance in joint space, too. Otherwise the interaction joint torques ought to be transformed into coressponding Cartesian forces/torques using the inverse of the Jacobian matrix which will result in problems in and near singularities.

$$\underline{F} = \underline{J}^{-T} \cdot \underline{\tau} \tag{21}$$

Another point why the Cartesian space approach may be unfavourable is the following: Unlike to the robot equipped with F/T wrist sensor the operator is able to move the motor current guided robot by pushing or pulling the manipulator arm at arbitrary location. Transforming such interaction forces into Cartesian forces/torques of robot tool frame the resulting robot motion may become anomalous. It is therefore proposed to perform human robot interaction based on motor currents in joint space.

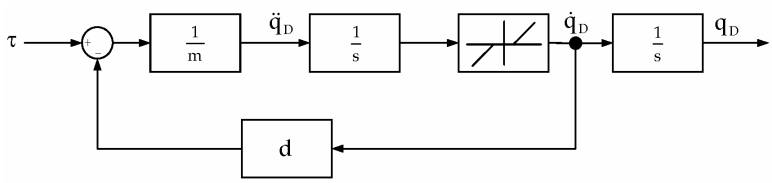


Figure 13. Mass damper desired behaviour with additionally dead zone nonlinearity

Because the estimated interaction torques differ somewhat from the measured values which are more accurate it is extremely important to implement the dead zone nonlinearity in the desired impedance behaviour. Besides its integration into the branch of the estimated joint torque very convenient place is also the branch of the desired joint velocity. Than the desired impedance behaviour based on the suggested mass damper system results in the signal flow diagram shown in Fig. 13. The dead zone nonlinearity prevents undesired motions of the robot caused by inaccurate estimation or measurement of the interaction forces and/or torques. A disadvantage of the implemented nonlinearity is that it distorts the specified parameters of the desired impedance behaviour, e.g. the value of damping will be increased.

For validation of force guidance without F/T sensor the pertaining algorithm was implemented in the WinDDC-Real-Time-Controller. Robot MANUTEC r3 was located at starting position given by $\underline{q}=[0\ 0\ 90°\ 0\ 0\ 0]^T$. Force guidance was activated for all joints. The operator acts on the manipulator arm to guide it throughout the work space. Fig. 14 shows

the estimated interaction joint torques caused by the operator. From these values the desired joint angles are calculated using the desired behaviour proposed in Fig. 13. The desired joint angles are connected to the joint position control loops which consist of simple proportional controllers with additional feed forward control of the joint velocities. The plots of the joint angle values as a result of robot force guidance based on motor currents are shown in Fig. 15. The diagram is subdivided into parts numbered by lower case letters. These letters can be found again in Fig. 16 where some snapshots of robot posture during force guidance are presented. It can be seen that it is possible to guide the robot throughout the workspace by the operator without F/T sensor. During the experiment also some singularities were crossed without any problem.

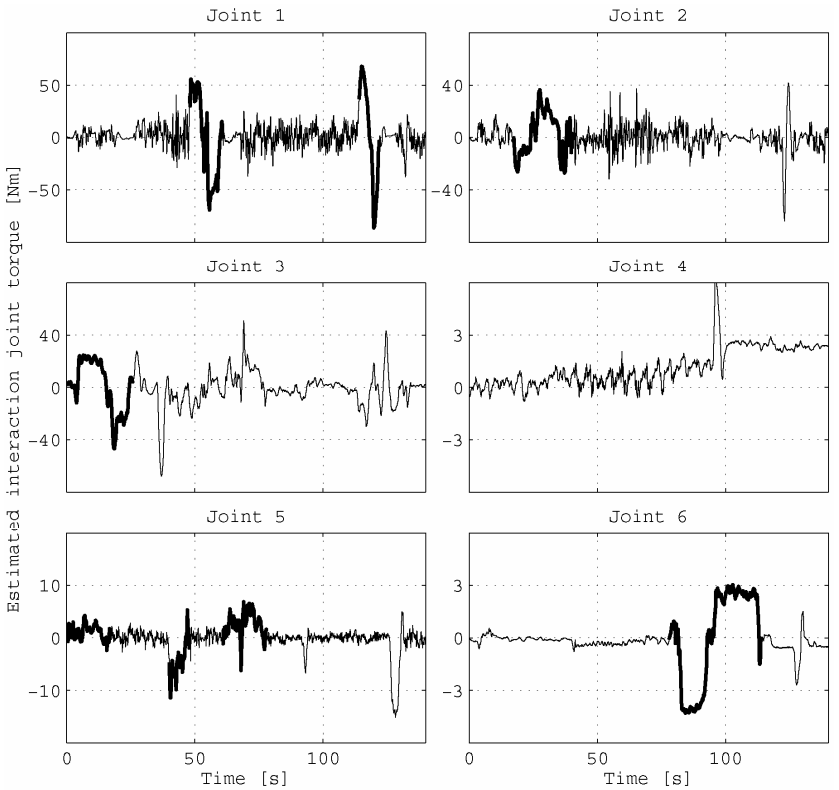


Figure 14. Estimated interaction joint torques during force guidance

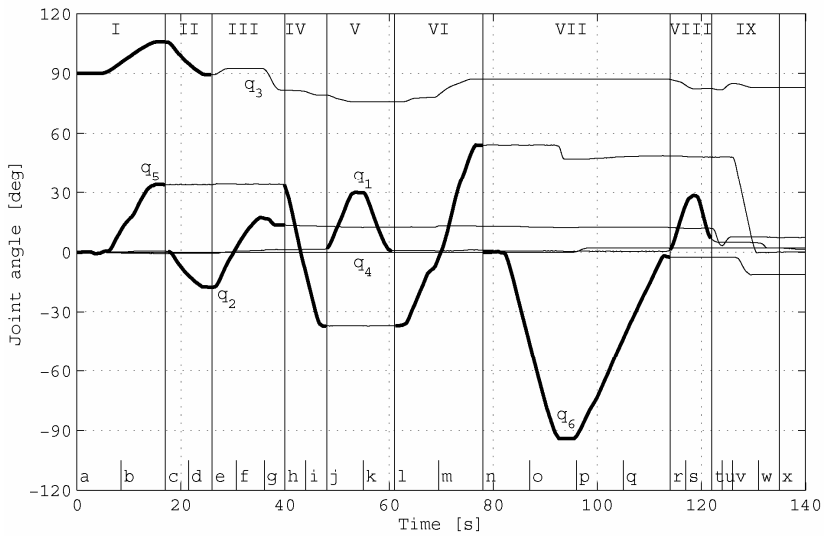


Figure 15. Joint angles during sensorless force guidance

4.3 Sensorless approaches to force guidance with industrial robot controllers

The just proposed approach to force based human robot interaction without F/T sensor requires the access to the actual or desired values of the joint motor currents. Using industrial robot controllers its functionality is often restricted to industrial applications like handling or assembly. Robot programmer can not expect that in the particular programming language any command is available which returns the motor currents or the joint torques. Nevertheless, some industrial robot controllers provide this functionality. An example is the RSI programming mode of the KUKA robot controller, already mentioned in the previous section, which enables sensor guided robot motion in real time. One special RSI object provides the values of the motor currents. They may be integrated into the corresponding controller structure consisting of RSI objects (Winkler & Suchý, 2006a).

Another functionality is the so called soft servo. It is available in several robot controllers, possibly under different name. Using the soft servo feature the stiffness of particular joints can be reduced. Commonly it is implemented by limiting the output signal of the velocity control loop in the joint position cascade controller. The soft servo is useful in some industrial applications to reduce high contact forces. However, soft servo is not favourable in force based human robot interaction. If the robot is moved and the motion is followed by change of the gravitational torques acting on the joints the robot can become unstable. Another disadvantage is that it is not possible to define the desired impedance behaviour. It is caused by robot mechanics with open joint brakes and without motor power. So it may be difficult to move a heavy payload robot by human force interaction.

Figure 16. Snapshots of robot posture during sensorless force guidance

5. Robot teleoperation with force feedback

One special kind of human robot interaction may be developed in robotic teleoperation. Teleoperation means the remote control of a robot manipulator by an operator. Fields of applications for such a system are in hazardous environments like nuclear power plants or chemical plants. Robot teleoperation becomes more and more important also in medicine or astronautics. The basic idea is to remotely control the slave robot by master which can be implemented as control panel, joystick, phantom robot or haptic interface. Operator on the

master side may be supported by audio-visual or haptic feedback from the slave side. If the robot is controlled by the so called haptic interface the operator gets a feedback of the contact forces/torques of the slave robot. The force feedback may be essential for some tasks.
Many structures of teleoperation systems were already published (Hirzinger et al., 1997; Sheridan, 1989; Sheridan, 1995). For the realization of a teleoperation system with force feedback it is also possible to use a force guided master robot instead of haptic interface. One proposed configuration is shown in Fig. 17.

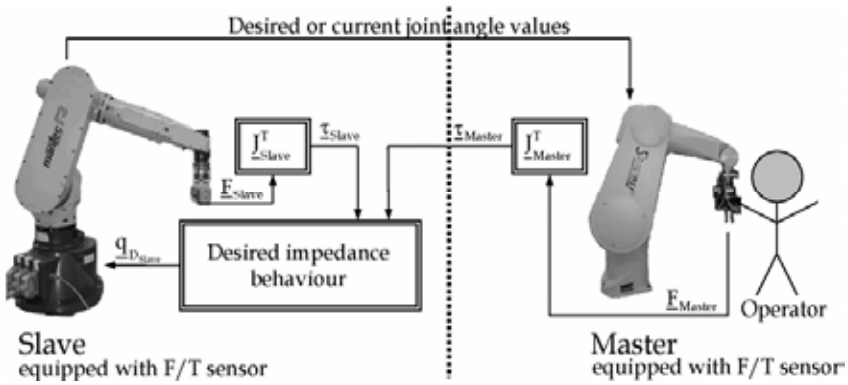


Figure 17. Proposal of a teleoperation system controlled in joint space by a force guided master robot

Both robots are equipped with F/T wrist sensors. On the other side, joint torque sensors or the schemes for estimation of the interaction or contact forces using the motor currents are also conceivable. The interaction forces of the master robot caused by the operator are transformed into joint space according to Eq. (5). The interaction joint torques are transmitted to the slave robot controller, e.g. by Ethernet connection. From the values of $\underline{\tau}_{Master}$ and the contact joint torques of the slave robot $\underline{\tau}_{Slave}$ the desired impedance behaviour computes the desired joint angle values of the slave robot $\underline{q}_{DSlave}$. The components of vector $\underline{q}_{DSlave}$ or the current joint positions are sent to the controller of the master robot where the corresponding motion will to be performed. The joint angle values have to be contingently adjusted to the particular master robot.
Very important for the stability of a teleoperation system is the time delay of data transmission. A lot of publications are dealing with this problem (Sheridan, 1989; Sheridan, 1995). One approach is based on predictive algorithms.

6. Conclusion

In this paper some possibilities and features of force based human robot interaction were presented. First, it was assumed that the robot manipulator is equipped with a six component force/torque wrist sensor to measure the interaction forces and torques caused by an operator. He or she tries to guide the robot throughout the work space by taking the

gripper and pushing or pulling it. It was distinguished between the task and the joint space approach to force guidance which resulted in different behaviour with respect to the robot motion. This was shown in particular with the kinematics of the Planar Two Link Manipulator.

For the specification of the robot dynamics during force guidance the desired impedance behaviour was introduced. Using force guidance for the comfortable teach-in, e.g., the behaviour of a simple mass damper system was proposed as the basis for the desired impedance behaviour. Of course, for the implementation of these ideas in the robot controller additional features had to be realized. Besides some standard features like robot velocity limits and the joint limit stops a very extensive functionality is the intuitive collision avoidance. It is based on the force potential fields around obstacles generated by virtual charges. For this purpose the corresponding algorithm was described. However, some more research activities seem to be necessary on this field. Apart from the desired impedance behaviour the dynamic behaviour of the robot manipulator together with its controller is important for the stability of the whole robot system for force based human robot interaction. Therefore, some common robot systems were regarded with respect to their possibilities of motion generation. This aspect is also crucial for robot force control in general and seems to be important for further research as only few robot controllers admit to set the desired joint torques and/or forces by programmer. On the other side this property is assumed by most of the published approaches to robot force control.

Because a six component force/torque sensor may be very expensive, an alternative approach for the determination of the contact forces/torques between robot and environment was suggested. It is based on the motor currents of the joint drives. From these values provided by the joint power amplifiers it was possible to estimate the forces and torques acting on the whole manipulator arm. The algorithm is especially suitable for low payload robots where the relationship between interaction and gravitational forces is high. In particular the estimation of frictional joint torques from the motor currents, which is very difficult, has to be investigated in more detail. Besides application of this approach to force based human robot interaction it may be also used in standard robot force control.

One special application of force based human robot interaction is robot teleoperation with force feedback. It may be e.g. realized with the joint space approach to force guidance. In this paper the basic structure of a teleoperation system is proposed. It consists of a slave and a master robot each equipped with force/torque sensor. The force guided master robot represents the input device for the operator and the slave robot works in the target environment.

7. References

Angeles, J. (2003). *Fundamentals of Robotic Mechanical Systems,* Springer, ISBN 0-387-95368-X, New York

Choset, H.; Lynch, K.; Hutchinson, S.; Kantor, G.; Burgard, W; Kavraki, L. & Thrun, S. (2005). *Principles of Robot Motion,* MIT Press, ISBN 0-262-03327-5, United States

Craig, J. J. (2005). *Introduction to Robotics Mechanics and Control,* Pearson Prentice Hall, ISBN 0-13-123629-6, United States

Deutsches Institut für Normung (1993). *Industrieroboter – Sicherheit,* DIN EN 775 (ISO 10218)

Gorinevsky, D. M.; Formalsky, A. M. & Schneider, A. Y. (1997). *Force Control of Robotic Systems,* CRC Press, ISBN 0-8493-2671-0, United States

Hirzinger, G. & Heindl, J. (1986). *Device for programming movements of robot manipulators,* US-Patent 4,589,810

Hirzinger, G.; Arbter, K.; Brunner, B.; Koeppe, R.; Landzettel, K. & Vogel, J. (1997). Telerobotic control and human robot-interaction, *Proc. of IEEE International Conference on Robotics and Automation,* Albuquerque, United States, April 1997,

McKerrow, P. J. (1995). *Introduction to Robotics,* Addison-Wesley, ISBN 0-201-18240-8

Sciavicco, L. & Siciliano, B. (2004). *Modelling and Control of Robot Manipulators,* Springer, ISBN 1-85233-221-2, London

Sheridan, T. B. (1989). Telerobotics, *Automatica,* Vol. 25, No. 4, pp. 487-507

Sheridan, T. B. (1995). Teleoperation, Telerobotics and Telepresence: A Progress Report, *Control Eng. Practice,* Vol. 3, No. 2, pp. 205-214

Som, F. (2004). Sichere Robotersteuerung für einen personensicheren Betrieb ohne trennende Schutzeinrichtungen, In: *VDI Berichte 1841,* pp. 745-754, VDI Verlag, ISBN 3-18-091841-1, Düsseldorf

Stadler, W. (1995). *Analytical Robotics and Mechatronics,* McGraw-Hill, ISBN 0-07-060608-0, United States

Türk, S. & Otter, M. (1987). Das DFVLR Modell Nr. 1 des Industrieroboters Manutec r3, *Robotersysteme,* Vol. 3, pp. 101-106

Winkler, A. & Suchý, J. (2005). Novel Joint Space Force Guidance Algorithm with Laboratory Robot System, *Proc. of 16th IFAC World Congress,* Prague, Czech Republic, July 2005

Winkler, A. & Suchý, J. (2006a). An Approach to Compliant Motion of an Industrial Manipulator, *Proc. of 8th International IFAC Symposium on Robot Control,* Bologna, Italy, September 2006

Winkler, A. & Suchý, J. (2006b). Force-guided motions of a 6-d.o.f industrial robot with a joint space approach, *Advanced Robotics,* Vol. 20, No. 9, pp. 1067-1084

Zeng, G. & Hemami, A. (1997). An overview of robot force control. *Robotica*, Vol. 15, pp. 473-482

26

Playing Games with Robots – A Method for Evaluating Human-Robot Interaction

Min Xin and Ehud Sharlin
University of Calgary
Canada

1. Introduction

Some of the acute technological challenges of the near future may relate to the coexistence of intelligent robots and humans. Robotic technology is quickly advancing and some believe that this rapid progress will have a huge effect on people and societies in the coming few decades (Moravec, 1999). Norman (Norman, 2004) suggests that we are already surrounded by simple robots, such as computerised dishwashers and cars. However, these devices still lack the capability and intelligence required for us to recognize them as "robots". Forlizzi and Disalvo (Forlizzi & Disalvo, 2006) demonstrated that even the introduction of the simple, popular and almost ubiquitous Roomba robotic vacuum cleaner had raised important human-robot interaction (HRI) questions, and changed social structures and patterns within domestic environments. Following, it is crucial that we understand the various issues and problems surrounding interaction with robots and be able to design effective interfaces that will allow us to work collaboratively with robotic interfaces.

Current designers of HRI paradigms no longer see robots as fully-controlled subordinates but rather as colleagues of sort, with a spectrum of social and emotional abilities (see for example (Breazeal, 2002)). It is logical that humans will find future autonomous robots more effective and collaborative if the robots act according to behavioural patterns that humans can easily recognize and relate to. Obviously, the challenge of creating robotic interfaces that will be fully aware of rich social settings, roles and proper action is enormous. However, future social robots may be integrated into everyday life tasks if they successfully exploit the human inclination to anthropomorphize animated phenomena and objects (Moravec, 1999), in a sense providing a task-limited illusion of social awareness and supporting a sociably accepted set of actions. Robots can use various methods in order to enhance their social acceptance skills, from the use of natural language, human-like or animal-like appearance and affordances, to animated movement and even cartoon art expression (Young et al., 2007).

How can we rapidly and efficiently design, implement and test sociable and other human-robot interfaces? The straightforward approach would be to design, implement and test for the actual, fully realistic robotic task. However, with robots being used for tasks such as space exploration, urban search and rescue, and battlefield support, developers may find themselves unable to be engaged in meaningful sociable HRI design in academic and

other laboratory settings. Many of the sociable HRI design dilemmas can be answered only after extensive testing and carefully controlled repetitive experiments. Since current robots are often engaged in difficult, dangerous and dirty (DDD) tasks, designing and testing sociable HRI paradigms can be a challenge which will arguably have to wait till sociable robotic platforms are more common and affordable.

In this paper we propose a meaningful and controlled HRI experimental testbed approach based on collaborative gameplay between robots and humans. We argue that our testbed approach supports rapid design, implementation and testing of various meaningful sociable robotic interaction techniques in relatively simple settings. How can we evaluate the validity of our suggested testbed? Similar to the psychology concept of *transfer* of cognitive skill (Singley, 1989), we consider the *transfer of robotic skills* from the testbed to real life. Humans can *transfer* cognitive knowledge from one experience to the other in various ways, with *transfer* being categorized as being either positive or negative; with the original experience enhancing or hindering the target experience, respectively. Similarly, we can assess the quality of an HRI testbed through its ability to *transfer* a set of robotic abilities from one experience to the other. A "good" testbed will be able to provide positive *transfer* of robotic abilities to its target application and inform of right answers to design dilemmas and challenges. On the other hand, a testbed can provide negative *transfer* of robotic abilities, pointing to design decisions that appear proper in experimental settings but fail once tested in real settings.

In the next sections we discuss the notion of human gameplay and its mappings to social interactions and tasks. We discuss the benefits and limitations of using games as HRI testbeds and suggest a set of simple heuristics for designing "good" HRI game-based testbeds which we believe will provide positive *transfer* to real-life settings. We then review several related efforts of using gameplay in HRI and attempt to analyse them using our suggested heuristics. Finally, we reflect on our own experience of designing, implementing and evaluating an HRI testbed based on a board game called *Sheep and Wolves*.

2. Gameplay and HRI

In order to design an effective testbed for human-robot interaction, we look to one of the most frequent human activities that have persisted through the development of civilization: playing games. Games are a staple of everyday life. Whether it is battling through a few games of Mario Kart, participating in a game of Bingo, or dressing up a Barbie doll, we play regardless of age or gender. Although many do not often consider playing games as an essential part of life, it is never the less a critical factor in human development. Through playing games, we interact with our world, communicate with other humans, and even explore brand new environments and experiences. Games are ripe with opportunities for interaction. What if we involve robots within our games? How will we design them to play with humans? What will we learn about human-robot interaction by playing games with robots? These questions motivate our exploration for using games as effective testbeds for evaluating human-robot interaction.

The goal of human-robot interaction is to investigate how to design robots for a variety of applications such as search and rescue or performing domestic duties. Compared to the other more important and practical applications, designing robots to play games seems

like a trivial exercise. How does playing games relate to other application areas? Although each application of robots in the real world has its unique challenges in terms of the mechanical controls required for operation, many applications share commonalities in the social aspects of interaction such as working together in a team. For robots of the future which will exist and work alongside us, the social aspects of interaction are as important if not more important than the electromechanical controls used to operate the robots. Certainly, the activity of playing games also includes such social aspects of interaction. For example, in team sports, teamwork and leadership are concepts often talked about and are critical for the success of the team. Therefore, we believe that by looking at the interaction involved in gameplay, we can explore the common social aspects of human-robot interaction shared with other application areas.

Huizinga and Caillois, two humanist theorists of play, detrivialized the idea of play by making it a central part of the history of human behaviour and culture. Huizinga states (Dovey, 2006):

> "Social life is endowed with suprabiological forms, in the shape of play, which enhances its value. It is through this playing that society expresses its interpretation of life and of the world."

The idea that playing is a reflection of culture, of life, and of the world serves to support our suggestion that games contain many of the essential aspects of interaction present in other applications of the real world. Through games played, we can get a glimpse of how people interact in real life. However, games are also more than just a mimicking of the real world. Playing games and the culture of the real world are in a formative relationship. Not only do games reflect existing cultural practices, but they also serve as a catalyst for generating new cultural practices. Turner calls play "the seedbeds of cultural creativity" (Dovey, 2006), where the generation of alternative social orders, political interventions, and utopian imaginings can take place. Online role playing games are good examples, where virtual societies are created with their own culture of play.

The generative and creative characteristic of games can be beneficial for the development of a human-robot interaction testbed. Although robots have been in use for decades, future robots will need vastly different ways of interacting with humans, in our homes and in our work places. Currently, little is known as to how such interaction will take place. Therefore, playing games with robots provides an excellent opportunity to explore a new social order and new culture of coexisting with robots. Silverstone mentions (Dovey, 2006):

> "Play enables the exploration of that tissue boundary between fantasy and reality, between the real and imagined, between the self and the other. In play we have license to explore, both ourselves and our society. In play we investigate culture, but we also create it."

The duality of playing games as both a way to take into account existing social practices and also to generate new ones is an important point for our suggested use of games as testbeds for human-robot interaction.

Now that we have established the importance of playing games for everyday life and for our human-robot interaction testbed, we will take a deeper look at exactly what is involved in games and what characteristics make playing games a suitable approach for

exploring human-robot interaction. Huizinga offers the following definition (Dovey, 2006):

> "Play is a voluntary activity or occupation executed within certain fixed limits of time and place, according to rules freely accepted but absolutely binding having its aim in itself and accompanied by a feeling of tension, joy, and the consciousness that it is 'different' from ordinary life."

First and foremost, games are played within a restricted domain. As indicated, games have fixed limits in terms of time, place, and rules. This is different from many other real life applications where the possibilities for interaction are endless. For example, a game of hide and seek can be set to be played only within a house, but a search and rescue mission requires a survey of a much larger space. These limits make games favourable for use in experimentation because they help to narrow the scope of exploration both in terms of implementation and also in terms of what is to be investigated. Rather than dealing with all the environmental variables in a real life application, it is much better to target an interesting point with a more focused experiment within a more controlled environment. Although games have many limits, it does not mean that they are rigid. In fact, games can be played however people wish them to be played. Certainly, there needs to be rules, but these can be created by people themselves, or in our case, the HRI application or experiment designers. Depending on the purpose of an experiment, new games can be created, and the rules of old games can be adapted to fit the needs of the experiment. Rules are also unquestionably accepted by the players of games. This is critical because it allows completely new social orders and playing practices to be imposed on the players for exploration. For example, in games, robots can play the role of superiors to humans, a scenario that is unlikely to happen in the interaction experiences of current real world HRI applications. Once again another duality of games in which they can be both rigid and flexible makes them a sensible choice for our testbed.

Finally, games are also a good choice for an HRI experimental testbed because good games are fun and engaging. When people play games, they are actively involved with the activity at hand, and some even become completely immersed in the game world. Games are also more accessible to most people because of their limited rule set. Often, no extensive training is required to play games. This can be beneficial for evaluating the common social aspects of human-robot interaction because if a more demanding, application specific activity is used we would need to account for the skill level of the participants. Plus, to realistically simulate real world social interaction, we need participants to believe that they are in such situations. Since people are used to playing games even in non-realistic scenarios, games can allow us to immerse participants in an envisioned setting even if it is less believable.

Although electronic games are becoming popular, many traditional games are played in the physical world and require tangible interaction. The physicality of games is important because robots are physical entities, and people will eventually interact with them within the physical world, but robots also have access to digital information and are capable of acting in the digital domain. Games can support this physical and digital duality. For example mixed reality techniques can help to design games that will allow interaction with robots in both the physical and the virtual realms.

We have outlined many advantages of using games for exploring human-robot interaction; however, there are certainly limitations to our approach. Most games are started and finished in a relatively short amount of time. This limited duration may not be enough to fully replicate some of the complex social scenarios present in the real world. Also, in games, people sometimes suspend their real world social beliefs and completely submit to the rules and goals of the game, pointing to a questionable quality of transfer to a realistic setting. For example, in games such as Grand Theft Auto (GTA, 2007), players actively participate in violent acts which would be completely disagreeable and detriment in their real life because this is how the game is played. Therefore, we must be careful as to which social aspects of interaction can and cannot be evaluated using a game-based testbed.

Based on the discussion above, we offer the following simple set of heuristics on how to design a "good" game-based testbed for HRI:

1. Tailor the game experience to the HRI design dilemma

When designing a game-based HRI testbed it is crucial to remember that the game is being played above all in order to reflect on the HRI experience. Game rules can and should be altered in order to allow the robots and the humans to interact in a manner that will inform on the HRI design question. A good game-based testbed will integrate into the game environment various aspects of the HRI problem at hand in order to provide better probability of transfer of the learned robotic skills from the game testbed to the target application.

2. Design a fun and engaging game experience

An effective game-based testbed should ideally provide an engaging and fun experience so players become immersed within the social scenario constructed. Highly engaged users will provide interaction insight that will better inform design decisions within the testbed, and will have a better probability of informing design decision in the real-life experience.

3. Design a game played within a bounded space with clearly defined rules

The game should be played within a bounded environment, where undesirable external variables can be filtered out. The game should also have clearly defined rules as this will help with both implementation and testing.

4. Design with the physical and digital robotic duality in mind

We believe good HRI testbeds will enable effective reflection on the robotic physical and digital duality. True, HRI testbeds can be designed to examine only the physical or only the virtual aspects of a specific interaction scenario. We however argue that good HRI testbeds are the ones capturing in their design the robotic "innate" ability to perceive and act in both the digital and physical realms. Without sensitivity to this duality the testbed is, arguably, either an electromechanical environment measuring physical-only aspects of the interaction, or a classic-HCI, software platform testing the virtual-only aspect of the interaction.

3. Game-Playing Robots

In this section we briefly overview a few current examples of the use of games in HRI, and reflect on each of these efforts using our heuristics. Probably the prime example for the use of games in the domain of robotics and HRI is Robocup. Robocup (Robocup, 2007) is an

international project to promote AI, robotics, and related fields. This project makes use of the soccer game to investigate many technical and social aspects of interactive gameplay with robots, exploring issues such as multi-agent collaboration and autonomous agents. Its goal is to develop a team of fully autonomous humanoid robots which by the year 2050 "can win against the human world soccer champion team" (Robocup, 2007). Technologies researched for Robocup are used in more practical applications such as search and rescue.

A related effort, Argall et al.'s work (Argall et al., 2006) with Segway Soccer between human-robot teams builds on the Robocup vision. In this effort, autonomous Segway Robotic Mobility Platforms (RMPs) play soccer alongside humans. The project explores a variety of technical challenges as well as issues which arise in peer-to-peer human-robot teams such as team coordination.

The long-term Robocup vision can be viewed as a strong example of our first design heuristics: robots that will play soccer alongside or against human players will provide illuminating insight to the nature of HRI, and can help explore fundamental robotic interaction challenges such as collaboration, task distribution and leadership. That said, Robocup currently is hardly an HRI effort as it challenges robots to play soccer-like games against other robots with no human interaction or intervention. As an essentially non-HRI effort, currently Robocup places higher emphasis on high fidelity to the game of soccer and its rules (our second and third heuristics) than to an HRI goal.

Bartneck et al.'s work (Bartneck et al., 2006) investigates the factors which influence the way people perceive robots as being alive. In their user study, the game of *Mastermind* is used to create an opportunity for the human participants to become engaged with the robot. The goal of the game is to select the right combination of colours. This task is completed by the robot and the human participant through cooperation and not competition. The robot would make suggestions to the human player as to what colours to pick, and the intelligence and agreeableness of the robot are manipulated for the purpose of the experiment. For example, Bartneck et al. found that humans tend to be more reluctant to switch off a robot that demonstrated intelligence and agreeable behaviour during the *Mastermind* gameplay.

Bartneck et al.'s use of gameplay is an excellent example of tailoring a gameplay experience to an HRI question (our first heuristic). The game being used, *Mastermind*, is very simple and so is the robot involved, limited to non-physical gameplay advice. However simple, the gameplay is sufficient for humans to directly perceive the robot's intelligence and agreeableness, and to act upon this behaviour. Since these, the robot's intelligence and agreeableness, are the experiment's independent variables, the gameplay allowed the designers to simulate a potentially quite complicated social setting through a very simple and engaging game. Since the game is simple, our second and third heuristics are obviously also satisfied in this example: the original *Mastermind* game is played according to its original rules using the original board and providing an, arguably, engaging experience (at least as engaging as the classic *Mastermind* gameplay goes).

Trafton et al.'s work (Trafton et al., 2006) on computational cognitive models for robots uses the children's game hide and seek as a way to understand how young children actually learn how to play hide and seek. This information is then used to create a robot which understands how to play hide and seek from a human perspective. Hide and seek allows the authors to work in a complex and dynamic environment and also allows them to explore

embodied cognition issues. Practically, robots that will be able to understand how to hide or seek can be extremely useful in various security and defence applications.

The hide and seek game provides, arguably, little interaction other than visual one (which can lead to the termination of the game in case the It spotted the hider). Reflecting on our simple design heuristics, Trafton et al.'s efforts seem to be directed more to the gaining insight and developing robot cognitive models based on a proper hide and seek gameplay (that is, our second and third heuristics) rather than a specific HRI question.

4. Sheep and Wolves

The *Sheep and Wolves* testbed (Fig. 1.) is our attempt to evaluate human-robot interaction through the use of games (Xin & Sharlin, 2006). We are particularly interested in the social aspects of collaboration between humans and robots such as teamwork and group dynamics. These social aspects of interaction are important for many applications where humans and robots must work together to solve problems. At the start of this exploration, we wanted to simulate real world scenarios of human-robot collaboration within the lab which motivated the construction of a human-robot interaction testbed. What we needed was an interactive environment where humans and robots can collaborate and also a believable interactive task which will facilitate collaboration. We chose not to follow a real world application such as investigating teamwork in search and rescue because of the scope and complexity of the implementation. Also, we wanted to explore collaboration between humans and robots in general and not just for one particular application. Therefore, our goal was to find a more universal interactive activity which can serve as a metaphor for a large set of human-robot interaction applications and encompass their common interactive qualities.

Figure 1. *Sheep and Wolves* testbed

The eventual inspiration for the testbed was the magical game of *Wizard's Chess* from the popular movie, *Harry Potter and the Philosopher Stone*. In the movie, human players played the game of chess on top of a large chess board moving and acting as game pieces. Not only did the game involve actual physical movement and battles, players also engaged in active communication with each other. For example, one child would use gestures and speech to tell another child to make a certain move. In concept, *Wizard's Chess* serves as an excellent metaphor for the interactive environment of our testbed. The large chess board offers a regular and bounded physical space where interaction can take place, and with the grid-like appearance of the chess board, board games were a natural choice to be used as the interactive task. When it comes to computers and technology, board games have been well explored. They often have well-defined domains and rules and allow for a multitude of potential tasks. Traditionally, interest in board games originated from a mathematical, game theory and AI research point of view, but with *Wizard's Chess* and the unique way in which it is played, we can use these games to construct realistic social HRI scenarios as well. However, chess is usually played between two players facing off against one another. In such a setup, there is very little potential for collaboration. Alternatively, if we attempt to place humans and robots as chess game pieces on the board we may end up with 32 entities which can make for a cumbersome and pricey apparatus. Therefore, we looked toward other board games which were simpler and could still support collaboration.

Following our goals, we decided on the use of another classic board game, *Sheep and Wolves*. This turn-based game is played on a checkerboard, and game pieces can only occupy and move on squares of the same color. The game involves five game pieces, four of which are the wolves, and one is the sheep. The wolves start on one end of the checkerboard, and the sheep starts on the other. The team of wolves are only allowed to move one wolf forward diagonally by one square during each turn. The team's objective is to surround the sheep so it cannot make any legal moves. Meanwhile, the sheep is allowed to move forward and backward diagonally by one square during each turn. Its objective is to move from one end of the checkerboard to the other. Obviously, while the sheep is more flexible in its moves, the wolves' strengths are in their numbers and ability to move as a pack. Traditionally, *Sheep and Wolves* is also played with two players, one playing the sheep and the other playing the team of wolves. Again, to make the game a more interactive and collaborative task we took a similar approach to *Wizard's Chess* and separated the team of four wolves into four separate player positions. This way, we can have humans and robots playing as independent members of the wolves' team.

We chose this game because it is simple yet able to support collaborative gameplay. The metaphor of the game can be extended to various applications where humans and robots are required to share information, opinions, and resources in order to effectively complete a task. By performing a collaborative task in a controlled physical game environment instead of the complex physical world, we are able to focus on interaction. Also, since implementing artificial intelligence for the game of *Sheep and Wolves* is relatively simple, we are able to easily adjust the intelligence of the robots in order to develop varying robotic behaviours.

In our game we have elected to use Sony's AIBO ERS-7 robot dogs as our robotic participants. These fairly capable commercial robots allow us to rapidly build prototype

interfaces for evaluation. For the physical environment of the game, we elected to use a 264cm (104'') by 264cm RolaBoard™ with the standard black and white checkerboard pattern. Each square measures 33cm (13'') by 33cm, providing sufficient room for an AIBO wolf to sit on or humans to stand on. This confined shared space is ideal for robots to navigate in. The lines and corners of the checkerboard serve as readily available navigation markers for movement on the checkerboard, and camera calibration can also be achieved using corner points to allow for augmented reality interfaces and localization of humans on the checkerboard.

In the first game we have created using this testbed concept, all four wolves are represented by the AIBOs and the sheep is a virtual entity (Fig. 1). We decided to include virtual entities within the game to highlight the multimodal nature of robots, being able to interact both in the physical world but also to percive and act in the digital domain. The use of virtual entities also serves to level the playing field for robots since humans must rely on the robots' senses when it comes to the virtual sheep, but for the robots the virtual entities are as real as the physical components of the game. The AIBOs physically move and sit down on the checkerboard to indicate movement of the wolves in the game. A human player controls a single AIBO wolf at a remote computer using a telepresence interface, personifying the robotic entity within the game. Other uncontrolled AIBO wolves are autonomous robotic teammates which the human player must collaborate with. Live video of the physical game environment from the controlled AIBO's point of view is provided to the remote human player, and mixed reality is utilized for visualizing the virtual sheep on top of the physical board. Winning the game as wolves requires teamwork. The human player has to provide suggestions to the team and consider propositions made by other teammates in order to help the team reach intelligent decisions on the moves the team should make. This setup effectively generates the collaborative scenarios intended.

With the first iteration of the testbed complete, we used it to perform a simple user study to evaluate the effect of two extreme robot behaviours on different aspects of collaboration. This study was exploratory in nature, we wanted to see if the game-based testbed is sensitive to the social aspects of interaction we wish to explore. The study involved two extreme robot behaviours for the autonomous AIBO wolves. In one test condition, all the AIBO teammates were programmed to be always submissive to the human player, and in the second condition they were programmed to be always assertive and make the human player feel inferior. We asked participants to play one game in each condition and assessed the gameplay experience with post-test questionnaires. We performed the pilot study with 5 participants and the actual study with 14 participants (Xin & Sharlin, 2006). One of the interesting results found in the pilot study was that human players trusted the assertive robots more than the submissive robots when it comes to decision making. However, this finding came up inconclusive when we performed the actual study. The other interesting finding was that when we asked the human players to evaluate their robotic teammates at the end of the game, most of them assessed their teammates as individuals and gave them different scores. This was surprising because all three autonomous AIBO wolves were programmed with the same behaviour. This finding was promising for our game-based testbed concept because it indicates that the game is able to produce a sociably immersive experience where players believe that they are participating in a collaborative

game with realistic team members even when in actuality the game is based on rather simplistic robot behaviours.

5. Discussion

From the description of our game-based HRI testbed above, we would like to provide some lessons learned in terms of the benefits and challenges of our approach and application. First and foremost, this testbed is relatively simple to construct and cost effective. Using readily available products such as the AIBO and the RolaBoard™, we were able to rapidly construct and prototype our testbed. This again speaks to the flexible nature of games which can be created with whatever is easily assessable or modified to make implementation easier. Second, because the game has simple and well defined rules and is played within a bounded environment, we can rapidly prototype new games and design new user studies. In fact, we are currently in the second iteration of our testbed which will feature a slightly modified game used to investigate a different research question. Third, we found the use of both physical and virtual entities to be useful for experimental design. Humans currently still have the advantage when it comes to interaction in the physical world. Robots, however, have the advantage when it comes to interacting with digital information. By playing with these factors, various social relationships can be generated such as trust. Finally, although our initial goal with the current testbed was to look at collaboration, we found that having a game which involves collaboration is beneficial for increasing the potential of other forms of social interaction. For example, when humans and robots need to collaborate, they are required to communicate with each other in a much more complex manner than simple command and execution.

Certainly, there are a couple of stumbling blocks with our testbed exploration as well. The biggest problem with using games for experimentation is that we can not really control how the game is played. Each participant will have a different gameplay experience based on the outcome of the game and the way the game was played. Therefore, it is difficult to compare data. For example, on the issue of trust, the outcome of the game played significantly affects the participant's opinion since winning tends to build trust. Scripting games is one solution to this problem, but this leads to the dilemma of having to disguise the scripting process to the participant, and in some situations, scripting is not possible. The other problem with our game-based testbed is that evaluation of game experiences in general is a difficult problem by itself. Generally it is hard to collect quantitative data for games, and most forms of gameplay evaluations are often vague. With exploratory studies, these issues are not critical, but with more focused studies, they can skew the data. Currently, we can not offer great solutions to these problems, but we are looking at methods to strike a balance between restrictive and more freeform styles of games. We also have not attempted to transfer the primitive results from our user study to other applications since more rigorous experimentation needs to be performed, but we feel the few results that we do have make sense for other applications as well. However, it is promising to see that the game-based testbed approach is able to explore critical social issues of human-robot interaction such as trust which can assist robot designers in developing future domestic and sociable robots.

6. Conclusion

The road for full integration of sociable robotic interfaces into the fabric of society is probably still long. However, robots with varying degrees of social ability are predicted to have a larger role in our everyday life in the near future. Arguably, many sociable HRI paradigms and ideas that seemed to belong not so long ago to science fiction literature can already be tested in lab settings. In this paper we suggested the use of gameplay and games as practical and attractive testbed platforms for the design, implementation and testing of sociable HRI concepts. We presented our simple set of heuristics for designing "good" game-based HRI testbeds and reflected on our heuristics' strengths and weaknesses vis-à-vis a number of recent related game-based sociable HRI projects. We discuss our ongoing efforts towards a sociable HRI game-based testbed using the mixed-reality *Sheep and Wolves* board game. We described the project technical realization, experimentation and current findings and discussed *Sheep and Wolves* strengths, drawbacks and future directions.

7. References

Argall, B.; Gu, Y.; Browning, B. & Veloso, M. (2006). The first segway soccer experience: towards peer-to-peer human-robot teams, *Proceeding of the 1st ACM conference on Human-robot interaction*, pp. 321-322, 1-59593-294-1, Salt Lake City, March 2006, ACM Press, New York

Bartneck, C.; Hoek, M.; Mubin, O. & Mahmud, A. (2007). Daisy, Daisy, give me your answer do!: switching off a robot, *Proceeding of the ACM/IEEE international conference on Human-robot interaction*, pp. 217-222, 978-1-59593-617-2, Arlington, March 2007, ACM Press, New York

Breazeal, C. L. (2002). *Designing Sociable Robots*, The MIT Press, 0262524317

Dovey, J. & Kennedy, H. W. (2006). *Game Cultures: Computer Games as New Media*, McGraw-Hill Education, 978-0-335-21357, New York

Forlizzi, J. & Disalvo, C. (2006). Service robots in the domestic environment: a study of the roomba vacuum in the home, *Proceeding of the 1st ACM conference on Human-robot interaction*, pp. 258-265, 1-59593-294-1, Salt Lake City, March 2006, ACM Press, New York

GTA, Grand Theft Auto (2007). Online: http://en.wikipedia.org/wiki/Grand_Theft_Auto_(series)

Moravec, H. (1999). *Robot: Mere Machine to Transcendent Mind*, Oxford University Press, Oxford

Norman, D. A. (2004). Emotional Design: *Why We Love (or Hate) Everyday Things*, Basic Books, New York

Robocup (2007). Online: http://www.robocup.org/

Singley, M. K. & Anderson, J. R. (1989). *The Transfer of Cognitive Skill*, Cognitive Science Series, Harvard University Press, Cambridge

Trafton, J. G. ; Shultz, A. C. ; Perznowski, D. ; Bugajska, M. D. ; Adams, W. ; Cassimatis, N. L. & Brock, D. P. (2006). Children and robots learning to play hid and seek, *Proceeding of the 1st ACM conference on Human-robot interaction*, pp. 242-249, 1-59593-294-1, Salt Lake City, March 2006, ACM Press, New York

Xin, M. & Sharlin, E. (2006). Exploring Human-Robot Interaction Through Telepresence Board Games, *Proceeding of the 16th Conference on Artificial Reality and Telexistence*, Hangzhou, November 2006, Springer

Young, J. E.; Xin, M. & Sharlin, E. (2007). Robot Expressionism Through Cartooning, *Proceeding of the ACM/IEEE international conference on Human-robot interaction*, pp. 309-316, 978-1-59593-617-2, Arlington, March 2007, ACM Press, New York

27

Designing Simple and Effective Expression of Robot's Primitive Minds to a Human

Seiji Yamada[1] and Takanori Komatsu[2]
[1]*National Institute of Informatics,* [2]*Future University-Hakodate*
Japan

1. Introduction

In recent years, home robots and entertainment robots like Roomba, AIBO have started to move out of laboratories and in people's homes. Almost all of them are for entertainment use to a user and the remaining ones are for a simple home task like rough sweeping by themselves. However home robots for a home task are expected to spread out in our dairy life rapidly because there are strong needs for the robots to be able to achieve various home tasks like sweeping, cooking, washing up, clearance and so on. In this situation, since a robot often cannot achieve such a task by itself, it needs to ask a user help it's task. For example, even though sweeping is a very simple home task, a robot can not remove a heavy or complicatedly structured obstacle like a chair, a table or a cart in order to sweep the floor under it. In this case, a robot should ask user helping behaviors to remove these obstacles .

The significant problem here is how to express the robot's internal state to a user. We call such an internal state the *robot's mind* because it may correspond to a human state of mind in the theory of mind (Baron-Cohen, 1995). We consider that these expressions should be designed depending on the robot's appearance because the appearance of the robots would significantly influences human impressions and interpretations of robot's expressions. Although it is an important problem to design how a robot expresses and informs its mind to a human depending on the appearance, few studies on designing robot's expression in such a way have been done thus far. In general, one of the simplest ways to express the mind is to use verbal communication with speech synthesis and it may be independent of a robot's appearance. However, in these days, such verbal communication significantly depends on the processing quality of natural language, and unfortunately, this quality is not mature for the actual use for our purpose. Hence we focused on nonverbal communication because various psychological researches showed that nonverbal communication also contain rich information than verbal ones.

In this chapter, we propose a policy to design expressions of robot's mind depending on its appearance, called "SE4PM: Simple Expression for Primitive Minds." According to this policy, we would like to argue that a designer should design simple information with a simple appearance to express primitive robot's minds (Yamada & Komatsu, 2006). We actually apply this SE4PM policy to express primitive minds of a robot with an appearance of a simple mobile robot implemented with LEGO MindStorms, and design beep sound as simple expression. We consider this beep sound is a promising way to express primitive

minds like negative, positive, and neutral because it was reported to be effective in human-computer interaction (Komatsu, 2006). To compare with our proposed expression, we utilized a pet robot, AIBO, which has a complicated appearance and can express its primitive minds by executing some complex behaviors with motion, light and sound. We investigate the effectiveness of SE4PM policy by a psychological experiment to compare the two robots. Finally we obtain results to support our SE4PM policy and find out it is a valid policy to design expression of robot's primitive minds depending on the appearance.

Different types of social robots have been developed to assist with various tasks in our daily life (Morishima et al, 1995; Ishiguro et al, 2001). In general, these robots have a particular appearance that is designed similar to that of humans or pet animals, i.e., beings that are familiar to us. Most humans who interact with these robots notice the familiarity of their appearances, and this makes it easier for them to communicate with these robots actively (Breazeal & Velasquez, 1998). However, a robot's appearance should not be the sole focus; designing the robot's expressed minds to enable better communication with users is also important. Based on this concept, Ono and Imai (Ono & Imai, 2000) developed an interactive robot that can express behaviors associated with frustration when it encounters certain obstacles that interrupt its pathway.

On the relationship between a robot's appearance and the function, Mori pioneered relationship between an appearance and a movement in a robot with the uncanny valley (Mori, 1970; 1982). His uncanny valley described a robot becomes more uncanny as it becomes more similar to a real human. Although the uncanny valley does not directly imply SE4PM, the basic consideration is close to it. We will discuss the relationship of the uncanny valley and our SE4PM with feasibility, familiarity and implementability in terms of engineering. Duffy also discussed anthropomorphism of a robot with much insight (Duffy, 2003), and he pointed out various important issues on relationship between anthropomorphism and a robot. In contrast with their studies, we propose the concrete design policy to express primitive minds, actually design them and verify the effectiveness.

Matsumoto et al. proposed a "Minimal Design" for interactive agents (Matsumoto et al, 2005); that is, agents should only have a minimalist appearance or express a minimal amount of information to users. In fact, they applied this minimal design policy in developing their interactive robots "Muu" (Okada et al, 2000) and life-like agent "Talking Eye" (Suzuki et al, 2000). Moreover, Reeves & Nass showed in their "Media Equation" studies that anthropomorphized agents or computers might induce natural behaviors in humans, such as those that we direct towards other people (Reeves & Nass, 1996). Although the policies of minimal design and Media Equation are similar to our hypothesis that a detailed and likable appearance and expressed information are not vital for informing us of primitive minds, they lack a concrete strategy, like "which kinds of appearance should agents have" or "which kinds of information should agents express to users." In contract, our study provides a concrete strategy for designing interactive agents by clarifying the relationship between the agent's appearance and its expressed information to make user understands these primitive minds.

Kanda et. al (Kanda et al, 2005) investigated human behaviors to humanoid robots with two different appearances, ASIMO, Robovie (Ishiguro et al, 2001), through a systematic psychological experiment with participant. As results, they found statistical significant difference in non-verbal behaviors like movement of arms, greeting motions, not in verbal

behaviors. Their results are interesting, however they do not propose any design policy to express robot's minds.

2. SE4PM: Design Policy to Express Robot 's Primitive Minds to a Human

We propose a policy to design expression of robot's minds depending on the appearance, called "SE4PM: Simple Expression for Primitive Mind". According to this SE4PM, we would like to argue that a designer should design simple information from a robot with a simple (e.g. robot-like) appearance to express its primitive minds. On the other hand, this policy is based on the following hypothesis about the relationship between the robot's appearance and its expressed information on the user's understanding of primitive minds: A robot with a human-like or animal-like appearance expressing complex and likable actions or behaviors is more confusing for users and is not really effective for conveying primitive minds. On the other hand, an robot with a more typical robot appearance conveying subtle expression (Liu & Picard, 2003), which shows simple but intuitive information that can be more readily understood, is much more effective for informing users of the robot's primitive minds (Fig. 1). If this hypothesis to support SE4PM was shown to be true, various interactive robots could be developed, ones that can interact naturally with users without the need for a huge budget to create a complex and likable appearance for these robots.

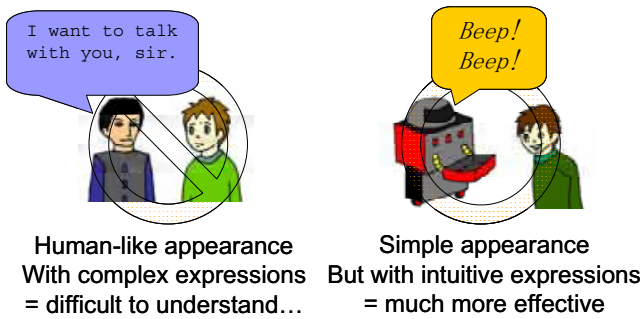


Figure 1. Concept of our hypothesis about the relationship between robot's expressed information and its appearance

3. Realizing SE4PM with Beep Sounds and Mobile Robot-Like Appearance

According to SE4PM, it is expected that we are able to design actual expression and appearance of a robot for express primitive minds. In this study, we realize simple expression with beep sounds and simple appearance with a mobile robot-like appearance. This realization is based on the following reasons.

Beep sounds: Komatsu (Komatsu, 2006) showed that people can estimate different primitive minds by means of simple beep-like sounds with different durations and inflections. He reported the following results.

- Sounds with decreasing intonation with shorter durations were perceived as a "positive mind, " such as "agreement."

- Sounds with increasing intonation regardless of its durations were perceived as a "negative mind, " such as "disagreement."
- Flat sounds with longer durations were estimated as a "neutral mind, " such as "hesitation."

These beep sounds were simple but intuitive and effective information for the user to understand primitive minds. We applied these beep sounds as expressed information from robots that did not have a life-like appearance and behaviors.

Mobile robot-like appearance: We formed the MindStorms as a mobile robot, thus Mobile robot-like appearance is utilized without cost of additional sensors and actuators. As well known, MindStorms is a kind of LEGO, thus we can easily configure the appearance with various simple sensors and actuators.

4. Experiments

4.1 Overview

As already mentioned, our SE4PM hypothesized that a robot with a typical robot appearance expressing simple but intuitive information regarding primitive minds is much more effectively to users. We then conducted a psychological experiment to investigate this hypothesis.

4.1.1 Expressing Minds

We focused on the following three primitive minds as primitive and important ones for a user: negative, positive, and neutral. These three minds correspond to a valence value that is the basic dimension of complex emotions or affect (Reeves & Nass, 1996). These minds were briefly explained to the participants so that they would have a rough idea of how to recognize the minds.

- Positive Mind: Agreement, e.g., acceptance.
- Negative Mind: Disagreement, e.g., surprising, doubting.
- Neutral Mind: Hesitation, e.g., being lost for words.
- These three primitive minds and interpretations were the same as the ones used in Komatsu's former study (Komatsu, 2006).

4.1.2 Appearance of Robots

We utilized the following two robots as robots in our experiment. One was AIBO (ESR-7, SONY corporation). It is a robot that has a detailed and animal-like appearance and behaviors. The other was MindStorms (Robotic Invention System 2.0, LEGO cooperation) which is is the robot that has a typical robotic appearance like "Star Wars' R2D2." AIBO is one of the most famous consumer pet robots, and MindStorms consists of LEGO blocks and Micro-computer modules. The user can then determine their preferred robot appearance by using various types of LEGO blocks. The appearances of AIBO and MindStoms are shown in Fig. 2.

In addition, for a control condition, we utilized a normal laptop PC (Let's Note, W2 CF-W2DW6AXR, Panasonic corporation) that was utilized to express beep sounds in the former study (Komatsu, 2006). The reason we utilized this laptop PC as a control was that it has a non-robot-like appearance compared with other robots (AIBO and MindStorms). In this

research, remind that we call this PC a robot too. Fig. 3 shows the actual appearance of these three robots. They actually have the nearly same body size.

Figure 2. The appearances of AIBO (left) and MindStorms (right)

Figure 3. Two robots and a PC utilized in this experiment: AIBO, MindStorms, and a laptop PC (from left to right)

4.1.3 Expressed Information

For expressing primitive minds to users, AIBO expresses the prepared dog-like behaviors, and MindStorms and the PC express the beep sounds that were utilized in Komatsu's former study (Komatsu, 2006).

- *Expressing information of AIBO* : SONY prepared utility software called the "AIBO entertainment player" for AIBO users, which offers about 80 basic preset motions, like "cheer up" and "good morning." Among these motions, we chose the following six motions (two motions for each mind) that were similar to typical dog-like behaviors and accorded them with three primitive minds.
  - Positive mind: "Happy 1" (wagging her tail cheer-fully), "Happy 3" (blinking face LED expresses smiling face)
  - Negative mind: "Angry 1" (howling action), "Un-happy 1" (moving her tail cheerlessly)
  - Neutral mind: "Incline her head", "Wondering" (looking doubtful while moving her tail)
- *Expressing information of MindStorms and PC* : The following six beep sounds (two sounds for each mind) showed higher interpretation rates (more than 80%) in the former study in each of the minds. These sounds were triangle waves generated by sound authoring software called "Cool Edit 2000," and they have the same F0 average of 131Hz.
  - Positive mind: Two beep sounds with decreasing intonation (One is a duration of 189ms and a de-creasing transition range in the F0 value between the onset and endpoint of 125Hz; the other is a duration of 418ms and a decreasing transition range of 125Hz)
  - Negative mind: Two beep sounds with increasing intonation (One is a duration of 189ms and an increasing transition range of 125Hz; the other is a duration of 819ms and an increasing transition range of 125Hz)
  - Neutral mind: Two beep sounds with a flat intonation (One is a duration of 639ms; the other is a duration of 819ms)

Just before AIBO expressed these behaviors to participants, the experimenter said "Ready" to them, and then AIBO started expressing the selected behaviors. Before MindStorms expressed these sounds, the experimenter started moving MindStorms backward by about 5 cm and then forward by about the same distance. And before the PC expressed its sounds, the experimenter flashed its display. These actions were meant to tell the participants that the "stimulus is about to be expressed."

4.2 Experimental Procedure

The participants were 18 Japanese university students (12 men and 6 women with a mean age of 21.2 years). All participants were not familiar with AIBO, MindStorms, and other robots in general.

First, the experimenter gave participants the following instructions: "the purpose of this experiment is to evaluate the three robots by means of a questionnaire. Specifically, these robots express certain information that includes one of three primitive minds (positive, negative, and neutral), and your tasks in this experiment are to answer "which kinds of mind were included with the expressed information," and to tell us your impression of these robots in the questionnaire."

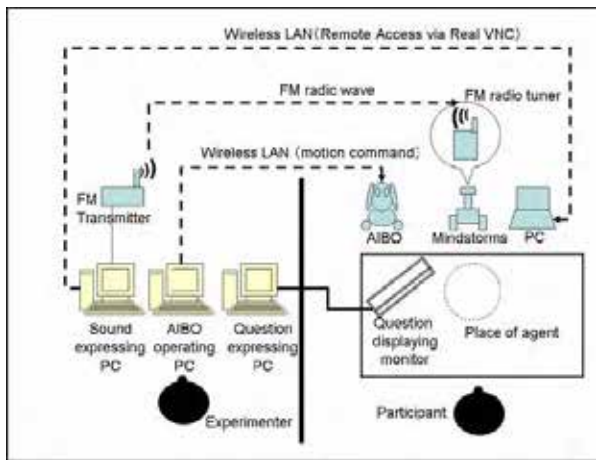


Figure 4. Experimental settings

Figure 5. Actual experimental scene (a participant facing AIBO)

The experimenter locating behind the partition used a wireless LAN to make AIBO express its behaviours in front of participants. To make MindStorms express its beep sounds, the experimenter played the sounds on a computer beside him, and then the sounds were transmitted as an FM radio wave. The FM radio tuner loaded on MindStorms received this radio wave and played the received sounds to participants. For the PC, the experimenter remotely controlled it by means of "Real VNC remote access system," and he started

playing the beep sounds at the appropriate time. The set up of the experiment is depicted in Fig. 4, and a photograph of the actual conditions in the experiment is Fig. 5. where a participant is facing with AIBO.

When the robots expressed the behaviors or sounds to participants, the display placed in front of them simultaneously showed the following questions, "*Did you feel that M was this robot's mind based on this presented information?*"; *M* was the randomly selected mind among the three primitive minds. Participants were asked to answer YES or NO on the questionnaire. Specifically, each participant went through 18 trials (6 parts of information X 3 minds) for each robot. The order of presentation for the stimulus-question pairs was counterbalanced, and all participants were assumed to have contingent tendencies for judging each of the trials.

After finishing 18 trials with one robot, the participants were asked to fill in a questionnaire about their impressions of these robots. The questions are shown in Table 1. After filling in this questionnaire, another 18 trials were conducted with the next robot, and then again with the last one. Thus, all participants worked with all three robots. The experiencing ordering of robots (AIBO, MindStorms and PC) was also counterbalanced.

Q1: Did you understand the robot's minds?
Q2: Was the expressed information easily understandable?
Q3: Did you enjoy the way that the robot expressed its information?
Q4: Do you think that this robot can be part of our daily life?
Q5: Do you think that this robot has emotions?
Q6: Do you think that you can communicate effectively with this robot?

Table 1. Questionnaires on impressions of a robot

5. Experimental Results

5.1 Can Participants Estimate the Robot's Mind Correctly?

The average number of correct answers (within 18 trials) was calculated for each robot to determine whether or not the participants could estimate the robot's minds correctly. The results were that participants showed an average of 8.50 answers correct with AIBO, 14.33 with MindStorms, and 13.78 answers with the PC as shown in Fig. 6.

From these results, it is evident that using MindStorms or a PC is a much more effective method of informing participants of a robot's minds compared with AIBO. Thus, these results support our hypothesis of SE4PM, which is, a robot with a typical robot appearance expressing simple but intuitive information regarding primitive minds is much more effectively to users than a robot that has a life-like appearance expressing more complex and likable behaviors. Although some concerns remain as to whether our hypothesis of SE4PM will stand up to further scientific analysis in an experiment, these results to support SE4PM

will likely have a significant impact on the traditional design policy, which has attempted to make the robot's appearance similar to those of humans or pets.

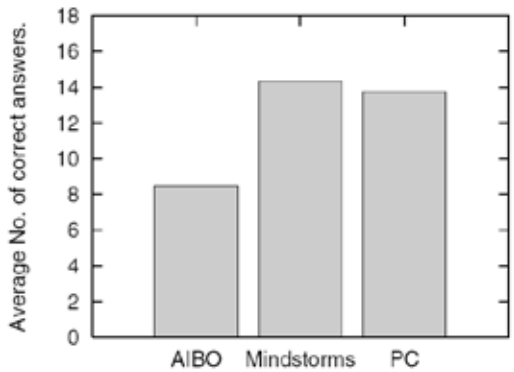


Figure 6. Average numbers of correct answers for each robot

5.2 Subjective Impressions of These Robots

We investigated the user's subjective impressions about these three robots by means of a questionnaire that was completed for each of the robots after the trials. Our investigation involved the use of an ANOVA of each of the aforementioned questions, which were answered using a six point likert scale (With lower points indicating poorer assessment: one point was the worst assessment, and six points was the best).

The average scores in evaluating the different robots for each question are depicted in Fig. 7. AIBO had the highest evaluations, and the PC had the lowest. However, the results of the ANOVA determined that four different relationships were present between the three robots.

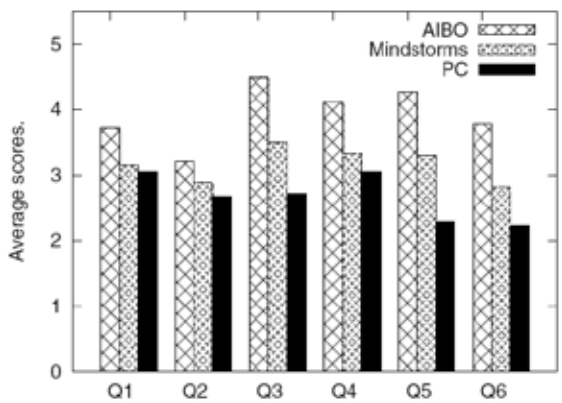


Figure 7. Average scores for each question in questionnaire

- *Relationship A*: AIBO received the highest overall evaluation: This relationship was observed in Q3 "Did you enjoy the way that the robot expressed its information?" and Q4 "Do you think that this robot can be part of our daily life?" Specifically, there were significant differences between AIBO and MindStorms and between AIBO and the PC (Q3: $F(2,51) = 10.33$, $p < .01$ (**), $Mse = 1.3845$, 5% *level*, Q4: $F(2,51) = 4.38$, $p < .05$ (*), $Mse = 1.2298$, 5% *level*). These results stem from the fact that AIBO is already well known as a sophisticated robot for entertainment purposes.
- *Relationship B*: AIBO received a higher evaluation compared with the PC: This relationship was observed in Q1 "Did you understand the robot's minds?" The only significant tendency was between AIBO and the PC ($F(2,51) = 3.16$, $p < .10$ (+), $Mse = 0.7265$, 5% *level*). However, the average number of correct answers for the PC's responses was significantly higher than that for AIBO, and it was nearly the same as that of MindStorms. Thus, a significant gap was evident between the effectiveness of the actual function (informing participants of the robot's minds) and the participants' impressions of the robots.
- *Relationship C*: Order of preference in the evaluation was AIBO, MindStorms, and PC: This relationship was observed in Q5 "Do you think that this robot has emotions?" and Q6 "Do you think that you can communicate with this robot?" Specifically, significant differences were evident between these three robots (Q5: $F(2,51) = 23.64$, $p < .01$ (**), $Mse = 0.7614$, 5% *level*, Q6: $F(2,51) = 14.56$, $p < .01$ (**), $Mse = 0.7492$, 5% *level*). Here, AIBO received the highest evaluation, just as in relationship A. Moreover, MindStorms received a higher evaluation than the PC.
- *Relationship D*: No differences among the two robots and the PC: This relationship was observed in Q2 "Was the expressed information easily understandable?" ($F(2,51) = 1.04$, *n.s.*). Here, although AIBO received higher evaluations on most questions, there were no significant differences between the three robots.

6. Discussion

6.1 Coverage of SE4PM

We conducted psychological experiment to verify the effectiveness of SE4PM design policy, and it can be said that the results eventually supported our proposed SE4PM. However these results are concerned with just case studies and just one example of various SE4PM realizations. Hence we need to discuss the coverage of the experimental results.

We consider the generality as to the following. First the results in this work show a concrete example that SE4PM-based robot design outperformed conventional one, with life-like and complicated appearance and expression, in expressing primitive minds. This also shows that another direction to design effective social robot without expensive appearance and actuators.

Second, by developing various simple expression based on SE4PM under a fixed robot-like appearance, the coverage of SE4PM can spread more and more. For example, we also developed and investigated a motion-based method to inform a user of robot's minds (e.g. a trouble with the front obstacle) (Kobayashi & Yamada, 2005). In this work, a simple back-and-forth behavior of a robot with a simple mobile robot's appearance is shown effective. We can utilize this behavior as simple expression and extend simple expression of SE4PM.

6.2 What Will the Gap between User's Impressions and Mind Estimation Cause?

The results of the experiment clarified that the evaluations of AIBO in the questionnaires were mostly higher than those of the other robots. However the average number of correct answers in interpreting AIBO's basic behaviors was significantly lower. At a glance, the first set of results indicates that AIBO is an appropriate robot for communicating with users. However, these superiorities are derived from its well-designed appearance as a commercial product or from participants' superficial impressions, such as "AIBO is a famous, cute, and clever pet robot," not from the fact that its behaviors are easily understandable. Yet, a serious gap has been demonstrated between the high evaluation participants gave AIBO and its inability to inform participants of its primitive minds. Specifically, the results of Q1 in table 1 are an obvious piece of evidence for this gap; AIBO received its highest evaluation on Q1 "Did you understand the robot's minds?" even though most participants perceived AIBO's expressed information incorrectly.

If these participants continued interacting with this robot, they would eventually notice the gap between its behavior and appearance, and then this gap might disappoint the participants and cause them to lose interest in communicating with it further. They would say something to the effect that "This robot looks very cute, but its behaviors are not really understandable..." This indication can be observed in the results of Q2 "Was the expressed information easily understandable?" No significant difference existed between the three robots on this question.

MindStorms, the other robot used in our test, received a lower evaluation from participants. However, the average number of correct answers was significantly higher; that is, MindStorms was better at informing participants of their primitive minds. If participants continuously communicated with it, they might notice that its behavior was more understandable, and subsequently, they might have a better subjective impression of the robot.

6.3 Influence of Robot's Appearance on Users

In our experiment, MindStorms and the PC expressed the same information (beep sounds) so that we could investigate the effects of the robots' appearance on the user's impressions and on their ability to estimate the robot's primitive minds.

In regards to estimating primitive minds, the average number of correct answers to MindStorms' expression was somewhat higher than that to PC's ones. However, the differences were not significant. The participants' impressions of MindStorms were significantly higher on the following two questions related to the participants' emotions: Q5 "Do you think that this robot has emotions?" and Q6 "Do you think that you can communicate with this robot?" These results were caused by the familiarity with the MindStorms' robot-like appearance, compared with the PC, which did not have a robot-like appearance. However, this does not automatically mean that pursuing a familiar appearance increased the evaluations of participants.

6.4 Relationship with Mori's Uncanny Valley

6.4.1 Life-Likeness and Familiarity

So far we discussed about the relationship between the robot's appearance and its expressed information for informing its minds for humans. On the other hand, about the effects of the robots' appearance (or human likeness) on the familiarities human users would feel, Mori

(Mori, 1970; 1982) proposed the pioneering hypothesis *uncanny valley*; that is, the appearance of the robot is getting similar to ones of human beings, in some point, humans suddenly start feeling uncanny or losing the familiarities with this robot because the subtle differences on the appearances between actual human beings and the robots are emphasized. Mori described this relationship between the robots' *life-likeness* to human beings and *familiarity* that human users would feel as the following qualitative diagram shown in Fig. 8.
Basically, we agree with this Mori's hypothesis because the relationship between the robots' appearance and the familiarities seems to succeed in explaining our hypothesis shown in Fig. 1: the robots with rich appearance (quite similar with actual humans or pet animals) expressing the likely information (verbal information or animal like behaviors) are more confusing to users and are not really effective for interacting with users: Instead, the robots without rich appearance expressing the simple but intuitive information such as subtle expressions are readily understood and are much more effective for their interaction.
We assumed that the former case in the above our hypothesis (the robots with rich appearance expressing the likely information) would correspond to the *bottom* (*B* in Fig. 8) of the uncanny valley, so that it is expected that users would feel uncomfortable as shown in the left of Fig. 1. On the other hand, we also assumed that the latter case (the robot without rich appearances expressing subtle expressions) would correspond to the *peak* (*P* in Fig. 8) just before starting the uncanny valley, so that it is expected that users would feel comfortable as shown in the right of the Fig. 1 (Komatsu & Yamada, 2007).

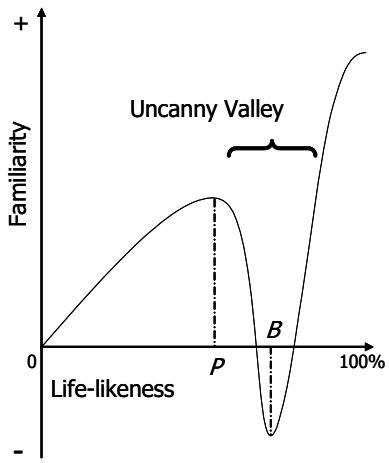


Figure 8. The concept diagram of Uncanny Valley (Mori, 1970)

6.4.2 Life-likeness and Implementability

In this chapter, we assumed that the robot without rich appearance expressing the simple but intuitive information, e.g., subtle expression, is much more effective for human users to understand the robots' minds. One of the reasons to focus on utilizing subtle expressions is

that it is technically and economically easy to implement these kinds of expressions into the robot. However, this does not imply that every kind of the robots should express the subtle expressions.

Corresponding to the results of this study, these subtle expressions should be designed according to the appearance of the robots, e.g., the beep sounds should be expressed from PC, not from robots. Therefore, each robot would have each appropriate subtle expression, and then its *implementability* would be getting decreasing according with increasing the *life-likeness* shown in Fig. 9. Simply saying, the robot without rich appearance (lower life-likeness) requires expressing rather simple expressions, while the robot with rich experience (higher life-likeness) does complex expressions. We assumed that the implementability is constantly decreasing according to increment of the life-likeness.

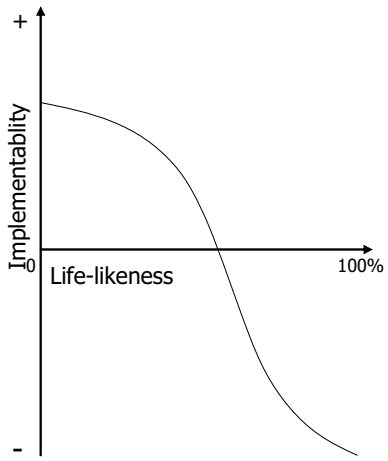


Figure9. The hypothesized implementability according to its life-likeness

6.4.2 Familiarity, Implementability and Life-likeness

So far we discussed about the relationship between the life-likeness of the robots and the familiarity human users would feel, and the one between the life-likeness and the implementability of expressed information. It can be said that the former relationship *familiarity* is about between the robot and the user, while the later one *implementability* is about between the robot and the designer. Thus we are able to describe them as the following two eqauations.

$$\text{Familiarity} = q(\text{robot, user}) \tag{1}$$

$$\text{Implementability} = w(\text{robot, designer}) \tag{2}$$

Here, we assumed that the factor *feasibility* of the intimate interaction between users and the robots can be proposed by the sum of the two factors *familiarity* and *implementability* as the following equation (3) and (4):

$$\text{Feasibility} = \text{Familiarity} + \text{Implementability} \tag{3}$$

$$= q(\text{robot, user}) + w(\text{robot, designer}) \tag{4}$$

From the equation (4), this feasibility could include the three factors, "user," "robot" and "designer" that are the important factors to form the interaction design. Therefore, we expect that this feasibility factor could literally indicate whether the user and the robot could create the intimate interaction or not.

We hypothesized that it is possible to superpose the familiarity and the implementability according to the life-likeness. We could then acquire the concept diagram of the *feasibility* shown in Fig. 10, by the sum of the graphs in Fig. 8 and Fig. 9. From this figure, at first its feasibility values are getting increase around *life-likeness* = *P*, however, this value suddenly decreases, as if the familiarity value falls into the uncanny valley. In addition, even though the life-likeness value is getting close to 100% life-likeness, the feasibility values could not recover to the same level of life-likeness *P*.

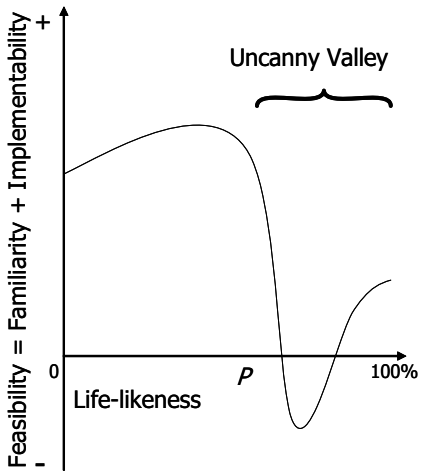


Figure 10. The hypothesized feasibility according to its life-likeness

Thus, it can be said that this figure shows that the robot with higher life-likeness would have the lower feasibilities to create an intimate relationship between the users and the robots, while the robot with lower life-likeness would have the higher feasibilities. This feasibility concept would support our hypothesis shown in Fig. 1, and accord the basic concept of the Mori's uncanny valley hypothesis.

However, there is one significant difference Mori's hypothesis: that is, once the feasibility value falls into the uncanny valley, this value never rises up to the peak of feasibility at life-likeness = *P* even though its life-likeness value is getting close to 100% life-likeness. We assumed that this phenomenon about the feasibility diagram would support again our hypothesis shown in Fig. 1.

As the consecutive studies based on the results of this study and our feasibility concept, we are planning to conduct the other experiments to reveal what is the most appropriate information according to the robots' appearance. For example, what is the appropriate information expressing from Mindstorms robot to inform its primitive attitudes? Are Starwars' R2D2 like behaviors appropriate? We expect that these consecutive studies would support our feasibility concept that can clarify the effective coupling between the appearance and its expressing information for realizing interactive robots readily and easily.

7. Conclusion

Various kinds of social robots have been developed to assist us with different tasks in our daily life. One of the most important issues in these studies is how to express the robot's primitive minds to a user for communication between them. This issue is strongly related to the robots' expressed information and its appearance. However few studies have investigated the relationship between these. Most studies applied human-like or animal-like appearance in the robots.

In this chapter, we proposed design policy of robot's expression of its primitive minds, SE4PM: Simple Expression for Primitive Mind, that means that a designer should design simple information with a simple appearance to express robot's primitive minds. To realize expression based on SE4PM, we designed mobile robot-like robot, MindStorms, with simple beep sound. We conducted a psychological experiment to clarify effectiveness of SE4PM by using AIBO entertainment robots with likely behaviors and MindStorms with beep sounds as simple expression. The results of our experiment supported SE4PM. Based on these results, we are able to create a design policy for simple and effective robots to interact with users. Eventually we discussed various properties of SE4PM and the relationship between feasibility (familiarity, implementability) and life-likeness based on Mori's uncanny valley.

8. Acknowledgements

This research has been supported by Grant-in-Aid for Exploratory Research (No. 18650034) from The Ministry of Education, Culture, Sports, Science and Technology, Japan and also NII Collaboration Grants (No. E-4) from National Institute of Informatics, Japan.

9. References

Baron-Cohen, S. (1995). *Mindblindness*, MIT Press, Cambridge, MA, USA.

Breazeal, C. & Velasquez, J. (1998). Toward teaching a robaot 'infant' using emotive communication acts. *In Proceedings of 1998 Simulation of Adaptive Behavior, Workshop on Socially Situated Intelligence*, pp. 25-40, Zurich, August.

Duffy, B. R. (2003) Anthropomorphism and the social robot. *Robotics and Autonomous Systems*, 42, 177-190

Ishiguro, H.; Ono, T.; Imai, M., Maeda, Kanda, T. & Nakatsu, R. (2001) Robovie: an interactive humanoid robot. *International Journal of Industrial Robot*, 28, 6, 498-503

Kanda, T.; Miyashita, T.; Osada, T.; Haikawa, Y. & Ishiguro, H. (2005) Analysis of humanoid appearances in human-robot interaction. *In Proceedings of 2005 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 62-69, Edmonton, August.

Kobayashi, K. & Yamada, S. (2005) Informing a user of robot's mind by motion. In *Proceedings of Third International Conference on Computational Intelligence, Robotics and Autonomous Systems*, SS4B-3, Singapore, December.

Komatsu. T. (2006) Audio subtle expressions affecting user's perceptions. *In Proceedings of 2006 International Conference on Intelligent User Interface*, pp. 306-308, San Diego, January.

Komatsu, T. & Yamada, S. (2007): Effects of robotic agents' appearances on users' interpretations of the agents' attitudes: towards an expansion of uncanny valley assumption, *In Proceedings of the 16th International Workshop on Robot and Human Interactive Communication*, to appear, Jeju Island, September.

Liu. K. & Picard, W. R. (2003) Subtle expressivity in a robotic computer. *In Proceedings of the CHI-2003 Workshop on Subtle Expressivity for Characters and Robots*, pp. 5-9, Florida, April.

Matsumoto, N.; Fujii, H.; Goan, M. & Okada, M. (2005) Minimal design strategy for embodied communication agents. *In The 14th IEEE International Workshop on Robot-Human Interaction*, pp. 335-340, Nashville, August.

Mori, M. (1970): Bukimi no tani (The Uncanny Valley). *Energy*, 7, 4, 33–35. (Originally in Japanese)

Mori, M. (1982) *Buddha in the Robot*. Carles E. Tuttle Co.

Morishima, S.; Iwasawa, S.; Sakaguchi, T.; Kawakami, F. & Ando, M. (1995). Better face communication. *In Visual Proceedings of SIGGRAPH'95 Interactive Communities*, page 117.

Okada, M.; Sakamoto, S. & Suzuki, N. (2000) Muu: Artificial creatures as an embodied interface. *In Proceedings of 27th International Conference on Computer Graphics and Interactive Techniques (SIGGRAPH 2000)*, page 91, New Orleans, July.

Ono, T. & Imai, M. (2000) Reading a robot's mind: A model of utterance understanding based on the theory of mind mechanism. *In Proceedings of the Seventeenth National Conference on Artificial Intelligence*, pp. 142-148, Austin, July.

Reeves, C. B. & Nass, C. (1996). *The Media Equation: How people treat computers, television, and new media like real people and places*. Cambridge, UK: Cambridge Press.

Suzuki, N.; Takeuchi, Y.; Ishii, K. & Okada, M. (2000) Talking eye: Autonomous creatures for augmented chatting, *Robotics and Autonomous Systems*, 31, 171-184, Elsevier.

Yamada, S. & Komatsu, T. (2006): Designing simple and effective expression of robot's primitive minds to a human, *In Proceedings of the 2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp.2614-2619, Beijing, October.

28

Hand Posture Segmentation, Recognition and Application for Human-Robot Interaction

Xiaoming Yin and Ming Xie
Singapore Institute of Manufacturing Technology, Nanyang Technological University
Singapore

1. Introduction

Human hand gestures provide the most important mean for non-verbal interaction among people. They range from simple manipulative gestures that are used to point at and move objects around to more complex communicative ones that express our feelings and allow us to communicate with others. Migrating the natural means that human employ to communicate with each other such as gestures, into Human-Computer Interaction (HCI) has been a long-term attempt. Numerous approaches have been applied to interpret hand gestures for HCI. In those approaches, two main categories of hand gesture models are used. The first group of models is based on appearance of the hand in the visual images. Gestures are modeled by relating the appearance of any gesture to that of the set of predefined template gestures [Pavlovic et al., 1996] [Ahmad et al., 1997]. Appearance-based approaches are simple and easy to implement in real time, but their application is limited to the recognition of a finite amount of hand gestures and they are mostly applicable to the communicative gestures.

The second group uses 3D hand models. 3D hand models offer a way to model hand gestures more elaborately. They are well suitable for modeling of both manipulative and communicative gestures. Several techniques have been developed in order to capture 3D hand gestures. Among those, glove-based devices are used to directly measure joint angles and spatial positions of the hand. Unfortunately, such devices remain insufficiently precise, too expensive and cumbersome, preventing the user from executing natural movements and interacting with the computer intuitively and efficiently. The awkwardness in using gloves and other devices can be overcome by using vision-based interaction techniques. These approaches suggest using a set of video cameras and computer vision techniques to reconstruct hand gestures [Lee and Kunii, 1995] [Lathuiliere and Herve, 2000]. Vision-based approaches are gaining more interest with the advantages of being intuitive, device-independent and non-contact.

Gesture-based interaction was firstly proposed by M. W. Krueger as a new form of human-computer interaction in the middle of the seventies [Krueger, 1991], and there has been a growing interest in it recently. As a special case of human-computer interaction, human-robot interaction is imposed by several constraints [Triesch and Malsburg, 1998]: the background is complex and dynamic; the lighting condition is variable; the shape of the human hand is deformable; the implementation is required to be executed in real time and

the system is expected to be user and device independent. Numerous techniques on gesture-based interaction have been proposed, but hardly any published work fulfills all the requirements stated above.

R. Kjeldsen and J. Render [Kjeldsen and Render, 1996b] presented a realtime gesture system which is used in place of the mouse to move and resize windows. In this system, the hand is segmented from the background using skin color and the hand's pose is classified using a neural net. A drawback of the system is that its hand tracking has to be specifically adapted for each user. The Perseus system developed by R. E. Kahn [Kahn et al., 1996] was used to recognize the pointing gesture. In the system, a variety of features, such as intensity, edge, motion, disparity and color have been used for gesture recognition. This system is implemented only in a restricted indoor environment. In the gesture-based human-robot interaction system proposed by J. Triesch and C. Ven Der Malsburg [Triesch and Malsburg, 1998], the combination of motion, color and stereo cues was used to track and locate the human hand, and the hand posture recognition was based on elastic graph matching. This system is person independent and can work in the presence of complex backgrounds in real time. But it is prone to noise and sensitive to the change of the illumination because its skin color detection is based on a defined prototypical skin color point in the HS plane.

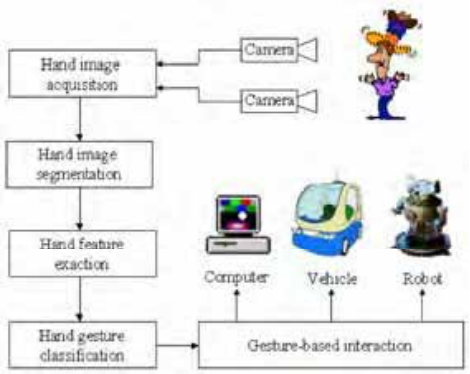


Figure 1. Process of hand gesture recognition

Usually the classical image processing pipeline as shown in Fig. 1 is used for hand gesture recognition. During the step of hand image acquisition, the pictures taken by the camera, in which a hand is to be seen, are digitized and prepared for further processing. It is usually automatically accomplished by a frame grabber. At the step of hand image segmentation, those areas in the picture, which represent the hand, are separated from the background. The aim of the step of hand feature extraction is to derive the smallest possible amount of features out of the segmented hand region, in order to differentiate the different given gestures. The last step is hand gesture classification, whereby the type of gesture shown in the picture are defined on the basis of extracted characteristics.

In this chapter, we present a novel hand posture recognition system. According to the process of hand gesture recognition, we first present a new color segmentation algorithm developed based on RCE neural network for hand image segmentation. Then we extract the topological features of the hand from the binary image of the segmented hand region. Based

on these features, we proposed a new method for accurate recognition of 2D hand postures. We also propose to use the stereo vision and 3D reconstruction techniques to recover 3D hand postures, and give a new approach to estimate the epipolar geometry between two uncalibrated hand images. Finally, we demonstrate the application of our hand gesture recognition system to human-robot interaction.

2. Hand Image Segmentation

Hand image segmentation separates the hand image from the background. It is the most important step in every hand gesture recognition system. All subsequent steps heavily rely on the quality of the segmentation. Two types of cues, color cues and motion cues, are often applied for hand image segmentation [Pavlovic et al., 1997]. Motion cues are used in conjunction with certain assumptions [Freeman and Weissman, 1995] [Maggioni, 1995]. For example, the gesturer is stationary with respect to the background that is also stationary. Such assumption restraints its application on occasion when the background is not stationary, which is the usual case for service robots. The characteristic color of human skin makes color a stable basis for skin segmentation [Quek et al., 1995] [Kjeldsen and Render, 1996a]. In this section, a novel color segmentation approach based on RCE neural network is presented for hand segmentation.

2.1 Skin Color Modeling

Color segmentation techniques rely on not only the segmentation algorithms, but also the color spaces used. RGB, HSI, and L*a*b* are the most commonly used color spaces in computer vision, and have all been applied in numerous proposed color segmentation techniques. After exploring the algorithm in these three color spaces respectively, we found L*a*b* color space is the most suitable for our hand segmentation algorithm.

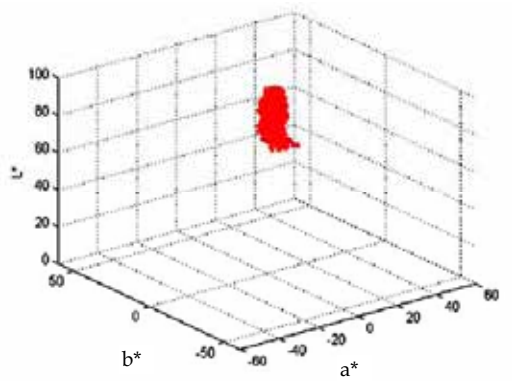


Figure 2. Skin color distribution in L*a*b* color space

L*a*b* color space is the uniform color space defined by the CIE (Commission International de 1'Eclairage) in 1979. It maps equal Euclidean distance in the color space to equal perceived color difference. The transformation from RGB to L*a*b* color space is defined as follows [Kasson and Plouffe, 1992]:

$$L^* = \begin{cases} 116\left(\frac{Y}{Y_n}\right)^{\frac{1}{3}} - 16 & \text{if } \frac{Y}{Y_n} > 0.008856 \\ 903.3\left(\frac{Y}{Y_n}\right) - 16 & \text{if } \frac{Y}{Y_n} \leq 0.008856 \end{cases} \quad (1)$$

$$a^* = 500\left[f\left(\frac{X}{X_n}\right) - f\left(\frac{Y}{Y_n}\right)\right] \quad (2)$$

$$b^* = 200\left[f\left(\frac{Y}{Y_n}\right) - f\left(\frac{Z}{Z_n}\right)\right] \quad (3)$$

where

$$f(t) = \begin{cases} t^{\frac{1}{3}} & \text{if } t > 0.008856 \\ 7.787 * t + 16/116 & \text{if } t \leq 0.008856 \end{cases} \quad (4)$$

X, Y and *Z* are tristimulus values of the specimen, and calculated from the values of *R, G* and *B* as follows:

$$X = 0.607R + 0.174G + 0.201B \quad (5)$$

$$Y = 0.299R + 0.587G + 0.114B \quad (6)$$

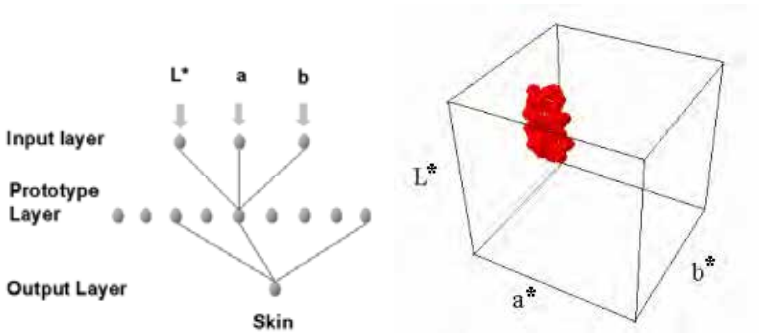
$$Z = 0.000R + 0.066G + 1.117B \quad (7)$$

X_n, Y_n and Z_n are tristimulus values of a perfect reflecting diffuser, which are selected to be 237.448, 244.073 and 283.478 respectively.

A common belief is that different people have different skin colors, but some studies show that such a difference lies largely in intensity than color itself [Yang et al., 1998] [Jones and Rehg, 1999]. We quantitatively investigated the skin color distribution of different human hands under different lighting conditions. It is found that skin colors cluster in a small region in the L*a*b* color space and have a translation along the lightness axis with the change of lighting conditions, as shown in Fig. 2.

Skin colors cluster in a specific small region in the color space, but the shape of the skin color distribution region is complicated and irregular. Common color segmentation techniques based on histogram are not effective enough to segment hand images from complex and dynamic backgrounds due to the difficulty of threshold selection. In our work, a new color segmentation algorithm based on RCE neural network has been developed.

RCE neural network was designed as a general-purpose, adaptive pattern classification engine [Reilly et al., 1982]. It consists of three layers of neuron cells, with a full set of connections between the first and second layers, and a partial set of connections between the second and third layers. Fig. 3(a) shows the network structure used for hand segmentation. Three cells on the input layer are designed to represent the L*a*b* color values of a pixel in the image. The middle layer cells are called prototype cells, and each cell contains color information of an example of the skin color class which occurred in the training data. The cell on the output layer corresponds to the skin color class.



a) b)

Figure 3. (a) Architecture of RCE neural network for hand segmentation, (b) Distribution region of skin colors in L*a*b* color space

2.2 Hand Segmentation

During training procedure, the RCE network allocates the positions of prototype cells and modifies the sizes of their corresponding spherical influence fields, so as to cover arbitrarily complex distribution region of skin colors in the color space. Fig 3(b) shows the distribution region of skin colors constructed by skin color prototype cells and their spherical influence fields in the L*a*b* color space. During running, the RCE network responds to input color signals in the fast response mode. If an input color signal falls into the distribution region of skin colors, this input color signal is classified into the skin color class, and the pixel represented by this color signal is identified as skin texture in the image.

During running, the RCE network identifies all the skin-tone pixels in the image. There are occasions that other skin-tone objects such as faces are segmented, or some non-skin pixels are falsely detected due to the effects of lighting conditions. We assume the hand is the largest skin-tone object in the image, and use the technique of grouping by connectivity of primitive pixels to further identify the region of the hand. With abundant skin color prototype cells together with their different spherical influence fields, the RCE network is capable of accurately characterizing the distribution region of skin colors in the color space and efficiently segment various hand images under variable lighting conditions from complex backgrounds after having been trained properly. Fig. 4 shows some segmentation results, in that the hand regions are separated perfectly from the complex backgrounds. The RCE neural network based hand image segmentation algorithm is described in more detail in our paper [Yin et al., 2001].

3. 2D Hand Posture Recognition

Hand segmentation is followed by feature extraction. Contour is the commonly used feature for accurate recognition of hand postures, and can be extracted easily from the silhouette of the segmented hand region. Segen and Kumar [Segen and Kumar, 1998] extracted the points along the boundary where the curvature reaches a local extremum as 'local features', and used those features that are labeled "peaks" or 'valleys' to classify hand postures. However, if the boundary is not smooth and continuous, it is difficult to identify peaks and valleys correctly.

In our study, we found it is difficult to extract the smooth and continuous contour of the hand because the segmented hand region is irregular, especially when the RCE neural network is not trained sufficiently. The topological features of the hand, such as the number and positions of fingers, are other distinctive features of hand postures. In this section, we present a new method for accurate recognition of hand postures, which extract topological features of the hand from the silhouette of the segmented hand region, and recognize hand postures on the basis of the analysis of these features.

3.1 Feature Extraction

In order to find the number and positions of fingers, the edge points of fingers are the most useful features. We extract these points using the following proposed algorithm:

1. Calculate the mass center of the hand from the binary image of the segmented hand region, in that pixel value 0 represents the background and 1 represents the hand image;
2. Draw the search circle with the radius r at the position of the center of mass;
3. Find all the points $\mathbf{E} = \{P_i, i = 0,1,2,..., n\}$ that have the transition either from pixel value 0 to 1, or 1 to 0 along the circle;
4. Delete P_i and P_{i-1}, if the distance between two conjoint points $D = |P_i P_{i-1}| <$ threshold λ_d;
5. Increment the radius r and iterate Step 2 to 4, until $r > 1/2$ (the width of the hand region).

Figure 4. Hand segmentation results

The purpose of Step 4 is to remove the falsely detected edge points resulted from imperfect segmentation. This step can removal most of falsely detected edge points. However, there are still occasions that one finger is divided into several branches because there are big holes in the image, or several fingers are merged into one branch because these fingers are too close. So we define the branch as follows:

Definition 3.1 *The branch is the segment between* P_{i-1} (0,1) *and* P_i (l,0). *Where* P_{i-1} (0,1) *and* P_i(l,0) *are two conjoint feature points detected on the search circle.* P_{i-1} (0,1) *has the transition from pixel value 0 to 1, and* P_i (l,0) *has the transition from 1 to 0.*

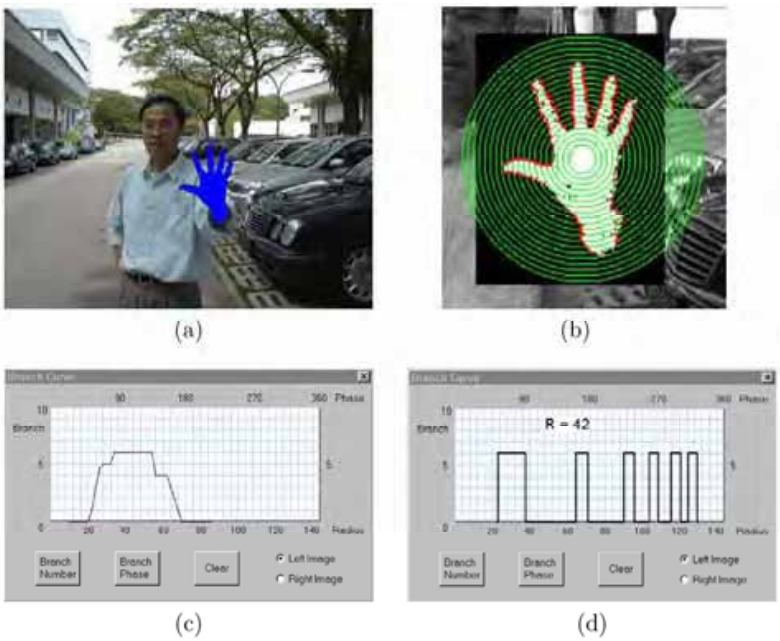


Figure 5. (a) segmented hand image, (b) Feature points extracted from the binary image of the segmented hand region, (c) Plot of branch number of the hand posture vs the radius of the search circle, (d) Plot of branch phase of the hand posture on the selected search circle

A branch indicates the possible presence of a finger. Then the extracted feature points accurately characterize the edge points of branches of the hand, including fingers and arm. Fig. 5 (a) shows a segmented hand image. Fig. 5(b) shows the part of Fig. 5(a) with the scale of 200%, in that the green circles represent the search circles and the red points represent the extracted feature points.

For each branch, two edge points can be found on the search circle, so half of the feature points found on the search circle just indicate the branch number of the hand posture. But the feature points on the different search circles are varied, how to determine the correct branch number is critical. In our method, we define the following function to determine the possibility p_i of each branch number:

$$p_i = a_i * \frac{C_i}{N}, i = 0, 1, 2, ..., 6 \tag{8}$$

Where C_i is the number of the search circles on that there are i branches; N is the total number of the search circles; a_i is the weight coefficient. We have $a_1 < a_2 < ... < a_i < ... < a_6$, because the number of the branches may decrease when the search circle is beyond the thumb or little finger. Then the branch number with the biggest possibility is selected as the most possible branch number *BN*.

In practice, the branch number *BN* can also be determined as follows:

1. Find all the branch numbers **K** (a set) whose occurrences are bigger than a threshold λ_n.
2. Choose the biggest one as the branch number *BN* among the numbers in **K**,

The biggest number in **K**, but not the number with the most occurrence, is selected as *BN*, because the biggest number may not have the most occurrence if there are some search circles beyond the fingers. But when its occurrence is bigger than the threshold, it should be the most possible branch number. For example, Fig. 5(c) shows the relationship between the branch number and the radius of the search circle. In this case, branch number 5 occurs 7 times, and 0 occurs 15 times. However, we select 5 but not 0 as *BN*. This method is easier to implement, and is very effective and reliable with the threshold λ_n selected to be 6 in our implementation.

After the branch number *BN* is determined, the branch phase can be obtained easily. Here we define the branch phase as follows:

Definition 3.2 *The branch phase is the positions of the detected branches on the search circle, described by angle.*

In our method, we selected the middle one of the search circles, on which there are *BN* branches, to obtain the branch phase. Fig. 5 (d) shows the radius of the selected search circle, and the branch phase on this circle.

Some morphological operations, such as dilation and erosion, are helpful for improvement of the binary image of the segmented hand region, but the branch number and phase obtained from the improved image are the same as those obtained from the original one. It indicates that our feature extraction algorithm has good robustness to noise, and can extract the correct branch number and phase reliably from the segmented hand image even though the segmentation is not very good.

3.2 Posture Recognition

After the branch phase is determined, the width of each branch *BWi* can be obtained easily from the branch phase. In most cases, the widest branch should be the arm. We use it as the base branch *BQ*. Then the distance from other branch *B^* to *BQ* can be calculated, that is just the distance between the finger and the arm *BD^*. Using these aforementioned parameters: the branch number *BN*, the width of the branch *BWi*, the distance between the finger and the arm *BD^*, the hand posture can be recognized accurately.

Although these parameters are all very simple and easy to estimate in real time, they are distinctive enough to differentiate those hand postures defined explicitly. In addition, the recognition algorithm also possesses the properties of rotational invariance and user independence because the topological features of human hands are quite similar and stable.

The postures shown in Fig. 6 all have distinctive features and are easy to recognize. We have used them for gesture-based robot programming and human-robot interaction of a real humanoid robot. The classification criterion of these postures is shown in Fig. 7. Preliminary experiments were conducted with users of different age, gender and skin color. The robot successfully recognized postures with the accuracy of more than 95% after the RCE network was trained properly.

The recognition accuracy may decrease in the case that the user or lighting condition changes too much, because the previous training of the RCE network becomes insufficient. But this problem can be solved easily by selecting parts of undetected hand sections as the training data using the mouse, and incrementally performing the online training. There is no need to re-present the entire training set to the network. In addition, the proposed posture recognition algorithm is only invariant to the hand rotation on the palm plane. If the hand is rotated more than 10 degree on the plane perpendicular to the palm, the posture recognition may be failed. The algorithms for topological feature extraction and hand posture recognition are described in more detail in our paper [Yin and Xie, 2007].

4. 3D Hand Posture Reconstruction

All of the 3D hand models employed so far use 3D kinematic models of the hand. Two sets of parameters are used in such models: angular (joint angles) and linear (phalange lengths and palm dimensions). However, The estimation of these kinematic parameters is a complex and cumbersome task because the human hand is an articulated structure with more than 27 degree of freedom. In this section, we propose to infer 3D information of the hand from the images taken from different viewpoints and reconstruct hand gestures using 3D reconstruction techniques.

4.1 Find robust matches

There are two approaches that can be used for the reconstruction of 3D vision models of hand postures. The first is to use calibrated stereo cameras, and the second is to use uncalibrated cameras. Camera calibration requires expensive calibration apparatus and elaborate procedures, and is only valid for the space near the position of the calibration object. Furthermore, the variation of focal lengths or relative positions of cameras will cause the previous calibration invalid. These drawbacks make camera calibration not feasible for gesture-based interaction, especially for human-robot interaction. Because service robots usually operate in dynamic and unstructured environments and their cameras need to be adjusted to track human hands frequently.

With uncalibrated stereo, there is an equivalence to the epipolar geometry which is presented by the fundamental matrix [Luong and Faugeras, 1996]. We have proposed a new method to estimate the fundamental matrix from uncalibrated stereo hand images. The proposed method consists of the following major steps: extracting points of interest; matching a set of at least 8 points; recovering the fundamental matrix.

In most approaches reported in the literature, high curvature points are extracted as points of interest. In our method, we use the edge points of the extended fingers, which are similar to those described in Section 3, as points of interest, and find robust matches from these points.

Matching different images of a single scene remains one of the bottlenecks in computer vision. A large amount of work has been carried out during the last decades, but the results are not satisfactory. The numerous algorithms for image matching that have been proposed can roughly be classified into two categories: correlation-based matching and feature-based matching. Correlation-based methods are not robust for hand image matching due to the ambiguity caused by the similar color of the hand. The topological features of the hand, such as the number and positions of the extended fingers that are described in the above section, are more distinct and stable in stereo hand images, only if the distance and angles between two cameras are not too big. In our method, we propose to take advantage of the topological features of the hand to establish robust correspondences between two perspective hand images.

We first detect fingertips by searching the furthest edge points from the mass center of the hand in the range between $B_i + BW_i$ and $B_i - BW_i$. Here B_i is the branch phase and BW_i is the branch width. The fingertips of two perspective hand images are found using this method , respectively. Simultaneously, their correspondences are established by the order of the finger. For example, the fingertip of B_1^r in the right image corresponds to the fingertip of B_1^l in the left image.

Then, we define the center of the palm as the point whose distance to the closest region boundary is maximum, and use the morphological erosion operation to find it. The procedure is as follows:

1. Apply dilation operation once to the segmented hand region.
2. Apply erosion operations until the area of the region becomes small enough. As a result, a small region at the center of the palm is obtained.
3. Calculate the center of mass of the resulting region as the center of the palm.

The purpose of the first step is to remove little holes in the imperfectly segmented hand image. These little holes can affect the result of erosion greatly. Fig. 8 shows the procedure to find the center of the palm by erosion operations.

The palm centers of two hand images are found by this method, respectively. In most case, they should correspond to each other because the shapes of the hand in two perspective images are almost the same under the assumption that the distance and angle between two cameras are small. However, because the corresponding centers of the palm are very critical for finding matches in our approach, we further use the following procedure to evaluate the accuracy of correspondence and determine the corresponding palm centers more robustly:

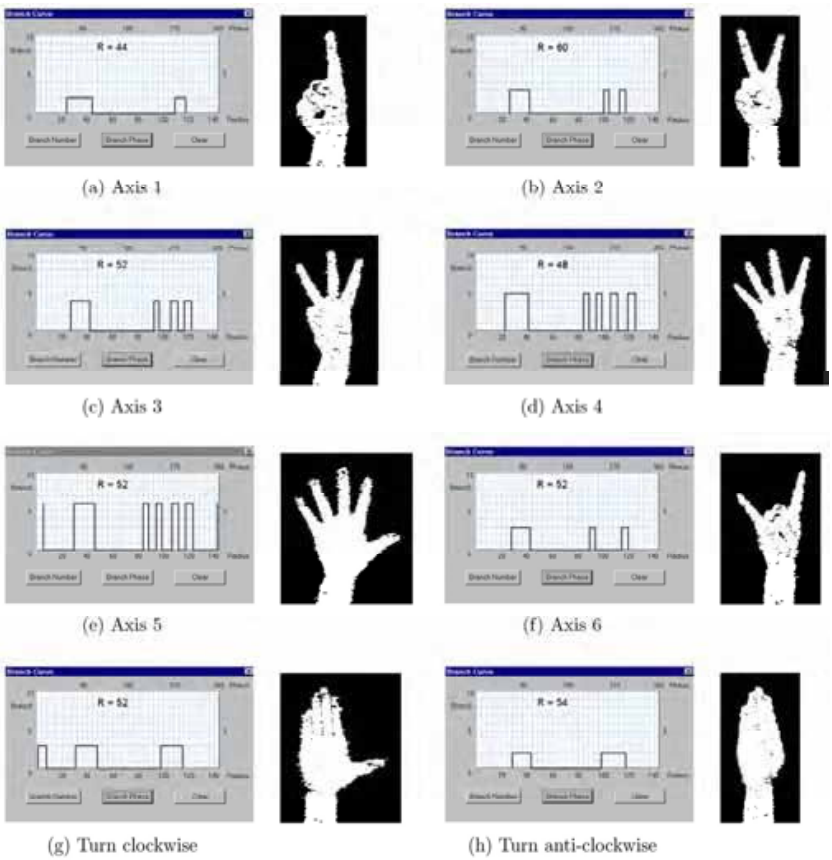
1. Find the fingertips T_i^l, T_i^r and the palm centers C^l, C^r for the left image and right image, respectively.
2. Calculate $d = \sum_1^{BN-1} |C^l T_i^l - C^r T_i^r|$. Here, $C^l T_i^l$ is the distance between the palm center and a fingertip in the left image, and $C^r T_i^r$ is that in the right image. $(BN - 1)$ represents the number of the extended fingers.
3. Take C^l and C^r as the corresponding palm centers if $d < \lambda_c$. λ_c is the threshold and is set to 2 pixels in our implementation.

The evaluation procedure above is used because we can assume $C^l T_i^l$ is equal to $C^r T_i^r$ according to projective invariance. If $d > \lambda_c$, we take the point, whose distance to each fingertip in the right image is the same as the distance between the palm center and each fingertip in the left image, as new C^r corresponding to C^l. Such a point is determined in theory by calculating the intersection of all the circles that are drawn in the right hand image

with the radius $C^l T_i^l$ at the positions of T_i^r. Referring to the coordinates of this point as x and y, they satisfy the following equation:

$$\min_{(x,\ y)} \sum_{i=1}^{BN-1} (x - (T_i^r)_x)^2 + (y - (T_i^r)_y)^2 - (C^l T_i^l)^2 \tag{9}$$

where, $(T_i^r)_x$ and $(T_i^r)_y$ denote the coordinates of a fingertip in the right image. Such an equation is difficult to be solved by mathematical methods. In practice, we can determine an intersection within the right hand region for every two circles, then calculate the mass center of all the intersections as new C^r.



(a) Axis 1 (b) Axis 2

(c) Axis 3 (d) Axis 4

(e) Axis 5 (f) Axis 6

(g) Turn clockwise (h) Turn anti-clockwise

Figure 6. Hand postures used for robot programming and human-robot interaction

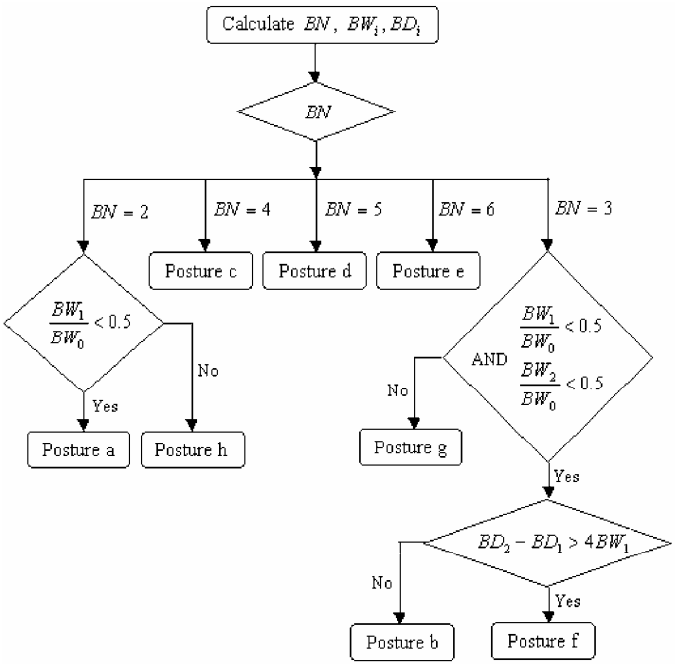


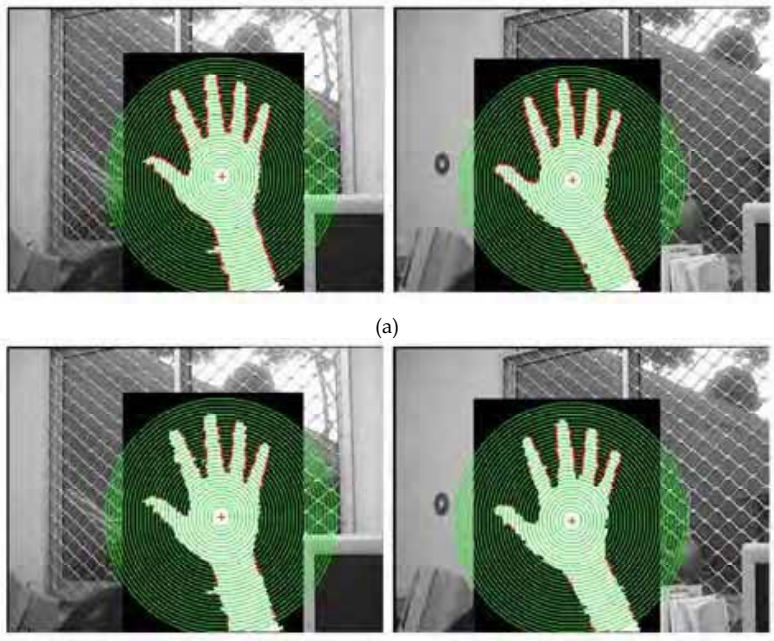
Figure 7. Classsification criterion of hand postures

Figure 8. Procedure for finding the center of the palm by the morphological erosion operation

After the corresponding palm centers are determined, matches can be found by comparing the edge points on the i^{th}(i = 1, ...,m) search circle of the left image with those of the right image. The criterion is as follows:

1. Calculate $d_a = |P^l_{i,j}P^l_{i,j-1} - P^r_{i,j}P^r_{i,j-1}|$ and $d_b = |P^l_{i,j}P^l_{i,j+1} - P^r_{i,j}P^r_{i,j+1}|$. Here, $P^l_{i,j}$ is the j^{th} edge point on the i^{th} search circle in the left image, and $P^r_{i,j}$ is that in the right image. $P^l_{i,j}P^l_{i,j-1}$ is the distance between the edge points $P^l_{i,j}$ and $P^l_{i,j-1}$
2. Calculate $d = (d_a + d_b)/2$. If d < threshold λ_m, $P^l_{i,j}$ and $P^r_{i,j}$ are taken as a pair of matches. λ_m is set to 2 pixels in our implementation.

The basic idea underlying this matching algorithm is to extract the edge points , whose distances to its previous and following points as well as to the center of the palm are almost identical in two images, as matches. The algorithm works very well under the situation that the distance and angle between two cameras are small. In Fig. 9, (a) shows the edge points extracted from the segmented hand regions of two perspective images and (b) shows the matches extracted from these edge points. The green circles represent the search circles and the red points are the extracted edge points.



(a)

(b)

Figure 9. (a) Edge points extractedfrom stereo hand images, (b) Matches extracted from edge points

4.2 Estimate the Fundamental Matrix

Using the set of matched points established in the previous step, the epipolar geometry between two uncali-brated hand images can be recovered. It contains all geometric information that is necessary for establishing correspondences between two perspective images, from which 3D structure of an object can be inferred.

The epipolar geometry is the basic constraint which arises from the existence of two viewpoints [Faugeras, 1993]. Considering the case of two cameras, we have the following fundamental equation:

$$(\tilde{\mathbf{m}}^r)^T \mathbf{F} \tilde{\mathbf{m}}^l = 0 \tag{10}$$

where $\tilde{\mathbf{m}}^l = [u^l, v^l, 1]^T$ and $\tilde{\mathbf{m}}^r = [u^r, v^r, 1]^T$ are the homogeneous image coordinates of a 3D point in the left and right images, respectively. $\mathbf{F}$ is known as the *fundamental matrix*. Geometrically, $\mathbf{F}\tilde{\mathbf{m}}^l$ defines the epipolar line of a left image point $\mathbf{m}^l$ in the right image. Equation (10) says no more than that the correspondence in the right image of point m' lies on the corresponding epipolar line. Transposing equation (10) yields the symmetric relation from the right image to the left image.

$\mathbf{F}$ is of rank 2. Besides, it is defined up to a scalar factor. Therefore, a fundamental matrix has only seven degrees of freedom. That is, there are only 7 independent parameters among the 9 elements of the fundamental matrix. Various techniques have been reported in the literature for estimation of the fundamental matrix (see [Zhang, 1996] for a review). The classical method for computing the fundamental matrix from a set of 8 or more point matches is the 8-point algorithm introduced by Longuet-Higgins in [Longuet-Higgins, 1981]. This method is the linear criterion and has the advantage of simplicity of implementation. However, it is quite sensitive to noise. In order to recover the epipolar geometry as accurately as possible, we use a combination of techniques such as input data normalization, rank-2 constraint, linear criterion, nonlinear criterion as well as robust estimator to yield an optimal estimation of the fundamental matrix. The algorithm is as follows:

1. Normalize pixel coordinates of matches.
2. Initialize the weights ω_i = *1* and γ_i = 1 for all matches.
3. For a number of iterations:
   3.1. Weight the *i*th linear equation by multiplying it by $\omega_i\sqrt{\gamma_i}$.
   3.2. Estimate the fundamental matrix $\mathbf{F}$ using the linear least-squares algorithm.
   3.3. Impose the rank-2 constraint to the estimated $\mathbf{F}$ by the singular value decomposition.
   3.4. Calculate the residuals of matches $r_i = (\tilde{\mathbf{m}}_i^r)^T \mathbf{F} \tilde{\mathbf{m}}_i^l$.
   3.5. Calculate the nonlinear method weight:

$$\omega_i = \left(\frac{1}{(l_{i1}^l)^2 + (l_{i2}^l)^2} + \frac{1}{(l_{i1}^r)^2 + (l_{i2}^r)^2} \right)^{1/2} \tag{11}$$

   here $\mathbf{l}_i^r = \mathbf{F}\tilde{\mathbf{m}}_i^l = [l_{i1}^r, l_{i2}^r, l_{i3}^r]^T$ is the corresponding epipolar line of point $\mathbf{m}_i^l$ and $\mathbf{l}_i^l = \mathbf{F}^T\tilde{\mathbf{m}}_i^r = [l_{i1}^l, l_{i2}^l, l_{i3}^l]^T$ the corresponding epipolar line of point $\mathbf{m}_i^r$.
   3.6. Calculate the distances between matching points and the corresponding epipolar lines $d_i = \omega_i r_i$.
   3.7. Calculate the robust method weight:

$$\gamma_i = \begin{cases} 1 & |d_i| \le \sigma \\ \sigma/|d_i| & \sigma < |d_i| \le 3\sigma \\ 0 & |d_i| > 3\sigma \end{cases} \tag{12}$$

By combining several simple methods together, the proposed approach becomes more effective and robust, but still easily to be implemented.

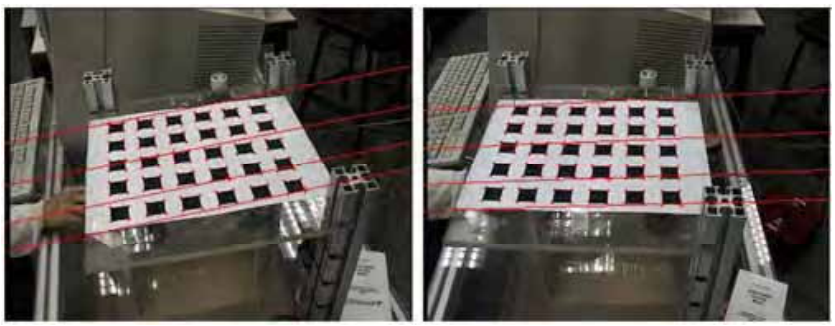
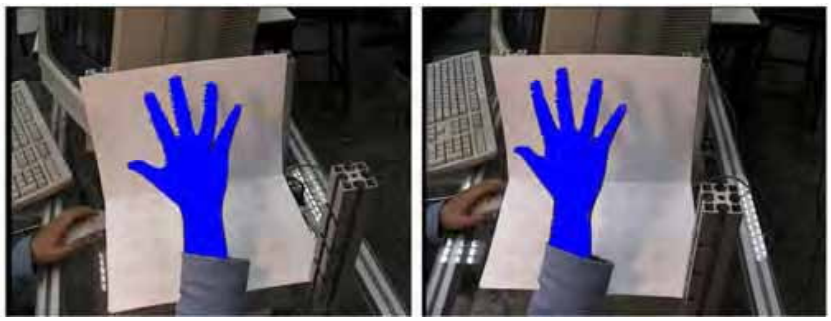
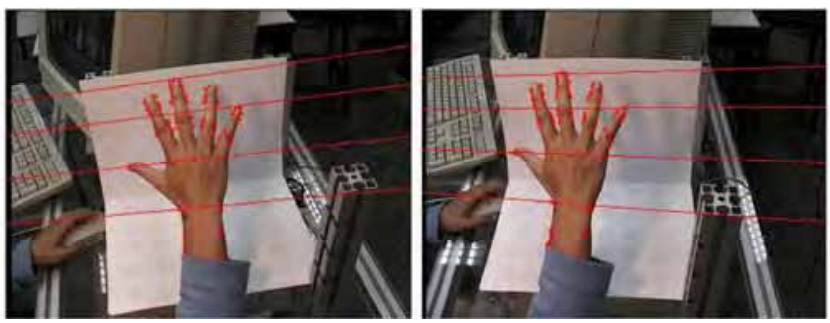


Figure 10. The epipolar geometry computed from the calibration matrices



a)



b)

Figure 11. (a) Segmentation results of one pair of calibrated hand images, (b) Extracted matches and the estimated epipolar geometry

Our experimental results demonstrate the performance of the proposed algorithm. We first use this algorithm to estimate the fundamental matrix between two calibrated cameras, and compare the obtained epipolar geometry with that computed from the calibration matrices

of the cameras. The epipolar geometry computed from the calibration matrices is shown in Fig. 10. It serves as a ground truth. Fig. 11 shows a pair of hand images taken by the calibrated cameras with the size of 384 x 288. In that, (a) shows the segmentation results of the hand images using the method presented in Section 2, and (b) shows the extracted corresponding points using the approach presented in Section 3 as well as the epipolar geometry estimated from these matches using the algorithm described in this section.

Sometimes, the matches extracted from the hand images may lie on a plane. This will cause degeneracy in the data, and affect the accuracy of the estimation of the fundamental matrix. We can take more hand images with the hand at different positions and use all the matches extracted from these images to get a more accurate estimation of the fundamental matrix. The epipolar geometry estimated using all the matches obtained from several hand images is shown in Fig. 12. The red solid lines represent the epipolar lines estimated from the extracted matches, and the green dash lines represent those computed from the calibration matrices. It can be observed the estimated epipolar geometry is very closed to the calibrated one.

Fig. 13 shows a pair of hand images taken by two uncalibrated cameras with the size of 384 x 288. In that, (a) shows the segmentation results of the hand images and (b) shows the extracted corresponding points as well as the epipolar geometry estimated from these matches. In order to avoid the problem of degeneracy, and obtain more accurate and robust estimation of the fundamental matrix, we take more than one pairs of hand images with the hand at different positions, and use all the matches found in these images to estimate the fundamental matrix. Fig. 14 shows another pair of images taken by the same cameras, where the epipolar geometry is estimated from all the matches obtained from several hand images. It can be observed that the estimated epipolar lines match the corresponding points well even though there is no point in this figure used for the estimation of the fundamental matrix. So at the beginning of hand gesture recognition, we can take several hand images with the hand at different positions, and use the matches extracted from these images to recovery the epipolar geometry of the uncalibrated cameras. Then the recovered epipolar geometry can be applied to match other hand images and reconstruct hand postures. If the parameters of the cameras change, the new fundamental matrix is easy to be estimated by taking some hand images again.

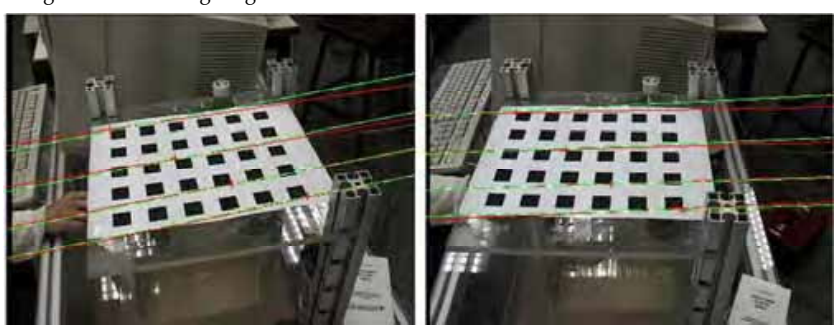


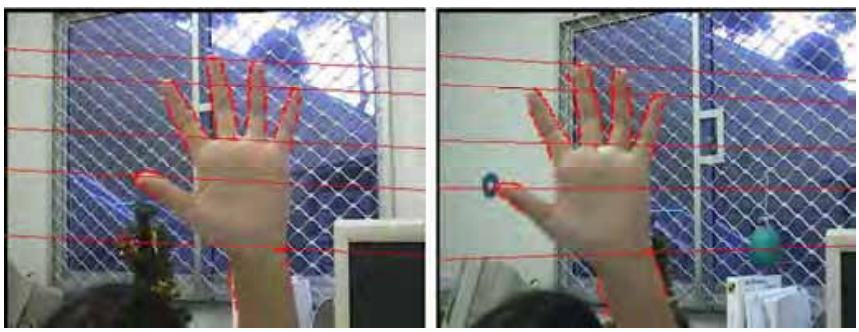
Figure 12. Comparison of the estimated epipolar geometry with the calibrated one

In [Zhang et al., 1995], Zhang proposed an approach to match images by exploiting the epipolar constraint. They extracted high curvature points as points of interest, and match

them using a classical correlation technique followed by a new fuzzy relaxation procedure. Then the fundamental matrix is estimated by using a robust method: the Least Median of Squares (Lmeds). Zhang provides a demo program to compute the epipolar geometry between two perspective images of a single scene using this method at his home page: http://www.inria.fr/robotvis /personnel/zzhang/zzhang-eng.html. We submitted the images in Figs. 13 and 14 to this program and obtain the results as shown in Figs. 15 and 16, where (a) shows the extracted correspondences which are marked by white crosses, and (b) shows the estimated epipolar geometry. It can be seen the epipolar lines are very far from the corresponding points on the hand.

The approach presented in this section can also be used for other practical applications. For example, at some occasions when the calibration apparatus is not available and the feature points of the scene, such as corners, are difficult to be extracted from the images, we can take advantage of our hands, and use the method presented above to derive the unknown epipolar geometry for the uncalibrated cameras. This method is described in more detail in our paper [Yin and Xie, 2003].

a)



b)

Figure 13. (a) Segmentation results of one pair of uncalibrated hand images, (b) Extracted matches and the estimated epipolar geometry

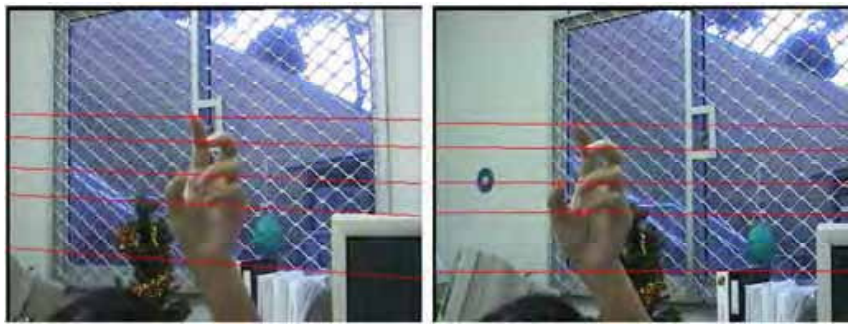


Figure 14. Application of the estimated epipolar geometry to one pair of uncalibrated hand images

a)

b)

Figure 15. (a) Extracted matches using the method proposed by Zhang from uncalibrated hand images shown in Fig. 13, (b) Estimated epipolar geometry from these matches

4.3 Reconstruct hand postures

After the epipolar geometry between two uncalibrated cameras are recovered, it can be applied to match other hand images and reconstruct 3D hand postures. Although stereo images taken by uncalibrated cameras allow reconstruction of 3D structure only up to a projective transformation, it is sufficient for hand gesture recognition, where the shape of the hand, not the scale, is important.

The epipolar geometry is the basic constraint which arises from the existence of two viewpoints. For a given point in one image, its corresponding point in the other image must lie on its epipolar line. This is known as the *epipolar constraint.* It establishes a mapping between points in the left image and lines in the right image and vice versa. So, if we determine the epipolar line $l^r_{m^l}$ in the right image for a point m^l in the left image, we can restrict the search for the match of m^l along $l^r_{m^l}$. The search for correspondences is thus reduced to a ID problem.

After the set of matching candidates $\mathbf{m}^r$ is obtained, the correct match of m^l in the right image, denoted by m^r, is further determined using correlation-based method. In correlation-based methods, the elements to match are image windows of fixed size, and the similarity criterion is a measure of correlation between windows in two images. The corresponding element is given by the window that maximizes the similarity criterion within a search region. For intensity images, the following cross-correlation is usually used [Faugeras, 1993]:

$$C(u^l,\ v^l,\ u^r,\ v^r) = \frac{1}{K\sigma^l(u^l,\ v^l)\sigma^r(u^r,\ v^r)} \sum_{i=-n}^{n}\sum_{j=-m}^{m} \left[I^l(u^l+i, v^l+j) - \overline{I^l(u^l, v^l)}\right]\left[I^r(u^r+i, v^r+j) - \overline{I^r(u^r, v^r)}\right] \tag{13}$$

with

$$K = (2n+1)(2m+1) \tag{14}$$

$$\overline{I^l(u^l,\ v^l)} = \frac{1}{(2n+1)(2m+1)} \sum_{i=-n}^{n}\sum_{j=-m}^{m} I^l(u^l+i,\ v^l+j) \tag{15}$$

$$\sigma^l(u^l,\ v^l) = \sqrt{\frac{\sum_{i=-n}^{n}\sum_{j=-m}^{m}\left[I^l(u^l+i,\ v^l+j) - \overline{I^l(u^l,\ v^l)}\right]^2}{(2n+1)(2m+1)}} \tag{16}$$

where, I^l and I^r are the intensity functions of the left and right images. $\overline{I^l(u^l,\ v^l)}$ and $\sigma^l(u^l,\ v^l)$ are the mean intensity and standard deviation of the left image at the point (u^l, v^l) in the window $(2n + 1) \times (2m + 1)$. $\overline{I^r(u^r,\ v^r)}$ and $\sigma^r(u^r,\ v^r)$ are similar to $\overline{I^l(u^l,\ v^l)}$ and $\sigma^l(u^l,\ v^l)$, respectively. The correlation C ranges from -1 for two correlation windows which are not similar at all, to 1 for two correlation windows which are identical. However, this cross-correlation method is unsuitable for color images, because in color images, a pixel is represented by a combination of three primary color components (*R* (red), *G* (green), *B* (blue)). One combination of *(R, G, B)* corresponds to only one physical color, and a same intensity value may correspond to a wide range of color combinations. In our method, we use the following color distance based similarity function to establish correspondences between two color hand images [Xie, 1997].

$$C(u^l,\ v^l,\ u^r,\ v^r) \quad = \quad 1 - \frac{1}{Ks} \sum_{i=-n}^{n} \sum_{j=-m}^{m} (A^2 + B^2 + C^2) \tag{17}$$

with

$$K \quad = \quad (2n+1)(2m+1) \tag{18}$$

$$s \quad = \quad 3 \cdot 255^2 \tag{19}$$

$$A \quad = \quad R^l(u^l + i,\ v^l + j) - R^r(u^r + i,\ v^r + j) \tag{20}$$

$$B \quad = \quad G^l(u^l + i,\ v^l + j) - G^r(u^r + i,\ v^r + j) \tag{21}$$

$$C \quad = \quad B^l(u^l + i,\ v^l + j) - B^r(u^r + i,\ v^r + j) \tag{22}$$

where, R^l, G^l and B^l are the color values of the left image corresponding to red, green and blue color components, respectively. R^r, G^r and B^r are those of the right image.

a)

b)

Figure 16. (a) Extracted matches using the method proposed by Zhang from uncalibrated hand images shown in Fig. 14, (b) Estimated epipolar geometry from these matches

The similarity function defined in Equation (17) varies in the range [0, 1]. Then stereo matching can be summarized as follows: Given a pixel $(u^l,\ v^l)$ in the left image, find a pixel $(\hat{u}^r,\ \hat{v}^r)$ in the right image which maximizes the similarity function in Equation (17):

$$C(u^l,\ v^l,\ \hat{u}^r,\ \hat{v}^r) = \max_{(u^r,\ v^r)\in W}\{C(u^l,\ v^l,\ u^r, v^r)\} \tag{23}$$

where, *W* denotes the searching area in the right image. In our implementation, the searching area is limited in the segmented hand region and on the epipolar line.
The computation of *C* is time consuming because each pixel involves three multiplications. In practice, a good approximation is to use the following similarity function.

$$C(u^l,\ v^l,\ u^r,\ v^r) = \frac{1}{3}(C_R + C_G + C_B) \tag{24}$$

where

$$C_R = 1 - \frac{\sum_{i=-n}^{n}\sum_{j=-m}^{m}|R^l(u^l+i,\ v^l+j) - R^r(u^r+i,\ v^r+j)|}{255\cdot(2n+1)(2m+1)} \tag{25}$$

$$C_G = 1 - \frac{\sum_{i=-n}^{n}\sum_{j=-m}^{m}|G^l(u^l+i,\ v^l+j) - G^r(u^r+i,\ v^r+j)|}{255\cdot(2n+1)(2m+1)} \tag{26}$$

$$C_B = 1 - \frac{\sum_{i=-n}^{n}\sum_{j=-m}^{m}|B^l(u^l+i,\ v^l+j) - B^r(u^r+i,\ v^r+j)|}{255\cdot(2n+1)(2m+1)} \tag{27}$$

The similarity function defined in Equation (24) also takes values in the range [0, 1].
As shown in Figure 17, for the points marked by red crosses in the left image, their matching candidates in the right image found by the technique described above are marked by red points. Figure 18 shows all detected corresponding points of the hand, and Figure 19 shows 4 views of the reconstructed 3D hand posture.

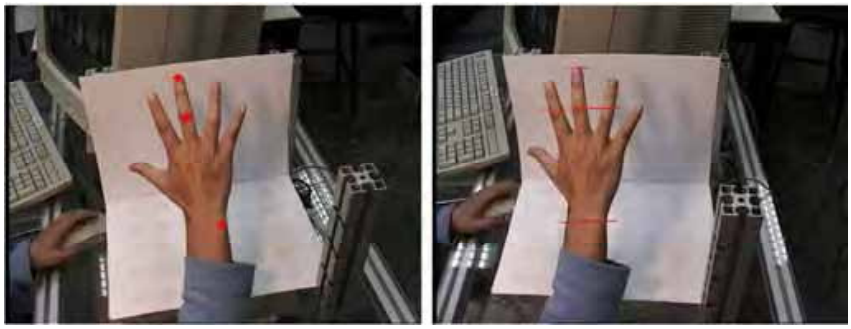


Figure 17. Find corresponding points in the right image which are marked by red points, for points in the left image which are marked by red crosses, using the color correlation and epipolar geometry

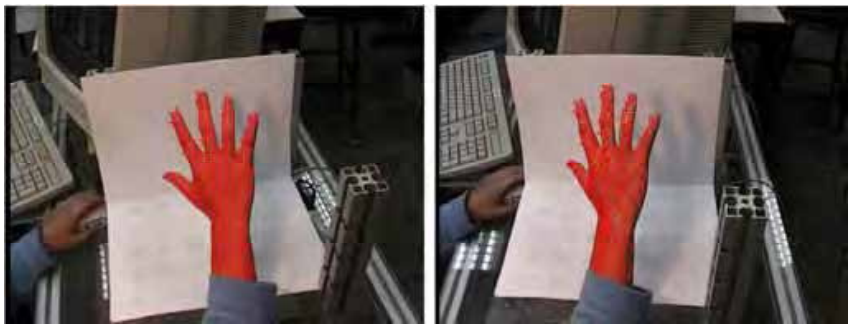
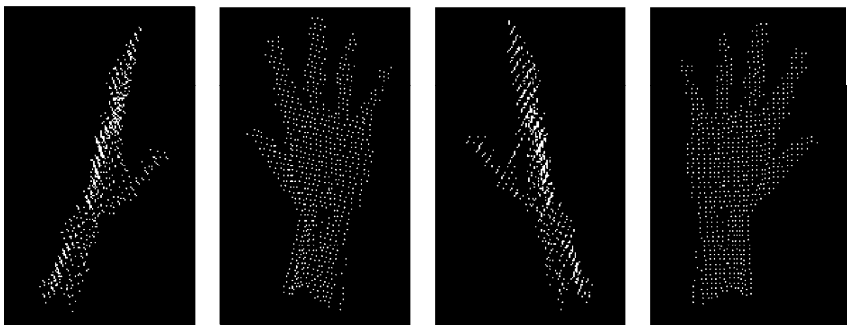


Figure 18. Detected corresponding points of the hand



(a) Right view (b) Front view (c) Left view (d) Back view

Figure 19. Different views of the reconstructed 3D hand posture

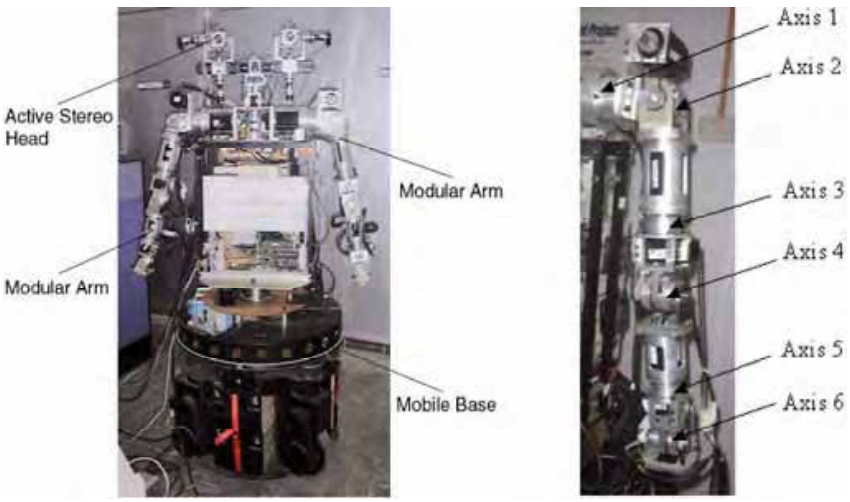
5. Gesture-Based Human-Robot Interaction

Our research on hand gesture recognition is a part of the project of Hybrid Service Robot System, in which we will integrate various technologies, such as real robot control, virtual robot simulation, human-robot interaction etc., to build a multi-modal and intelligent human-robot interface. Fig. 20(a) shows the human-alike service robot HARO-1 at our lab. It was designed and developed by ourselves, and mainly consists of an active stereo vision head on modular neck, two modular arms with active links, an omnidirectional mobile base, dextrous hands under development and the computer system. Each modular arm has 3 serially connected active links with 6 axes, as shown in 20 (b).

5.1 Gesture-Based Robot Programming

In order to carry out a useful task, the robot has to be programmed. Robot programming is the act of specifying actions or goals for the robot to perform or achieve. The usual methods of robot programming are based on the keyboard, mouse and teach-pendant [Sing and Ikeuchi, 1997]. However, service robots necessitate new programming techniques because they operate in everyday environment, and have to interact with people that are not

necessarily skilled in communicating with robots. Gesture-based programming offers a way to enable untrained users to instruct service robots easily and efficiently.



a) b)

Figure 20. (a) Humanoid service robot HARO-1; (b) Modular robot arm with 6 axes

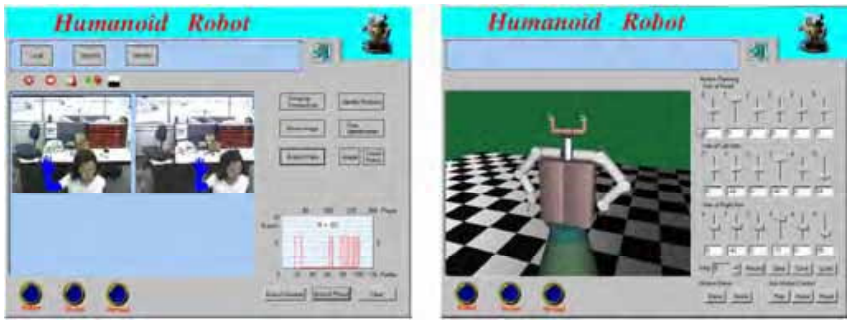
Based on our approach of 2D hand posture recognition, we have proposed a posture programming method for our service robot. In this method, we define task postures and corresponding motion postures respectively, and associate them during the training procedure, so that the robot will perform all the motions associated with a task if that task posture is presented to the robot by the user. Then, the user can interact with the robot and guide the behavior of the robot by using various task postures easily and efficiently.

The postures shown in Fig. 6 is used for both robot programming and human-robot interaction. In the programming mode, Postures *a to f* represent the six axes of the robot arm respectively, Posture *g* means 'turn clockwise', and Posture *h* means 'turn anti-clockwise'. We use them as motion gestures to control the movements of the six axes of either robot arm. Using these postures, we can guide the robot arm to do any motion, and record any motion sequence as a task.

In the interaction mode, these postures are redefined as task postures and associated with corresponding tasks. For example, some motion sequence is defined as Task 1, and is associated with Posture a. When Posture a is presented to the robot in the interaction stage, the robot will move its arm according to the predefined motion sequence. A task posture is easy to be associated with different motion sequences in different applications by programming using corresponding motion postures.

5.2 Gesture-Based Interaction System

Fig. 21 shows the Graphic User Interface (GUI) of the gesture-based interaction system implemented on robot HARO-1, in that (a) represents the Vision section of the interface, and (b) shows the virtual robot developed using Open GL.



a) b)

Figure 21. Graphic user interface of the robot HARO-1: (a) Posture recognition; (b) Virtual robot

As shown in Fig. 21 (a), live images with the size of 384x288 are captured through two CCD video cameras (EVID31, SONY) in the system. At the end of each video field the system processes the pair of images, and output the detected hand information. The processing is divided into two phases: hand tracking phase and posture recognition phase. At the beginning, we have to segment the whole image to locate the hand, because we have no any information about the position of the hand. After the initial search, we do not need to segment the whole image, but a smaller region surrounding the hand, since we can assume continuity of the position of the hand during the tracking. At the tracking phase, the hand is segmented using the approach described in Section 2 from a low resolution sampling of the image, and can be tracked reliably at 4-6Hz on a normal 450MHz PC.

The system also detects the motion features of the hand such as pauses during the tracking phase. Once a pause is confirmed, the system stops the tracking, crops a high resolution image tightly around the hand and performs a more accurate segmentation based on the same techniques. Then the topological features of the hand is extracted from the segmented hand image and the hand posture is classified based on the analysis of these features as described in Section 3. If the segmented hand image is recognized correctly as one of the postures defined in Fig. 6, the robot will perform motions associated with this posture. If the segmented image can not be recognized because of the presence of noises, the robot will not output any response. The time spent on the segmentation of the high resolution image is less than 1 second, and the whole recognition phase can be accomplished within 1.5 seconds. After the posture recognition phase is finished, the system continues to track the hand until another pause is detected.

6. Conclusions

Vision-based hand gesture recognition provide a more nature and powerful way for human-computer interaction. In the chapter, we present some new approaches for hand image segmentation, 2D hand posture recognition and 3D hand posture reconstruction. We segment hand images using the color segmentation approach which is based on the RCE neural network. Then we extract topological features of the hand from the binary image of the segmented hand region, and recognize 2D hand postures base on the analysis of these features. We also propose to use the stereo vision and 3D reconstruction techniques to recover 3D hand postures and present a new method to estimate the fundamental matrix from uncalibrated stereo hand images in this chapter. A human-robot interaction system has been developed to demonstrate the application of our hand posture recognition approaches.

7. References

Ahmad, T., Taylor, C. J., Lanitis, A., and Cootes, T. F. (1997). Tracking and recognizing hand gestures using statistical shape models. *Image and Vision Computing,* 15:345-352. [Ahmad et al., 1997]

Faugeras, O. (1993). *Three-dimensional computer vision: a geometric viewpoint.* MIT Press, Cambridge, Massachusetts. [Faugeras, 1993]

Freeman, W. T. and Weissman, C. D. (1995). Television control by hand gestures. In *Proceedings of International Workshop on Automatic Face and Gesture Recognition,* pages 179-183, Zurich, Switzerland. [Freeman and Weissman, 1995]

Jones, M. J. and Rehg, J. M. (1999). Statistical color models with application to skin detection. In *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition,* volume 1, pages 274-280, Fort Collins, CO. [Jones and Rehg, 1999]

Kahn, R. E., Swain, M. J., Prokopowicz, P. N., and Firby, R. J. (1996). Gesture recognition using the perseus architecture. In *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition,* pages 734—741, San Francisco. [Kahn et al., 1996]

Kasson, J. K. and Plouffe, W. (1992). A analysis of selected computer interchange color spaces. *ACM Transaction on Graphics,* 11(4):373-405. [Kasson and Plouffe, 1992]

Kjeldsen, R. and Render, J. (1996a). Finding skin in color images. In *Proceedings of International Conference on Automatic Face and Gesture Recognition,* pages 312—317, Killington, Vt. [Kjeldsen and Render, 1996a]

Kjeldsen, R. and Render, J. (1996b). Toward the use of gesture in traditional user interfaces. In *Proceedings of International Conference on Automatic Face and Gesture Recognition,* pages 151-156, Killington, Vt. [Kjeldsen and Render, 1996b]

Krueger, M. W. (1991). *Artificial Reality H.* Addison-Wesley. [Krueger, 1991]

Lathuiliere, F. and Herve, J.-Y. (2000). Visual hand posture tracking in a gripper guiding application. In *Proceedings of IEEE International Conference on Robotics and Automation,* pages 1688-1694, San Francisco, CA. [Lathuiliere and Herve, 2000]

Lee, J. and Kunii, T. L. (1995). Model-based analysis of hand posture. *IEEE Transactions on Computer Graphics and Application,* 15(5):77—86. [Lee and Kunii, 1995]

Longuet-Higgins, H. C. (1981). A computer algorithm for reconstructing a scene from two projections. *Nature,* 293:133-135. [Longuet-Higgins, 1981]

Luong, Q.-T. and Faugeras, O. D. (1996). The fundamental matrix: Theory, algorithms and stability analysis. *International Journal of Computer Vision,* 1(17):43—76. [Luong and Faugeras, 1996]

Maggioni, C. (1995). Gesturecomputer - new ways of operation a computer. In *Proceedings of International Workshop on Automatic Face and Gesture Recognition,* Zurich, Switzerland. [Maggioni, 1995]

Pavlovic, V. I., Sharma, R., and Huang, T. S. (1996). Gestural interface to a visual computing environment for molecular biologists. In *Proceedings of International Conference on Face and Gesture Recognition,* pages 30-35, Killington, Vt. [Pavlovic et al., 1996]

Pavlovic, V. L., Sharma, R., and Huang, T. S. (1997). Visual interpretation of hand gestures for human-computer interaction: A review. *IEEE Transactions on Pattern Analysis and Machine Intelligence,* 19(7):677-695. [Pavlovic et al., 1997]

Quek, F. K. H., Mysliviec, T., and Zhao, M. (1995). Finger mouse: A freehand pointing interface. In *Proceedings of International Workshop on Automatic Face and Gesture Recognition,* pages 372-377, Zurich, Switzerland. [Quek et al., 1995]

Reilly, D. L., Cooper, L. N., and Elbaum, C. (1982). A neural network mode for category leaning. *Biological Cybernetics,* 45:35-41. [Reilly et al., 1982]

Segen, J. and Kumar, S. (1998). Fast and accurate 3d gesture recognition interface. In *Proceedings of International Conference on Pattern Recognition,* pages 86—91, Brisbane, Australia. [Segen and Kumar, 1998]

Sing, B. K. and Ikeuchi, K. (1997). Toward automatic robot instruction from perception—mapping human grasps to manipulator grasps. *IEEE Transaction on Robotics and Automation,* 13(l):81-95. [Sing and Ikeuchi, 1997]

Triesch, J. and Malsburg, C. V. D. (1998). A gesture interface for human-robot interaction. In *Proceedings of 3rd IEEE International Conference on Automatic Face and Gesture Recognition,* pages 546-551. [Triesch and Malsburg, 1998]

Xie, M. (1997). Automatic feature matching in uncalibrated stereo vision through the use of color. *Robotic and Autonomous System,* 21(4):355-364. [Xie, 1997]

Yang, J., Lu, W., and Waibel, A. (1998). Skin-color modeling and adaptation. In *Proceedings of ACCV98,* pages 687-694, Hong Kong. [Yang et al., 1998]

Yin, X., Guo, D., and Xie, M. (2001). Hand image segmentation using color and rce neural network. *International journal of Robotics and Autonomous System,* 34(4):235-250. [Yin et al., 2001]

Yin, X. and Xie, M. (2003). Estimation of the fundamental matrix from uncalibrated stereo hand images for 3d hand gesture recognition. *Pattern Recognition,* 36(3):23-40. [Yin and Xie, 2003]

Yin, X. and Xie, M. (2007). Finger identification and hand posture recognition for human-robot interaction. *Image and Vision Computing,* 25(8):1291-1300. [Yin and Xie, 2007]

Zhang, Z. (1996). Determining the epipolar geometry and its uncertainty: A review. Technical Report 2927, INRIA Sophia-Antipolis, France. [Zhang, 1996]

Zhang, Z., Deriche, R., Faugeras, O., and Luong, Q.-T. (1995). A robust technique for matching two uncalibrated images through the recovery of the unknown epipolar geometry. *Artificial Intelligence Journal,* 78:87-119. [Zhang et al., 1995]

Edited by Nilanjan Sarkar

Human-robot interaction research is diverse and covers a wide range of topics. All aspects of human factors and robotics are within the purview of HRI research so far as they provide insight into how to improve our understanding in developing effective tools, protocols, and systems to enhance HRI. For example, a significant research effort is being devoted to designing human-robot interface that makes it easier for the people to interact with robots. HRI is an extremely active research field where new and important work is being published at a fast pace. It is neither possible nor is it our intention to cover every important work in this important research field in one volume. However, we believe that HRI as a research field has matured enough to merit a compilation of the outstanding work in the field in the form of a book. This book, which presents outstanding work from the leading HRI researchers covering a wide spectrum of topics, is an effort to capture and present some of the important contributions in HRI in one volume. We hope that this book will benefit both experts and novice and provide a thorough understanding of the exciting field of HRI.

ISBN 978-953-51-5818-9

Photo by ankarb / iStock

IntechOpen