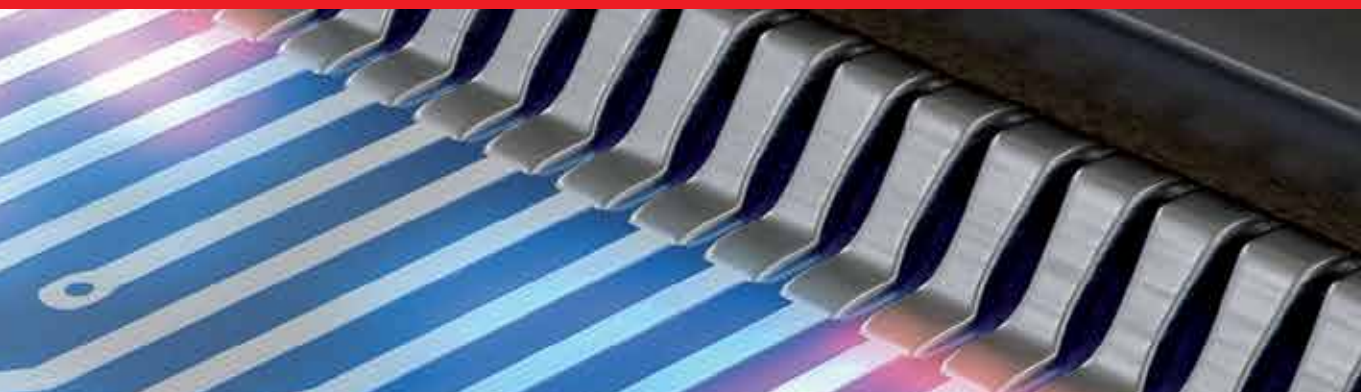
IntechOpen

From Natural to Artificial Intelligence

Algorithms and Applications

Edited by Ricardo Lopez-Ruiz



FROM NATURAL TO ARTIFICIAL INTELLIGENCE - ALGORITHMS AND APPLICATIONS

Edited by **Ricardo Lopez-Ruiz**

From Natural to Artificial Intelligence - Algorithms and Applications
http://dx.doi.org/10.5772/intechopen.71252
Edited by Ricardo Lopez-Ruiz

Contributors

Sabur Ajibola Alim, Nahrul Khair Alang Rashid, Josep Llorca, Héctor Zapata, Jesús Alba, Ernest Redondo, David Fonseca, Vikram Singh, Jay Patel, Chih-Wei Lin, Jesus Olivares-Mercado, Karina Toscano-Medina, Gabriel Sanchez, Mariko Nakano Miyatake, Hector Perez Meana, Luis Carlos Castro Madrid, Monica Dascalu, Guadalupe Elizabeth Morales-Martinez, Ernesto Lopez-Ramirez, Jocelyn Garcia-Duran, Claudia Jaquelina Gonzalez-Trujillo, Yanko Mezquita-Hoyos, Michael Vielhaber, Rajesh Aggarwal, Vishal Passricha, Hiromi Miyajima, Hirofumi Miyajima, Noritaka Shigei

Notice

First published in London, United Kingdom, 2018 by IntechOpen
eBook (PDF) Published by IntechOpen, 2019
IntechOpen is the global imprint of INTECHOPEN LIMITED, registered in England and Wales, registration number: 11086078, The Shard, 25th floor, 32 London Bridge Street
London, SE19SG – United Kingdom
Printed in Croatia

British Library Cataloguing-in-Publication Data
A catalogue record for this book is available from the British Library

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

From Natural to Artificial Intelligence - Algorithms and Applications
Edited by Ricardo Lopez-Ruiz

p. cm.

Print ISBN 978-1-78984-702-4

Online ISBN 978-1-78984-703-1

eBook (PDF) ISBN 978-1-83881-548-6

Meet the editor

Ricardo López-Ruiz, M.S., Ph.D., works as an Associate Professor in the Department of Computer Science and Systems Engineering, Faculty of Science, University of Zaragoza, Spain. He also serves as an Associate Researcher in Complex Systems at the School of Mathematics, University of Zaragoza. He also worked as a Lecturer in the University of Navarra, the Public University of Navarra and the UNED of Calatayud. He completed his postdoc with Prof. Yves Pomeau at the École Normale Supérieure of Paris and with Prof. Gabriel Mindlin at the University of Buenos Aires. His areas of interest include statistical complexity and nonlinear models, chaotic maps and applications, multiagent systems, and econo-physics.

Contents

Preface

Artificial intelligence (AI) is a field experiencing constant growth and change, with a long history. The challenge to reproduce human behavior in machines requires the interaction of many fields, from engineering to mathematics, from neurology to biology, from computer science to robotics, from web search to social networks, from machine learning to game theory, etc. Numerous applications and possibilities of AI are already a reality but other ones are needed to reduce the human limitations and to expand the human capability to limits beyond our imagination. This book brings together researchers working on areas related to AI such as face and speech recognition, representation of learning and acoustic scenarios, fuzzy inference and data exploration, cellular automata applications with a special interest in the tools and algorithms that can be applied in these different branches of the AI discipline. The book provides a new reference to an audience interested in the development of this field.

The first three sections of the book present different algorithms and models with a variety of applications in speech and face recognition and also in learning and acoustic scenarios. In Chapter 1, Alim and Rashid explain some extraction techniques used for speech identification and recognition. In Chapter 2, Passricha and Aggarwal discuss an acoustic model based on convolutional neural networks to decode raw speech signals. In Chapter 3, Llorca et al. evaluate the subjective impact of immersive acoustic features in the representation of urban environments. In Chapter 4, Morales-Martínez et al. present tools of cognitive science to improve adaptive e-learning systems. In Chapter 5, Lin discusses local pattern descriptors for face recognition. The last chapter of this section, Chapter 6 by Olivares-Mercado et al., analyzes the performance of different texture descriptor algorithms for face feature extraction tasks.

The last two sections of the book present some investigations in fuzzy inference and data exploration and also different cellular automata applications. In Chapter 7, Miyajima et al. study the improvement of different fuzzy methods for pattern classification. In Chapter 8, Patel and Singh propose a query reformulation method for a relevant data exploration. In Chapter 9, Dascalu discusses algorithms and architectures of cellular automata used as random number generators. In the last chapter, Chapter 10, Vielhaber studies the randomness in cellular automata evolving under asynchronous clocking schemes.

As the editor of this book, I would like to thank all the authors who have contributed to this volume as well as the reviewers for their assessment. Also, I must express my gratitude to the IntechOpen Editorial Staff for their invitation to be editor for a third time, and Author Service Managers (ASM) Mr. Markus Mattila and Ms. Ljerka Bilan, who were both of particular help in realizing this new IntechOpen book. Finally, at this moment where life is a sweet time flow, I want to dedicate this book to my very good friend Enrique Lozano Corbi, recently retired from his position as Professor of Roman Law after forty-four years working at the University of Zaragoza. Of course, all my family from Villafranca, Navarra, Spain, and all my friends and advisors are not forgotten in this dedicatory final paragraph.

Ricardo López-Ruiz
University of Zaragoza, Spain

Section 1

Speech Extraction and Recognition

Chapter 1

Some Commonly Used Speech Feature Extraction Algorithms

Sabur Ajibola Alim and Nahrul Khair Alang Rashid

Additional information is available at the end of the chapter

http://dx.doi.org/10.5772/intechopen.80419

Abstract

Speech is a complex naturally acquired human motor ability. It is characterized in adults with the production of about 14 different sounds per second via the harmonized actions of roughly 100 muscles. Speaker recognition is the capability of a software or hardware to receive speech signal, identify the speaker present in the speech signal and recognize the speaker afterwards. Feature extraction is accomplished by changing the speech waveform to a form of parametric representation at a relatively minimized data rate for subsequent processing and analysis. Therefore, acceptable classification is derived from excellent and quality features. Mel Frequency Cepstral Coefficients (MFCC), Linear Prediction Coefficients (LPC), Linear Prediction Cepstral Coefficients (LPCC), Line Spectral Frequencies (LSF), Discrete Wavelet Transform (DWT) and Perceptual Linear Prediction (PLP) are the speech feature extraction techniques that were discussed in these chapter. These methods have been tested in a wide variety of applications, giving them high level of reliability and acceptability. Researchers have made several modifications to the above discussed techniques to make them less susceptible to noise, more robust and consume less time. In conclusion, none of the methods is superior to the other, the area of application would determine which method to select.

Keywords: human speech, speech features, mel frequency cepstral coefficients (MFCC), linear prediction coefficients (LPC), linear prediction cepstral coefficients (LPCC), line spectral frequencies (LSF), discrete wavelet transform (DWT), perceptual linear prediction (PLP)

1. Introduction

Human beings express their feelings, opinions, views and notions orally through speech. The speech production process includes articulation, voice, and fluency [1, 2]. It is a complex

naturally acquired human motor abilities, a task categorized in regular adults by the production of about 14 different sounds per second via the harmonized actions of roughly 100 muscles connected by spinal and cranial nerves. The simplicity with which human beings speak is in contrast to the complexity of the task, and that complexity could assist in explaining why speech can be very sensitive to diseases associated with the nervous system [3].

There have been several successful attempts in the development of systems that can analyze, classify and recognize speech signals. Both hardware and software that have been developed for such tasks have been applied in various fields such as health care, government sectors and agriculture. Speaker recognition is the capability of a software or hardware to receive speech signal, identify the speaker present in the speech signal and recognize the speaker afterwards [4]. Speaker recognition executes a task similar to what the human brain undertakes. This starts from speech which is an input to the speaker recognition system. Generally, speaker recognition process takes place in three main steps which are acoustic processing, feature extraction and classification/recognition [5].

The speech signal has to be processed to remove noise before the extraction of the important attributes in the speech [6] and identification. The purpose of feature extraction is to illustrate a speech signal by a predetermined number of components of the signal. This is because all the information in the acoustic signal is too cumbersome to deal with, and some of the information is irrelevant in the identification task [7, 8].

Feature extraction is accomplished by changing the speech waveform to a form of parametric representation at a relatively lesser data rate for subsequent processing and analysis. This is usually called the front end signal-processing [9, 10]. It transforms the processed speech signal to a concise but logical representation that is more discriminative and reliable than the actual signal. With front end being the initial element in the sequence, the quality of the subsequent features (pattern matching and speaker modeling) is significantly affected by the quality of the front end [10].

Therefore, acceptable classification is derived from excellent and quality features. In present automatic speaker recognition (ASR) systems, the procedure for feature extraction has normally been to discover a representation that is comparatively reliable for several conditions of the same speech signal, even with alterations in the environmental conditions or speaker, while retaining the portion that characterizes the information in the speech signal [7, 8].

Feature extraction approaches usually yield a multidimensional feature vector for every speech signal [11]. A wide range of options are available to parametrically represent the speech signal for the recognition process, such as perceptual linear prediction (PLP), linear prediction coding (LPC) and mel-frequency cepstrum coefficients (MFCC). MFCC is the best known and very popular [9, 12]. Feature extraction is the most relevant portion of speaker recognition. Features of speech have a vital part in the segregation of a speaker from others [13]. Feature extraction reduces the magnitude of the speech signal devoid of causing any damage to the power of speech signal [14].

Before the features are extracted, there are sequences of preprocessing phases that are first carried out. The preprocessing step is pre-emphasis. This is achieved by passing the signal

http://dx.doi.org/10.5772/intechopen.80419

through a FIR filter [15] which is usually a first-order finite impulse response (FIR) filter [16]. This is succeeded by frame blocking, a method of partitioning the speech signal into frames. It removes the acoustic interface existing in the start and end of the speech signal [17].

The framed speech signal is then windowed. Bandpass filter is a suitable window [15] that is applied to minimize disjointedness at the start and finish of each frame. The two most famous categories of windows are Hamming and Rectangular windows [18]. It increases the sharpness of harmonics, eliminates the discontinuous of signal by tapering beginning and ending of the frame zero. It also reduces the spectral distortion formed by the overlap [17].

2. Mel frequency cepstral coefficients (MFCC)

Mel frequency cepstral coefficients (MFCC) was originally suggested for identifying monosyllabic words in continuously spoken sentences but not for speaker identification. MFCC computation is a replication of the human hearing system intending to artificially implement the ear's working principle with the assumption that the human ear is a reliable speaker recognizer [19]. MFCC features are rooted in the recognized discrepancy of the human ear's critical bandwidths with frequency filters spaced linearly at low frequencies and logarithmically at high frequencies have been used to retain the phonetically vital properties of the speech signal. Speech signals commonly contain tones of varying frequencies, each tone with an actual frequency, f (Hz) and the subjective pitch is computed on the Mel scale. The mel-frequency scale has linear frequency spacing below 1000 Hz and logarithmic spacing above 1000 Hz. Pitch of 1 kHz tone and 40 dB above the perceptual audible threshold is defined as 1000 mels, and used as reference point [20].

MFCC is based on signal disintegration with the help of a filter bank. The MFCC gives a discrete cosine transform (DCT) of a real logarithm of the short-term energy displayed on the Mel frequency scale [21]. MFCC is used to identify airline reservation, numbers spoken into a telephone and voice recognition system for security purpose. Some modifications have been proposed to the basic MFCC algorithm for better robustness, such as by lifting the log-mel-amplitudes to an appropriate power (around 2 or 3) before applying the DCT and reducing the impact of the low-energy parts [4].

2.1. Algorithm description, strength and weaknesses

MFCC are cepstral coefficients derived on a twisted frequency scale centerd on human auditory perception. In the computation of MFCC, the first thing is windowing the speech signal to split the speech signal into frames. Since the high frequency formants process reduced amplitude compared to the low frequency formants, high frequencies are emphasized to obtain similar amplitude for all the formants. After windowing, Fast Fourier Transform (FFT) is applied to find the power spectrum of each frame. Subsequently, the filter bank processing is carried out on the power spectrum, using mel-scale. The DCT is applied to the speech signal

after translating the power spectrum to log domain in order to calculate MFCC coefficients [5]. The formula used to calculate the mels for any frequency is [19, 22]:

$$mel(f) = 2595\, x\, \log_{10}(1 + f/700) \tag{1}$$

where mel(f) is the frequency (mels) and f is the frequency (Hz).

The MFCCs are calculated using this equation [9, 19]:

$$\hat{C}_n = \sum_{n=1}^{k} \left(\log \hat{S}_k\right) cos\left[n\left(k - \frac{1}{2}\right)\frac{\pi}{k}\right] \tag{2}$$

where k is the number of mel cepstrum coefficients, $\hat{S}_k$ is the output of filterbank and $\hat{C}_n$ is the final mfcc coefficients.

The block diagram of the MFCC processor can be seen in **Figure 1**. It summarizes all the processes and steps taken to obtain the needed coefficients. MFCC can effectively denote the low frequency region better than the high frequency region, henceforth, it can compute formants that are in the low frequency range and describe the vocal tract resonances. It has been generally recognized as a front-end procedure for typical Speaker Identification applications, as it has reduced vulnerability to noise disturbance, with minute session inconsistency and easy to mine [19]. Also, it is a perfect representation for sounds when the source characteristics are stable and consistent (music and speech) [23]. Furthermore, it can capture information from sampled signals with frequencies at a maximum of 5 kHz, which encapsulates most energy of sounds that are generated by humans [9].

Cepstral coefficients are said to be accurate in certain pattern recognition problems relating to human voice. They are used extensively in speaker identification and speech recognition [21]. Other formants can also be above 1 kHz and are not efficiently taken into consideration by the large filter spacing in the high frequency range [19]. MFCC features are not exactly accurate in the existence of background noise [14, 24] and might not be well suited for generalization [23].

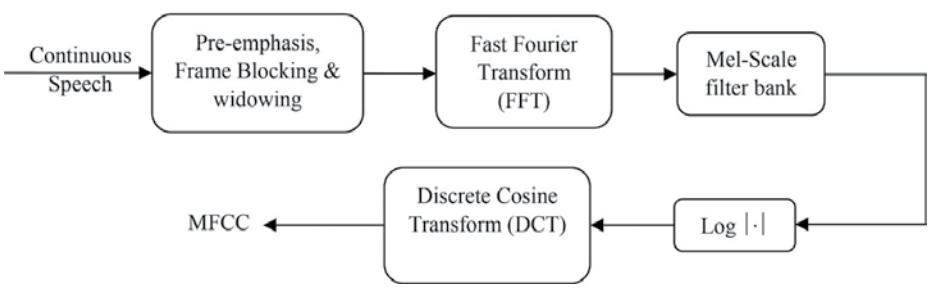


Figure 1. Block diagram of MFCC processor.

http://dx.doi.org/10.5772/intechopen.80419

3. Linear prediction coefficients (LPC)

Linear prediction coefficients (LPC) imitates the human vocal tract [16] and gives robust speech feature. It evaluates the speech signal by approximating the formants, getting rid of its effects from the speech signal and estimate the concentration and frequency of the left behind residue. The result states each sample of the signal as a direct incorporation of previous samples. The coefficients of the difference equation characterize the formants, thus, LPC needs to approximate these coefficients [25]. LPC is a powerful speech analysis method and it has gained fame as a formant estimation method [17].

The frequencies where the resonant crests happen are called the formant frequencies. Thus, with this technique, the positions of the formants in a speech signal are predictable by calculating the linear predictive coefficients above a sliding window and finding the crests in the spectrum of the subsequent linear prediction filter [17]. LPC is helpful in the encoding of high quality speech at low bit rate [13, 26, 27].

Other features that can be deduced from LPC are linear predication cepstral coefficients (LPCC), log area ratio (LAR), reflection coefficients (RC), line spectral frequencies (LSF) and Arcus Sine Coefficients (ARCSIN) [13]. LPC is generally used for speech reconstruction. LPC method is generally applied in musical and electrical firms for creating mobile robots, in telephone firms, tonal analysis of violins and other string musical gadgets [4].

3.1. Algorithm description, strength and weaknesses

Linear prediction method is applied to obtain the filter coefficients equivalent to the vocal tract by reducing the mean square error in between the input speech and estimated speech [28]. Linear prediction analysis of speech signal forecasts any given speech sample at a specific period as a linear weighted aggregation of preceding samples. The linear predictive model of speech creation is given as [13, 25]:

$$\hat{s}(n) = \sum_{k=1}^{p} a_k s(n-k) \tag{3}$$

where $\hat{s}$ is the predicted sample, s is the speech sample, p is the predictor coefficients.

The prediction error is given as [16, 25]:

$$e(n) = s(n) - \hat{s}(n) \tag{4}$$

Subsequently, each frame of the windowed signal is autocorrelated, while the highest autocorrelation value is the order of the linear prediction analysis. This is followed by the LPC analysis, where each frame of the autocorrelations is converted into LPC parameters set which consists of the LPC coefficients [26]. A summary of the procedure for obtaining the LPC is as seen in **Figure 2**. LPC can be derived by [7]:

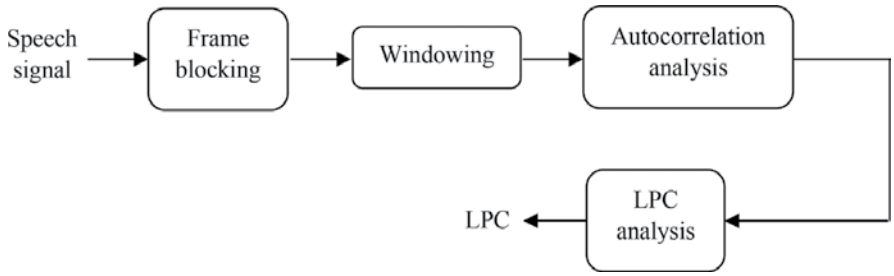


Figure 2. Block diagram of LPC processor.

$$a_m = \log\left[\frac{1-k_m}{1+k_m}\right] \tag{5}$$

where a_m is the linear prediction coefficient, k_m is the reflection coefficient.

Linear predictive analysis efficiently selects the vocal tract information from a given speech [16]. It is known for the speed of computation and accuracy [18]. LPC excellently represents the source behaviors that are steady and consistent [23]. Furthermore, it is also be used in speaker recognition system where the main purpose is to extract the vocal tract properties [25]. It gives very accurate estimates of speech parameters and is comparatively efficient for computation [14, 26]. Traditional linear prediction suffers from aliased autocorrelation coefficients [29]. LPC estimates have high sensitivity to quantization noise [30] and might not be well suited for generalization [23].

4. Linear prediction cepstral coefficients (LPCC)

Linear prediction cepstral coefficients (LPCC) are cepstral coefficients derived from LPC calculated spectral envelope [11]. LPCC are the coefficients of the Fourier transform illustration of the logarithmic magnitude spectrum [30, 31] of LPC. Cepstral analysis is commonly applied in the field of speech processing because of its ability to perfectly symbolize speech waveforms and characteristics with a limited size of features [31].

It was observed by Rosenberg and Sambur that adjacent predictor coefficients are highly correlated and therefore, representations with less correlated features would be more efficient, LPCC is a typical example of such. The relationship between LPC and LPCC was originally derived by Atal in 1974. In theory, it is relatively easy to convert LPC to LPCC, in the case of minimum phase signals [32].

4.1. Algorithm description, strength and weaknesses

In speech processing, LPCC analogous to LPC, are computed from sample points of a speech waveform, the horizontal axis is the time axis, while the vertical axis is the amplitude axis [31].

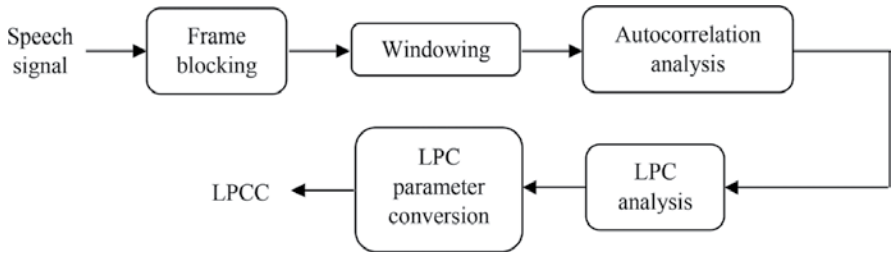


Figure 3. Block diagram of LPCC processor.

The LPCC processor is as seen in **Figure 3**. It pictorially explains the process of obtaining LPCC. LPCC can be calculated using [7, 15, 33]:

$$C_m = a_m + \sum_{k=1}^{m-1} \left[\frac{k}{m}\right] c_k a_{m-k} \tag{6}$$

where a_m is the linear prediction coefficient, C_m is the cepstral coefficient.

LPCC have low vulnerability to noise [30]. LPCC features yield lower error rate as compared to LPC features [31]. Cepstral coefficients of higher order are mathematically limited, resulting in an extremely extensive array of variances when moving from the cepstral coefficients of lower order to cepstral coefficients of higher order [34]. Similarly, LPCC estimates are notorious for having great sensitivity to quantization noise [35]. Cepstral analysis on high-pitch speech signal gives small source-filter separability in the quefrency domain [29]. Cepstral coefficients of lower order are sensitive to the spectral slope, while the cepstral coefficients of higher order are sensitive to noise [15].

5. Line spectral frequencies (LSF)

Individual lines of the Line Spectral Pairs (LSP) are known as line spectral frequencies (LSF). LSF defines the two resonance situations taking place in the inter-connected tube model of the human vocal tract. The model takes into consideration the nasal cavity and the mouth shape, which gives the basis for the fundamental physiological importance of the linear prediction illustration. The two resonance situations define the vocal tract as either being completely open or completely closed at the glottis [36]. The two situations begets two groups of resonant frequencies, with the number of resonances in each group being deduced from the quantity of linked tubes. The resonances of each situation are the odd and even line spectra correspondingly, and are interwoven into a singularly rising group of LSF [36].

The LSF representation was proposed by Itakura [37, 38] as a substitute to the linear prediction parametric illustration. In the area of speech coding, it has been realized that this illustration has an improved quantization features than the other linear prediction parametric illustrations

(LAR and RC). The LSF illustration has the capacity to reduce the bit-rate by 25–30% for transmitting the linear prediction information without distorting the quality of synthesized speech [38–40]. Apart from quantization, LSF illustration of the predictor are also suitable for interpolation. Theoretically, this can be inspired by the point that the sensitivity matrix linking the LSF-domain squared quantization error to the perceptually relevant log spectrum is diagonal [41, 42].

5.1. Algorithm description, strength and weaknesses

LP is established on the point that a speech signal can be defined by Eq. (3). Recall

$$\hat{s}(n) = \sum_{k=1}^{p} a_k s(n-k)$$

where k is the time index and p is the order of the linear prediction, $\hat{s}(n)$ is the predictor signal and a_k is the LPC coefficients.

The a_k coefficients are determined in order to reduce the prediction error by method of autocorrelation or covariance. Eq. (3) can be modified in the frequency domain with the z-transform. As such, a small part of the speech signal is anticipated to be given as an output to the all-pole filter $H(z)$. The new equation is

$$H(z) = \frac{1}{A(z)} = \frac{1}{1 - \sum_{i=1}^{p} a_i z^{-1}} \tag{7}$$

where $H(z)$ is the all-pole filter and $A(z)$ is the LPC analysis filter.

In order to compute the LSF coefficients, an inverse polynomial filter is split into two polynomials $P(z)$ and $Q(z)$ [36, 38, 40, 41]:

$$P(z) = A(z) + z^{-(p+1)} A\left(z^{-1}\right) \tag{8}$$

$$Q(z) = A(z) - z^{-(p+1)} A\left(z^{-1}\right) \tag{9}$$

where $P(z)$ is the vocal tract with the glottis closed, $Q(z)$ is the LPC analysis filter of order p.

In order to convert LSF back to LPC, the equation below is used [36, 41, 43, 44]:

$$A(z) = 0.5[P(z) + Q(z)] \tag{10}$$

The block diagram of the LSF processor is as seen in **Figure 4**. The most prominent application of LSF is in the area of speech compression, with extension into the speaker recognition and speech recognition. This technique has also found restricted use in other fields. LSF have been investigated for use in musical instrument recognition and coding. LSF have also been applied to animal noise identification, recognizing individual instruments and financial market analysis. The advantages of LSF include their ability to localize spectral sensitivities, the fact that

http://dx.doi.org/10.5772/intechopen.80419

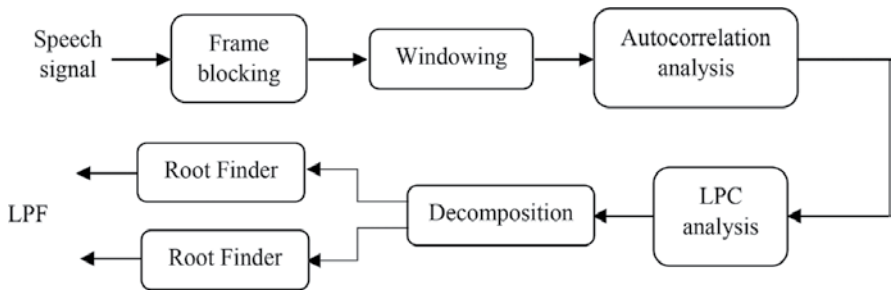


Figure 4. Block diagram of LSF processor.

they characterize bandwidths and resonance locations and lays emphasis on the important aspect of spectral peak location. In most instances, the LSF representation provides a near-minimal data set for subsequent classification [36].

Since LSF represents spectral shape information at a lower data rate than raw input samples, it is reasonable that a careful use of processing and analysis methods in the LSP domain could lead to a complexity reduction against alternative techniques operating on the raw input data itself. LSF play an important role in the transmission of vocal tract information from speech coder to decoder with their widespread use being a result of their excellent quantization properties. The generation of LSP parameters can be accomplished using several methods, ranging in complexity. The major problem revolves around finding the roots of the P and Q polynomials defined in Eqs. (8) and (9). This can be obtained through standard root solving methods, or more obscure methods and it is often performed in the cosine domain [36].

6. Discrete wavelet transform (dwt)

Wavelet Transform (WT) theory is centered around signal analysis using varying scales in the time and frequency domains [45]. With the support of theoretical physicist Alex Grossmann, Jean Morlet introduced wavelet transform which permits high-frequency events identification with an enhanced temporal resolution [45–47]. A wavelet is a waveform of effectively limited duration that has an average value of zero. Many wavelets also display orthogonality, an ideal feature of compact signal representation [46]. WT is a signal processing technique that can be used to represent real-life non-stationary signals with high efficiency [33, 46]. It has the ability to mine information from the transient signals concurrently in both time and frequency domains [33, 45, 48].

Continuous wavelet transform (CWT) is used to split a continuous-time function into wavelets. However, there is redundancy of information and huge computational efforts is required to calculate all likely scales and translations of CWT, thereby restricting its use [45]. Discrete wavelet transform (DWT) is an extension of the WT that enhances the flexibility to the decomposition process [48]. It was introduced as a highly flexible and efficient method for sub band breakdown of signals [46, 49]. In earlier applications, linear

discretization was used for discretizing CWT. Daubechies and others have developed an orthogonal DWT specially designed for analyzing a finite set of observations over the set of scales (dyadic discretization) [47].

6.1. Algorithm description, strength and weaknesses

Wavelet transform decomposes a signal into a group of basic functions called wavelets. Wavelets are obtained from a single prototype wavelet called mother wavelet by dilations and shifting. The main characteristic of the WT is that it uses a variable window to scan the frequency spectrum, increasing the temporal resolution of the analysis [45, 46, 50].

WT decomposes signals over translated and dilated mother wavelets. Mother wavelet is a time function with finite energy and fast decay. The different versions of the single wavelet are orthogonal to each other. The continuous wavelet transform (CWT) is given by [33, 45, 50]:

$$W_x(a,b) = \frac{1}{\sqrt{a}} \int_{-\infty}^{\infty} x(t)\psi^*\left(\frac{t-b}{a}\right)dt \tag{11}$$

where $\psi(t)$ is the mother wavelet, a and b are continuous parameters.

The WT coefficient is an expansion and a particular shift represents how well the original signal corresponds to the translated and dilated mother wavelet. Thus, the coefficient group of CWT (a, b) associated with a particular signal is the wavelet representation of the original signal in relation to the mother wavelet [45]. Since CWT contains high redundancy, analyzing the signal using a small number of scales with varying number of translations at each scale, i.e. discretizing scale and translation parameters as $a = 2^j$ *and* $b = 2^j k$ gives DWT. DWT theory requires two sets of related functions called scaling function and wavelet function given by [33]:

$$\phi(t) = \sum_{n=0}^{N-1} h[n]\sqrt{2}\phi(2t-n) \tag{12}$$

$$\psi(t) = \sum_{n=0}^{N-1} g[n]\sqrt{2}\phi(2t-n) \tag{13}$$

where $\phi(t)$ is the scaling function, $\psi(t)$ is the wavelet function, h[n] is the an impulse response of a low-pass filter, and g[n] is an impulse response of a high-pass filter.

There are several ways to discretize a CWT. The DWT of the continuous signal can also be given by [45]:

$$(DWT)(m,p) = \int_{-\infty}^{+\infty} x(t)\cdot\psi_{m,p}dt \tag{14}$$

where $\psi_{m,p}$ is the wavelet function bases, m is the dilation parameter and p is the translation parameter.

http://dx.doi.org/10.5772/intechopen.80419

Thus, $\psi_{m,p}$ is defined as:

$$\psi_{m,p} = \frac{1}{\sqrt{a_0^m}} \psi\left(\frac{t - pb_0a_0^m}{a_0^m}\right) \tag{15}$$

The DWT of a discrete signal is derived from CWT and defined as:

$$(DWT)(m,k) = \frac{1}{\sqrt{a_0^m}} \sum_n x[n] \cdot g\left(\frac{n - nb_0a_0^m}{a_0^m}\right) \tag{16}$$

where $g(*)$ is the mother wavelet and $x[n]$ is the discretized signal. The mother wavelet may be dilated and translated discretely by selecting the scaling parameter $a = a_0^m$ and translation parameter $b = nb_0a_0^m$ (with constants taken as $a_0 > 1$, $b_0 > 1$, while m and n are assigned a set of positive integers).

The scaling and wavelet functions can be implemented effectively using a pair of filters, h[n] and g[n], called quadrature mirror filters that confirm with the property $g[n] = (-1)^{1-n}h[n]$. The input signal is filtered by a low-pass filter and high-pass filter to obtain the approximate components and the detail components respectively. This is summarized in **Figure 5**. The approximate signal at each stage is further decomposed using the same low-pass and high-pass filters to get the approximate and detail components for the next stage. This type of decomposition is called dyadic decomposition [33].

The DWT parameters contain the information of different frequency scales. This enhances the speech information obtained in the corresponding frequency band [33]. The ability of the DWT to partition the variance of the elements of the input on a scale by scale basis is an added advantage. This partitioning leads to the opinion of the scale-dependent wavelet variance, which in many ways is equivalent to the more familiar frequency-dependent Fourier power spectrum [47]. Classic discrete decomposition schemes, which are dyadic do not fulfill all the requirements for direct use in parameterization. DWT does provide adequate number of frequency bands for effective speech analysis [51]. Since the input signals are of finite length, the wavelet coefficients will have unwantedly large variations at the boundaries because of the discontinuities at the boundaries [50].

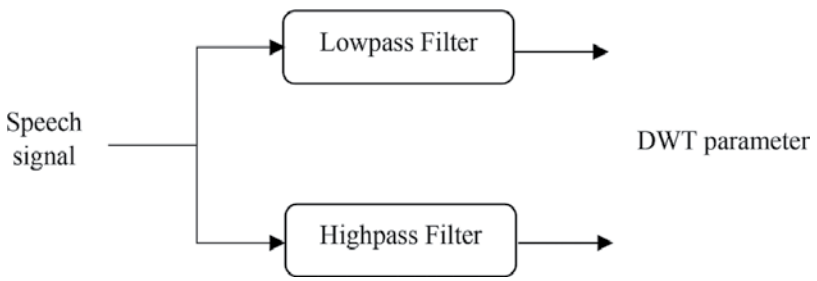


Figure 5. Block diagram of DWT.

7. Perceptual linear prediction (PLP)

Perceptual linear prediction (PLP) technique combines the critical bands, intensity-to-loudness compression and equal loudness pre-emphasis in the extraction of relevant information from speech. It is rooted in the nonlinear bark scale and was initially intended for use in speech recognition tasks by eliminating the speaker dependent features [11]. PLP gives a representation conforming to a smoothed short-term spectrum that has been equalized and compressed similar to the human hearing making it similar to the MFCC. In the PLP approach, several prominent features of hearing are replicated and the consequent auditory like spectrum of speech is approximated by an autoregressive all–pole model [52]. PLP gives minimized resolution at high frequencies that signifies auditory filter bank based approach, yet gives the orthogonal outputs that are similar to the cepstral analysis. It uses linear predictions for spectral smoothing, hence, the name is perceptual linear prediction [28]. PLP is a combination of both spectral analysis and linear prediction analysis.

7.1. Algorithm description, strength and weaknesses

In order to compute the PLP features the speech is windowed (Hamming window), the Fast Fourier Transform (FFT) and the square of the magnitude are computed. This gives the power spectral estimates. A trapezoidal filter is then applied at 1-bark interval to integrate the overlapping critical band filter responses in the power spectrum. This effectively compresses the higher frequencies into a narrow band. The symmetric frequency domain convolution on the bark warped frequency scale then permits low frequencies to mask the high frequencies, concurrently smoothing the spectrum. The spectrum is subsequently pre-emphasized to approximate the uneven sensitivity of human hearing at a variety of frequencies. The spectral amplitude is compressed, this reduces the amplitude variation of the spectral resonances. An Inverse Discrete Fourier Transform (IDCT) is performed to get the autocorrelation coefficients. Spectral smoothing is performed, solving the autoregressive equations. The autoregressive coefficients are converted to cepstral variables [28]. The equation for computing the bark scale frequency is:

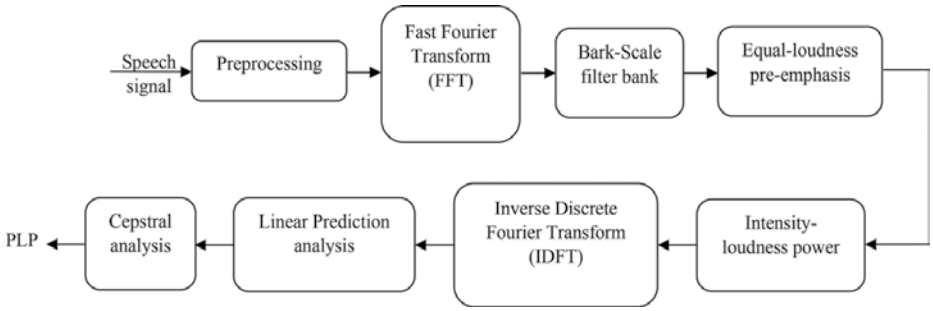


Figure 6. Block diagram of PLP processor.

	Type of Filter	Shape of filter	What is modeled	Speed of computation	Type of coefficient	Noise resistance	Sensitivity to quantization/additional noise	Reliability	Frequency captured
Mel frequency cepstral coefficient (MFCC)	Mel	Triangular	Human Auditory System	High	Cepstral	Medium	Medium	High	Low
Linear prediction coefficient (LPC)	Linear Prediction	Linear	Human Vocal Tract	High	Autocorrelation Coefficient	High	High	High	Low
Linear prediction cepstral coefficient (LPCC)	Linear Prediction	Linear	Human Vocal Tract	Medium	Cepstral	High	High	Medium	Low & Medium
Line spectral frequencies (LSF)	Linear Prediction	Linear	Human Vocal Tract	Medium	Spectral	High	High	Medium	Low & Medium
Discrete wavelet transform (DWT)	Lowpass & highpass	—	—	High	Wavelets	Medium	Medium	Medium	Low & High
Perceptual linear prediction (PLP)	Bark	Trapezoidal	Human Auditory System	Medium	Cepstral & Autocorrelation	Medium	Medium	Medium	Low & Medium

Table 1. Comparison between the feature extraction techniques.

http://dx.doi.org/10.5772/intechopen.80419

$$bark(f) = \frac{26.81\,f}{1960 + f} - 0.53 \tag{17}$$

where bark(f) is the frequency (bark) and f is the frequency (Hz).

The identification achieved by PLP is better than that of LPC [28], because it is an improvement over the conventional LPC because it effectively suppresses the speaker-dependent information [52]. Also, it has enhanced speaker independent recognition performance and is robust to noise, variations in the channel and microphones [53]. PLP reconstructs the autoregressive noise component accurately [54]. PLP based front end is sensitive to any change in the formant frequency.

Figure 6 shows the PLP processor, showing all the steps to be taken to obtain the PLP coefficients. PLP has low sensitivity to spectral tilt, consistent with the findings that it is relatively insensitive to phonetic judgments of the spectral tilt. Also, PLP analysis is dependent on the result of the overall spectral balance (formant amplitudes). The formant amplitudes are easily affected by factors such as the recording equipment, communication channel and additive noise [52]. Furthermore, the time-frequency resolution and efficient sampling of the short-term representation are addressed in an ad-hoc way [54].

Table 1 shows a comparison between the six feature extraction techniques that have been explicitly described above. Even though the selection of a feature extraction algorithm for use in research is individual dependent, however, this table has been able to characterize these techniques based on the main considerations in the selection of any feature extraction algorithm. The considerations include speed of computation, noise resistance and sensitivity to additional noise. The table also serves as a guide when considering the selection between any two or more of the discussed algorithms.

8. Conclusion

MFCC, LPC, LPCC, LSF, PLP and DWT are some of the feature extraction techniques used for extracting relevant information form speech signals for the purpose speech recognition and identification. These techniques have stood the test of time and have been widely used in speech recognition systems for several purposes. Speech signal is a slow time varying signal, quasi-stationary, when observed over an adequately short period of time between 5 and 100 msec, its behavior is relatively stationary. As a result of this, short time spectral analysis which includes MFCC, LPCC and PLP are commonly used for the extraction of important information from speech signals. Noise is a serious challenge encountered in the process of feature extraction, as well as speaker recognition as a whole. Subsequently, researchers have made several modifications to the above discussed techniques to make them less susceptible to noise, more robust and consume less time. These methods have also been used in the recognition of sounds. The extracted information will be the input to the classifier for identification purposes. The above discussed feature extraction approaches can be implemented using MATLAB.

Author details

Sabur Ajibola Alim[1]* and Nahrul Khair Alang Rashid[2]

*Address all correspondence to: moaj1st@yahoo.com

1 Ahmadu Bello University, Zaria, Nigeria

2 Universiti Teknologi Malaysia, Skudai, Johor, Malaysia

References

[1] Hariharan M, Vijean V, Fook CY, Yaacob S. Speech stuttering assessment using sample entropy and Least Square Support vector machine. In: 8th International Colloquium on Signal Processing and its Applications (CSPA). 2012. pp. 240-245

[2] Manjula GN, Kumar MS. Stuttered speech recognition for robotic control. International Journal of Engineering and Innovative Technology (IJEIT). 2014;**3**(12):174-177

[3] Duffy JR. Motor speech disorders: Clues to neurologic diagnosis. In: Parkinson's Disease and Movement Disorders. Totowa, NJ: Humana Press; 2000. pp. 35-53

[4] Kurzekar PK, Deshmukh RR, Waghmare VB, Shrishrimal PP. A comparative study of feature extraction techniques for speech recognition system. International Journal of Innovative Research in Science, Engineering and Technology. 2014;**3**(12):18006-18016

[5] Ahmad AM, Ismail S, Samaon DF. Recurrent neural network with backpropagation through time for speech recognition. In: IEEE International Symposium on Communications and Information Technology (ISCIT 2004). Vol. 1. Sapporo, Japan: IEEE; 2004. pp. 98-102

[6] Shaneh M, Taheri A. Voice command recognition system based on MFCC and VQ algorithms. World academy of science. Engineering and Technology. 2009;**57**:534-538

[7] Mosa GS, Ali AA. Arabic phoneme recognition using hierarchical neural fuzzy petri net and LPC feature extraction. Signal Processing: An International Journal (SPIJ). 2009;**3**(5): 161

[8] Yousefian N, Analoui M. Using radial basis probabilistic neural network for speech recognition. In: Proceeding of 3rd International Conference on Information and Knowledge (IKT07), Mashhad, Iran. 2007

[9] Cornaz C, Hunkeler U, Velisavljevic V. An Automatic Speaker Recognition System. Switzerland: Lausanne; 2003. Retrieved from: http://read.pudn.com/downloads60/sourcecode/multimedia/audio/209082/asr_project.pdf

[10] Shah SAA, ul Asar A, Shaukat SF. Neural network solution for secure interactive voice response. World Applied Sciences Journal. 2009;**6**(9):1264-1269

[11] Ravikumar KM, Rajagopal R, Nagaraj HC. An approach for objective assessment of stuttered speech using MFCC features. ICGST International Journal on Digital Signal Processing, DSP. 2009;**9**(1):19-24

[12] Kumar PP, Vardhan KSN, Krishna KSR. Performance evaluation of MLP for speech recognition in noisy environments using MFCC & wavelets. International Journal of Computer Science & Communication (IJCSC). 2010;**1**(2):41-45

[13] Kumar R, Ranjan R, Singh SK, Kala R, Shukla A, Tiwari R. Multilingual speaker recognition using neural network. In: Proceedings of the Frontiers of Research on Speech and Music, FRSM. 2009. pp. 1-8

[14] Narang S, Gupta MD. Speech feature extraction techniques: A review. International Journal of Computer Science and Mobile Computing. 2015;**4**(3):107-114

[15] Al-Alaoui MA, Al-Kanj L, Azar J, Yaacoub E. Speech recognition using artificial neural networks and hidden Markov models. IEEE Multidisciplinary Engineering Education Magazine. 2008;**3**(3):77-86

[16] Al-Sarayreh KT, Al-Qutaish RE, Al-Kasasbeh BM. Using the sound recognition techniques to reduce the electricity consumption in highways. Journal of American Science. 2009;**5**(2):1-12

[17] Gill AS. A review on feature extraction techniques for speech processing. International Journal Of Engineering and Computer Science. 2016;**5**(10):18551-18556

[18] Othman AM, Riadh MH. Speech recognition using scaly neural networks. World academy of science. Engineering and Technology. 2008;**38**:253-258

[19] Chakroborty S, Roy A, Saha G. Fusion of a complementary feature set with MFCC for improved closed set text-independent speaker identification. In: IEEE International Conference on Industrial Technology, 2006. ICIT 2006. pp. 387-390

[20] de Lara JRC. A method of automatic speaker recognition using cepstral features and vectorial quantization. In: Iberoamerican Congress on Pattern Recognition. Berlin, Heidelberg: Springer; 2005. pp. 146-153

[21] Ravikumar KM, Reddy BA, Rajagopal R, Nagaraj HC. Automatic detection of syllable repetition in read speech for objective assessment of stuttered Disfluencies. In: Proceedings of World Academy Science, Engineering and Technology. 2008. pp. 270-273

[22] Hasan MR, Jamil M, Rabbani G, Rahman MGRMS. Speaker Identification Using Mel Frequency cepstral coefficients. In: 3rd International Conference on Electrical & Computer Engineering, 2004. ICECE 2004. pp. 28-30

[23] Chu S, Narayanan S, Kuo CC. Environmental sound recognition using MP-based features. In: IEEE International Conference on Acoustics, Speech and Signal Processing, 2008. ICASSP 2008. IEEE; 2008. pp. 1-4

[24] Rao TB, Reddy PPVGD, Prasad A. Recognition and a panoramic view of Raaga emotions of singers-application Gaussian mixture model. International Journal of Research and Reviews in Computer Science (IJRRCS). 2011;**2**(1):201-204

[25] Agrawal S, Shruti AK, Krishna CR. Prosodic feature based text dependent speaker recognition using machine learning algorithms. International Journal of Engineering Science and Technology. 2010;**2**(10):5150-5157

[26] Paulraj MP, Sazali Y, Nazri A, Kumar S. A speech recognition system for Malaysian English pronunciation using neural network. In: Proceedings of the International Conference on Man-Machine Systems (ICoMMS). 2009

[27] Tan CL, Jantan A. Digit recognition using neural networks. Malaysian Journal of Computer Science. 2004;**17**(2):40-54

[28] Kumar P, Chandra M. Speaker identification using Gaussian mixture models. MIT International Journal of Electronics and Communication Engineering. 2011;**1**(1):27-30

[29] Wang TT, Quatieri TF. High-pitch formant estimation by exploiting temporal change of pitch. IEEE Transactions on Audio, Speech, and Language Processing. 2010;**18**(1):171-186

[30] El Choubassi MM, El Khoury HE, Alagha CEJ, Skaf JA, Al-Alaoui MA. Arabic speech recognition using recurrent neural networks. In: Proceedings of the 3rd IEEE International Symposium on Signal Processing and Information Technology (IEEE Cat. No.03EX795). Ieee; 2003. pp. 543-547. DOI: 10.1109/ISSPIT.2003.1341178

[31] Wu QZ, Jou IC, Lee SY. On-line signature verification using LPC cepstrum and neural networks. IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics. 1997; **27**(1):148-153

[32] Holambe R, Deshpande M. Advances in Non-Linear Modeling for Speech Processing. Berlin, Heidelberg: Springer Science & Business Media; 2012

[33] Nehe NS, Holambe RS. DWT and LPC based feature extraction methods for isolated word recognition. EURASIP Journal on Audio, Speech, and Music Processing. 2012;**2012**(1):7

[34] Young S, Evermann G, Gales M, Hain T, Kershaw D, Liu X, et al. The HTK Book, Version 3.4. Cambridge, United Kingdom: Cambridge University; 2006

[35] Ismail S, Ahmad A. Recurrent neural network with backpropagation through time algorithm for arabic recognition. In: Proceedings of the 18th European Simulation Multiconference (ESM). Magdeburg, Germany; 2004. pp. 13-16

[36] McLoughlin IV. Line spectral pairs. Signal Processing. 2008;**88**(3):448-467

[37] Itakura F. Line spectrum representation of linear predictor coefficients of speech signals. The Journal of the Acoustical Society of America. 1975;**57**(S1):S35-S35

[38] Silva DF, de Souza VM, Batista GE, Giusti R. Spoken digit recognition in portuguese using line spectral frequencies. Ibero-American Conference on Artificial Intelligence. Vol. 7637. Berlin, Heidelberg: Springer; 2012. pp. 241-250

[39] Kabal P, Ramachandran RP. The computation of line spectral frequencies using Chebyshev polynomials. IEEE Transactions on Acoustics, Speech and Signal Processing. 1986;**34**(6):1419-1426

[40] Paliwal KK. On the use of line spectral frequency parameters for speech recognition. Digital Signal Processing. 1992;**2**(2):80-87

[41] Alang Rashid NK, Alim SA, Hashim NNWNH, Sediono W. Receiver operating characteristics measure for the recognition of stuttering Dysfluencies using line spectral frequencies. IIUM Engineering Journal. 2017;**18**(1):193-200

[42] Kleijn WB, Bäckström T, Alku P. On line spectral frequencies. IEEE Signal Processing Letters. 2003;**10**(3):75-77

[43] Bäckström T, Pedersen CF, Fischer J, Pietrzyk G. Finding line spectral frequencies using the fast Fourier transform. In: 2015 IEEE International Conference on in Acoustics, Speech and Signal Processing (ICASSP). 2015. pp. 5122-5126

[44] Nematollahi MA, Vorakulpipat C, Gamboa Rosales H. Semifragile speech watermarking based on least significant bit replacement of line spectral frequencies. Mathematical Problems in Engineering. 2017. 9 p

[45] Oliveira MO, Bretas AS. Application of discrete wavelet transform for differential protection of power transformers. In: IEEE PowerTech. Bucharest: IEEE; 2009. pp. 1-8

[46] Gupta D, Choubey S. Discrete wavelet transform for image processing. International Journal of Emerging Technology and Advanced Engineering. 2015;**4**(3):598-602

[47] Lindsay RW, Percival DB, Rothrock DA. The discrete wavelet transform and the scale analysis of the surface properties of sea ice. IEEE Transactions on Geoscience and Remote Sensing. 1996;**34**(3):771-787

[48] Turner C, Joseph A. A wavelet packet and mel-frequency cepstral coefficients-based feature extraction method for speaker identification. In: Procedia Computer Science. 2015. pp. 416-421

[49] Reig-Bolaño R, Marti-Puig P, Solé-Casals J, Zaiats V, Parisi V. Coding of biosignals using the discrete wavelet decomposition. In: International Conference on Nonlinear Speech Processing. Berlin Heidelberg: Springer; 2009. pp. 144-151

[50] Tufekci Z, Gowdy JN. Feature extraction using discrete wavelet transform for speech recognition. In: IEEE Southeastcon 2000. 2000. pp. 116-123

[51] Gałka J, Ziółko M. Wavelet speech feature extraction using mean best basis algorithm. In: International Conference on Nonlinear Speech Processing Berlin. Heidelberg: Springer; 2009. pp. 128-135

[52] Hermansky H. Perceptual linear predictive (PLP) analysis of speech. The Journal of the Acoustical Society of America. 1990;**87**(4):1738-1752

[53] Picone J. Fundamentals of Speech Recognition: Spectral Transformations. 2011. Retrieved from: http://www.isip.piconepress.com/publications/courses/msstate/ece_8463/lectures/current/lecture_17/lecture_17.pdf

[54] Thomas S, Ganapathy S, Hermansky H. Spectro-temporal features for automatic speech recognition using linear prediction in spectral domain. In: Proceedings of the 16th European Signal Processing Conference (EUSIPCO 2008), Lausanne, Switzerland. 2008

Chapter 2

Convolutional Neural Networks for Raw Speech Recognition

Vishal Passricha and Rajesh Kumar Aggarwal

Additional information is available at the end of the chapter

http://dx.doi.org/10.5772/intechopen.80026

Abstract

State-of-the-art automatic speech recognition (ASR) systems map the speech signal into its corresponding text. Traditional ASR systems are based on Gaussian mixture model. The emergence of deep learning drastically improved the recognition rate of ASR systems. Such systems are replacing traditional ASR systems. These systems can also be trained in end-to-end manner. End-to-end ASR systems are gaining much popularity due to simplified model-building process and abilities to directly map speech into the text without any predefined alignments. Three major types of end-to-end architectures for ASR are attention-based methods, connectionist temporal classification, and convolutional neural network (CNN)-based direct raw speech model. In this chapter, CNN-based acoustic model for raw speech signal is discussed. It establishes the relation between raw speech signal and phones in a data-driven manner. Relevant features and classifier both are jointly learned from the raw speech. Raw speech is processed by first convolutional layer to learn the feature representation. The output of first convolutional layer, that is, intermediate representation, is more discriminative and further processed by rest convolutional layers. This system uses only few parameters and performs better than traditional cepstral feature-based systems. The performance of the system is evaluated for TIMIT and claimed similar performance as MFCC.

Keywords: ASR, attention-based model, connectionist temporal classification, CNN, end-to-end model, raw speech signal

1. Introduction

ASR system has two important tasks—phoneme recognition and whole-word decoding. In ASR, the relationship between the speech signal and phones is established in two different steps [1]. In the first step, useful features are extracted from the speech signal on the basis of

prior knowledge. This phase is known as information selection or dimensionality reduction phase. In this, the dimensionality of the speech signal is reduced by selecting the information based on task-specific knowledge. Highly specialized features like MFCC [2] are preferred choice in traditional ASR systems. In the second step, discriminative models estimate the likelihood of each phoneme. In the last, word sequence is recognized using discriminative programming technique. Deep learning system can map the acoustic features into the spoken phonemes directly. A sequence of the phoneme is easily generated from the frames using frame-level classification.

Another side, end-to-end systems perform acoustic frames to phone mapping in one step only. End-to-end training means all the modules are learned simultaneously. Advanced deep learning methods facilitate to train the system in an end-to-end manner. They also have the ability to train the system directly with raw signals, i.e., without hand-crafted features. Therefore, ASR paradigm is shifting from cepstral features like MFCC [2], PLP [3] to discriminative features learned directly from raw speech. End-to-end model may take raw speech signal as input and generates phoneme class conditional probabilities as output. The three major types of end-to-end architectures for ASR are attention-based method, connectionist temporal classification (CTC), and CNN-based direct raw speech model.

Attention-based models directly transcribe the speech into phonemes. Attention-based encoder-decoder uses the recurrent neural network (RNN) to perform sequence-to-sequence mapping without any predefined alignment. In this model, the input sequence is first transformed into a fixed length vector representation, and then decoder maps this fixed length vector into the output sequence. Attention-based encoder-decoder is much capable of learning the mapping between variable-length input and output sequences. Chorowski and Jaitly proposed speaker-independent sequence-to-sequence model and achieved 10.6% WER without separate language models and 6.7% WER with a trigram language model for Wall Street Journal dataset [4]. In attention-based systems, the alignment between the acoustic frame and recognized symbols is performed by attention mechanism, whereas CTC model uses conditional independence assumptions to efficiently solve sequential problems by dynamic programming. Attention model has shown high performance over CTC approach because it uses the history of the target character without any conditional independence assumptions.

Another side, CNN-based acoustic model is proposed by Palaz et al. [5–7] which processes the raw speech directly as input. This model consists of two stages: feature learning stage, i.e., several convolutional layers, and classifier stage, i.e., fully connected layers. Both the stages are learned jointly by minimizing a cost function based on relative entropy. In this model, the information is extracted by the filters at first convolutional layer and modeled between first and second convolutional layer. In classifier stage, learned features are classified by fully connected layers and softmax layer. This approach claims comparable or better performance than traditional cepstral feature-based system followed by ANN training for phoneme recognition on TIMIT dataset.

This chapter is organized as follows: In Section 2, the work performed in the field of ASR is discussed with the name of related work. Section 3 covers the various architectures of ASR. Section 4 presents the brief introduction about CNN. Section 5 explains CNN-based direct raw

speech recognition model. In Section 6, available experimental results are shown. Finally, Section 7 concludes this chapter with the brief discussion.

2. Related work

Traditional ASR system leveraged the GMM/HMM paradigm for acoustic modeling. GMM efficiently processes the vectors of input features and estimates emission probabilities for each HMM state. HMM efficiently normalizes the temporal variability present in speech signal. The combination of HMM and language model is used to estimate the most likely sequence of phones. The discriminative objective function is used to improve the recognition rate of the system by the discriminatively fine-tuned methods [8]. However, GMM has a shortcoming as it shows inability to model the data that is present on the boundary line. Artificial neural networks (ANNs) can learn much better models of data lying on the boundary condition. Deep neural networks (DNNs) as acoustic models tremendously improved the performance of ASR systems [9–11]. Generally, discriminative power of DNN is used for phoneme recognition and, for decoding task, HMM is preferred choice. DNNs have many hidden layers with a large number of nonlinear units and produce a very large number of outputs. The benefit of this large output layer is that it accommodates the large number of HMM states. DNN architectures have densely connected layers. Therefore, such architectures are more prone to overfitting. Secondly, features having the local correlations become difficult to learn for such architectures. In [12], speech frames are classified into clustered context-dependent states using DNNs. In [13, 14], GMM-free DNN training process is proposed by the researchers. However, GMM-free process demands iterative procedures like decision trees, generating forced alignments. DNN-based acoustic models are gaining much popularity in large vocabulary speech recognition task [10], but components like HMM and n-gram language model are same as in their predecessors.

GMM or DNN-based ASR systems perform the task in three steps: feature extraction, classification, and decoding. It is shown in **Figure 1**. Firstly, the short-term signal s_t is processed at time "t" to extract the features x_t. These features are provided as input to GMM or DNN acoustic model which estimates the class conditional probabilities $P_e(i|x_i)$ for each phone class $i \in \{1, \ldots, I\}$. The emission probabilities are as follows:

$$p_e(x_t|i) \propto \frac{p(x_t|i)}{p(x_t)} = \frac{P(i|x_t)}{p(i)} \quad \forall_i \in i, \ldots, I \tag{1}$$

The prior class probability $p(i)$ is computed by counting on the training set.

DNN is a feed-forward NN containing multiple hidden layers with a large number of hidden units. DNNs are trained using the back-propagation methods then discriminatively fine-tuned for reducing the gap between the desired output and actual output. DNN-/HMM-based hybrid systems are the effective models which use a tri-phone HMM model and an n-gram language model [10, 15]. Traditional DNN/HMM hybrid systems have several independent components that are trained separately like an acoustic model, pronunciation model, and

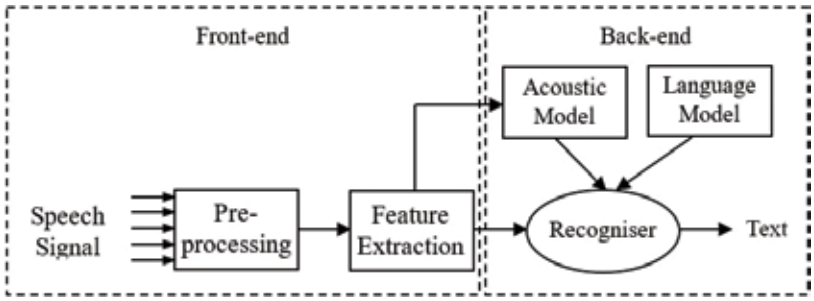


Figure 1. General framework of automatic speech recognition system.

Figure 2. Hybrid DNN/HMM phoneme recognition.

language model. In the hybrid model, the speech recognition task is factorized into several independent subtasks. Each subtask is independently handled by a separate module which simplifies the objective. The classification task is much simpler in HMM-based models as compared to classifying the set of variable-length sequences directly. **Figure 2** shows the hybrid DNN/HMM phoneme recognition model.

On the other side, researchers proposed end-to-end ASR systems that directly map the speech into labels without any intermediate components. As the advancements in deep learning, it has become possible to train the system in an end-to-end fashion. The high success rate of deep learning methods in vision task motivates the researchers to focus on classifier step for speech recognition. Such architectures are called deep because they are composed of many layers as compared to classical "shallow" systems. The main goal of end-to-end ASR system is to simplify the conventional module-based ASR system into a single deep learning framework. In earlier systems, divide and conquer approaches are used to optimize each step independently, whereas deep learning approaches have a single architecture that leads to more optimal system. End-to-end speech recognition systems directly map the speech to text without requiring predefined alignment between acoustic frame and characters [16–24]. These systems are generally divided into three broad categories: attention-based model [19–22], connectionist temporal classification [16–18, 25], and CNN-based direct raw speech method [5–7, 26]. All these models have a capability to address the problem of variable-length input and output sequences.

Attention-based models are gaining much popularity in a variety of tasks like handwriting synthesis [27], machine translation [28], and visual object classification [29]. Attention-based models directly map the acoustic frame into character sequences. However, this model differs from other machine translation tasks by requesting much longer input sequences. This model generates a character based on the inputs and history of the target character. The

http://dx.doi.org/10.5772/intechopen.80026

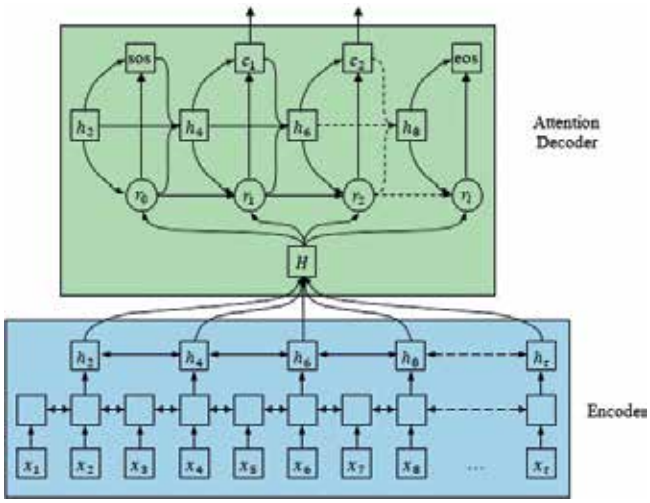


Figure 3. Attention-based ASR model.

attention-based models use encoder-decoder architecture to perform the sequence mapping from speech feature sequences to text as shown in **Figure 3**. Its extension, i.e., attention-based recurrent networks, has also been successfully applied to speech recognition. In the noisy environment, these models' results are poor because the estimated alignment is easily corrupted by noise. Another issue with this model is that it is hard to train from scratch due to misalignment on longer input sequences. Sequence-to-sequence networks have also achieved many breakthroughs in speech recognition [20–22]. They can be divided into three modules: an encoding module that transforms sequences, attention module that estimates the alignment between the hidden vector and targets, and decoding module that generates the output sequence. To develop successful sequence-to-sequence model, the understanding and preventing limitations are required. The discriminative training is a different way of training that raises the performance of the system. It allows the model to focus on most informative features with the risk of overfitting.

End-to-end trainable speech recognition systems are an important application of attention-based models. The decoder network computes a matching score between hidden states generated by the acoustic encoder network at each input time. It processes its hidden states to form a temporal alignment distribution. This matching score is used to estimate the corresponding encoder states. The difficulty of attention-based mechanism in speech recognition is that the feature inputs and corresponding letter outputs generally proceed in the same order with only small deviations within word. However, the different length of input and output sequences makes it more difficult to track the alignment. The advantage of attention-based mechanism is that any conditional independence assumptions (Markov assumption) are not required in this mechanism. Attention-based approach replaces the HMM with RNN to perform the sequence prediction. Attention mechanism automatically learns alignment between the input features and desired character sequence.

CTC techniques infer the speech-label alignment automatically. CTC [25] was developed for decoding the language. Firstly, Hannun et al. [17] used it for decoding purpose in Baidu's deep speech network. CTC uses dynamic programming [16] for efficient computation of a strictly monotonic alignment. However, graph-based decoding and language model are required for it. CTC approaches use RNN for feature extraction [28]. Graves et al. [30] used its objective function in deep bidirectional long short-term memory (LSTM) system. This model successfully arranges all possible alignments between input and output sequences during model training, not on the prior.

Two different versions of beam search are adopted by [16, 31] for decoding CTC models. **Figure 4** shows the working architecture of the CTC model. In this, noisy and not informative frames are discarded by the introduction of the blank label which results in the optimal output sequence. CTC uses intermediate label representation to identify the blank labels, i.e., no output labels. CTC-based NN model shows high recognition rate for both phoneme recognition [32] and LVCSR [16, 31]. CTC-trained neural network with language model offers excellent results [17].

End-to-end ASR systems perform well and achieve good results, yet they face two major challenges. First is how to incorporate lexicons and language models into decoding. However, [16, 31, 33] have incorporated lexicons for searching paths. Second, there is no shared experimental platform for the purpose of benchmark. End-to-end systems differ from the traditional system in both aspects: model architecture and decoding methods. Some efforts were also made to model the raw speech signal with little or no preprocessing [34]. Palaz et al. [6] showed in his study that CNN [35] can calculate the class conditional probabilities from raw speech signal as direct input. Therefore, CNNs are the preferred choice to learn features from the raw speech. Two stages of learned feature process are as follows: initially, features are learned by the filters at first convolutional layer, and then learned features are modeled by second and higher-level convolutional layers. An end-to-end phoneme sequence

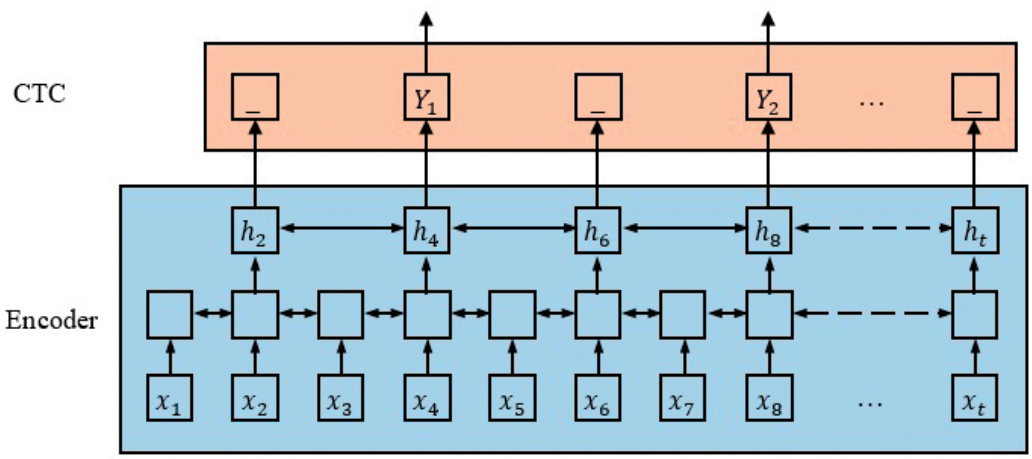


Figure 4. CTC model for speech recognition.

http://dx.doi.org/10.5772/intechopen.80026

recognizer directly processes the raw speech signal as inputs and produces a phoneme sequence. The end-to-end system is composed of two parts: convolutional neural networks and conditional random field (CRF). CNN is used to perform the feature learning and classification, and CRFs are used for the decoding stage. CRF, ANN, multilayer perceptron, etc. have been successfully used as decoder. The results on TIMIT phone recognition task also confirm that the system effectively learns the features from raw speech and performs better than traditional systems that take cepstral features as input [36]. This model also produces good results for LVCSR [7].

3. Various architectures of ASR

In this section, a brief review on conventional GMM/DNN ASR, attention-based end-to-end ASR, and CTC is given.

3.1. GMM/DNN

ASR system performs sequence mapping of T-length speech sequence features, $X = \{X_t \in \mathbb{R}^D | t = 1, \ldots, T\}$, into an N-length word sequence, $W = \{w_n \in v | n = 1, \ldots, N\}$ where X_t represents the D-dimensional speech feature vector at frame t and w_n represents the word at position n in the vocabulary, v.

The ASR problem is formulated within the Bayesian framework. In this method, an utterance is represented by some sequence of acoustic feature vector X, derived from the underlying sequence of words W, and the recognition system needs to find the most likely word sequence as given below [37]:

$$\hat{W} = \arg\max_{w} p(W|X) \tag{2}$$

In Eq. (2), the argument of $p(W|X)$, that is, the word sequence W, is found which shows maximum probability for given feature vector, X. Using Bayes' rule, it can be written as

$$\hat{W} = \arg\max_{w} \frac{p(X|W)p(W)}{p(X)} \tag{3}$$

In Eq. (3), the denominator $p(X)$ is ignored as it is constant with respect to W. Therefore,

$$\hat{W} = \arg\max_{w} p(X|W)p(W) \tag{4}$$

where $p(X|W)$ represents the sequence of speech features and it is evaluated with the help of acoustic model. $p(W)$ represents the prior knowledge about the sequence of words W and it is determined by the language model. However, current ASR systems are based on a hybrid

HMM/DNN [38], which is also calculated using Bayes' theorem and introduces the HMM state sequence S, to factorize $p(W|X)$ into the following three distributions:

$$\arg \max_{w \in v^*} p(W|X) \tag{5}$$

$$= \arg \max_{w \in v^*} \sum_{S} p(X|S, W)p(S|W)p(W) \tag{6}$$

$$\approx \arg \max_{w \in v^*} \sum_{S} p(X|S), p(S|W)p(W) \tag{7}$$

where $p(X|S)$, $p(S|W)$, and $p(W)$ represent acoustic, lexicon, and language models, respectively. Equation (6) is changed into Eq. (7) in a similar way as Eq. (4) is changed into Eq. (5).

3.1.1. Acoustic models $p(X|S)$

$p(X|S)$ can be further factorized using a probabilistic chain rule and Markov assumption as follows:

$$p(X|S) = \prod_{t=1}^{T} p(x_t|x_1, \ldots, x_{t-1}, S) \tag{8}$$

$$\approx \prod_{t=1}^{T} p(x_t|s_t) \propto \prod_{t=1}^{T} \frac{p(s_t|x_t)}{p(s_t)} \tag{9}$$

In Eq. (9), framewise likelihood function $p(x_t|s_t)$ is changed into the framewise posterior distribution $\frac{p(s_t|x_t)}{p(s_t)}$ which is computed using DNN classifiers by pseudo-likelihood trick [38]. In Eq. (9), Markov assumption is too strong. Therefore, the contexts of input and hidden states are not considered. This issue can be resolved using either the recurrent neural networks (RNNs) or DNNs with long-context features. A framewise state alignment is required to train the framewise posterior which is offered by an HMM/GMM system.

3.1.2. Lexicon model $p(S|W)$

$p(S|W)$ can be further factorized using a probabilistic chain rule and Markov assumption (first order) as follows:

$$p(S|W) = \prod_{t=1}^{T} p(s_t|s_1, \ldots, s_{t-1}, W) \tag{10}$$

$$\approx \prod_{t=1}^{T} p(s_t|\, s_{t-1}, W) \tag{11}$$

An HMM state transition represents this probability. A pronunciation dictionary performs the conversion from w to HMM states through phoneme representation.

3.1.3. Language model $p(W)$

Similarly, $p(W)$ can be factorized using a probabilistic chain rule and Markov assumption ((m-1) th order) as an m-gram model, i.e.,

$$p(W) = \prod_{n=1}^{N} p(w_n|w_1, \ldots, w_{n-1}) \tag{12}$$

$$\approx \prod_{n=1}^{N} p(w_n|w_{n-m-1}, \ldots, w_{n-1}) \tag{13}$$

The issue of Markov assumption is addressed using recurrent neural network language model (RNNLM) [39], but it increases the complexity of decoding process. The combination of RNNLMs and m-gram language model is generally used and it works on a rescoring technique.

3.2. Attention mechanism

The approach based on attention mechanism does not make any Markov assumptions. It directly finds the posterior $p(C|X)$, on the basis of a probabilistic chain rule:

$$p(C|X) = \underbrace{\prod_{l=1}^{L} p(c_l|c_1, \ldots, c_{l-1}, X)}_{\triangleq p_{att}(C/X)} \tag{14}$$

where $p_{att}(C|X)$ represents an attention-based objective function. $p(c_l|c_1, \ldots, c_{l-1}, X)$ is obtained by

$$\boldsymbol{h}_t = Encoder(X), \tag{15}$$

$$a_{lt} = \begin{cases} ContentAttention\left(\boldsymbol{q}_{l-1}, \boldsymbol{h}_t\right) \\ LocationAttention\left(\{a_{l-1}\}_{t=1}^{T}, \boldsymbol{q}_{l-1}, \boldsymbol{h}_t\right) \end{cases}, \tag{16}$$

$$\boldsymbol{r}_l = \sum_{t=1}^{T} a_{lt} \boldsymbol{h}_t, \tag{17}$$

$$p(c_l|c_1, \ldots, c_{l-1}, X) = Decoder\left(\boldsymbol{r}_l, \boldsymbol{q}_{l-1}, c_{l-1}\right) \tag{18}$$

Eq. (15) represents the encoder and Eq. (18) represents the decoder networks. a_{lt} represents the soft alignment of the hidden vector, $\boldsymbol{h}_t$. Here, $\boldsymbol{r}_l$ represents the weighted letter-wise hidden vector that is computed by weighted summation of hidden vectors. Content-based attention mechanism with or without convolutional features are shown by $ContentAttention(.)$ and $LocationAttention(.)$, respectively.

3.2.1. Encoder network

The input feature vector X is converted into a framewise hidden vector, $\boldsymbol{h}_t$ using Eq. (15). The preferred choice for an encoder network is BLSTM, i.e.,

$$Encoder(X) \triangleq BLSTM_t(X) \tag{19}$$

It is to be noted that the computational complexity of the encoder network is reduced by subsampling the outputs [20, 21].

3.2.2. Content-based attention mechanism

$ContentAttention(.)$ is shown as

$$e_{lt} = g^T \tanh\left(Lin\left(\boldsymbol{q}_{l-1}\right) + LinB(\boldsymbol{h}_t)\right) \tag{20}$$

$$a_{lt} = Softmax\left(\{e_{lt}\}_{t=1}^{T}\right) \tag{21}$$

g represents a learnable parameter. $\{e_{lt}\}_{t=1}^{T}$ represents a T-dimensional vector. $\tanh(.)$ and $Lin(.)$ represent the hyperbolic tangent activation function and linear layer with learnable matrix parameters, respectively.

3.2.3. Location-aware attention mechanism

It is an extended version of content-based attention mechanism to deal with the location-aware attention. If $a_{l-1} = \{a_{l-1}\}_{t=1}^{T}$ is replaced in Eq. (16), then $LocationAware(.)$ is represented as follows:

$$\left\{\boldsymbol{f}_t\right\}_{t=1}^{T} = \boldsymbol{\mathcal{K}} * \boldsymbol{a}_{l-1} \tag{22}$$

$$e_{lt} = g^T \tanh\left(Lin\left(\boldsymbol{q}_{l-1}\right) + Lin(\boldsymbol{h}_t) + LinB\left(\boldsymbol{f}_t\right)\right) \tag{23}$$

$$a_{lt} = softmax\left(\{e_t\}_{t=1}^{T}\right) \tag{24}$$

Here, * denotes 1-D convolution along the input feature axis, t, with the convolution parameter, $\boldsymbol{\mathcal{K}}$, to produce the set of T features $\left\{f_t\right\}_{t=1}^{T}$.

3.2.4. Decoder network

The decoder network is an RNN that is conditioned on previous output C_{l-1} and hidden vector $\boldsymbol{q}_{l-1}$. LSTM is preferred choice of RNN that represented as follows:

$$Decoder(.) \triangleq softmax(LinB(LSTM_l(.)) \tag{25}$$

$LSTM_l(.)$ represents uniconditional LSTM that generates hidden vector $\boldsymbol{q}_l$ as output:

$$\boldsymbol{q}_l = LSTM_l\left(\boldsymbol{r}_l, \boldsymbol{q}_{l-1}, c_{l-1}\right) \tag{26}$$

$\boldsymbol{r}_l$ represents the concatenated vector of the letter-wise hidden vector; c_{l-1} represents the output of the previous layer which is taken as input.

http://dx.doi.org/10.5772/intechopen.80026

3.2.5. Objective function

The objective function of the attention model is computed from the sequence posterior

$$p_{att}(C|X) \approx \prod_{l=1}^{L} p\left(c_l|c_1^*, \ldots, c_{l-1}^*, X\right) \triangleq p_{att}^*(C|X) \tag{27}$$

where c_l^* represents the ground truth of the previous characters. Attention-based approach is a combination of letter-wise objectives based on multiclass classification with the conditional ground truth history $c_l^*, \ldots, c_{l-1}^*$ in each output l.

3.3. Connectionist temporal classification (CTC)

The CTC formulation is also based on Bayes' decision theory. It is to be noted that L-length letter sequence,

$$C' = \{< b >, c_1, < b >, c_2, < b >, \ldots, c_L, < b >\} = \left\{c_l' \in \mathcal{U} \cup \{< b >\} | l = 1, \ldots, 2L+1\right\} \tag{28}$$

In C', c_l' is always $" < b > "$ and letter when l is an odd and an even number, respectively. Similar as DNN/HMM model, framewise letter sequence with an additional blank symbol

$$Z = \{z_t \in \mathcal{U} \cup \{< b >\} | t = 1, \ldots, T\} \tag{29}$$

is also introduced. The posterior distribution, $p(C|X)$, can be factorized as

$$p(C|X) = \sum_z p(C|Z, X) p(Z|X) \tag{30}$$

$$\approx \sum_z p(C|Z).p(Z|X) \tag{31}$$

Same as Eq. (3), CTC also uses Markov assumption, i.e., $p(C|Z, X) \approx p(C|Z)$, to simplify the dependency of the CTC acoustic model, $p(Z|X)$, and CTC letter model, $p(C|Z)$.

3.3.1. CTC acoustic model

Same as DNN/HMM acoustic model, $p(Z|X)$ can be further factorized using a probabilistic chain rule and Markov assumption as follows:

$$p(Z|X) = \prod_{t=1}^{T} p(z_t|z_1, \ldots, z_{t-1}, X) \tag{32}$$

$$\approx \prod_{t=1}^{T} p(z_t|\ X) \tag{33}$$

The framewise posterior distribution, $p(z_t|X)$ is computed from all inputs, X, and it is directly modeled using bidirectional LSTM [30, 40]:

$$p(z_t|X) = Softmax(LinB(\boldsymbol{h}_t)), \tag{34}$$

$$\boldsymbol{h}_t = BLSTM_t(X) \tag{35}$$

where $Softmax(.)$ represents the softmax activation function. $LinB(.)$ is used to convert the hidden vector, $\boldsymbol{h}_t$, to a $(|\mathcal{U}|+1)$ dimensional vector with learnable matrix and bias vector parameter. $BLSTM_t(.)$ takes full input sequence as input and produces hidden vector $(\boldsymbol{h}_t)$ at t.

3.3.2. CTC letter model

By applying Bayes' decision theory probabilistic chain rule and Markov assumption, $p(Z|X)$ can be written as

$$p(C/Z) = \frac{p(Z/C)p(C)}{p(Z)} \tag{36}$$

$$= \prod_{t=1}^{T} p(z_t|z_1, \ldots, z_{t-1}, C)\frac{p(C)}{p(Z)} \tag{37}$$

$$\approx \prod_{t=1}^{T} p(z_t|z_{t-1}, C)\frac{p(C)}{p(Z)} \tag{38}$$

where $p(z_t|z_{t-1}, C)$ represents state transition probability. $p(C)$ represents letter-based language model, and $p(Z)$ represents the state prior probability. CTC architecture incorporates letter-based language model. CTC architecture can also incorporate a word-based language model by using letter-to-word finite state transducer during decoding [18]. The CTC has the monotonic alignment property, i.e.,

when $z_{t-1} = c'_m$, then $z_t = c'_l$ where $l \geq m$.

Monotonic alignment property is an important constraint for speech recognition, so ASR sequence-to-sequence mapping should follow the monotonic alignment. This property is also satisfied by HMM/DNN.

3.3.3. Objective function

The posterior, $p(C|X)$, is represented as

$$p(C|X) \approx \underbrace{\sum_z \prod_{t=1}^{T} p(z_t|z_{t-1}, C)p(z_t|X)}_{\triangleq p_{ctc}(C/X)} . \frac{p(C)}{p(Z)} \tag{39}$$

Viterbi method and forward-backward algorithm are dynamic programming algorithm which is used to efficiently compute the summation over all possible Z. CTC objective function $p_{CTC}(C|X)$ is designed by excluding the $p(C)/p(Z)$ from Eq. (23).

The CTC formulation is also same as HMM/DNN. The minute difference is that Bayes' rule is applied to $p(C|Z)$ instead of $p(W|X)$. It has also three distribution components like HMM/DNN, i.e., framewise posterior distribution, $p(z_t|X)$; transition probability, $p(z_t|z_{t-1}, C)$; and letter model, $p(C)$. It also uses Markov assumption. It does not fully utilize the benefit of end-to-end ASR, but its character output representation still possesses the end-to-end benefits.

4. Convolutional neural networks

CNNs are the popular variants of deep learning that are widely adopted in ASR systems. CNNs have many attractive advancements, i.e., weight sharing, convolutional filters, and pooling. Therefore, CNNs have achieved an impressive performance in ASR. CNNs are composed of multiple convolutional layers. **Figure 5** shows the block diagram of CNN. LeCun and Bengio [41] describe the three states of convolutional layer, i.e., convolution, pooling, and nonlinearity.

Deep CNNs set a new milestone by achieving approximate human level performance through advanced architectures and optimized training [42]. CNNs use nonlinear function to directly process the low-level data. CNNs are capable of learning high-level features with high complexity and abstraction. Pooling is the heart of CNNs that reduces the dimensionality of a feature map. Maxout is widely used nonlinearity and has shown its effectiveness in ASR tasks [43, 44].

Pooling is an important concept that transforms the joint feature representation into the valuable information by keeping the useful information and eliminating insignificant information. Small frequency shifts that are common in speech signal are efficiently handled using pooling. Pooling also helps in reducing the spectral variance present in the input speech. It maps the input from p adjacent units into the output by applying a special function. After the element-wise nonlinearities, the features are passed through pooling layer. This layer executes the downsampling on the feature maps coming from previous layer and produces the new feature maps with a condensed resolution. This layer drastically reduces the spatial dimension of input. It serves the two main purposes. The first is that the amount of parameters or weight is reduced by 65%, thus lessening the computational cost. The second is that it controls the overfitting. This term refers to when a model is so tuned to the training examples.

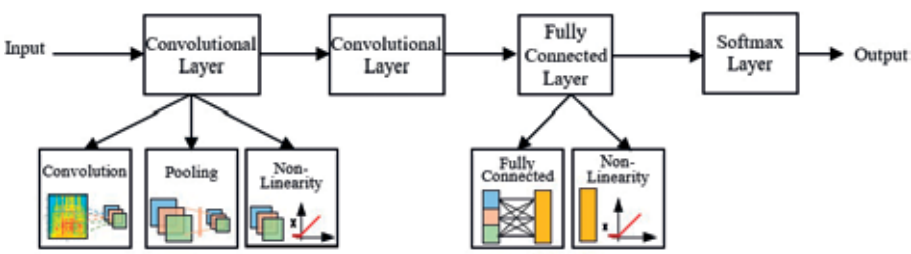


Figure 5. Block diagram of convolutional neural network.

5. CNN-based end-to-end approach

A novel acoustic model based on CNN is proposed by Palaz et al. [5] which is shown in **Figure 6**. In this, raw speech signal is segmented into input speech signal $s_t^c = \{s_{t-c}, \ldots, s_t, \ldots, s_{t+c}\}$ in the context of 2c frames having spanning window w_{in} milliseconds. First convolutional layer learns the useful features from the raw speech signal, and remaining convolutional layers further process these features into the useful information. After processing the speech signal, CNN estimates the class conditional probability, i.e., $P(i/s_t^c)$, which is used to calculate emission scaled-likelihood $P(s_t^c/i)$. Several filter stages are present in the network before the classification stage. A filter stage is a combination of convolutional layer, pooling layer, and a nonlinearity. The joint training of feature stage and classifier stage is performed using the back-propagation algorithm.

The end-to-end approach employs the following understanding:

1. Speech signals are non-stationary in nature. Therefore, they are processed in a short-term manner. Traditional feature extraction methods generally use 20–40 ms sliding window size. Although in the end-to-end approach, short-term processing of signal is required. Therefore, the size of the short-term window is taken as hyperparameter which is automatically determined during training.

2. Feature extraction is a filter operation because its components like Fourier transform, discrete cosine transform, etc. are filtering operations. In traditional systems, filtering is applied on both frequency and time. So, this factor is also considered in building convolutional layer in end-to-end system. Therefore, the number of filter banks and their parameters are taken as hyperparameters that are automatically determined during training.

3. The short-term processing of speech signal spread the information across time. In traditional systems, this spread information is modeled by calculating temporal derivatives and contextual information. Therefore, intermediate representation is supplied to classifier and calculated by taking long time span of input speech signal. Therefore, w_{in}, the size of input window, is taken as hyperparameter, which is estimated during training.

The end-to-end model estimates $P(i/s_t^c)$ by processing the speech signal with minimal assumptions or prior knowledge.

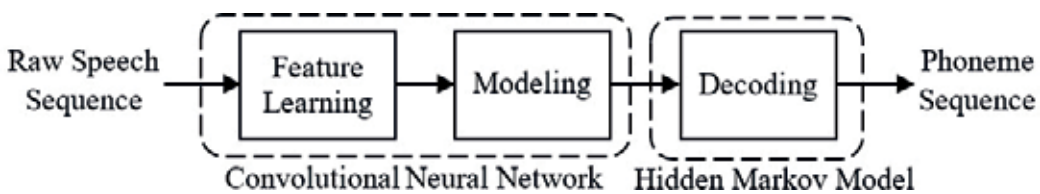


Figure 6. CNN-based raw speech phoneme recognition system.

6. Experimental results

In this model, a number of hyperparameters are used to specify the structure of the network. The number of hidden units in each hidden layer is very important; hence, it is taken as hyperparameter. w_{in} represents the time span of input speech signal. kW represents the kernel and temporal window width. dW represents the shift of temporal window. kW_{mp} represents max-pooling kernel width and dW_{mp} represents the shift of max-pooling kernel. The value of all hyperparameters is estimated during training based on frame-level classification accuracy on validation data. The range of hyperparameters after validation is shown in **Table 1**.

The experiments are conducted for three convolutional layers. The speech window size (w_{in} is taken 250 ms with a shift of temporal window (dW) 10 ms. **Table 2** shows the comparison of existing end-to-end speech recognition model in the context of PER. The results of the experiments conducted on TIMIT dataset for this model are compared with already existing techniques, and it is shown in **Table 3**. The main advantages of this model are that it uses only few parameters and offers better performance. It also increases the generalization capability of the classifiers.

Hyperparameter	Units	Range
Input window size (w_{in})	ms	100–700
Kernel width of the first ConvNet layer (kW_1)	Samples	10–90
Kernel width of the nth ConvNet layer (kW_n)	Samples	1–11
Number of filters per kernel (d_{out}t)	Filters	20–100
Max-pooling kernel width (kW_{mp})	Frames	2–6
Number of hidden units in the classifier	Units	200–1500

Table 1. Range of hyperparameter for TIMIT dataset during validation.

End-to-end speech recognition model	PER (%)
CNN-based speech recognition system using raw speech as input [7]	33.2
Estimating phoneme class conditional probabilities from raw speech signal using convolutional neural networks [36]	32.4
Convolutional neural network-based continuous speech recognition using raw speech signal [6]	32.3
End-to-end phoneme sequence recognition using convolutional neural networks [5]	27.2
CNN-based direct raw speech model	**21.9**
End-to-end continuous speech recognition using attention-based recurrent NN: First results [19]	18.57
Toward end-to-end speech recognition with deep convolutional neural networks [44]	18.2
Attention-based models for speech recognition [20]	17.6
Segmental recurrent neural networks for end-to-end speech recognition [45]	17.3

Bold value and text represent the performance of the CNN-based direct raw speech model.

Table 2. Comparison of existing end-to-end speech model in the context of PER (%).

Methods	PER (%)
GMM-/HMM-based ASR system [46]	34
CNN-based direct raw speech model	**21.9**
Attention-based models for speech recognition [20]	17.6
Segmental recurrent neural networks for end-to-end speech recognition [45]	17.3
Combining time and frequency domain convolution in convolutional neural network-Based phone recognition [47]	16.7
Phone recognition with hierarchical convolutional deep maxout networks [48]	16.5

Bold value and text represent the performance of the CNN-based direct raw speech model.

Table 3. Comparison of existing techniques with CNN-based direct raw speech model in the context of PER (%).

7. Conclusion

This chapter discusses the CNN-based direct raw speech recognition model. This model directly learns the relevant representation from the speech signal in a data-driven manner and calculates the conditional probability for each phoneme class. In this, CNN as an acoustic model consists of a feature stage and classifier stage. Both the stages are trained jointly. Raw speech is supplied as input to first convolutional layer, and it is further processed by several convolutional layers. Classifiers like ANN, CRF, MLP, or fully connected layers calculate the conditional probabilities for each phoneme class. After that decoding is performed using HMM. This model shows the similar performance as shown by MFCC-based conventional mode.

Author details

Vishal Passricha and Rajesh Kumar Aggarwal*

*Address all correspondence to: rka15969@gmail.com

National Institute of Technology, Kurukshetra, India

References

[1] Rabiner LR, Juang B-H. Fundamentals of Speech Recognition. Englewood Cliffs: PTR Prentice Hall; 1993

[2] Davis SB, Mermelstein P. Comparison of Parametric Representations for Monosyllabic Word Recognition in Continuously Spoken Sentences. Readings in Speech Recognition. Elsevier; 1990. pp. 65-74

[3] Hermansky H. Perceptual linear predictive (PLP) analysis of speech. The Journal of the Acoustical Society of America. 1990;**87**(4):1738-1752

[4] Chorowski J, Jaitly N. Towards better decoding and language model integration in sequence to sequence models. 2016. arXiv preprint arXiv:161202695

[5] Palaz D, Collobert R, Doss MM. End-to-end phoneme sequence recognition using convolutional neural networks. 2013. arXiv preprint arXiv:13122137

[6] Palaz D, Doss MM, Collobert R. Convolutional neural networks-based continuous speech recognition using raw speech signal. In: Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE International Conference on. IEEE; 2015

[7] Palaz D, Collobert R. Analysis of CNN-Based Speech Recognition System Using Raw Speech as Input. In Proceeding of Interspeech 2015 (No. EPFL-Conf-210029); 2015

[8] O'Shaughnessy D. Automatic speech recognition: History, methods and challenges. Pattern Recognition. 2008;**41**(10):2965-2979

[9] Dahl GE, Yu D, Deng L, Acero A. Context-dependent pre-trained deep neural networks for large-vocabulary speech recognition. IEEE Transactions on Audio, Speech and Language Processing. 2012;**20**(1):30-42

[10] Hinton G, Deng L, Yu D, Dahl GE, Mohamed AR, Jaitly N, et al. Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups. IEEE Signal Processing Magazine. 2012;**29**(6):82-97

[11] Seide F, Li G, Yu D. Conversational speech transcription using context-dependent deep neural networks. In: Twelfth Annual Conference of the International Speech Communication Association; 2011

[12] Abdel-Hamid O, Mohamed AR, Jiang H, Penn G. Applying convolutional neural networks concepts to hybrid NN-HMM model for speech recognition. In: 2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP); March 2012. IEEE; 2012

[13] Senior A, Heigold G, Bacchiani M, Liao H. GMM-free DNN acoustic model training. In: Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on. IEEE; 2014

[14] Bacchiani M, Senior A, Heigold G. Asynchronous, online, GMM-free training of a context dependent acoustic model for speech recognition. In: Fifteenth Annual Conference of the International Speech Communication Association; 2014

[15] Gales M, Young S. The application of hidden Markov models in speech recognition. Foundations and Trends® in Signal Processing. 2008;**1**(3):195-304

[16] Graves A, Jaitly N. Towards end-to-end speech recognition with recurrent neural networks. In: International Conference on Machine Learning; 2014

[17] Hannun A, Case C, Casper J, Catanzaro B, Diamos G, Elsen E, et al. Deepspeech: Scaling up end-to-end speech recognition. 2014. arXiv preprint. arXiv preprint arXiv:14125567

[18] Miao Y, Gowayyed M, Metze F. EESEN: End-to-end speech recognition using deep RNN models and WFST-based decoding. In: Automatic Speech Recognition and Understanding (ASRU), 2015 IEEE Workshop on. IEEE; 2015

[19] Chorowski J, Bahdanau D, Cho K, Bengio Y. End-to-end continuous speech recognition using attention-based recurrent NN: First results. 2014. arXiv preprint arXiv:14121602

[20] Chorowski JK, Bahdanau D, Serdyuk D, Cho K, Bengio Y. Attention-based models for speech recognition. In: Advances in Neural Information Processing Systems. 2015

[21] Chan W, Jaitly N, Le Q, Vinyals O. Listen, attend and spell: A neural network for large vocabulary conversational speech recognition. In: Acoustics, Speech and Signal Processing (ICASSP), 2016 IEEE International Conference on. IEEE; 2016

[22] Bahdanau D, Chorowski J, Serdyuk D, Brakel P, Bengio Y. End-to-end attention-based large vocabulary speech recognition. In: Acoustics, Speech and Signal Processing (ICASSP), 2016 IEEE International Conference on. IEEE; 2016

[23] Lu L, Zhang X, Renais S. On training the recurrent neural network encoder-decoder for large vocabulary end-to-end speech recognition. In: Acoustics, Speech and Signal Processing (ICASSP), 2016 IEEE International Conference on. IEEE; 2016

[24] Chan W, Lane I. On online attention-based speech recognition and joint Mandarin Character-Pinyin training. In: INTERSPEECH; 2016

[25] Graves A, Fernández S, Gomez F, Schmidhuber J. Connectionist temporal classification: Labelling unsegmented sequence data with recurrent neural networks. In: Proceedings of the 23rd International Conference on Machine Learning; ACM; 2006

[26] Golik P, Tüske Z, Schlüter R, Ney H. Convolutional neural networks for acoustic modeling of raw time signal in LVCSR. In: Sixteenth Annual Conference of the International Speech Communication Association. 2015

[27] Graves A. Generating sequences with recurrent neural networks. 2013. arXiv preprint arXiv:13080850

[28] Bahdanau D, Cho K, Bengio Y. Neural machine translation by jointly learning to align and translate. 2014. arXiv preprint arXiv:14090473

[29] Mnih V, Heess N, Graves A. Recurrent models of visual attention. In: Advances in Neural Information Processing Systems; 2014

[30] Graves A, Jaitly N, Mohamed AR. Hybrid speech recognition with deep bidirectional LSTM. In: Automatic Speech Recognition and Understanding (ASRU), 2013 IEEE Workshop on. IEEE; 2013

[31] Hannun AY, Maas AL, Jurafsky D, Ng AY. First-pass large vocabulary continuous speech recognition using bi-directional recurrent DNNs. 2014. arXiv preprint arXiv:14082873

[32] Graves A, Mohamed AR, Hinton G. Speech recognition with deep recurrent neural networks. In: 2013 IEEE International Conference on Acoustics, Speech and Signal Processing; May 2013. IEEE; 2013

[33] Maas A, Xie Z, Jurafsky D, Ng A. Lexicon-free conversational speech recognition with neural networks. In: Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics. Human Language Technologies; 2015

[34] Jaitly N, Hinton G. Learning a better representation of speech soundwaves using restricted boltzmann machines. In: Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on. IEEE; 2011

[35] LeCun Y. Generalization and network design strategies. Connectionism in Perspective. 1989:143-155

[36] Palaz D, Collobert R, Doss MM. Estimating phoneme class conditional probabilities from raw speech signal using convolutional neural networks. 2013. arXiv preprint arXiv:13041018

[37] Rabiner LR, Juang B-H. Speech recognition: Statistical methods. Encyclopedia of Linguistics. 2006:1-18

[38] Bourlard HA, Morgan N. Connectionist Speech Recognition: A Hybrid Approach. Vol. 247. Springer Science & Business Media; 2012

[39] Mikolov T, Karafiát M, Burget L, Černocký J, Khudanpur S. Recurrent neural network based language model. In: Eleventh Annual Conference of the International Speech Communication Association; 2010

[40] Hochreiter S, Schmidhuber J. Long short-term memory. Neural Computation. 1997;**9**(8): 1735-1780

[41] LeCun Y, Bengio Y. Convolutional networks for images, speech, and time series. The Handbook of Brain Theory and Neural Networks. 1995;**3361**(10):1995

[42] Krizhevsky A, Sutskever I, Hinton GE. Imagenet classification with deep convolutional neural networks. In: Advances in Neural Information Processing Systems; 2012

[43] Zhang X, Trmal J, Povey D, Khudanpur S. Improving deep neural network acoustic models using generalized maxout networks. In: Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on. IEEE; 2014

[44] Zhang Y, Pezeshki M, Brakel P, Zhang S, Bengio CLY, Courville A. Towards end-to-end speech recognition with deep convolutional neural networks. 2017. arXiv preprint arXiv: 170102720

[45] Lu L, Kong L, Dyer C, Smith NA, Renals S. Segmental recurrent neural networks for end-to-end speech recognition. In: INTERSPEECH 2016; 8 September 2016. ISCA; 2016

[46] Fauziya F, Nijhawan G. A Comparative study of phoneme recognition using GMM-HMM and ANN based acoustic modeling. International Journal of Computer Applications. 2014:12-16

[47] Toth L. Combining time- and frequency-domain convolution in convolutional neural network-based phone recognition. In: Acoustics, Speech and Signal Processing (ICASSP). 2014 IEEE International Conference on May 4 2014. IEEE; pp. 190-194

[48] Toth L. Phone recognition with hierarchical convolutional deep maxout networks. EURASIP Journal on Audio, Speech, and Music Processing. 2015;**2015**(1):25

Section 2

Acoustic and Learning Applications

Chapter 3

Evaluation between Virtual Acoustic Model and Real Acoustic Scenarios for Urban Representation

Josep Llorca, Héctor Zapata, Jesús Alba,
Ernest Redondo and David Fonseca

Additional information is available at the end of the chapter

http://dx.doi.org/10.5772/intechopen.78330

Abstract

Audio representation is critical for immersive virtual environments. This article presents a quasi-experiment based on architecture students evaluating the immersive impact of 3D audio in the representation of urban environments. In the framework of acoustic urban heritage preservation, a set of city squares with varying acoustic features were used as case studies in a two-step process: an objective analysis of the acoustic properties of these spaces; and the users' subjective perceptions of the virtual environment of the squares. The study shows that we can gain a better understanding of the objective parameters through the subjective views of users. Acoustic heritage can be assessed subjectively using an immersive system such as virtual reality, in which audio representation is a key factor.

Keywords: virtual reality, architect, urban planner, sound design

1. Introduction

The current chapter is based on the answers to two basic questions arising in evaluations of digital cultural resources, and more specifically, sonic cultural heritage: "what" and "how." Firstly, *what* are we evaluating when we refer to digital cultural resources? Is it a tangible issue or an intangible perception? Does it consist of a series of personal impressions or can we establish an objective parameter? The United Nations' Millennium Ecosystem Assessment was carried out from 2003 to 2005, and the subcategory of cultural heritage in Cultural Ecosystem Services (CES) was introduced over a decade later. Yet there is a lack of consensus about what cultural heritage refers to within the Ecosystem Services (ES) context [1]. Secondly, *how* should cultural heritage be evaluated: through abstract concepts or immersive

experience? It seems practically impossible to imagine an evaluation of cultural resources that is only based on abstract concepts. However, the evaluation also requires a common basis to enable comparisons of the results. This second answer supports the immersive experience as a powerful method for cultural resources evaluation.

An explanation is required to understand the methodology used in this paper. In general, people know very little about decibels of sound, and much less about sound roughness, musical clarity or speech intelligibility. Only a small group of scientists understand the operation of acoustic science. Therefore, when urban acoustic heritage is evaluated, why are ordinary people forced to refer to numbers and graphs? Surely the evaluation would be more reasonable if it were made using real sound samples? In this context, Virtual Reality (VR) provides an easy, interactive framework for ordinary people to evaluate urban acoustic heritage.

The interest in conservation of tangible and intangible cultural heritage has been rising notably in recent years. Apart from its own value, cultural heritage fosters economic and social growth. The Heritage Research National Plan, drawn up by the Spanish Cultural Heritage Institute, highlights the importance of cultural heritage as a local development engine and a stimulus for tourism, and its relevance as a generator of culture and knowledge. However, the Plan also stresses the complexity of research in this field, due to a range of characteristics and problems, and because of the high number of factors involved that make it necessary to apply human and experimental sciences in interdisciplinary teams.

1.1. Evaluation of acoustic quality in outdoor spaces

The evaluation of the acoustic quality of a space is fundamental to determine possible interventions in it and the suitability of its future uses. Several studies have established optimal indexes and ranges for the various measurable parameters [2, 3]. Nevertheless, as in current regulations, the focus has been concert halls, which have different features and requirements from outdoor spaces.

Very few objective and subjective tests have been undertaken in these kinds of spaces, due to the difficulty in installing measuring instruments and the variable conditions of the environments. In this study, four outdoor spaces were tested and a great effort was made to find the best environmental conditions. Preliminary work was done in the studied environments [4].

The application of new technologies in cultural heritage is a practice that has become increasingly widespread. The construction of virtual models allows us to reproduce environments for their study, avoiding direct intervention in these spaces and encouraging their conservation. After some data collection in the actual place, a model is designed and calibrated in which the environment can be recreated as many times as desired, without the need to travel there. This methodology could overcome the major difficulty that an in-place test might present.

Some authors have attempted to investigate urban sound propagation. They have centered on the complexity of the medium: irregular faces, interconnection with adjacent canyons, and a large variety of materials and boundary conditions. Moreover, a predominant characteristic of the urban environment is that it is open to the sky, and induces large radiative losses [5–7]. Much of the literature is focused on propagation in a single urban canyon [8–13]. A few authors attempted

http://dx.doi.org/10.5772/intechopen.78330

to model wave propagation in parallel or intersecting streets, [14–17] or in larger urban areas [18], but often limited to 2D geometries. Others have used a coupled modal-finite elements method to address the problem, while others have introduced the frontier finite elements method.

1.2. Spatial audio in architectural representation

Spatial audio in virtual reality has received increasing attention in recent years, due to its impact on the immersive experience. Spatial audio is the representation of audio features of reality that intentionally exploit sound localization. It has many possible uses in the gaming industry, entertainment or military applications. Most of these uses rely on both acoustic and spatial information about the sound. However, although spatial information is addressed, architectural design representation does not currently pay much attention to spatial audio as a factor in spatial representation.

Many other factors that have been considered in architectural design representation are linked to visual features [19, 20]. Natural light modeling and rendering [21, 22], artificial light control [23, 24], texture cognition and representation [25, 26], color discernment [27, 28] or material visualisation [29–34] are some of the countless details that an architect must manage when they represent a building. However, although the effect of sound on spatial cognition is recognizable [35], it has received little attention in architectural representation.

In 2003, Kang et al. highlighted the introduction of new EU noise policies [36] and noted that noise-mapping software/techniques are being widely used in European cities [37]. Nevertheless, they noted that these techniques can provide an overall picture for macro-scale urban areas, but the study of the micro-scale, for example an urban street or a square, could be more appropriate with the use of detailed acoustic simulation techniques. In addition, applications that predict and measure micro-scale environments [38] are still not sufficiently user-friendly, and the computation time is rather long. Kang et al. presented two computer models based on the radiosity and image source methods in an attempt to present to urban designers an interface that could be useful in the design stage, using simple formulae that can estimate sound propagation in micro-scale urban areas.

This paper presents a set of criteria for implementing 3D audio in virtual urban environments. The study is based on the definition of a new virtual audio format, generated from the combination of objects and ambisonic formats. This new audio format was explained in 2017 [4]. Using these criteria, we then describe the preparation of a set of experiments with architecture students. The results of the experiments confirm that the implementation of 3D audio enhances the immersive experience in the environments.

2. The case study environments

Four main performance environments located within the heart of the Ciutat Vella of Barcelona, the area surrounded by the former Roman Walls, were studied: Plaça Sant Felip Neri, the corner between Carrer del Bisbe and Carrer Santa Llúcia, Plaça Sant Iu, and Plaça del Rei.

2.1. Plaça Sant Felip Neri

This quiet and secluded public square, located at the end of Montjuïc del Bisbe street, is one of a set of closed squares in the Ciutat Vella of Barcelona. Its floor plan shows an irregular pentagon boundary figure with a central fountain. The 505 sqm plan presents a uniform stone floor material and is completed by five façades made of stone material as well. Of the five façades, one is Sant Felip Neri church, while the others house a school, a hotel, some dwellings and the parish stances. Three big, old trees with an asymmetric distribution in plan cover the square with their foliage. Their trunks serve as irregular columns that support the green ceiling, enclosing the square and preventing people from seeing the open sky. No sound of traffic is heard, because the Plaça is far from main roads. However, the noise of shouting children fills the square every morning, when a group play during breaktime in their beautiful schoolyard: Plaça Sant Felip Neri. During the rest of the day, a few groups of tourists arrive and look to the pockmarked stones on the church façade; marks that remind us of the Spanish Civil War. At any time of day, a street musician may use the square to play the guitar or violin in the most distant corner or near the central fountain, accompanying with music couples who are out walking, in a romantic scene.

2.2. Carrer del Bisbe-Carrer de Santa Llúcia

This little crossroads near the cathedral square seems an ordinary place. Nevertheless, a closer examination reveals that some factors come together in this single crossing. Geometrically, the floorplan forms a T pattern in which the crossing point coincides with the bishop's palace door. This door, when opened, reveals an interior courtyard that enlarges Carrer de Santa Llúcia, leading into this peaceful enclosure. The façade of Santa Llúcia chapel in the same corner gives a monumental and ceremonial character to the place. On the opposite side, the entrance to the Casa de La Ardiaca museum is prolonged by a ramp. During the day, some street vendors invade the corner and try to sell their products in front of Santa Llúcia chapel or at the beginning of the ramp. However, the Bishop's palace door is always fully clear, because of the presence of a guard when the door is open, or even on account of the large number of people who circulate through Carrer del Bisbe. Only at night, and particularly on Saturday nights, people tend to fill the area in front of the closed door of the bishop's palace standing up and looking in the opposite direction. There, an old man sings opera arias and recitatives over an amplified orchestral base. His voice invades the corner and goes beyond those limits, turning the old streets into an urban opera theatre.

2.3. Plaça de Sant Iu

Like a widening of the street, Plaça de Sant Iu is located in front of the eastern door of the cathedral. A gigantic gothic door crowned by an octagonal bell tower constitutes the west façade of this little square. On the opposite façade, a gallery formed by five stone arches closes the square. The north façade is formed by the classic-style entrance of Frederic Marès museum, whilst the south façade presents a flat wall without any door that serves as a perfect backstage for street musicians. This is the preferred point for buskers in Plaça de Sant Iu, not only because of the presence of the big wall behind them, but also because of a long stone bench on the east façade that allows listeners to sit down. In this privileged environment, groups of one, two or three musicians sing or play their instruments. A unique stone material covers the four façades of the square and the floor and gives it a uniform appearance.

http://dx.doi.org/10.5772/intechopen.78330

2.4. Plaça del Rei

Some meters behind Plaça de Sant Iu is Plaça del Rei, a totally different environment both in size and proportions. There are no trees in this 745 sqm area of stone pavement, whilst its four façades enclose the square up to a height of 20 m. In the north corner of the square, a monumental staircase rises from the floor to the Museu de la Ciutat door. Usually, street musicians enliven the atmosphere with their instruments every day, and crowds of tourists occupy the entire square looking at the real shields on the walls, the pointed arches of the windows, or the tower of Santa Àgata chapel. The everyday life of the square is always very busy, and total silence only occurs when the square transforms into a concert hall for choir, orchestra or band performances. At these times, the players are usually situated on the corner stairs and the public occupy the rest of the square. When this happens, the sound of the musicians can be heard bouncing on the hard stone of the rear walls creating a sense of spatiality that envelopes the audience.

3. The methodology used for the in-site measurements

The four environments have some features of open-air places. However, bearing in mind studies on the evaluation of outdoor space acoustics [39], we analysed them using a closed concert hall acoustic method. This decision was taken after considering three factors. The first concerns the openness of the places: the four environments can be seen as boxes in which the floor and walls are made of stone, and the ceiling of the most absorbent material that ever existed, because no sound will bounce in the open air. The second consideration for the decision concerns size: the smallest environment, the Carrer de Santa Llúcia, holds an air volume of 1800 m^3, which makes it similar to a typical hall for speeches; the largest environment, the Plaça del Rei with a volume of 12,000 m^3, does not exceed the volume of a big concert hall such as the Berlin Philharmonic Concert Hall. Finally, the third consideration explains that in an open-air environment, the sound sources change their position every moment. This situation could be definitive if we were studying the soundscape of an everyday configuration, with running children, singers, street vendors or even police sirens. However, we are recording the place in a street concert configuration, and this means that there is one player at a fixed point and the listeners stand up in the quietest mode.

The measurement methodology was *previous controlled reproduction*. This method consists of the previous recording of an acoustic signal in an anechoic chamber and the following recording of the same signal in the environment. Subsequently, the two signals are compared. The first of the recordings in the anechoic chamber were made with a calibrated reproduction system and a calibrated recording system. The reproduction system consisted of a directional speaker LD 90 W connected to a 230 V power supply. It was positioned in one of the corners of the anechoic chamber. The recording system consisted of RODE NT-55 pair-matched microphones connected to a ZOOM H6 handy recorder on a stand. This recording system was positioned in the middle of the chamber, which was 2.5 m from the speaker. Additionally, the anechoic measurements were recorded with a HATS system connected to a laptop (**Figures 1** and **2**).

The in-site measurements were performed with the same equipment as the anechoic chamber measurements, except the HATS system. The distance of the measurements varied in each environment, as shown in **Figure 3**:

Figure 1. Recordings in the anechoic chamber of Universitat Politècnica de València, Gandia.

Figure 2. Recordings in Plaça de Sant Felip Neri, Barcelona.

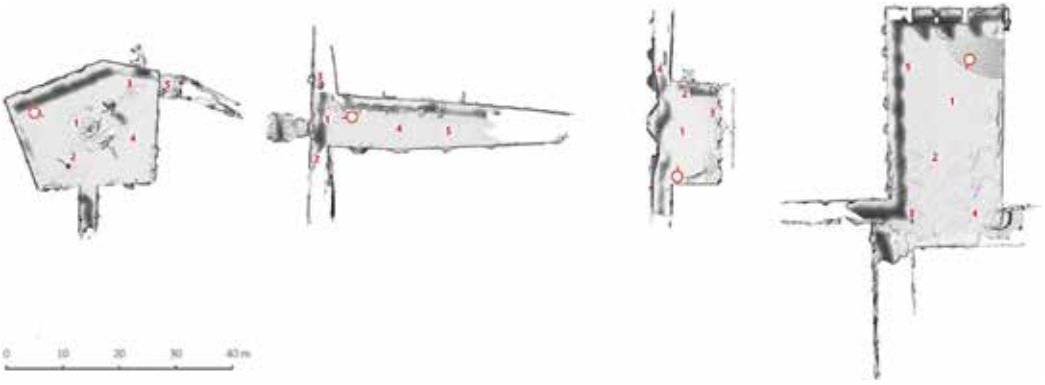


Figure 3. The plans of the four environments: Plaça de Sant Felip Neri, Carrer de Santa Llúcia, Plaça de Sant Iu, Plaça del Rei.

To ensure uniform measurements, some were repeated in the four environments, while others were recorded in specific places, considering the normal musical use of the environments. Tracks 1–13 were repeated in the four environments. However, tracks 17 (the final part of Stravinsky's *Firebird Suite*) and 18 (a solo harp piece by Lucien) were reproduced

only in Plaça de Sant Felip. Track 16 (the initial measures of Puccini's *Nessum Dorma*) was reproduced in Carrer de Santa Llúcia due to the usual type of music played in this environment. Tracks 14 (initial measures of Victoria's choral work *O sacrum convivium*) and 15 (a flute-guitar piece by Dvorak) were played in Plaça de Sant Iu. Finally, track 19 (initial measures of Orff's *Carmina Burana*) was recorded in Plaça del Rei. The dates of the recording are as follows (**Table 1**):

Track	Plaça de Sant Felip Neri	Carrer de Santa Llúcia	Plaça de Sant Iu	Plaça del Rei
1. La oboe	30/3/2017	24/04/2017	07/04/2017	06/04/2017
2. Brahms	30/3/2017	24/04/2017	07/04/2017	06/04/2017
3. Mendelssohn	30/3/2017	24/04/2017	07/04/2017	06/04/2017
4. Tchaikovsky	30/3/2017	24/04/2017	07/04/2017	06/04/2017
5. 63 Hz	30/3/2017	24/04/2017	07/04/2017	06/04/2017
6. 160 Hz	30/3/2017	24/04/2017	07/04/2017	06/04/2017
7. 400 Hz	30/3/2017	24/04/2017	07/04/2017	06/04/2017
8. 1000 Hz	30/3/2017	24/04/2017	07/04/2017	06/04/2017
9. 2000 Hz	30/3/2017	24/04/2017	07/04/2017	06/04/2017
10. 4000 Hz	30/3/2017	24/04/2017	07/04/2017	06/04/2017
11. White noise	30/3/2017	24/04/2017	07/04/2017	06/04/2017
12. Pink noise	30/3/2017	24/04/2017	07/04/2017	06/04/2017
13. Shotgun	30/3/2017	24/04/2017	07/04/2017	06/04/2017
14. Victoria			07/04/2017	
15. Dvorak			07/04/2017	
16. Puccini		24/04/2017		
17. Stravinsky	30/3/2017			
18. Lucien	30/3/2017			
19. Orff				06/04/2017

Table 1. Relation of tracks played and recordings at the environments.

4. *What* is evaluated?

The four environments were studied using a reproduction-recording system. In this system, an impulse signal previously calibrated in the Anechoic Chamber was emitted in the environment. This signal was captured in different positions in each environment, as already mentioned. **Figure 3** shows the measured positions. Each of the recording points was subdivided into two channels corresponding to left (L) and right (R). Therefore, the naming of each recording consists of the number of the recording point, followed by an underscore and capital letter L or R: 1_L, 1_R, 2_L, 2_R, etc.

4.1. Acoustic framework

Once the recordings had been made, different parameters of acoustic quality were obtained by signal processing. The following parameters were studied:

Reverberation time (T60): when a sonorous source that is continually radiating suddenly stops in a determined enclosure, a listener in the hall will continue to hear the sound for a period of time in which its energy is being absorbed by the surfaces of the enclosure's limits [39]. The T60 value corresponds to the falling time of the sound associated with the angle for the first 60 dB decrease. The T60 for an empty hall varies with the frequency. Generally, for music halls, the Ts is higher for low frequencies and decreases when the frequency increases. This typical spectrum of reverberation is known as the *tonal curve*.

Early decay time (EDT): this considers the reverberation time for the first 10 dB of decrease. EDT is more closely related to the subjective impression of the reverberation in an enclosure than Ts [40]. To ensure good diffusion of sound in a hall, it is imperative that EDT corresponding to 500 Hz and 1 kHz is in the same order as Ts [40].

Speech clarity (C50): registered C50 values vary with the listening point. According to Carrión Isbert [39], the recommended value of C50 associated with each point in an occupied hall must fulfill C50 > 2 dB. The higher the value, the greater is the speech intelligibility and sonority in the considered point.

Definition (D50): if the definition increases, the hall is better prepared for speech, as may be the case in theatres or conference halls. Thus, a D50 value that is over 65% is an appropriate value for this kind of hall. A concert hall with good acoustics has a definition index lower than 50% in central frequencies of 500 and 1000 Hz. In concert halls, the higher the definition index is, the worse quality is the acoustics [3].

Musical clarity (C80): registered C80 values vary with the listening point. Beranek [41] recommends an average of $-4 \leq C80 \leq 0$ dB for C80 in the 500 Hz, 1 kHz and 2 kHz frequencies for an empty hall. Values over +1 dB should be avoided.

Strength (G): G values remain similar at each of the measurement points. They approximately correspond to a decreasing line from low frequencies (G = 30) to high frequencies (G = 10). UNE-EN ISO 3382 [40] recommends G values between 4 and 5.5.

4.2. Acoustics quality parameters in the environments

4.2.1. Reverberation time (T60)

In Sant Felip Neri square (top left from **Figure 4**) we can observe an increase in reverberation time when we step away from the source. The Ts values vary from a conference hall (T60 = 0.7–1.9: point 1), an opera theatre (T60 = 1.2–1.5: point 2), a chamber music concert hall (T60 = 1.3–1.7: points 3 and 4) and a symphonic concert hall (T60 = 1.8–2.0), according to the recommended values of Carrión [42]. Plaça del Rei (top right) presents typical values of a symphonic hall when we step away from the source. Meanwhile, Plaça Sant Iu (bottom left) shows a typical curve of a speech hall in all the interior points of the square, except for the point in the

http://dx.doi.org/10.5772/intechopen.78330

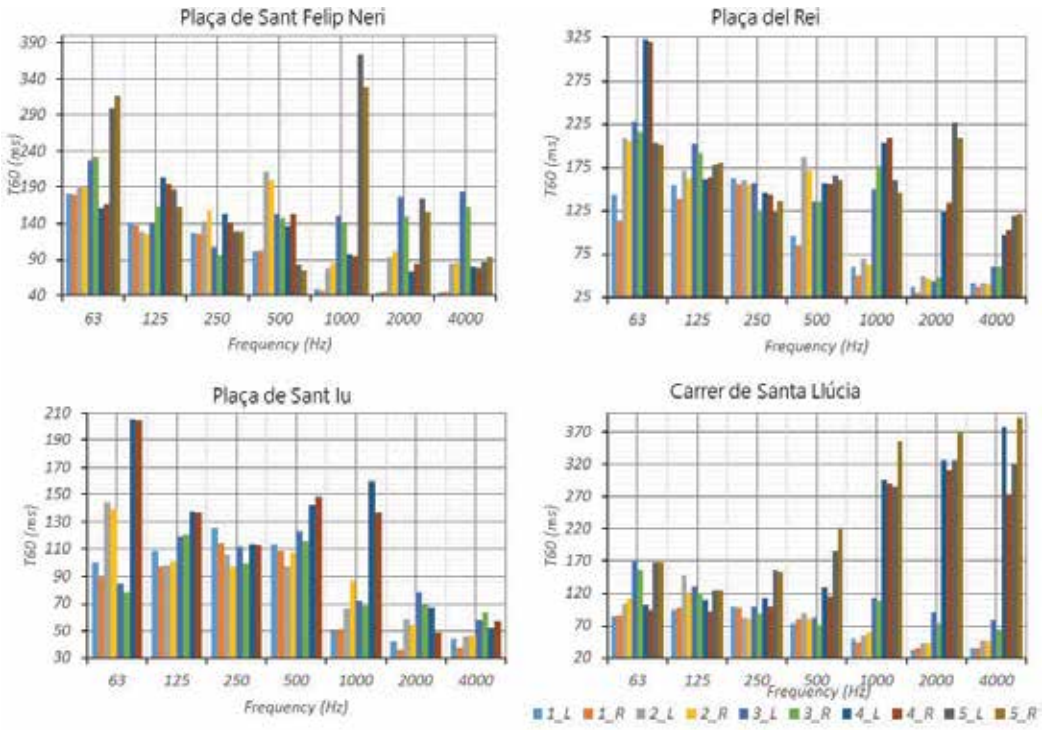


Figure 4. Reverberation time (T60) of the different environments: Sant Felip Neri (top left), Plaça del Rei (top right), Plaça Sant Iu (bottom left), and Carrer Santa Llúcia (bottom right).

alley, which presents Ts values like a chamber music hall. Carrer de Santa Llúcia (bottom right) presents some features of an opera theatre for points 1, 2 and 3, that is, in the frontal points to the source, whilst in the rear part to the source, some symphonic hall features are presented.

4.2.2. Speech clarity (C50)

In Plaça de Sant Felip Neri (top left from **Figure 5**) only the recordings at points 1, 2 and 3 exceed 2 dB of C50 for high frequencies. Note that Points 1, 2 and 3 are the nearest points to the source and it is natural that clarity is better near the speaker. Thus, we can deduce that this is not a square with clear acoustics for speech in most of the recording locations and frequencies. In Plaça del Rei (top right) we find a similar situation at first glance. However, clarity is very appropriate at point 1 for mid-high frequencies. Moreover, as we step away from the source, that is, at points 1, 2 and 3, clarity is restricted only to high frequencies, whilst in lateral points, the clarity is below accepted levels. Plaça de Sant Iu (bottom left) presents a similar scheme to those seen above: a lack of clarity for low-mid frequencies and better clarity for high frequencies. Note that point 4 is the only one that does not follow the typical curve of the other points. This is due to its position in the access alley to the square rather than inside the enclosure. Thus, its behavior is different from the others. In Carrer de Santa Llúcia (bottom right) we can observe a progressive decrease in speech clarity from point 1 to point 5 for

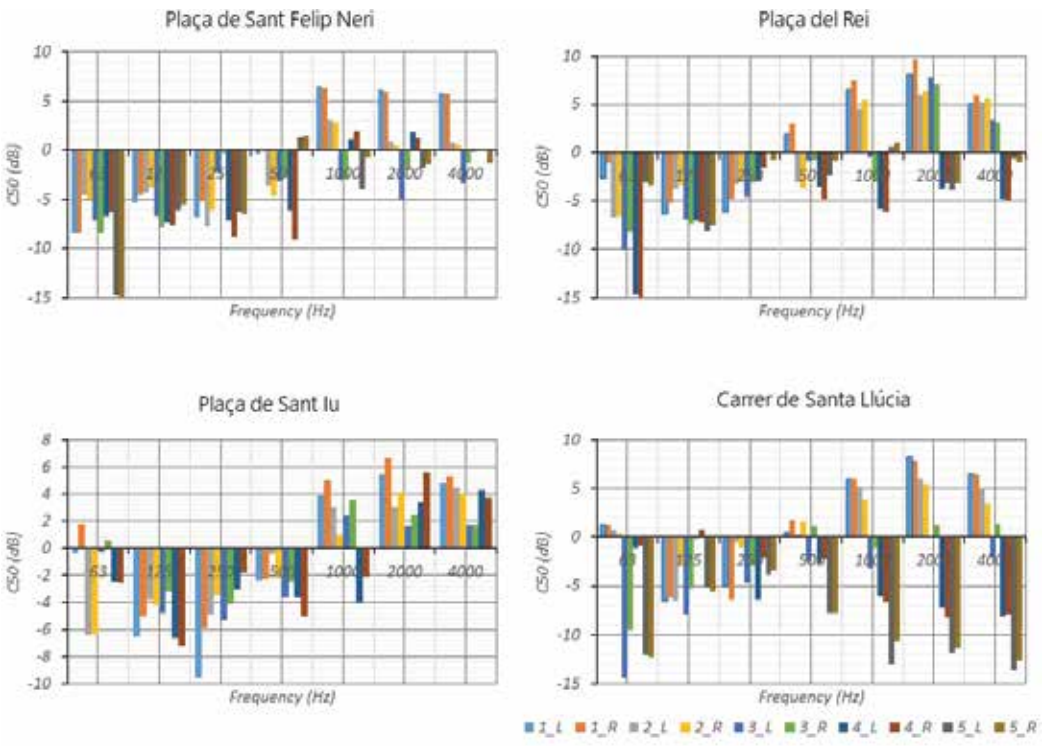


Figure 5. Speech clarity (C50) of the different environments: Sant Felip Neri (top left), Plaça del Rei (top right), Plaça Sant Iu (bottom left), and Carrer Santa Llúcia (bottom right).

mid-high frequencies. This knowledge indicates that speech clarity at points 1 and 2 is only acceptable for mid-high frequencies. Considering that this environment is generally used by opera singers, and that most of the audience occupies the zone in points 1 and 2, we can say that the acoustics of this space are extremely favorable to its use.

4.2.3. Musical clarity (C80)

In Sant Felip Neri Square (top left from **Figure 6**) we can see that all the C80 values remain above −4 dB, but they are higher than +1 dB from 2 kHz at points 1, 2 and 4. If we compare C50 and C80 values, we can deduce that this square is clearer for music than for speech. In Plaça del Rei (top right), C80 values at points 2, 4 and 5 are within the desired limits. However, at points 1 and 2, C80 values exceed +1 dB, thus those points are not optimum for musical clarity. This square holds good musical clarity at the points furthest from the source, that is to say, points that belong to the reverberant field and not to the direct field. In contrast, Plaça de Sant Iu (bottom left) only shows C80 values within the desired limit when we consider low and mid frequencies. These recommended frequencies hold C80 values exceeding the +1 dB criteria. These data, compared with C50, make us think that the square is more appropriate for speech than for music. It is a square in which the spoken word

http://dx.doi.org/10.5772/intechopen.78330

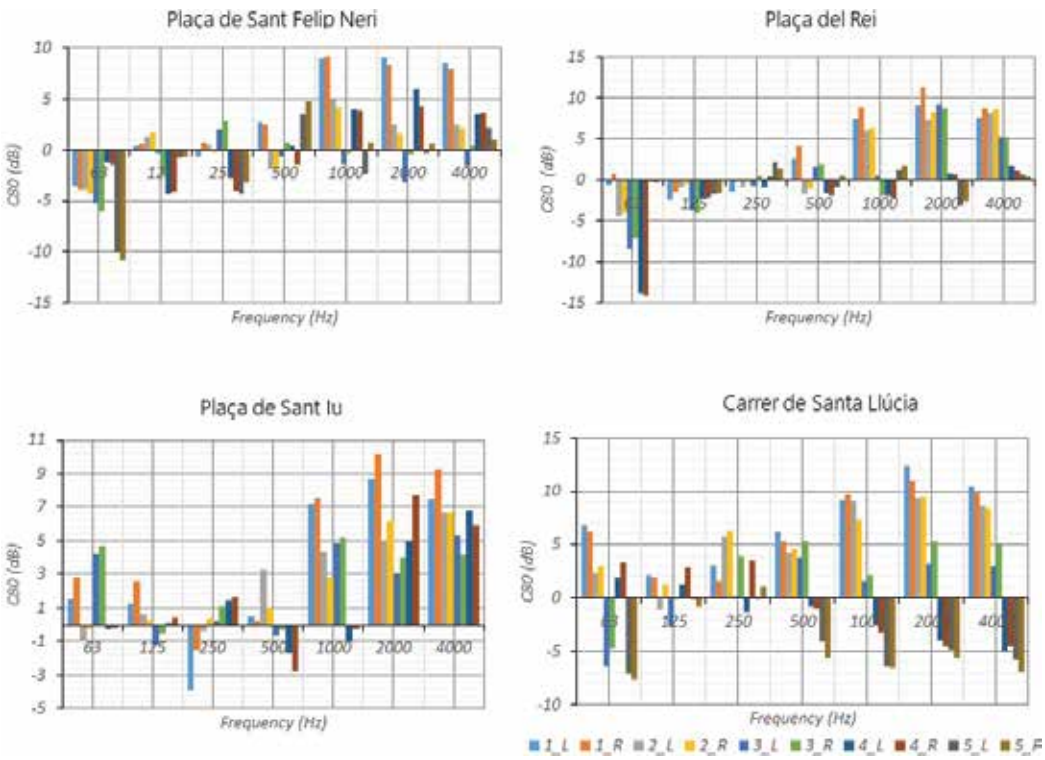


Figure 6. Music clarity (C80) of the different environments: Sant Felip Neri (top left), Plaça del Rei (top right), Plaça Sant Iu (bottom left), and Carrer Santa Llúcia (bottom right).

is correctly understood, although music is not underprivileged. Finally, Carrer de Santa Llúcia (bottom right) has better musical clarity at points 3, 4 and 5. Curiously enough, these points are the same that held the aforementioned bad speech clarity (C50). This fact suggests that the points with better qualities for speech are not the optimum ones for music, and vice-versa.

4.2.4. Definition (D50)

In Plaça Sant Felip Neri (top left from **Figure 7**), we can see that D50 values were below 50% for low frequencies, but above 50% for high frequencies (from 500 Hz). This was true particularly at points 1, 2 and 4. Thus, this square is very appropriate for music at points 3 and 5 (lateral and distant from the source) and better for speech at points 1, 2 and 4 (points near to or centered with the source). Similarly, the Plaça del Rei (top right) had D50 parameters that exceeded 50% for high frequencies at points 1, 2 and 3 (these points were aligned frontally with the source). The lateral points maintained D50 values under 50%. Thus, these lateral points are appropriate for music. These data, together with those revised about C50 and C80, again indicate that this square has good acoustics for music and worse acoustics for speech. Conversely, Plaça Sant Iu (bottom left) had a similar tendency for almost all the measured

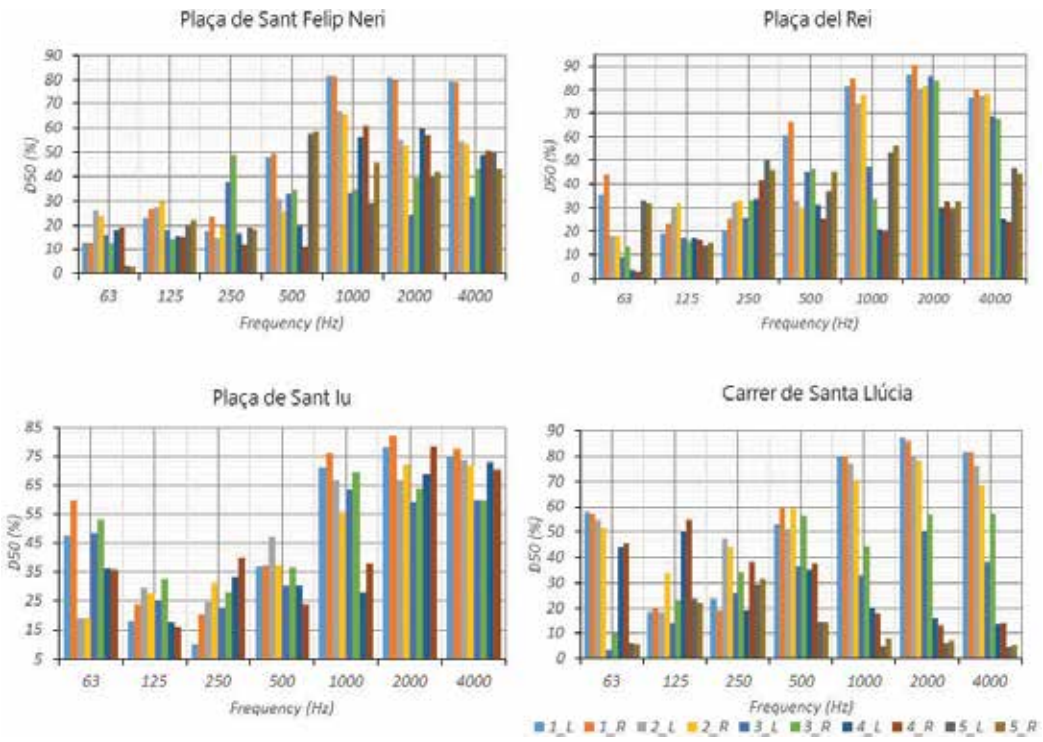


Figure 7. Definition (D50) of the different environments: Sant Felip Neri (top left), Plaça del Rei (top right), Plaça Sant Iu (bottom left), and Carrer Santa Llúcia (bottom right).

points, except for the point measured in the alley. In particular, D50 was above 65% for high frequencies and under 50% for mid and low frequencies. Therefore, we restate that this square works better for speech and has too much definition for music. Perhaps for this reason, and because of its size, the square is ideal for solo singers or those accompanied with chamber instruments. Finally, Carrer de Santa Llúcia (bottom right) has an inherent tendency to lower definition when we move away from the source or we are behind it. Particularly, points 1 and 2 are more suitable for speech or opera, whilst points 3, 4 and 5 have better features for music. Again, we can note that the audience zone belongs to points 1 and 2.

4.2.5. Early decay time (EDT)

In Plaça de Sant Felip Neri (top left from **Figure 8**) we can see that EDT values for 500 Hz and 1 kHz are similar to Ts values, except for the Ts peak at point 5, which is the result of a measurement error due to the high amount of background noise at that time. Similarly, in Plaza del Rei (top right), Sant Felip Neri (bottom left) and Carrer Santa Llúcia (bottom right), the EDT levels are very similar to the Ts levels, which indicates that there is a good sound diffusion in these environments.

http://dx.doi.org/10.5772/intechopen.78330

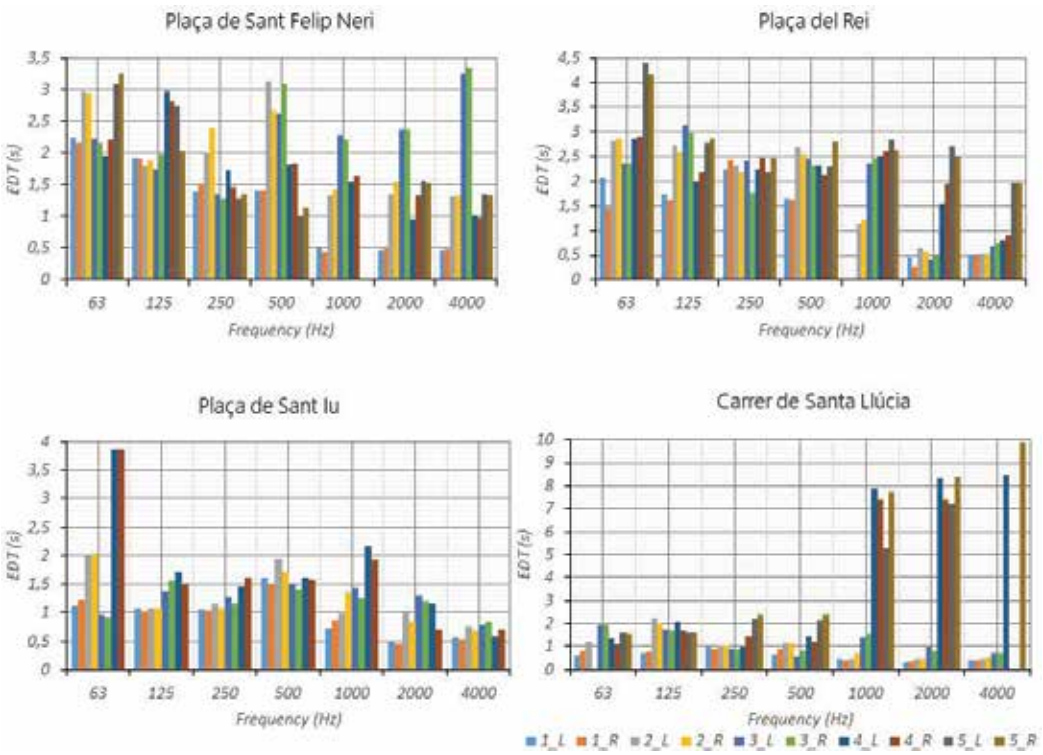


Figure 8. Early decay time (EDT) of the different environments: Sant Felip Neri (top left), Plaça del Rei (top right), Plaça Sant Iu (bottom left), and Carrer Santa Llúcia (bottom right).

5. *How* is it evaluated?

Quantitative approaches are the main methods of scientific research. They focus on analysing the degree of association between quantified variables, promulgated by logical positivism. Therefore, possible answers need to be constrained in order to evaluate results objectively [43]. Some evaluation investigations have already been done with architecture students [44, 45].

Qualitative research is less common in education areas, because it focuses on detecting and processing intentions. Unlike quantitative methods, qualitative approaches require deduction to interpret the results. The qualitative approach is subjective, because it is assumed that reality is multifaceted and cannot be reduced to a universal parameter. Interviewers are passive observers, they take notes and classify them [46]. These methods have traditionally been related with social sciences, due to their association with human factors and the user's experience. In fact, *User Experience*, UX, is a discipline focused on the study of behavioral patterns in working environments. Our case study is framed in teaching process usability [47]. Thus, a brief discussion of what usability means is mandatory.

5.1. Usability evaluation

We could define usability as a general quality that indicates the suitability for a specific purpose of a particular artefact (appropriateness for a purpose) [48].

This term is linked with the development of products (which could be systems, technologies, tools, applications or devices) that can be easy to learn, effective and enjoyable in the user's experience. Nevertheless, usability can be considered another factor in a wider process called the acceptability of a system. Thus, acceptability defines whether a system is good enough to meet all a user's needs [49].

In ISO/IEC 9126, usability is defined as "software product capability to be understood, learned, used and attractive for the user, when it is used under specific conditions." However, usability is not limited to computer systems. It is a concept that can be applied to any element in which an interaction between a human and an artefact occurs.

In addition, in ISO/IEC 9241-11, the guidelines for the usability of a particular product are described. Here, usability is defined as "the level in which a product can be used by particular users in order to reach specified goals with effectivity, efficiency and satisfaction in a particular context of use." In our research, the effectivity of a system is related with its goals, efficiency is related with the performance of the used resources to reach the goals, and satisfaction is related with its acceptability and commodity [50]. This definition is based on the concept of *quality in use*, and describes how the user does particular tasks in particular environments in an effective way [51]. For Bevan, the quality of use, measured in terms of efficiency, efficacy and satisfaction, is not only determined by the product, but also by the context (kind of users, tasks of the users and physical environment). Therefore, the usability, understood as the quality in use of a product is the interaction between a user and a product while a task is being accomplished in a technical, physical, social and organisational environment.

In our study, usability defines the general quality, indicating the suitability for educational purposes of an immersive scenario. In a similar line as [52], the goal is to evaluate the student motivation before and after the use of such technologies. Users are asked to evaluate the quality of the soundscape representation in this scenario. Both visual and acoustic data have a direct impact on the perception of the space and the realism of this representation is the focus of the evaluation.

5.2. Experiments with architecture students

Two kinds of experiments were carried out with architecture students. The first group of experiments considered a quantitative approach to the evaluation, whereas the second group was performed according to a qualitative approach.

5.2.1. The quantitative test

A set of multiple choice questions about several audio sequences were administered to a group of people (17 in total). In this test, some urban acoustic features were analysed. This first test, as shown in **Table 2**, was given once the recordings and acoustic analysis had been carried out. The test had two objectives: to characterise people's perception and knowledge of

E. Code	Description	Option A	Mention index MI (%) for option A	Option B	Mention index, MI, (%) for option B
1. I	Speech clarity (C50)	C50 (500 Hz) = 2 dB (from **Figure 5**: Carrer Santa Llúcia, recording point 1, Puccini track)	**100**	C50 (500 Hz) = −7.5 dB (from **Figure 5**: Carrer Santa Llúcia, recording point 3: Puccini track)	**0**
2. S	Sense of space	(Plaça de Sant Iu, recording point 3, Tchaicovsky track)	**82.4**	(Plaça del Rei, recording point 1, Tchaicovsky track)	**17.6**
3. EDT	Early decay time	EDT (500 Hz) = 1.3 s (from **Figure 8**: Plaça Sant Felip Neri, recording point 1, Mendelssohn track)	**29.4**	EDT (500 Hz) = 2.3 s (from **Figure 8**: Plaça del Rei, recording point 4, Mendelssohn track)	**70.6**
4. Br	Brightness	(Plaça Sant Iu, recording point 1, Dvorak track)	**58.8**	(Plaça Sant Iu, recording point 4, Dvorak track)	**41.2**
5. T60	Reverberation time	T60 (500 Hz) = 0.65 s (from **Figure 4**, Plaça del Rei, recording point 1, Orff track)	**88.2**	T60 (500 Hz) = 1.75 s (from **Figure 4**, Plaça del Rei, recording point 4, Orff track)	**11.8**
6. BR	Bass ratio	(Plaça Sant Felip Neri, recording point 1, Lucien track)	**82.4**	(Plaça Sant Felip Neri, recording point 4, Lucien track)	**17.6**

Table 2. Description and definition of the options in the quantitative test.

acoustic and sonic features of the outdoor space, and to obtain feedback on the most relevant aspects of street music.

The results should allow for an initial approximation of whether people are aware of the nuances and differences between a street music recording and a concert hall music recording. Above all, it should be possible to test people on the big differences between the acoustics of the different public spaces. Therefore, different recordings of the same music but from different spatial points were compared. A total of six questions, each with a new melody, covered the following topics: speech intelligibility, sense of space, reverberation time, timbre modification, EDT and bass amplification.

The first dataset (**Tables 2** and **3**) shows the description of the analysed elements and the individual responses of the different users.

After an analysis of the survey results, we can highlight some important findings. First, all the questions in **Table 2** are balanced to A or B over 70%, except the fourth question, which is more ambiguous. This shows the high consensus about the acoustic features that were being evaluated. Second, we highlight the presence of users who had professional or higher music qualifications in **Table 3**. These users agreed unanimously, or almost unanimously (except one) in their decisions. The only question on which they disagreed with each other concerns the sense of space.

E. Code	Male										Female						
USERS	**U 1**	**U 2 (M)**	**U 3 (M)**	**U 4**	**U 5**	**U 6**	**U 7**	**U 8**	**U 9**	**U 10 (M)**	**U 11**	**U 12**	**U 13 (M)**	**U 14**	**U 15 (M)**	**U 16**	**U 17**
1. I	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
2. S	A	A	B	B	A	A	A	A	A	A	A	A	A	A	B	A	A
3. EDT	B	B	B	B	B	B	B	B	A	B	B	A	B	A	B	A	A
4. Br	B	A	B	A	A	B	A	B	B	A	B	B	A	A	A	A	A
5. RT	A	A	A	A	A	B	A	A	A	A	A	A	A	A	B	A	A
6. BR	B	B	B	B	B	B	A	B	B	B	B	B	B	B	A	A	B

Table 3. Individual options for the quantitative test.

5.2.2. The qualitative test

In this context, the analysis of some typical street music locations in Ciutat Vella of Barcelona was included in this soundscape evaluation. One of the environments studied here was the Plaça Sant Felip Neri. Various recordings were made in the environment according to different positions of the listeners. For the current study, one recording was selected: *The Fountain* by Marcel Lucien was reproduced in Plaça de Sant Felip Neri in five positions. **Figure 1** shows the emitting point with the enumeration of the different recording points in each square.

To create the test conditions, Plaça Sant Felip Neri needed to be reproduced as faithfully as possible, in terms of visual and auditory aspects.

First, a 3D model was created using photogrammetric processing of digital images to generate 3D spatial data. This method is relatively fast to implement, does not require specialised hardware like laser scanning and, if performed correctly, produces high quality results that are not as precise as other techniques, but more than enough to transport the user to a faithful 3D recreation of the square.

The next goal was to recreate the soundscape. Two options were developed and presented to the test subjects. The first option was to use the original concert hall recording and present it to the users as it is, without any distortion, reverberation or additional ambient sounds from the square.

The second option was more difficult to create. As stated, five recordings of the song were made from different locations in the square. These recordings captured the subtleties a user would experience listening in the real square. The challenge was to allow the test subjects to move freely around the square and still be able to listen to the song in conditions as close as possible to the real conditions at any point in the square, not only the five recording points.

To extend the experience to any given point, a mixing algorithm was used to perform a logarithmic interpolation between the nearest recordings, to provide an experience that was identical to the original when the user was exactly at the recording point, and faded seamlessly as they moved closer to the next one.

http://dx.doi.org/10.5772/intechopen.78330

Figure 9. Three students in the experiment.

Finally, a head mounted display needed to be used to show the virtual reality to the test subjects. The Oculus Rift was selected for this task due to its compact size, high quality display and integrated headphones. This provided a fully immersive environment with a great sense of presence for the users (**Figure 9**).

The Oculus Rift allows for room scale tracking. This means that the user can move around the real space, and that movement translates to the virtual world. This greatly amplifies the sense of presence and makes the experience a lot more realistic and comfortable. However, it has limitations: the length of the cable and the resolution of the tracking cameras only allow the user to move around a space 3 m long and 2 m wide, approximately.

To improve this aspect, a teleport system was created. Using the Oculus Rift touch controllers, the user could point to any location in the square, and instantly teleport to that location. This provided the necessary freedom to move around the square, while maintaining all the benefits of room tracking.

Figure 10. Two screenshots of what students can see with the glasses.

The result of this process was an experience that allowed free movement around a realistic 3D recreation of the original square. Users perceived two distinctly different audio environments: one that recreated concert hall conditions, and a second one to experience as closely as possible the square's sound conditions (**Figure 10**).

For the qualitative study, a sample of 18 students (11 men and 8 women) who agreed to participate was selected.

The BLA method works on positive and negative poles to define the strengths and weaknesses of the product. Once the element is obtained, the laddering technique can be applied to define the relevant details of the product. The object of a laddering interview was to uncover how product attributes, usage consequences, and personal values are linked in a person's mind. The characteristics obtained through the laddering application will define which specific factor contributes to the consideration of an element as either a strength or a weakness. The BLA process consisted of three steps, following a similar method to Fonseca, Redondo and Villagrasa [53]:

1. Elicitation of elements. The implementation of the test started with a blank template for the positive (most favorable) and negative (least favorable) elements. The interviewer (in this case the professor) asked the users (the student) to mention a positive and a negative aspect of the two types of music that could be heard (Option A and Option B). Thus, we obtained two positive aspects and two negative aspects.
2. Marking of elements. Once the list of positive and negative elements has been completed, the interviewer asked the user to mark each one from 0 (lowest possible level of satisfaction) to 10 (maximum level of satisfaction).
3. Element definition. Once the elements had been assessed, the qualitative phase started. The interviewer asked for justification of each one of the elements by performing the laddering technique. Questions were asked such as "Why is it a positive element?" "Why did you give it this mark?" The answers had to be specific explanations of the exact characteristics that made the mentioned element a strength or weakness of the product.

From the results obtained, the next step was to polarize the elements based on two criteria:

1. Positive (Px)/Negative (Nx): the student had to differentiate between elements perceived as strong points of the experience that helped them to consider the music as satisfactory, compared to negative aspects that were not satisfactory or simply needed to be modified to be satisfactory.
2. Common Elements (xC)/Particular (xP): finally, the positive and negative elements that were repeated in the students' answers (common points) and the responses that were only given by one of the students (particular points) were separated according to the coding scheme shown in **Tables 1–3**.

The common elements that were mentioned at a higher rate were the most important aspects to use, improve or modify (according to their positive or negative sign). Particular elements, which were mentioned by only one user, could be ruled out or treated in later stages for development (**Table 4**).

E. code	Description	Av. score (Av)	Mention index (MI) (%)
1PC (A)	Clarity of music	7.7	44.4
2PC (A)	Guiding thread for music	8.3	16.6
3PC (A)	Quality of sound	8.3	16.6
4PC (A)	Focused on the music	9	11.1
1PP (A)	Peaceful music	9	5.6
1NC (A)	Not realistic	3	33.3
2NC (A)	No sense of space	4	22.2
3NC (A)	No background	4.5	11.1
4NC (A)	Movement too fast	4	11.1
1NP (A)	No variance of echo	4	5.6
2NP (A)	Like a television	4	5.6
3NP (A)	Too loud	4	5.6
1PC (B)	Realistic	8.4	50
2PC (B)	Sense of the place	8.7	33.3
1PP (B)	Alive	9	5.6
2PP (B)	Softer and modulated	7	5.6
3PP (B)	More natural	7	5.6
1NC (B)	No clarity of music	3.8	22.2
2NC (B)	Relation between background and vision	4.7	16.7
3NC (B)	Disturbing background	3.7	16.7
4NC (B)	Problems with volume	3.7	16.7
5NC (B)	It is not real enough	3.5	11.1
1NP (B)	Quality of hardware	5	5.56
2NP (B)	Sudden changes in sound	3	5.56

Table 4. Positive common (PC), particular (PP), negative common (NC) and negative particular (NP) elements for option A (concert hall) and option B (public square).

The individual values obtained for positive and negative indicators are shown in **Table 5**. Once the features mentioned by the students were identified and given values, the third step defined by the BLA initiated the qualitative stage in which the students described and provided solutions or improvements for each of their contributions in the format of an open interview.

Table 6 shows the main improvements or changes that the students proposed for both positive and negative elements.

At this point, we can identify the most relevant items obtained from the BLA, which had high rates of citation, high scores or a combination of both. It is important to separate the types of results obtained. The first group belongs to option A (concert hall recording), and the second

E. Code	Male									Female								
USERS	**U 1**	**U 2**	**U 3**	**U 4**	**U 5**	**U 6**	**U 7**	**U 8**	**U 9**	**U 10**	**U 11**	**U 12**	**U 13**	**U 14**	**U 15**	**U 16**	**U 17**	**U 18**
1PC (A)	—	—	—	9	8	—	8	—	7	7	8	—	8	—	—	—	7	—
2PC (A)	9	—	—	—	—	—	—	8	—	—	—	—	—	—	—	8	—	—
3PC (A)	—	8	—	—	—	7	—	—	—	—	—	—	—	9	—	—	—	8
4PC (A)	—	—	8	—	—	—	—	—	—	—	—	10	—	—	—	—	—	—
1PP (A)	—	—	—	—	—	—	—	—	—	—	—	—	—	—	9	—	—	—
1NC (A)	2	—	—	—	3	4	—	—	—	—	—	—	—	3	—	5	—	1
2NC (A)	—	2	4	—	—	—	—	5	—	—	—	—	—	—	—	—	5	—
3NC (A)	—	—	—	—	—	—	5	—	—	—	—	—	4	—	—	—	—	—
4NC (A)	—	—	—	5	—	—	—	—	3	—	—	—	—	—	—	—	—	—
1NP (A)	—	—	—	—	—	—	—	—	—	—	4	—	—	—	—	—	—	—
2NP (A)	—	—	—	—	—	—	—	—	—	—	—	4	—	—	—	—	—	—
3NP (A)	—	—	—	—	—	—	—	—	—	3	—	—	—	—	—	—	—	—
1PC (B)	—	9	—	—	10	9	7	—	5	9	—	10	—	—	—	8	—	9
2PC (B)	10	—	—	9	—	—	—	9	—	—	4	—	—	10	—	—	10	—
1PP (B)	—	—	—	—	—	—	—	—	—	—	—	—	—	—	9	—	—	—
2PP (B)	—	—	7	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
3PP (B)	—	—	—	—	—	—	—	—	—	—	—	—	7	—	—	—	—	—
1NC (B)	—	—	—	—	—	—	4	3	—	4	4	—	—	—	—	—	—	—
2NC (B)	—	5	4	5	—	—	—	—	—	—	—	—	—	—	—	—	—	—
3NC (B)	—	—	—	—	—	4	—	—	—	—	—	3	—	—	4	—	—	—
4NC (B)	3	—	—	—	—	—	—	—	—	—	—	—	—	—	—	5	3	—
5NC (B)	—	—	—	—	3	—	—	—	2	—	—	—	—	—	—	—	—	—
1NP (B)	—	—	—	—	—	—	—	—	—	—	—	—	5	—	—	—	—	—
2NP (B)	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	3

Table 5. Individual scores for PC, PP, NC and PC elements for option A (concert hall) and option B (public square).

group to option B (public square recording). After the elicitation of the most relevant features of each of them, we are going to end by comparing them.

Option A (concert hall recording). We can highlight that this kind of recording has good clarity of music (MI: 44.4%, Av: 7.8), it favors the guidance of the thread for music (MI: 16.6%, Av: 8.3), and the quality of the sound is valued (MI: 16.6%, Av: 8.3). In terms of the main negative comments, students clearly identified a lack of realism in this kind of experience (MI: 33.3%, Av: 3), that was related to the lack of sense of space (MI: 22.2%, Av: 4) and they missed the background noise (MI: 11.1, Av: 4.5), aspects that were directly related to the design of the application.

http://dx.doi.org/10.5772/intechopen.78330

E. Code	Description	Mention index (MI) (%)
1CI (A)	Improve the relation with the environment	66.7
2CI (A)	Improve the background sound	22.2
3CI (A)	Change the position of the sounds	11.1
2PI (A)	Decrease the volume of the sound	5.7
3PI (A)	Improve the relation with the musician	5.7
4PI (A)	Improve the quality of the sound	5.7
1CI (B)	Improve sound quality	27.8
2CI (B)	The changes between position could be softer	22.2
3CI (B)	Balance the volume levels between different points	11.1
4CI (B)	Improve the relation between vision and sound	11.1
5CI (B)	Decrease the background noise	11.1
1PI (B)	Improve the clarity of sound	5.7
1PI (B)	Improve the relation with the place	5.7

Table 6. Proposed common improvements (CI) and particular improvements (PI) for both positive and negative elements for common and particular items in A recording (concert hall) and B recording (public square).

Option B (public square recording). Two main positive aspects were highlighted by students: the high degree of realism of the application both in visual and acoustic terms (MI: 50%, Av: 8.4), and the good relation between sound and place (MI: 33.3%, Av: 8.7). Conversely, some negative comments were pointed out: a lack of clarity in the music (MI: 22.2%, Av: 3.8), a bad relation between background and vision (MI: 16.7%, Av: 4.7), which could be solved with the position of different visual avatars, and the presence of some disturbing background (MI: 16.7%, Av: 3.7), due to the different times of the original recordings. Technically, these would be the main aspects to modify in future iterations of the proposed method.

In summary, two clear opinions about the experiment were shown, which confirm the first question of the survey: Which recording do you prefer, A or B? Most people (61.1%) agreed that option B was better than option A (38.9%). The reasons for this answer were clearly explained in the rest of the survey. Although there was a high valuation of the realism of the application both in visual and acoustic terms in option B (MI: 50%, Av: 8.4), it was also certain that clarity of music in option B was not as good as in option A, as we can see if we compare 1PC (A) with 1NC (B). This confirms that the street music recording implies a decrease of quality in the music played. This loss could be a drawback for musicians who want to perform in the middle of the city. However, the survey reveals another feature that must be taken into account: a third of the students (MI: 33.3%) evaluated option B with an almost excellent score (Av: 8.7) for the sense of space quality (2PC [B]). This shows the hidden potential of spatial sound, that is, sound spatialization. Several attempts can be found in the history of music in which composers wrote their music bearing in mind the spatial features of the places in which it was going to be played. However, all these compositions tend to be limited to closed spaces, and the spatial possibilities are limited to the specific space. A wide range of possibilities arise when a closed concert hall is replaced by the openness of squares

and public spaces. Coupled volumes, streets, galleries, balconies or even stairs now belong to this new stage for music that can be explored in infinite ways.

6. Conclusions

The study aimed to highlight the questions of *What is going to be evaluated* and *How is it going to be evaluated* within the context of cultural heritage evaluation. In this case study of a higher education evaluation, we have explained that both questions can be answered in three words: objectivity through subjectivity. In fact, what was evaluated with the quantitative and qualitative tests was not far from what was studied with the acoustic parameters. Furthermore, the subjective opinions were based on the objective parameters. This built a bridge over the big gap between these two poles, and helped us to understand that no objective parameters can be evaluated without subjective insight. For the cultural evaluation, a scientific basis must be established to achieve reliable results. Nevertheless, a unilateral evaluation that only relies on these scientific data would overlook the valuable opinions of users. What is more, without the user's insight, the analysis would neglect the term "cultural," because no culture is possible without the action of humans, that is, the users. Here, cultural is defined as the opposite of natural, as a synonym of artificial, as something that is evaluated by a human.

However, some limitations of the study need to be addressed in further research. The number of participants in the surveys should be increased, and a pre- and post-test evaluation of satisfaction with the process introduced. Similarly, the immersive experiment should be extended to other outdoor environments (Carrer Santa Llúcia, Plaça de Sant Iu and Plaça del Rei) so that the objective parameters can be compared with the subjective users' opinions.

Further research must also be carried out on the implementation of these representation techniques in the higher education system, especially in Architectural Degree courses, in which spatial understanding is crucial. In this context, it is clear that architecture students should be able to deal with spatial representations that not only cover visual features, but also sonic or even thermic components of architecture. Today's technology has reached such a high level of representation capabilities, that a vague idea of what an environment looks like is no longer acceptable. An architect should manage these tools when they present a new building, and protect existing constructions that are regarded as cultural heritage.

Acknowledgements

This research was supported by the National Programme of Research, Development and Innovation aimed to the Society Challenges BIA2016-77464-C2-1-R & BIA2016-77464-C2-2-R of the National Plan for Scientific Research, Development and Technological Innovation 2013-2016, Government of Spain, titled "*Gamificación para la enseñanza del diseño urbano y la integración en ella de la participación ciudadana* (EduGAME4CITY)," and "*Diseño Gamificado de visualización 3D con sistemas de realidad virtual para el studio de la mejora de competencias motivacionales, sociales y espaciales del usuario* (EduGAME4CITY)."

Author details

Josep Llorca[1]*, Héctor Zapata[1], Jesús Alba[2], Ernest Redondo[1] and David Fonseca[3]

*Address all correspondence to: josep.llorca@upc.edu

1 AR&M, Barcelona School or Architecture, Universitat Politècnica de Catalunya, Spain

2 Universitat Politècnica de Valencia, Centro de Tecnologías Físicas, Escola Politècnica Superior Gandia, Valencia, Spain

3 Grup de Recerca en Technology Enhanced Learning (GRETEL), La Salle-Ramon Llull University, Barcelona, Spain

References

[1] Hølleland H, Skrede J, Holmgaard SB. Cultural heritage and ecosystem services: A literature review. Conservation and Management of Archaeological Sites. Jul. 2017;**19**(3): 210-237

[2] Barron M. Auditorium Acoustics and Architectural Design. New York: E & FN Spon; 1993

[3] Beranek L. Concert Halls and Opera Houses. New York, NY: Springer New York; 2004

[4] Molerón M, Félix S, Pagneux V, Richoux O. Sound propagation in periodic urban areas. Journal of Applied Physics. Jun. 2012;**111**(11):114906

[5] Kang J. Sound propagation in street canyons: Comparison between diffusely and geometrically reflecting boundaries. Feb. 2000. DOI: 10.1121/1.428580

[6] Pelat A, Félix S, Pagneux V. On the use of leaky modes in open waveguides for the sound propagation modeling in street canyons. The Journal of the Acoustical Society of America. Dec. 2009;**126**(6):2864-2872

[7] Richoux, Ayrault, Pelat, Félix, Lihoreau. Effect of the open roof on low frequency acoustic propagation in street canyons. Applied Acoustics. Aug. 2010;**71**(8):731-738

[8] Picaut. Numerical modeling of urban sound fields by a diffusion process. Applied Acoustics. Sep. 2002;**63**(9):965-991

[9] Bullen and Fricke. Sound propagation at a street intersection in an urban environment. Journal of Sound and Vibration. Sep. 1977;**54**(1):123-129

[10] Van Renterghem T, Salomons E, Botteldooren D. Parameter study of sound propagation between city canyons with a coupled FDTD-PE model. Applied Acoustics. Jun. 2006;**67**(6):487-510

[11] Can A, Leclercq L, Lelong J, Botteldooren D. Traffic noise spectrum analysis: Dynamic modeling vs. experimental observations. Applied Acoustics. Aug. 2010;**71**(8):764-770

[12] Hornikx M, Forssén J. The 2.5-dimensional equivalent sources method for directly exposed and shielded urban canyons. The Journal of the Acoustical Society of America. 2007;**122**(5):2532

[13] Hornikx M. Acoustic modelling for indoor and outdoor spaces. Journal of Building Performance Simulation. Jan. 2015;**8**(1):1-2

[14] Picaut J, Hardy J, Simon L. Sound propagation in urban areas: A periodic disposition of buildings. Physical Review E. Oct. 1999;**60**(4):4851-4859

[15] Albert DG, Liu L, Moran ML. Time reversal processing for source location in an urban environment. The Journal of the Acoustical Society of America. Aug. 2005;**118**(2):616-619

[16] Albert DG, Liu L. The effect of buildings on acoustic pulse propagation in an urban environment. The Journal of the Acoustical Society of America. Mar. 2010;**127**(3):1335-1346

[17] Heimann D. Three-dimensional linearised Euler model simulations of sound propagation in idealised urban situations with wind effects. Applied Acoustics. Feb. 2007;**68**(2): 217-237

[18] Pallasmaa J. The Eyes of the Skin : Architecture and the Senses. Hoboken, New Jersey: Wiley; 2012

[19] Perez R, Seals R, Michalsky J. All-weather model for sky luminance distribution—Preliminary configuration and validation. Solar Energy. Mar. 1993;**50**(3):235-245

[20] Kittler R, Kocifaj M, Darula S. Daylight Science and Daylighting Technology. New York, NY: Springer New York; 2012

[21] Crow FC. Shadow algorithms for computer graphics. In: Proceedings of the 4th Annual Conference on Computer Graphics and Interactive Techniques - SIGGRAPH '77, vol. 11(2). 1977. pp. 242-248

[22] Verbeck CP, Greenberg DP. A comprehensive light-source description for computer graphics. IEEE Computer Graphics and Applications. Jul. 1984;**4**(7):66-75

[23] Newhall SM, Nickerson D, Judd DB. Final report of the OSA Subcommittee on the spacing of the Munsell colors. Journal of the Optical Society of America. Jul. 1943;**33**(7):385

[24] Julesz B. Foundations of Cyclopean Perception. MIT Press; 2006

[25] Pointer MR, Attridge GG. The number of discernible colours. Color Research and Application. Feb. 1998;**23**(1):52-54

[26] Linhares JMM, Pinto PD, Nascimento SMC. The number of discernible colors in natural scenes. Journal of the Optical Society of America. A, Optics, Image Science, and Vision. Dec. 2008;**25**(12):2918-2924

[27] Lafortune EPF, Foo S-C, Torrance KE, Greenberg DP. Non-linear approximation of reflectance functions. In: Proceedings of the 24th Annual Conference on Computer Graphics and Interactive Techniques - SIGGRAPH '97; 1997. pp. 117-126

[28] Ashikhmin M, Shirley P. An anisotropic Phong BRDF model. Journal of Graphics Tools. Jan. 2000;**5**(2):25-32

[29] Nelson WT, Bolia RS, Ericson MA, McKinley RL. Spatial audio displays for speech communications: A comparison of free field and virtual acoustic environments. In: Proceedings of the Human Factors and Ergonomics Society Annual Meeting, Vol. 43(22); Sep. 1999. pp. 1202-1205

[30] Hofman PM, Van Riswick JGA, Van Opstal J. Relearning sound localization with new ears. Nature Neuroscience. 1998;**1**(5)

[31] Warren JD, Griffiths TD. Neural mechanisms underlying melodic perception and memory for pitch. The Journal of Neuroscience. 1994;14(4):1908-1919

[32] Grothe B, Pecka M, McAlpine D. Mechanisms of sound localization in mammals. Physiological Reviews. Jul. 2010;**90**(3):983-1012

[33] Plack CJ. The Sense of Hearing. 2nd ed. London: Taylor & Francis Group; 2014

[34] Al-barrak L, Kanjo E, Younis EMG. NeuroPlace: Categorizing urban places according to mental states. PLoS One. Sep. 2017;**12**(9):e0183890

[35] Union E. Directive 2002/49/EC of the European Parliament and of the Council of 25 June 2002 Relating to the Assessment and Management of Environmental Noise. EUR-Lex; 2002. [Online]. Available: http://eur-lex.europa.eu/legal-content/GA/TXT/?qid=1399875039336&uri=CELEX%3A32002L0049 [Accessed: 09-Nov-2017]

[36] Welcome to Schal. [Online]. Available: http://www.tpsconsult.co.uk/schal.aspx. [Accessed: 09-Nov-2017]

[37] Jing Y, Xiang N. A modified diffusion equation for room-acoustic predication. The Journal of the Acoustical Society of America. Jun. 2007;**121**(6):3284-3287

[38] Llorca J, Redondo E, Valls F, Fonseca D, Villagrasa S. Acoustic filter. In: Learning and Collaboration Technologies. Novel Learning Ecosystems. Cham: Springer; 2017. pp. 22-33. http://sci-hub.tw/10.1007/978-3-319-58509-3_3

[39] Arau Puchades H. ABC de la acústica arquitectónica. Barcelona: Ediciones CEAC; 1999

[40] Isbert AC. Diseño acústico de espacios arquitectónicos. Barcelona: Edicions UPC; 1998

[41] AENOR. AENOR: Norma UNE-EN ISO 3382:2001. http://www.aenor.es/

[42] Fonseca Escudero D, Pifarré M, Redondo Domínguez E, Alitany A, Sánchez Riera A. Combinación de técnicas cuantitativas y cualitivas en el análisis de la implantación de nuevas tecnologías en el ámbito docente. Uso de la Realidad Aumentada en la visualización del proyecto arquitectónico. Sist. e Tecnol. Informaçao. Atas da 8a Conferência Ibérica Sist. e Tecnol. Informaçao. 19-22 junho 2013; Lisboa, Port. 2013. pp. 205-211

[43] Conde MÁ, García-Peñalvo FJ, Alier M, Piguillem J. The implementation, deployment and evaluation of a mobile personal learning environment. Journal of Universal Computer Science. 2013;**19**(7)

[44] Llorca J, Zapata H, Redondo E, Alba J, Fonseca D. Bipolar laddering assessments applied to urban acoustics education. In: WorldCIST'18 2018: Trends and Advances in Information Systems and Technologies; 2018. pp. 287-297. http://sci-hub.tw/10.1007/978-3-319-77700-9_29

[45] Campanyà C, Fonseca D, Martí N, Peña E, Ferrer A, Llorca J. Identification of significant variables for the parameterization of structures learning in architecture students. In: WorldCIST'18 2018: Trends and Advances in Information Systems and Technologies. 2018. pp. 298-306. http://sci-hub.tw/10.1007/978-3-319-77700-9_30

[46] Hassenzahl M, Tractinsky N. User experience—A research agenda. Behaviour & Information Technology. Mar. 2006;**25**(2):91-97

[47] Brooke J. SUS—A quick and dirty usability scale—WebSM. In: Jordan PW, Thomas B, Lyall I, Weerdmeester A, Thomas A, editors. London: Taylor and Francis, Usability Evaluation in Industry; 1996. pp. 189-194

[48] Nielsen J, Jakob. Usability Engineering. Cambridge, Massachusetts: Academic Press; 1993

[49] Martín Gutiérrez J. Estudio y evaluación de contenidos didácticos en el desarrollo de las habilidades espaciales en el ámbito de la ingeniería. Valencia (Spain): Universitat Politècnica de València; 2010

[50] Bevan N. Practical issues in usability measurement. Interactions. Nov. 2006;**13**(6):42

[51] Fonseca D, Redondo E, Villagrasa S. Mixed-methods research: A new approach to evaluating the motivation and satisfaction of university students using advanced visual technologies. Universal Access in the Information Society. Aug. 2015;**14**(3):311-332. http://sci-hub.tw/10.1007/s10209-014-0361-4

[52] Fonseca D et al. Student motivation assessment using and learning virtual and gamified urban environments. In: Proceedings of the 5th International Conference on Technological Ecosystems for Enhancing Multiculturality - TEEM 2017; 2017. pp. 1-7. DOI: 10.1145/3144826.3145422

[53] Llorca J, Alba J, Mendoza H, Redondo E. Un acercamiento a los paisajes sonoros de la ciutat vella de Barcelona. In: 48o Congreso Español de Acústica, TECNIACÚSTICA 2017; 2017. ISBN: 978-84-87985-29-4

Chapter 4

Formative E-Assessment of Schema Acquisition in the Human Lexicon as a Tool in Adaptive Online Instruction

Guadalupe Elizabeth Morales-Martinez,
Yanko Norberto Mezquita-Hoyos,
Claudia Jaquelina Gonzalez-Trujillo,
Ernesto Octavio Lopez-Ramirez and
Jocelyn Pamela Garcia-Duran

Additional information is available at the end of the chapter

http://dx.doi.org/10.5772/intechopen.81623

Abstract

This chapter presents a comprehensive method of implementing e-assessment in adaptive e-instruction systems. Specifically, a neural net classifier capable of discerning whether a student has integrated new schema-related concepts from course content into her/his lexicon is used by an expert system with a database containing natural mental representations from course content obtained from students and teachers for adapting e-instruction. Mental representation modeling is used to improve student modeling. Implications for adaptive hypermedia systems and hypertext-based instructions are discussed. Furthermore, it is argued that the current research constitutes a new cognitive science empirical direction to evaluate knowledge acquisition based on meaning information.

Keywords: adaptive instruction, technology-enhanced assessment, human lexicon, formative e-assessment

1. Introduction

A significant number of cognitive oriented adaptive hypermedia systems (AHSs) for learning have been developed. Due to the alternative formative character of AHSs emphasizing

learning processes during learning [1], many of these systems are developed mainly by considering users′ cognitive styles, learning style [2–5], previous knowledge before or during an AHS learning [6–8], or intellectual [9].

Typically, an AHS approach demands two types of information processing to achieve two goals [10]. One process consists of gathering information (dependent variables typifying personal and psychological attributes of a user [6–8]), which is used to assign a user to one of several learner models (cognitive classification). Based on the results, a second process adapts the hypermedia instruction (e.g., adaptive content selection, adaptive presentation, and adaptive navigation [11]). **Figure 1** illustrates these processes.

Note from **Figure 1** that achieving the second goal depends completely on achieving the first goal, that is, selecting a learner model. Thus, any weakness in achieving proper student classification demands urgent corrective behavior within the adaptation process to accurately infer the user goals and thus offer navigation support and content adaptation during instruction. Unfortunately, more often than not, the construction of user models is based on weak data collection (descriptive and/or psychological data), and this weakness leads to the implementation of mechanisms to enhance adaptation processes by minimizing the cost of adaptive behavior and increasing user control over adaptation [12], improving [13], and addressing user variability [14], etc. In other words, this process is driven by the corrective adaptivity of the system rather than adaptability in which the user can consciously participate in the adaptation [15].

It is assumed that weaknesses in student modeling frequently stem from using cognitive tools that are controversial, either poorly structured or poorly developed, and many tools are famous for lacking robust empirical support (e.g., learning style/cognitive style instruments [16]). Generally, these tools do not have a good reputation in cognitive science.

Thus, from a cognitive science point of view, there is clearly much to say about student modeling. As we will discuss in the next sections, by digital implementation of more sophisticated cognitive science tools to study human learning and by introducing a third goal regarding assessment in typical adaptive instruction systems, research directions can be expanded to provide innovation in student modeling to enhance AHS.

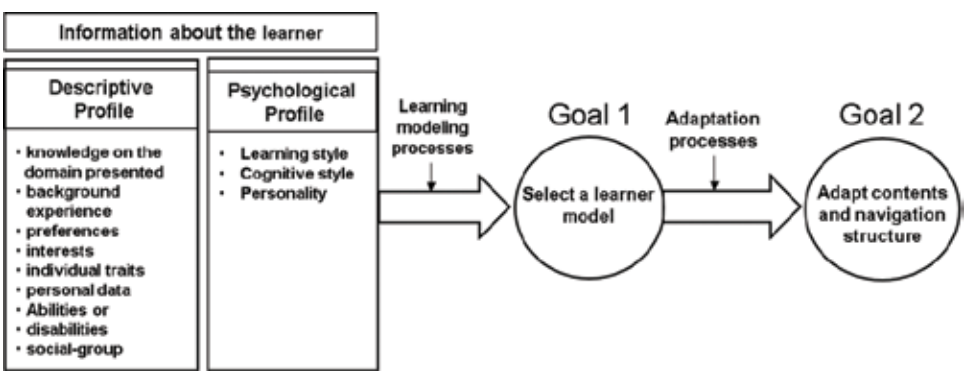


Figure 1. Typical approach in developing adaptive hypermedia technologies.

http://dx.doi.org/10.5772/intechopen.81623

2. Considerations on cognitive science of human learning and cognitive modeling of students

Common sense in formal education assumes that the better we understand learners cognitive functioning during learning, the more effective the instruction can be achieved. Inside the educational technology fields, many intelligent tutoring systems (ITSs) claim to do this by modeling the way students take decisions [17] and solve problems while they are socializing [18] or by considering users emotional states during instruction [19]. Even when this approach has many positive implications to research and development on cognitive ergonomics and engineering psychology [20], it is our strong belief that the current state on ITS is still far from inheriting positive implications from cognitive science research advances. For instance, AHS innovation, instead of considering cognitive research to innovate student modeling to improve error-type analysis of learner's performance during learning, has rested on corrective adaptability of instruction to support learning outcomes [17]. This kind of evaluating a learner's performance resembles summative assessment of learning where the goal is to specify what a student does not know at the end of a course rather than knowing what a student knows during and after learning like in formative assessment of learning approaches [21]. This approach to evaluate learning can be extrapolated to many fields of digital educational technology [22, 23].

To our current paper goals and to illustrate this point in a deeper way, we will introduce next a discussion inside the context of adaptive hypermedia systems (AHSs) to emphasize how education technology development is strengthened when contextualized by basic cognitive research. Here the main goal is to speak in favor of:

a. Innovating education technology by constantly binding basic cognitive science research advances to develop education technology.

b. Considering new empirical directions to integrate assessment of learning and instruction into single parallel formulations to support adaptability of instruction.

Thus, the following description of a formative-oriented AHS computational system is brought as an example on how improvement opportunities are at disposal for ITS research and development. This is achieved by focusing our attention on considering the human lexicon as the starting point to develop an AHS to support constructive learning outcomes.

3. The human lexicon as a potential cognitive construct to implement AHSs

The human mental lexicon is considered a memory capacity to store and meaningfully organize single concepts by connecting them through different types of semantic relations (a mental dictionary). This definition of one of our mental capacities was first appointed by Treisman in 1961 [24], and it is considered a central cognitive structure for language description and human learning (e.g., learning a language).

As it has been the case for most cognitive constructs introduced to explain the human mind, to consider a human lexicon as part of our cognitive architecture has not been an easy task. After heated academic debates, several views (cognitive models) regarding the lexicon have emerged, leaving different research groups to enroll into different theoretical considerations or views ranging from the possibility of a mental dictionary-like system up to the possibility of a no-lexicon view. Thus, currently, three dominant views prevail to guide academic research on this topic [25]: the multiple lexicons view implying different system stores for different lexical information like sensorimotor information, emotion or spatial information [26, 27], the single-lexicon view where all lexical levels are integrated [28], and the no-lexicon view (lexical knowledge without a mental lexicon [29]).

In spite of controversy regarding this topic, the concept of a human lexicon has been appealing enough to bring attention from education technology developers. For instance, Salcedo et al. [30] presented an adaptive hypermedia model (LEXMATH) that can be used as an opportunity to illustrate this point. Specifically, these authors argued that by considering a student's lexicon, learner modeling is optimized. In this AHS model, students´ lexicons regarding general or specific topics are obtained through surveys and are maintained in a database. An ideal lexical domain is obtained from teachers, and during instruction, an expert system optimizes learning paths by adapting navigation support and teaching activities to minimize differences between students´ lexicons and the provided ideal lexical domain in field of mathematics.

These types of models point to a more robust direction to innovate student modeling since it empowers the AHS technology with a developed theoretical framework regarding human mental representation but still incomplete. However, notice that LEXMATH does not subscribe to a specific view or specific model within an academic view of the human lexicon. This model seems to rest on a commonsense view of considering a dictionary-like view of the human lexicon. This excludes the system from using robust methodology to assess specific assumptions of lexical behavior (especially regarding learning) promoted by a cognitive model. Rather, LEXMATH again describes a kind of error-type analysis approach to minimize differences between an expert and a learner where lexical knowledge acquisition (modification) uses indicators unfamiliar to robust cognitive lexicon views. As pointed before, this is not so uncommon since this approach to support cognitive-based instruction based on minimizing differences is frequently used inside modern approaches of ITS or AHS.

As we will describe next, alternative new empirical research directions that impose a strongest connection between basic cognitive research and education technology implementation empower innovation without losing our old tricks to ITS and AHS development. Specifically, to continue with our lexicon model discussion, it is described a cognitive constructive-chronometric system to asses human lexical oriented learning and at the same time improving student modeling to minimize corrective adaptability.

Interestingly, this model subscribes to the third view of the human lexicon, which is the no-lexicon view. As it is expected, whenever an academic effort subscribes to a specific view, it immediately inherits academic criticisms form alternative views. However, by taking this step forward, some advantages are obtained:

a. Methodology is obtained to measure specific assumptions about how lexical knowledge is acquired.

b. In contrast to other alternative lexicon views, the no-lexicon proposal is computational plausible under consideration of recent advances in computer science to model learners, namely connectionism models of mental knowledge representation.

c. Most importantly, as it will be described, the use of artificial neural net classifiers (ANNs) allows researchers to deal with cognitive theoretical developments, suggesting that schemata to assimilate knew knowledge does not really exist in memory but knowledge schema emerges as required for learning and thinking purposes.

Finally, by embedding these cognitive precepts about the human lexicon into AHS development, a prominent role is given to articulate dynamic assessment of learning to adapt and support digital instruction. This requires another way to explore adaptability of an ITS.

4. Adaptive instruction and the constructive/chronometric e-assessment approach

Instruction and assessment are integral parts of teaching to improve students′ experiences [31]. For instance, learning-oriented assessment (also referred to as formative assessment or assessment for learning) requires active participation of students in using feedback and self-monitoring from instruction and assessment as keys to successfully acquire appropriate new knowledge from a course [32]. It is assumed that assessment provides explicit and implicit messages to facilitate a student's academic performance.

Let us first present a general framework of implementing dynamic assessment inside the context of AHS development. In this proposal, assessment is assumed to exert effects at various levels, and it constitutes by itself a domain and a goal. **Figure 2** illustrates this point.

An e-assessment system that complies with these evaluation requirements, implementation viability, and cognitive science principles was first presented by Morales and colleagues [33–35]. At the core of their assessment system (EVCOG, for cognitive evaluator), there is a neural net classifier capable of identifying students who have integrated schema-related concepts from a school course into their lexicon (this schema-related concepts are obtained by using natural semantic nets; **Figure 3A**). The neural net classification capacity is based on the cognitive fact that once a student has integrated new knowledge into her/his long-term memory, a semantic priming effect (in a semantic priming study) is obtained from schema-related words only if meaningful long-term learning has occurred (single-word schemata priming [36, 37]). Thus, the classifier uses a student's schema-related word-recognition times to assess whether the student has integrated new knowledge into long-term memory or has retained information in her/his short-term memory (e.g., to pass a test) or no new schemata were acquired at all. **Figure 3B** shows the role of this net classifier within a cognitive constructive-responsive/chronometric assessment of learning [38, 39].

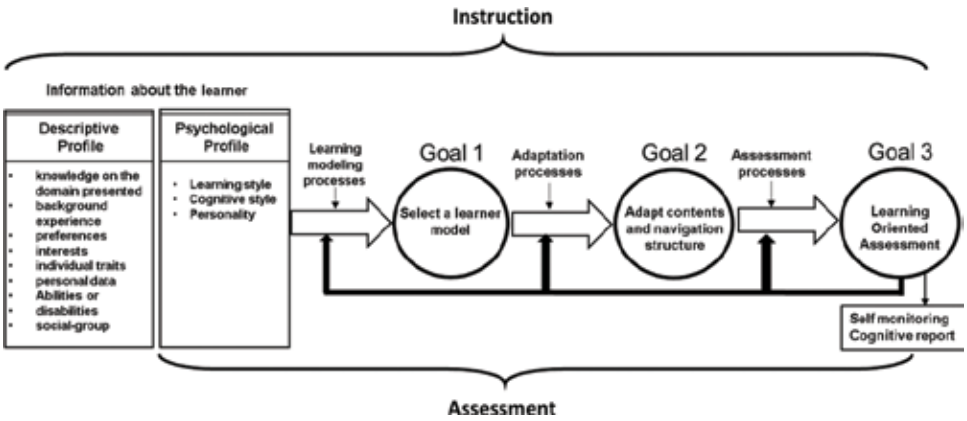


Figure 2. Diagram of how continuous assessment of student knowledge acquisition affects various levels of processing in an AHS learning session.

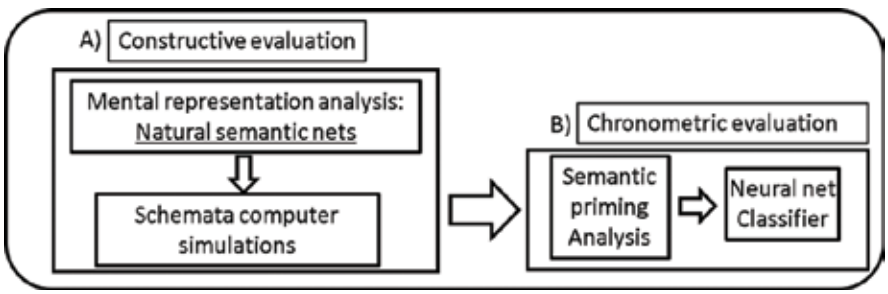


Figure 3. Concepts related to schema course content are selected during a constructive evaluation (A). These concepts can be used to assess whether a student integrated new information into her/his long-term memory using digitized cognitive semantic priming techniques (chronometric evaluation; B). Word-recognition latency patterns are used by the neural net to discriminate between successful and unsuccessful learners.

To train a classifier, hundreds of successful and unsuccessful learners´ schema-related word-recognition patterns are presented to it. Achieving this requires first obtaining schema-related concepts from students before and after a course (after learning).

Figure 4A shows a computer system for obtaining students' and teachers' concept definers to target schema-related concepts using a technique called natural semantic net mapping. This technique produces definitions (using single concept definers such as nouns and adjectives) for represented objects based on their meanings and not on free associations or pure semantic category memberships [40, 41].

In this technique, the 10 highest-ranked definers of each target concept (SAM group) can be used to draw a semantic net, if desired. Some concepts serve as definers for more than one target concept. These are common definers, and other definers and target concepts are interconnected through them. Numerous common definers tend to emerge whenever there are close links among target concepts (schemata).

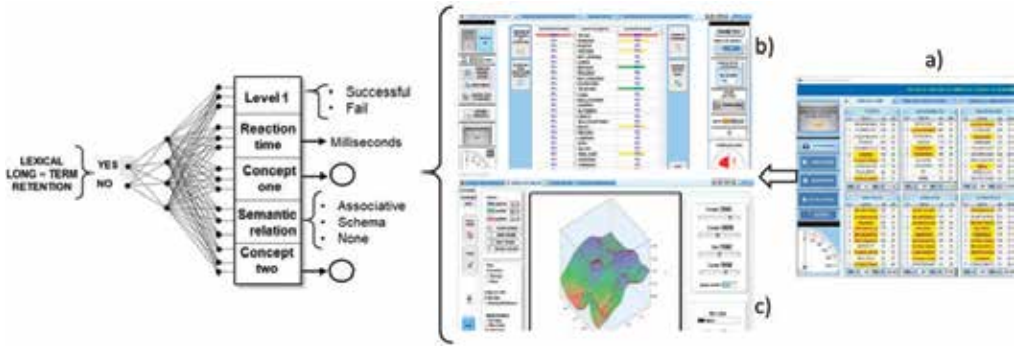


Figure 4. The schema-related concepts (Left) used to train a neural net through semantic priming studies are obtained from simulated connectionist schemata behavior (b), (c), that is based on teachers' and students' conceptual semantic nets (a).

A constraint satisfaction neural net (CSNN) is developed from concept cooccurrence through SAM groups such that the probability that two concepts cooccur or do not cooccur becomes their weight association in a rectangular matrix, with k possible connections with N concepts such that $k = N(N - 1)/2$. Thus, the weight association between two concepts (W) is calculated using the following derivative of the Bayesian formula:

$$Wij = -\ln \left[p(X = 0 \,\&\, Y = 1)\; p(X = 1 \,\&\, Y = 0)\right] \left[p(X = 1 \,\&\, Y = 1)\; p(X = 0 \,\&\, Y = 0)\right] 1, \qquad (1)$$

where X represents one concept in a pair of concepts to be associated, and Y represents another concept. In determining association values among concepts in a natural semantic network such as the one selected earlier, the joint probability value p(X = 1 and Y = 0) can be obtained by calculating how often the definer X of a pair of concepts appears in a list of definers in which Y does not appear, and likewise for the other probability values. These association values are used as an input matrix to the CSNN to simulate schemata of interest [42] (**Figure 4B** and **C**), and a large set of metrics for concept organization and structure can be obtained [41]. From schema simulations and semantic net analysis, schema-related word pairs are selected to implement semantic priming studies. Thus, students' word-recognition latencies to these word pairs are presented to the classifier for student classification (**Figure 4**, left).

5. Empirical support for e-assessment based on the human lexicon

To better describe these concepts, we will describe data resulting from application of constructive-chronometric assessment in an undergraduate psychology course on the computational mind. **Figure 5A** and **B** shows partial instances of definitions obtained from a set of 10 schema-related target concepts relevant to this course before learning (**Figure 5**, top panels) and after learning (**Figure 5**, bottom panels). The following target concepts were provided by the teacher of the course: mind, computation, von Neumann, Turing machine, connectionism, memory, computational mind, working memory, long-term memory, and HPI (human information processing).

In developing a natural semantic net, participants are allowed 60 seconds to provide concept definers. Then, following each definition task, they rank each concept definer (between 1 and 10) in terms of how well they define the target concept. After the system has randomly presented target concepts, it calculates the 10 highest-ranking definers for each target (SAM group; **Figure 5A**). For later consideration in building an expert system, note that **Figure 5A** shows that the M value corresponds to the sum of ranks assigned by all the participants to each definer concept. This value is a measure of the definition relevance for the target concept. Other values, such as the density of the net (G value) and the richness of the definers for each target (J value), are also calculated [40].

Note also that in **Figure 5B**, before learning (top panel), some of the targets lacked complete definitions. Moreover, lower common definers are obtained before learning. This lack of connectivity is reflected when a weight association matrix among concepts is calculated using Eq. (1) (**Figure 5C**, top). This is not the case for the weight association symmetric matrix obtained after learning, shown in **Figure 5C** (bottom). In turn, these connectivity matrixes can be used as an input matrix to many visualization tools, as shown in **Figure 5D**. Before learning, the visual concept organization allows one to immediately note that all the definers were arranged in two main groups connected by a single central one (PROCESSES). In contrast, at the end of the course, the net consists of a more sophisticated concept organization resembling a small world structure characterized by a set of highly clustered neighborhoods and a short average path length in which a small number of well-connected nodes serve as hubs. This net is a normal result of learning when using this technique [41].

This approach to evaluating learning emphasizes two aspects. First, the semantic net focuses on identifying meaning formation. For instance, at the end of the course, students centered

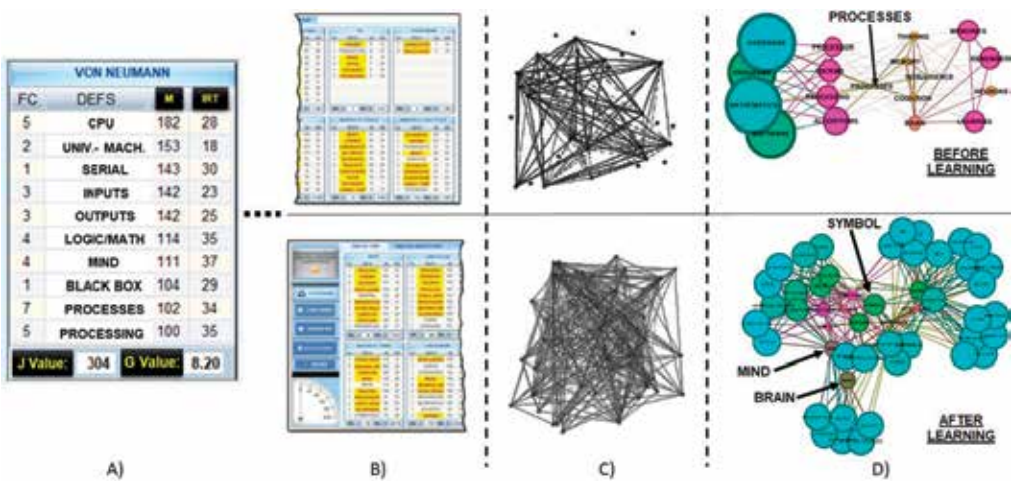


Figure 5. Ten relevant concept definers (SAM group) used to define schema concepts (A) are obtained by a computer system to obtain natural semantic nets (B) before and after a course on the computational mind. Cooccurrence weight associations among concepts (C) and GEPHI analysis (using the Yifan Hu algorithm and (D) can be produced using this semantic mapping technique.

http://dx.doi.org/10.5772/intechopen.81623

their meaning formation around the core concepts of the computational mind: symbol; mind and brain; and the leading figures in this academic field, Turing and von Neumann. The teacher confirmed that this was the intent.

The weight matrix is used by a CSNN to simulate schema behavior, as shown in **Figure 6**. Here, there is 100% activation of the SYMBOLS input. As a result, MIND and DURATION were the only output activated concepts. When the students were asked about this result, they argued that according to what they learned from the course, a core concept in cognitive theory is that all mind activities occur in time, even symbol processing and construction. This schema acquisition was also intended by the teacher. In addition, note from the surface plot in **Figure 6** that balanced positivity and negativity of weight association values (from +10 to −60) enhanced correct discrimination among the schema-related concepts.

By selecting schema-related concepts from the computer models and semantic definers relevant to meaning formation (e.g., emergence of common definers in SAM groups or concepts relevant to a schema), schema word pairs can be selected to perform a semantic word priming study.

Schema-related concepts following the course involve longer word-recognition times since a whole schema is activated (not simply a lexical association).

To illustrate this point, **Figure 7** shows interaction graphs describing a frequent result on schemata word-related time recognition. **Figure 7A** shows that at the beginning of the course, schemata related are not significantly differentiated from other semantically related word pairs. This is not the case at the end of the course where students required significant higher processing time to recognize schemata words (schemata priming).

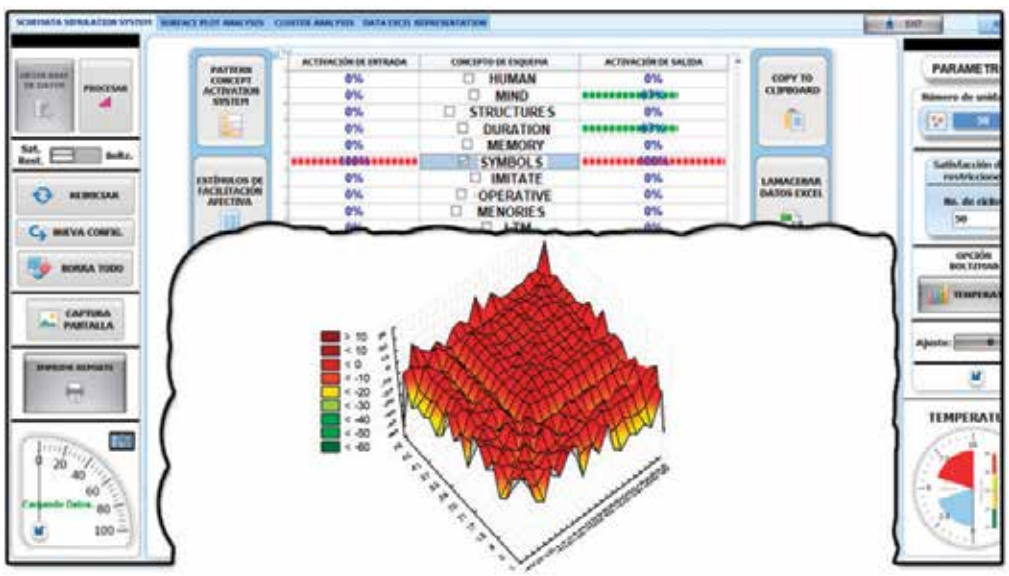


Figure 6. User interface for modeling schema-based behavior, and a surface plot of its underlying weight association matrix (bottom center).

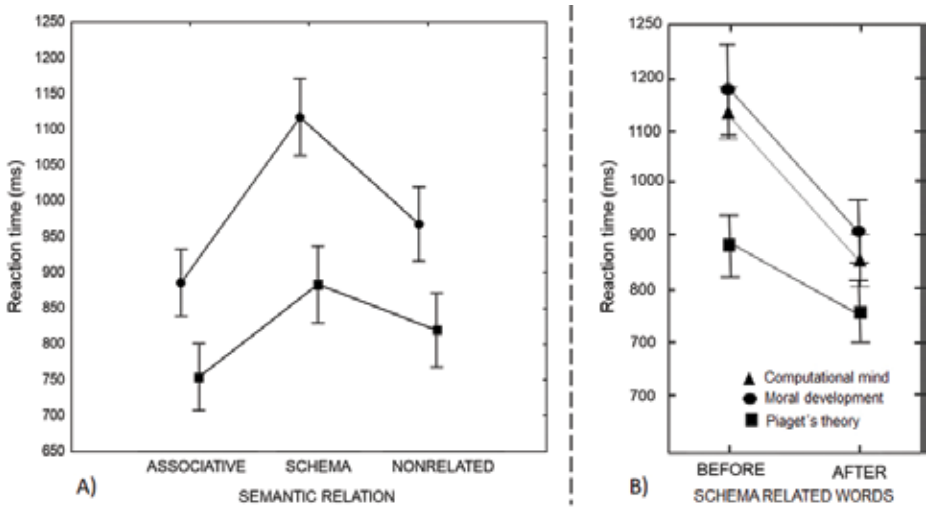


Figure 7. Students' word-recognition latency times corresponding to associative, schema-related, and nonrelated words (A). Comparison of schema priming effects obtained from this study and from similar studies involving other knowledge domains (B).

This schema priming is assumed to occur when schema information is stored in long-term memory, which likely explains why the neural net (after training) is useful for discriminating between successful and unsuccessful learners.

This is relevant because even when we cannot see the existence of a schema in the lexicon, we can track its footsteps as evidence that long-term learning has occurred. On the other hand, it is not necessary to specify a lexicon; it is enough to say that lexical information is obtained and organized as proposed by a no-lexicon view. **Figure 7B** shows that this effect might vary depending on the knowledge domain and the effect of instruction [34, 37, 39].

Several possibilities are introduced by considering a cognitive assessment of learning like the one just described. Let us consider a study [43] carried over 60 first-semester bachelor engineering students who took a course on computer usability. Here, 15 students failed to pass the course, but after a post-season corrective course, they succeeded and achieved the course credit. **Figure 8** shows the mental concept representations obtained by a constructive-chronometric assessment of learning before and after the corrective course.

Note that at the beginning of the course, the EVCOG system shows that students have a mental representation with separated concept clusters (A). This leads to confusion in terms of meaning of a topic. After a corrective course, students presented a single unified schemata knowledge where DESIGN, INTERFACE, and USER showed meaningful centrality to knowledge representation (B). The teacher in charge of the course argued that after looking at the system cognitive report at the beginning of the course, she tried meaningful integration of topics by having the concept of DESIGN as the main reference for meaning formation. Chronometric assessment provided support to this learning process since schemata priming was not obtained at the beginning of the learning period but appeared at the end of classes supporting the idea that students not only successfully passed the course but also obtained long-term retention of schemata.

http://dx.doi.org/10.5772/intechopen.81623

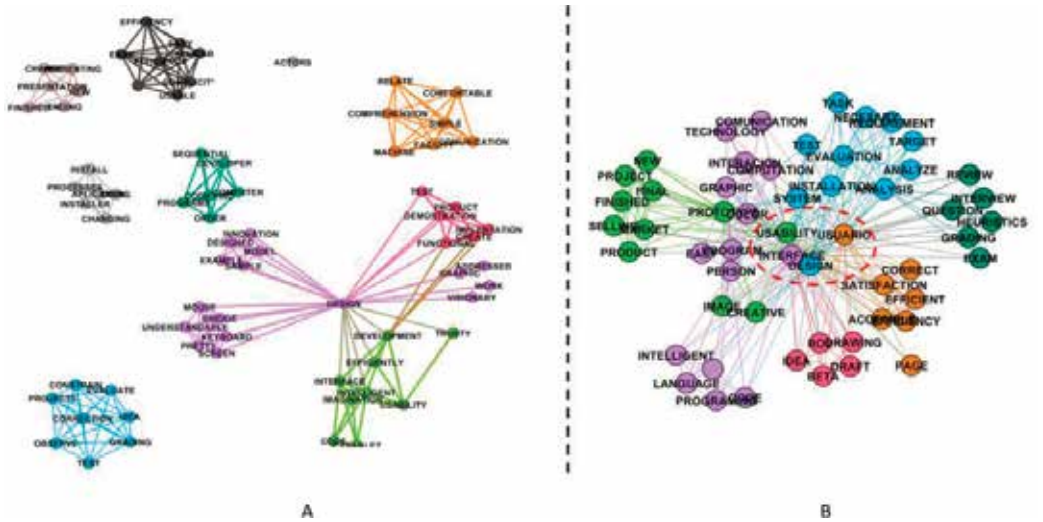


Figure 8. A fractured mental representation on computer usability (A), changing after a corrective course (B).

Is it possible to obtain the same results with an ITS-AHS? Well, notice that now the problem is not cognitive modeling students but instructors. As it will be described next, current academic efforts are being made on this direction.

6. Adaptive e-instruction through e-assessment in e-learning environments: a proposal

Up to this point, the discussion of applying e-assessment to navigation support and adaptation of content has focused only on AHS. Note, however, that the same arguments can be applied to alternative e-instruction systems or alternative e-instruction. For instance, adaptive navigation support for AHS or in e-instruction can be implemented using the same assumptions by considering the model presented in **Figure 9**.

6.1. The student model

In a functional adaptive instruction system such as the one shown in **Figure 8**, the student model is a domain-specific well-trained classifier. Empirical research in several knowledge domains has shown that this type of classifier yields successful classification in 95–98% of instances [38].

6.2. Expert model: determining concept organization of meaning formation

During the defining of a target concept (in natural semantic net mapping), after a student decides which is the highest-ranking concept definer (indicated by its M value), the next-highest-ranked concept from the set of definers depends on the concept frequency (F) in the definition task and the time required to produce it, that is, its interresponse time (ITR; see

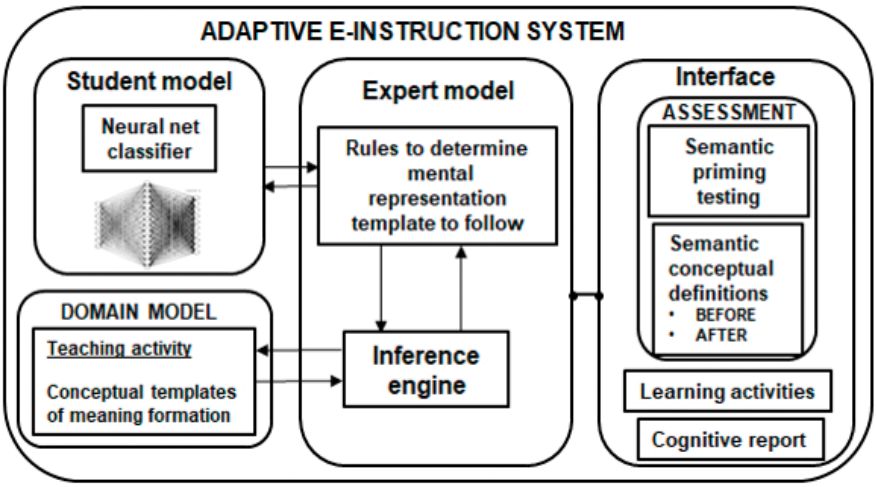


Figure 9. Proposal for adaptive instruction/assessment instruction.

right column in the SAM group in **Figure 5A**). Thus, the M value of each definer can be correctly predicted (98% accuracy) using the following equation [41]:

$$M = \mathrm{A} * e^{(\mathrm{B}/F + \mathrm{C} * ITR)} + \mathrm{D} * \ln(F) \tag{2}$$

where A, B, C, and D are constants obtained from fit analysis. Here, word position in a SAM group is needed only to identify which definer ranks higher since the concept frequency has already been used to filter the SAM group.

Consider the case of a user searching for information on a web page (information foraging). This page must contain linked concepts sufficient for meaning formation (obtained using a natural semantic net). Then, after calculating the M values of selected concepts (considering the time taken by the user to select an available concept; ITR), a comparison can be made to check if the M values corresponding to searching for information on a web page correspond to a proper path of optimized M values corresponding to ideal meaning formation [44].

To illustrate this point, consider **Figure 9**. Here, a user has an initial representation state or initial meaning of web contents. This initial user conceptual organization is not assumed to be identical to the concept organization in a web page (isomorphic) but homomorphic. Information foraging through time (R) is based on a user cognitive strategy to obtain meaning from contents. Thus, transforming conceptual organization (T) and acquiring new concepts serve to obtain valid homomorphic representation of contents such that T′R = RT. A transformation path can be specified as:

$$\mathrm{R}[\mathrm{T}(\mathrm{S}(t), \mathrm{O}(t))] = \mathrm{T}'[\mathrm{R}(\mathrm{S}(t), \mathrm{O}(t))], \tag{3}$$

where O(t) relates to a specific conceptual organization (defined by natural semantic net parameters), which in turns defines R. Furthermore, by using some basic notation from automata theory [45], it is possible to specify a transition rule from 3 as follows:

$$\partial'(q, w) = \partial(\partial'(q, x), a) = T'[R(S(t), O(t))], \quad (4)$$

where meaning formation implies regulation of a transition rule $\partial'(q, w) = T'$ (**Figure 10**).

For example, consider a set of 10 highest-ranked concepts that provide most of connectivity in a natural semantic network [q_0, q_1, q_2, q_3, q_4, q_5, q_6, q_7, q_8, q_9]. Here, proper meaning formation requires going from q0 to q9. Now suppose that after information foraging, a user produces a transition set like [q_1, q_6, q_0, q_3, q_5, q_7, q_4, q_9, q_8, q_{10}] such that:

Natural semantic net ∩ information foraging [q_0, q_1, q_3, q_4, q_5, q_6, q_7, q_8, q_9]

Since user exploration of contents only missed one relevant semantic concept (q_2), it is assumed the user obtained a valid homomorphic mental representation of the meaning implied inside the web page even when the concept path position (estimated M value) is high (by considering Eq. (2)).

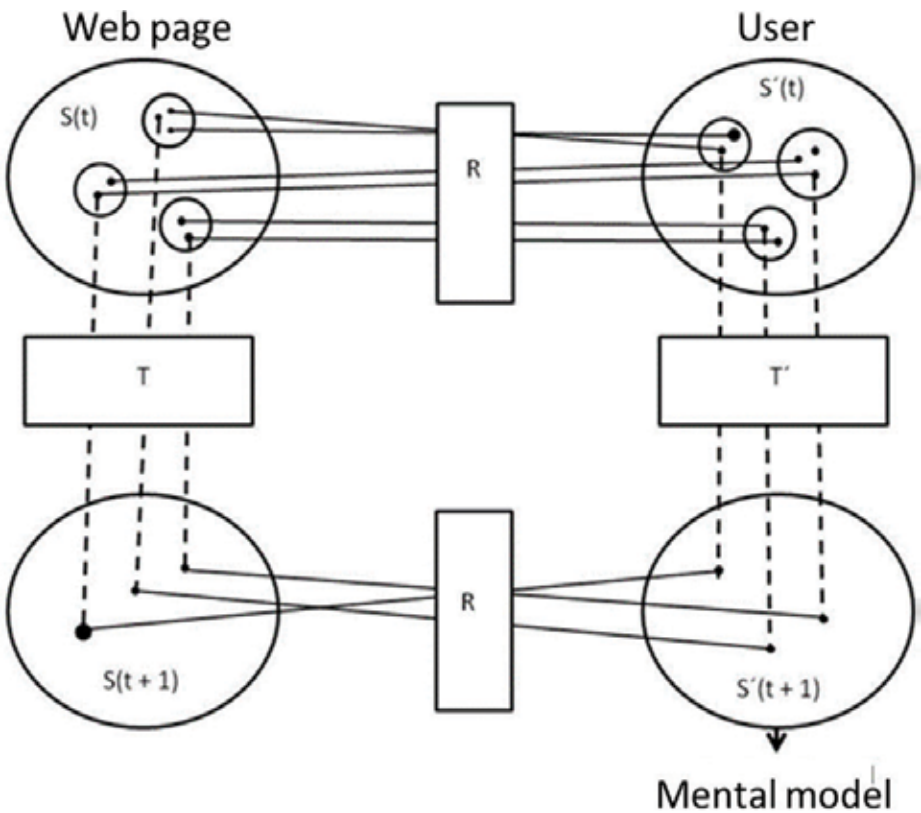


Figure 10. Building a mental model from web page contents through meaning formation [44].

The expert system control mechanisms adapt navigation links that minimize differences between information foraging values and meaning formation (determined by transition rules specified by Eq. (4)), as well as by using the neural net classifier information (successful vs. unsuccessful integration of information in the user's lexicon).

6.3. Expert model: inference engine

The expert system includes a PROLOG backward-chaining inference engine that allows the system to build a valid "mental representation (GOAL)" based on natural semantic net data structures in the knowledge domain (templates) by request of a decision rule. This rule system considers whether schema priming for a specific module has been achieved by consulting the neural net classifier and by comparing the obtained path M values against an ideal descending organization of M values. If a semantic effect is not obtained, then the following events occur:

1. The subsequent knowledge modules remain disabled.
2. The system instructs the inference engine to use the database to construct the closest mental representation based on the user's concept path (link set). Then, the navigation is modified based on the template that best approximates the user's initial exploration, and the user is prompted to try again.

Currently, research is being performed to achieve a dynamic optimization of search information by adapting navigational support based on minimization of differences between meaning values of the user and knowledge domains rather than waiting for the user to complete a knowledge module.

6.4. Knowledge domain

An adaptive e-instruction system (AHS/hypertext) within the present scope requires a database containing natural semantic networks similar to those described earlier. Here, templates are data structures containing SAM groups and their semantic values in which information can be accessed by a PROLOG-based inference engine. As the sample for developing these SAM groups is enlarged, better predictions for adapting navigational support can be achieved.

6.5. Interface

As shown in **Figure 11A**, when a student begins a learning session, she/he is presented with a menu of options of the course content.

Before and after exploring each module, a semantic priming study must be performed to provide the expert system module with information for adapting the navigation support by modifying the link structure in a module based on a meaning formation template. After a

http://dx.doi.org/10.5772/intechopen.81623

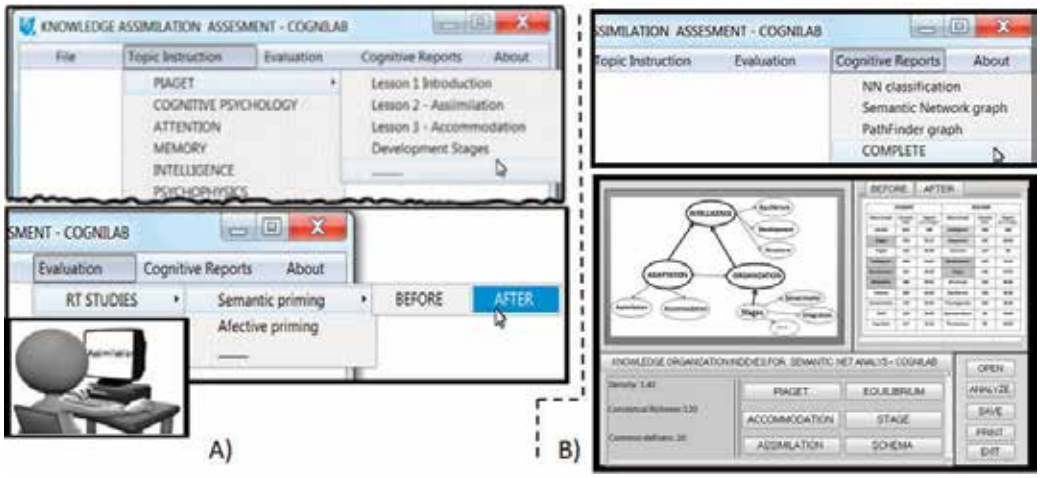


Figure 11. Interface showing a menu of options of instruction (A), and assessment (B).

module or an entire course is completed, the user can obtain a cognitive performance report (**Figure 11B**). This report serves as explicit assessment results that empower a user to adapt the searching of content (encouraging adaptability), whereas the link modification is an implicit message of corrective adaptability to improve proper meaning formation of the content. Selection of learning activities either using hypermedia or by modification of knowledge content depends on the expert model's evaluation of the user's meaning formation.

7. Conclusions

The goal of the proposed system is to promote assessment tasks as learning tasks, student involvement in assessment, and forward-looking feedback in adaptive e-instruction systems [32]. This system reduces the enormous delay in e-assessment innovation: e-assessment has been limited to mere digitization of traditional, sometimes ancient, evaluation methods [35].

A new empirical research line is opened in which student modeling is improved by using tools of cognitive science in adaptive e-learning systems in ways that were not possible before. We believe that research exploring the human lexicon as a way to adapt instruction will be at the center of future developments in AHS/hypertext.

Acknowledgements

This study was supported by DGAPA-UNAM with a grant from the Support Program of Research and Technological Innovation Projects (PAPIIT) <<TA400116>>.

Author details

Guadalupe Elizabeth Morales-Martinez[1]*, Yanko Norberto Mezquita-Hoyos[2], Claudia Jaquelina Gonzalez-Trujillo[3], Ernesto Octavio Lopez-Ramirez[4] and Jocelyn Pamela Garcia-Duran[1]

*Address all correspondence to: gemoramar@hotmail.com

1 Cognitive Science Laboratory, Institute of Research in the University and Education, National Autonomous University of Mexico (UNAM, IISUE), Mexico City, Mexico

2 Department of Psychology, Universidad Autonoma de Yucatan, Merida City, Mexico

3 Department of Education, University of Monterrey (UDEM), Monterrey, Mexico

4 Cognitive Science Laboratory, Department of Psychology, Nuevo Leon Autonomous University (UANL), Monterrey, Mexico

References

[1] Gouli E, Papanikolaou K, Grigoriadou M. Personalizing assessment in adaptive educational hypermedia systems. In: De Bra P, Brusilovsky P, Conejo R, editors. Adaptive Hypermedia and Adaptive Web-based Systems. Vol. 2347. Heidelberg: Springer Verlag; 2002. pp. 153-163. DOI: 10.1007/3-540-47952-X_17

[2] Mampadi F, Chen SY, Ghinea G, Chen MP. Design of adaptive hypermedia learning systems: A cognitive style approach. Computers & Education. 2011;**56**:1003-1011. DOI: 10.1016/j.compedu.2010.11.018

[3] Triantafillou E, Pomportsis A, Demetriadis S. The design and the formative evaluation of an adaptive educational system based on cognitive styles. Computers & Education. 2003;**41**:87-103. DOI: 10.1016/S0360-1315(03)00031-9

[4] Bernard J, Chang TW, Popescu E, Graf S. Learning style identifier: Improving the precision of learning style identification through computational intelligence algorithms. Expert Systems with Applications. 2017;**75**:94-108. DOI: 10.1016/j.eswa.2017.01.021

[5] Popescu E. A unified learning style model for technology enhanced learning: What, why and how? International Journal of Distance Education Technologies. 2010;**8**:65-81. DOI: 10.4018/jdet.2010070105

[6] Martins AC, Faria L, Vaz de Carvalho C, Carrapatoso E. User modeling in adaptive hypermedia educational systems. Educational Technology & Society. 2008;**11**:194-207 Available from: https://www.jstor.org/stable/jeductechsoci.11.1.194 [Accessed: May 07, 2018]

[7] Mampadi F, Ghinea G, Huang PR, Chen SY. Influence of prior knowledge and cognitive styles in adaptive hypermedia learning systems. In: Mizoguchi R, Sitthisak O, Hirashima

T, Biswas G, Supnithi T, Yu FY, editors. Proceedings of the 19th International Conference on Computers in Education. Thailand: Asia-Pacific Society for Computers in Education; 2011. pp. 1-3

[8] Triantafillou E, Georgiadou E, Economides AA. Adaptive hypermedia systems: A review of adaptivity variables. In: Proceedings of the Fifth Panhellenic Conference on Information and Communication Technologies in Education. Greece: Thessaloniki; 2006. pp. 75-82

[9] Greene JA, Costa LJ, Robertson J, Pan Y, Deekens VM. Exploring relations among college students' prior knowledge, implicit theories of intelligence, and self-regulated learning in a hypermedia environment. Computers & Education. 2010;**55**:1027-1043. DOI: 10.1016/j.compedu.2010.04.013

[10] Popescu E. Dynamic adaptive hypermedia systems for e-learning. Education [thesis]. France: Universite de Technologie de Compiegne HAL archives; 2008

[11] Tsianos N, Germanakos P, Lekkas Z, Mourlas C. An assessment of human factors in adaptive hypermedia environments. In: Mourlas C, Germanakos P, editors. Intelligent User Interfaces: Adaptation and Personalization Systems and Technologies. Greece: National & Kapodistrian University of Athens; 2009. pp. 1-34. DOI: 10.4018/978-1-60566-032-5.ch001

[12] Tsandilas T, Schraefel MC. Usable adaptive hypermedia systems. New Review of Hypermedia and Multimedia. 2004;**10**:5-29. DOI: 10.1080/13614560410001728137

[13] De Bra P. Pros and cons of adaptive hypermedia in web-based education. CyberPsychology & Behavior. 2000;**3**:71-77. DOI: 10.1089/109493100316247

[14] Conati C, Gertner A, VanLehn K. Using Bayesian networks to manage uncertainty in student modeling. Journal of User Modeling and User-Adapted Interaction. 2002;**12**: 371-417. DOI: 10.1023/A:1021258506583

[15] Rodríguez V, Ayala G. Adaptivity and adaptability of learning object's interface. International Journal of Computer Applications. 2012;**37**:6-13. Available from: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.259.2896&rep=rep1&type=pdf [Accessed 2018-06-07]

[16] Reynolds M. Learning styles: A critique. Management Learning. 1997;**28**:115-133. DOI: 10.1177/1350507697282002

[17] Mitrovic A. Modeling domains and students with constraint-based modeling. In: Nkambou R, Bourdeau J, Mizoguchi R, editors. Advances in Intelligent Tutoring Systems. Studies in Computational Intelligence. Berlin Heidelberg: Springer Verlag; 2015. pp. 63-80. DOI: 10.1007/978-3-642-14363-2_4

[18] Olsen JK, Belenky DM, Aleven V, Rummel N. Intelligent tutoring systems for collaborative learning: Enhancements to authoring tools. In: Lane HC, Yacef K, Mostow J, Pavlik P, editors. Artificial Intelligence in Education. 16th International Conference. AIED 2013. Berlin Heidelberg: Springer; 2013. pp. 900-903. DOI: 10.1007/978-3-642-39112-5_141

[19] San Pedro MO, Baker RS, Gowda SM, Heffernan NT. Towards an understanding of affect and knowledge from student interaction with an intelligent tutoring system. In: Lane HC, Yacef K, Mostow J, Pavlik P, editors. Artificial Intelligence in Education. 16th International Conference. AIED 2013. Berlin Heidelberg: Springer; 2013. pp. 41-50. DOI: 10.1007/978-3-642-39112-5_5

[20] Harris D. Engineering psychology and cognitive ergonomics. In: 12th International Conference EPCE International Conference on Engineering Psychology and Cognitive Ergonomics (EPCE) Held as Part of HCI International 2015; 2-7 August 2015; Las Vegas. USA: Proceedings; 2015. pp. 365-372. DOI: 10.1007/978-3-642-21741-8

[21] Arieli-Attali M. Formative assessment with cognition in mind: The cognitively based assessment of, for and as learning (CBALTM) research initiative at Educational Testing Service. In: Proceeding of the 39th Annual Conference on Educational Assessment 2.0: Technology in Educational Assessment. 2013. pp. 1-11

[22] Rainer L. Using semantic networks for assessment of learners' answers. In: The Sixth IEEE International Conference on Advanced Learning Technologies (ICALT '06); July 2005; Netherlands: Kerkrade; pp. 1070-1072. DOI: 10.1109/ICALT.2006.1652631

[23] Tu LY, Hsu WL, Wu SH. A cognitive student model—An ontological approach. In: Proceedings of the International Conference on Computers in Education (ICCE '02); 3-6 December 2002; Auckland, New Zealand: Proceedings; 2002. pp. 163-164. DOI: 10.1109/CIE.2002.1185877

[24] Coltheart MR, Perry C, Langdon R, Ziegler J. DRC: A dual route cascaded model of visual word recognition and reading aloud. Psychological Review. 2001;**108**:204-256. Available from: http://psycnet.apa.org/buy/2001-16162-009 [Accessed: May 07, 2018]

[25] De Sousa LB, Gabriel R. Does the mental lexicon exist? Revista de Estudos da Linguagem. 2015;**23**:335-361. DOI: 10.17851/2237.2083.23.2.335-361

[26] Huth AG, de Heer WA, Griffiths TL, Theunissen FE, Gallant JL. Natural speech reveals the semantic maps that tile human cerebral cortex. Nature. 2016;**532**:453-468. DOI: 10.1038/nature17637

[27] Ullman MT. The biocognition of the mental lexicon. In: Gaskell MG, editor. The Oxford Handbook of Psycholinguistics. Oxford, UK: Oxford University Press; 2007. pp. 267-286. DOI: 10.1093/oxfordhb/9780198568971.001.0001

[28] Rogers T, McClelland JL. Semantic Cognition: A Parallel Distributed Processing Approach. Massachusetts, USA: MIT Press; 2004. DOI: 10.1017/S0140525X0800589X

[29] Elman JL. On the meaning of words and dinosaur bones: Lexical knowledge without a lexicon. Cognitive Science. 2009;**33**:1-36. DOI: 10.1111/j.1551-6709.2009.01023.x

[30] Salcedo P, Pinninghoff MA, Contreras R, Figueroa JF. An adaptive hypermedia model based on student's lexicon. Expert Systems. 2017;**34**:e12222. DOI: 10.1111/exsy.12222

[31] Joughin G. Introduction: Refocusing assessment. In: Gordon J, editor. Assessment, Learning and Judgement in Higher Education. New York: Springer; 2009. pp. 1-11. DOI: 10.1007/978-1-4020-8905-3_1

[32] Keppel M, Carless D. Learning-oriented assessment: A technology-based case study. Assessment in Education: Principles, Policy & Practice. 2006;**13**:179-191. DOI: 10.1080/09695940600703944

[33] Lopez RE, Morales MG, Hedlefs AM, Gonzalez TC. New empirical directions to evaluate online learning. International Journal of Advances in Psychology. 2014;**3**:40-47. DOI: 10.14355/ijap.2014.0302.03

[34] Morales MG, Lopez RE, Velasco MD. Alternative e-learning assessment by mutual constrain of responsive and constructive techniques of knowledge acquisition evaluation. International Journal for Infonomics. 2016;**9**:1195-1200. DOI: 10.20533/iji.1742.4712.2016.0145

[35] Morales MG, Lopez RE, Castro C, Villarreal G, Gonzales TC. Cognitive analysis of meaning and acquired mental representations as an alternative measurement method technique to innovate e-assessment. European Journal of Educational Research. 2017;**6**: 455-464. DOI: 10.12973/eu-jer.6.4.455

[36] Lopez RE, Theios J. Single word schemata priming: A connectionist approach. In: The 69th Annual Meeting of the Midwestern Psychological Association, Chicago, IL. 1996

[37] Gonzalez CJ, Lopez RE, Morales GE. Evaluating moral schemata learning. International Journal of Advances in Psychology. 2013;**2**:130-136

[38] Morales MG, Lopez RE, Lopez GA. New approaches to e-cognitive assessment of e-learning. International Journal for e-Learning Security (IJeLS). 2015;**5**:449-453. DOI: 10.20533/ijels.2046.4568.2015.0057

[39] Morales MG, Lopez RE. Cognitive responsive e-assessment of constructive e-learning. Journal of e-Learning and Knowledge Society. 2016;**12**:10-19. DOI: https:10.20368/1971-8829/1187

[40] Figueroa JG, Gonzales GE, Solis V. An approach to the problem of meaning: Semantic networks. Journal of Psycholinguistic Research. 1976;**5**:107-115. DOI: 10.1007/BF01067252

[41] Morales MG, Santos AM. Alternative empirical directions to evaluate schemata organization and meaning. Advances in Social Sciences Research Journal. 2015;2:51-58. DOI: http://dx.doi.org/10.14738/assrj.29.2015

[42] Rumelhart DE, Smolensky P, McClelland JL, Hinton GE. Schemata and sequential thought processes. In: McClelland JL, Rumelhart DE, The PDP Research Group, editors. Parallel Distributed Processing: Explorations in the Microstructure of Cognition. Volume 2: Psychological and Biological Models. Massachusetts: MIT Press; 1986. pp. 7-57

[43] Gonzales CJ, Lopez RE, Hedlefs MI. A new empirical approach to assess learning outcomes due to meaning formation and constructive knowledge. In: International Conference on Education Technology Management, University of Barcelona, Spain; December 19-21, 2018

[44] Torres GF, Lopez RE. Rastreo de la Información en Páginas web a través del Significado (foraging information in webpages through meaning). Daena: International Journal of Good Conscience. 2010;**5**:308-323. Available from: http://www.spentamexico.org/v5-n2/5(2)308-323.pdf [Accessed: June 07, 2018]

[45] Hopcroft JE, Motwani R, Ullman JD. Teoría de autómatas, Lenguajes y computación (Introduction to Automata Theory, Languages, and Computation). 3rd ed. New York: PEARSON Addison-Wesley; 2008. 434 p

Section 3

Face Recognition

Chapter 5

Local Patterns for Face Recognition

Chih-Wei Lin

Additional information is available at the end of the chapter

http://dx.doi.org/10.5772/intechopen.76571

Abstract

The main objective of the local pattern is to describe the image with high discriminative features so that the local pattern descriptors are more suitable for face recognition. The word "local" represents the measured image with the subregion and is the key in this chapter. Regardless of the techniques proposed, the local pattern is one of the most interesting areas in face recognition. The local facial descriptor is a local pattern that generates the descriptor by considering the subregion of an image. Techniques based on various combination methods from the local facial descriptors are not unusual. This chapter is concerned primarily to help the reader to develop a basic understanding of the local pattern descriptors and how they apply to face recognition. We begin to describe the outline of the local pattern in face recognition and its relative facial descriptors. Next, we give an introduction to the popular local patterns and establish examples to demonstrate the process of each method. To the end of this chapter, we conclude those methods with a discussion of issues related to the properties of the local patterns.

Keywords: local pattern, micropattern, face recognition, descriptor

1. Introduction

Due to the intelligence security monitoring is more popular in recent years, the automatically recognizing face is needed for various visual surveillance systems, for example the accessing control system for personal or company to verify the legal/illegal people, policing system for identifying the thief and the robber who presents the illegal behavior in public or private space. To construct an efficient face recognition system, the facial descriptor with discriminated characteristic is required.

The facial descriptor refers to the process of extracting the discriminative features to represent a given face image. Numerous methodologies are proposed to recognize face and those can be classified as global and local facial descriptors. The global facial descriptor describes the facial characteristics with the whole face image, such as principal component analysis (PCA) [1, 2] and linear discriminant analysis (LDA) [3, 4]. PCA converts the global facial descriptor from high dimension to low dimension by using the linear transform methodology to reduce the computational cost. Linear discriminant analysis (LDA) also called the Fishers Linear Discriminant is similar to PCA, while it is a supervised methodology. Although the global facial descriptor can extract the principal component from the facial images, reduces the computational cost, and maintains the variance of the facial image, the performance is sensitive to the change of the environment, such as the change of light.

The flexibilities of the local facial descriptors are better than the global facial descriptors because they successfully and effectively represent the spatial structure information of an input image. A well local facial descriptor generates discriminative and robust features to achieve good recognition results with computational simplicity. In this chapter, we represent a number of approaches in the local facial descriptor including the local binary pattern (LBP), local derivation pattern (LDP), local tetra pattern (LTrP), local vector pattern (LVP) and local clustering pattern (LCP).

2. Local pattern descriptor

A local pattern considers the variations of subregion in an image, which is also called a micropattern. In this section, we introduce the basic and several popular techniques of local pattern descriptor for facial recognition.

2.1. Local binary pattern

Local binary pattern (LBP) [5] is designed to describe the texture in a local neighborhood is an invariant texture measure and has been various comparative studies, such as fingerprint recognition [6], face recognition [7], and license plate recognition [8]. The main characteristics of LBP are: (1) highly discriminative capability (2) and computational efficiency.

The basic LBP encodes the pixels of an image by thresholding 3×3 neighborhood of each pixel with the given referenced pixel X_c and concatenates the results to form a binary expression, as shown in **Figure 1**. The equation of basic LBP operator is formulated as follows:

$$LBP_P(X_c) = \left\{ f_1\left(I\left(X_p\right), I(X_c)\right)\right\}\big|_{p=1,2,\ldots,P} \tag{1}$$

where $X_p, p = 1, \ldots, 8$ is a neighborhood of the referenced pixel X_c in the local subregion of an image I. $f_1(\cdot,\cdot)$ is the coding scheme which decides the binary number of each neighborhood, called the *threshold function* and this can be expressed as

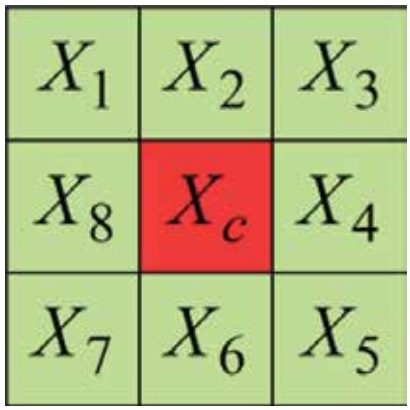


Figure 1. Example of 8-neighborhood surrounding a referenced pixel X_c.

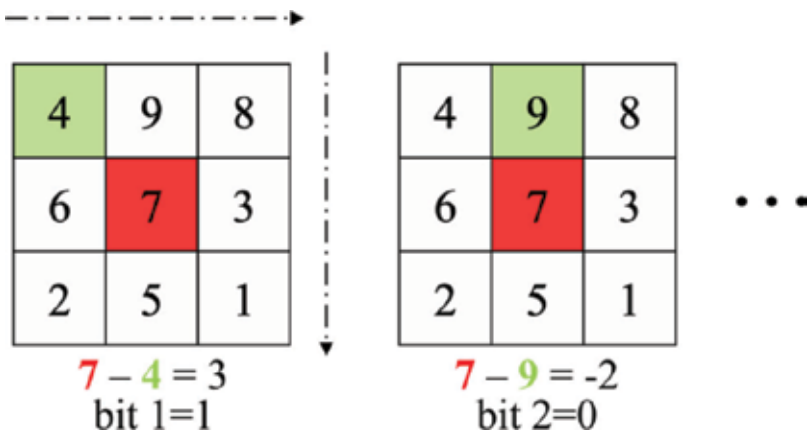


Figure 2. Example of generating LBP for local region.

$$f_1\left(I\left(X_p\right), I(X_c)\right) = \begin{cases} 1, & if\left(I\left(X_p\right) - I(X_c)\right) \geq T_1 \\ 0, & else \end{cases} \tag{2}$$

where p is the index of the neighborhoods which is surrounding the referenced pixel X_c. and T_1 is the threshold. $f_1(\cdot,\cdot)$ represents the gradient variation between a given referenced pixel X_c and its neighborhoods. In practical, the threshold T_1 can be set to 0, if $f_1\left(I\left(X_p\right), I(X_c)\right) = 0$, it means the neighborhoods X_p have higher gradient information compared with referenced pixel X_c. **Figure 2** is an example of generating an LBP micropattern. **Figure 2** demonstrates that LBP is generated by using Eqs. (1) and (2) from X_1 to X_8 and encodes the binary pattern of a give reference pixel X_c as 10011111. **Figure 3** demonstrates the spatial distribution of the example of LBP as shown in **Figure 2** in one-dimensional. In **Figure 3**, the neighborhoods, which are encoded as 1, are arranged on the right of reference pixel X_c, and the others are

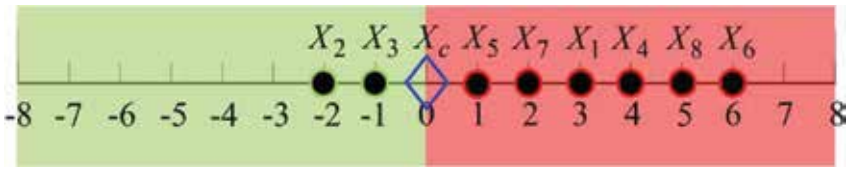


Figure 3. Spatial distribution of example of LBP.

arranged on the left of reference pixel X_c. The distance is the gradient variant between reference pixel and its neighborhoods as shown in **Figure 3**.

Furthermore, to address the problem of the textures at different scales, there are some followers which extend to use neighborhoods with various scales [9, 10]. To compare with basic LBP, the local neighborhoods are evenly spaced on a circle centered at the reference pixel X_c, and the formulation of Eqs. (1) and (2) is re-formulated as follows:

$$LBP_{P,R}(X_c) = \left\{ f_1\left(I\left(X_{p,r}\right), I(X_c)\right)\right\}\big|_{p=1,2,\cdots,P;\, r=1,..R} \tag{3}$$

$$f_1\left(I\left(X_{p,R}\right), I(X_c)\right) = \begin{cases} 1, & if\left(I\left(X_{p,R}\right) - I(X_c)\right) \geq T_1 \\ 0, & else \end{cases} \tag{4}$$

where r is the radius between the referenced pixel X_c and its neighborhood pixels X_p. **Figure 4** illustrates examples of circular neighborhoods with any radius and number of sampling points. The neighbor that does not fall in the center of a pixel is estimated by using bilinear interpolation.

2.2. Local derivative pattern

LBP is a nondirectional first-order local derivative pattern of images and fails to extract more detailed information, such as the directions between neighborhoods and referenced pixel, and the high-order gradient information. Local derivative pattern (LDP) can be considered as an

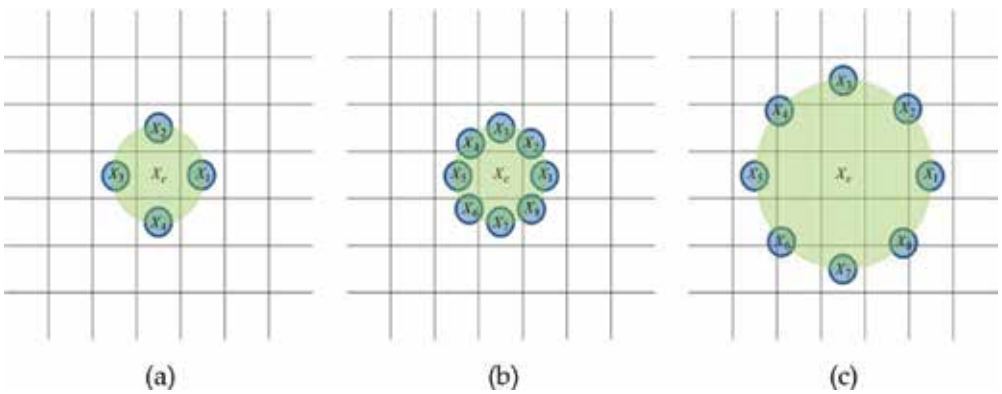


Figure 4. Circularly symmetric neighborhoods sets for different (P, R). (a) (P=4,R=1), (b) (P=8,R=1), (c) (P=8,R=2).

extension of LBP with directional high-order local derivative pattern [11]. To encode the n^{th}-order LDP, the $(n-1)^{th}$-order local derivative variations with various distinctive spatial relationships along 0˚, 45˚, 90˚, and 135˚ directions are used. The first-order derivatives of the referenced pixel X_c along 0˚, 45˚, 90˚, and 135˚ directions can be written as

$$I'_{0^\circ} = I(X_c) - I(X_4) \tag{5}$$

$$I'_{45^\circ} = I(X_c) - I(X_3) \tag{6}$$

$$I'_{90^\circ} = I(X_c) - I(X_2) \tag{7}$$

$$I'_{135^\circ} = I(X_c) - I(X_1) \tag{8}$$

where I is a given image, X_c is the referenced pixel and X_p, $p = 1, 2, 3, 4$ are the neighborhoods of X_c as shown in **Figure 1**. Then, the second-order LDP can be encoded as,

$$LDP^2_\alpha(X_c) = \left\{ f_2\left(I'_\alpha(X_c), I'_\alpha(X_1)\right), f_2 I'_\alpha(X_c), I'_\alpha(X_2), \cdots, f_2 I'_\alpha(X_c), I'_\alpha(X_p) \right\}|_{p=1,2,\cdots P} \tag{9}$$

where α is derivative direction at referenced pixel X_c along 0˚, 45˚, 90˚, and 135˚ directions and $f_2(\cdot,\cdot)$ is the binary coding function which describes the spatial relationship between referenced pixel X_c and its neighborhoods X_p in various derivative directions, and that can be expressed as

$$f_2\left(I'_\alpha(X_c), I'_\alpha(X_p)\right) = \begin{cases} 1, & \text{if } I'_\alpha(X_c) \cdot I'_\alpha(X_p) \le 0 \\ 0, & \text{if } I'_\alpha(X_c) \cdot I'_\alpha(X_p) > 0 \end{cases} \quad p = 1, 2, \cdots, P \tag{10}$$

The spatial relationship between two pixels includes the conditions of turning and monotonically increasing/decreasing and be coded as 1 and 0 in LDP, respectively.

Finally, the second-order LDP is defined as the concatenation of the four directional LDPs

$$LDP^2(X_c) = \left\{ LDP^2_\alpha(X_c) | \alpha = 0^\circ, 45^\circ, 90^\circ, 135^\circ \right\}. \tag{11}$$

The $(n-1)^{th}$-order derivatives along 0˚, 45˚, 90˚, and 135˚ directions are calculated by modifying Eqs. (5)–(8) and be re-formulated as

$$I^{n-1}_{0^\circ} = I^{n-2}_{0^\circ}(X_c) - I^{n-2}_{0^\circ}(X_{4,R}) \tag{12}$$

$$I^{n-1}_{45^\circ} = I^{n-2}_{45^\circ}(X_c) - I^{n-2}_{45^\circ}(X_{3,R}) \tag{13}$$

$$I^{n-1}_{90^\circ} = I^{n-2}_{90^\circ}(X_c) - I^{n-2}_{90^\circ}(X_{2,R}) \tag{14}$$

$$I^{n-1}_{135^\circ} = I^{n-2}_{135^\circ}(X_c) - I^{n-2}_{135^\circ}(X_{1,R}) \tag{15}$$

Then, the n^{th}-order LDP, $LDP^n_\alpha(X_c)$, in α derivative direction at referenced pixel X_c is expressed as

$$LDP_{\alpha}^{n}(X_c) = \left\{ f_2\left(I_{\alpha}^{n-1}(X_c), I_{\alpha}^{n-1}\left(X_{p,R}\right)\right)\right\} | p = 1, 2, \cdots, P; R = 1, \alpha = 0^{\circ}, 45^{\circ}, 90^{\circ}, 135^{\circ} \quad (16)$$

An example of high-order derivative is shown in **Figure 5**. **Figure 5(a)** is the original value of image, **Figure 5(b)** is the first-order derivative in 0° direction by using Eq. (5), and **Figure 5(c)** is the second-order derivative in 0° direction by using Eq. (12) with the value in **Figure 5(b)**.

Figure 6 demonstrates an example to encode the second-order LDP in 0° direction. To encode the second-order LDP, the results of first-order derivatives are needed. Taking the bit 1 as an example, the results of first-order derivatives of referenced pixel X_c and the neighborhood X_1 are $(7-3) = 4$ and $(4-9) = -5$, respectively. The spatial relationship between neighborhood X_1 and referenced pixel X_c is turning $(7-3) \times (4-9) = -20 \leq 0$. Therefore, we encode the bit 1 as 1 by Eq. (10). Similarly, the spatial relationship between referenced pixel X_c and neighborhoods pixels X_p,$p = 4, 5, 7, 8$ presents the turning and be encoded as "1". The reset of neighborhoods pixels is encoded as "0". The second-order LDP in 0° direction, $LDP^2_{P,R=1,\alpha=0^{\circ}}$, is encoded as "10011011". According to the same encoding process, the results of second-order LDP in 45°, 90° and 135° are $LDP^2_{P,R=1,\alpha=45^{\circ}} = 01110100$, $LDP^2_{P,R=1,\alpha=90^{\circ}} = 11100001$, and $LDP^2_{P,R=1,\alpha=135^{\circ}} = 10011101$, respectively. Finally, $LDP^2_{P,R=1} = 10011011011101001110000110011101$ with 32-bit is generated by concatenating the four 8-bit LDPs with various derivative directions.

Figure 7 demonstrates the spatial distribution of example of LDP in 0° direction in one-dimensional. In **Figure 7**, the evaluation results of LDP in 0° direction are normalized into the region of $[-8\ 8]$, the neighborhoods that are encoded as 1 are arranged on the left of 0, and the others are arranged on the right of 0. The distance is the magnitude of gradient variant between reference pixel X_c and its neighborhoods.

2.3. Local tetra pattern

Local tetra pattern (LTrP) [12] adopts the concepts of LBP and LDP which extends the spatial relationship from one-dimensional to two-dimensional. LTrP uses two high-order derivative

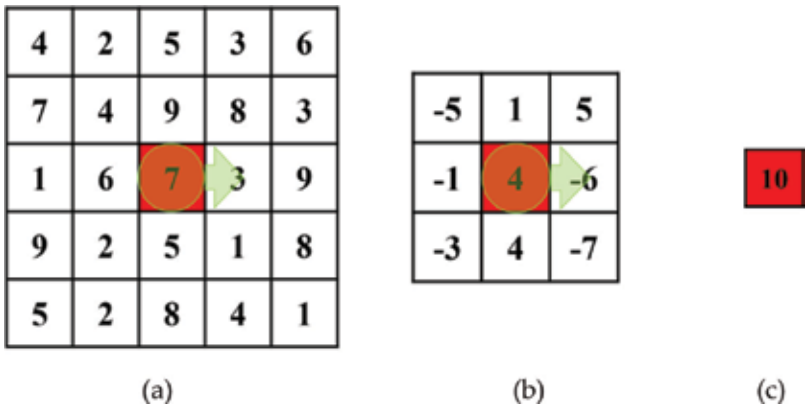


Figure 5. Example of high-order derivative in 0° direction. (a) Original values (b) First-order (c) Second-order.

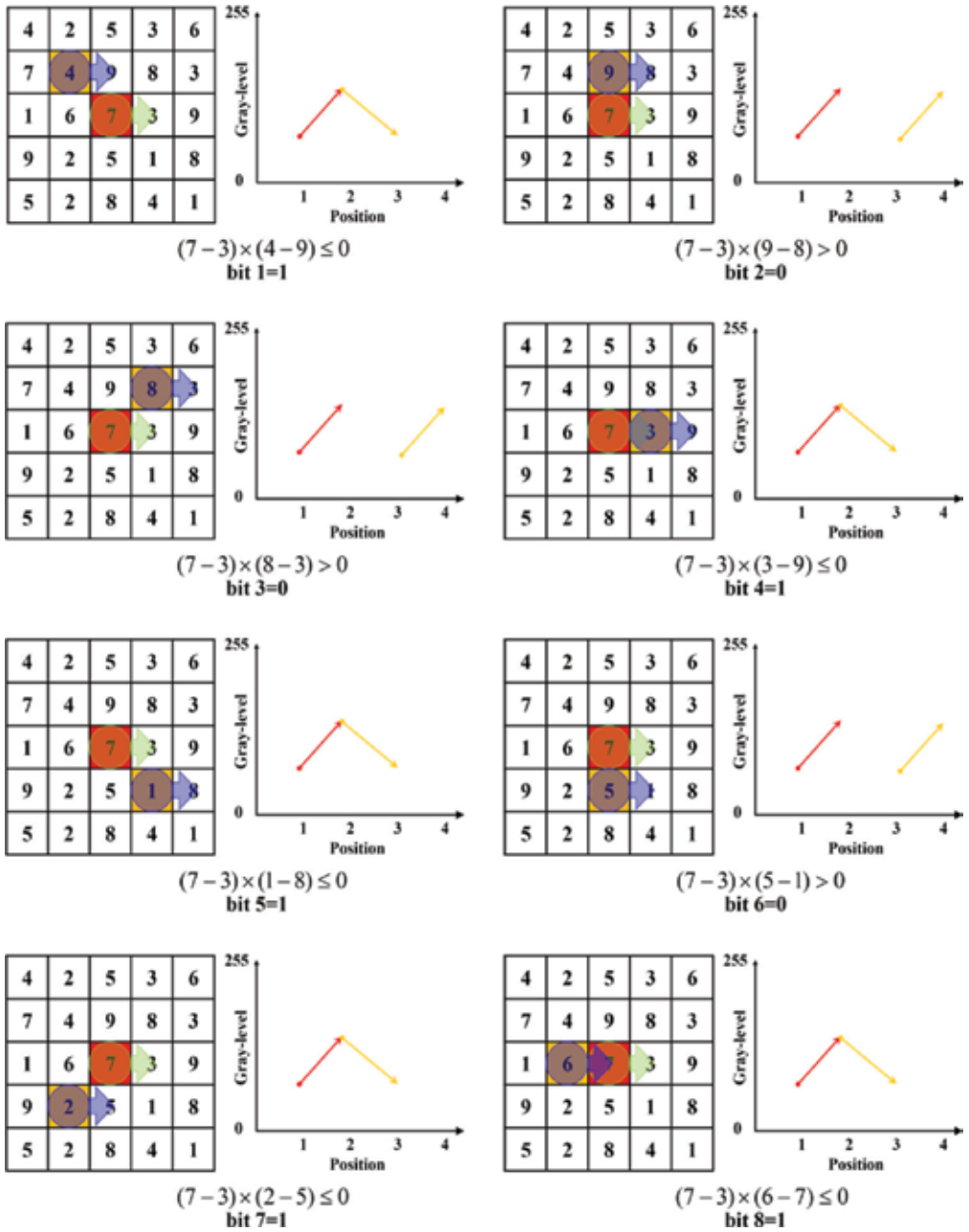


Figure 6. Example to encode second-order LDP in 0° direction.

directions with four distinct values to encode the micropattern for extract more discriminative information. The n^{th}-order LTrP is derivative from $(n-1)^{th}$-order derivatives along 0° and 90° which can be written as

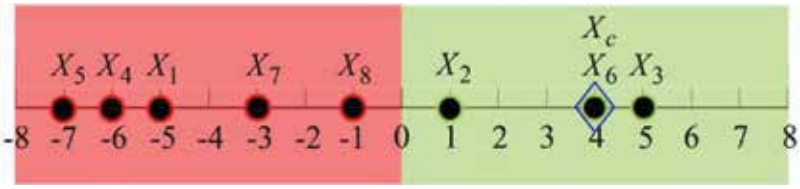


Figure 7. Spatial distribution of example of LDP in 0° direction.

$$I_{0^\circ}^{n-1} = I_{0^\circ}^{n-2}(X_{h,R}) - I_{0^\circ}^{n-2}(X_c) \tag{17}$$

$$I_{90^\circ}^{n-1} = I_{90^\circ}^{n-2}(X_{v,R}) - I_{90^\circ}^{n-2}(X_c) \tag{18}$$

where X_c is the referenced pixel, $X_{h.R}$ and $X_{v.R}$ horizontal and vertical neighborhoods of referenced pixel X_c, respectively; R is the distance between reference pixel X_c and its neighborhood; $I_{0^\circ}^{n-2}(\cdot)$, $I_{90^\circ}^{n-2}(\cdot)$ are the $(n-2)$-order derivatives in 0° and 90° directions, respectively; $I_{0^\circ}^{n-1}(\cdot)$, and $I_{90^\circ}^{n-1}(\cdot)$ are the $(n-1)$-order derivatives in 0° and 90° directions, respectively. Then, the direction of the referenced pixel X_c can be expressed as the quadrant representation and be defined as

$$I_{Dir}^{n-1}(X_c) = \begin{cases} 1, \ I_{0^0}^{n-1}(X_c) \geq 0 \ and \ I_{90^0}^{n-1}(X_c) \geq 0 \\ 2, \ I_{0^0}^{n-1}(X_c) < 0 \ and \ I_{90^0}^{n-1}(X_c) \geq 0 \\ 3, \ I_{0^0}^{n-1}(X_c) < 0 \ and \ I_{90^0}^{n-1}(X_c) < 0 \\ 4, \ I_{0^0}^{n-1}(X_c) \geq 0 \ and \ I_{90^0}^{n-1}(X_c) < 0 \end{cases} \tag{19}$$

where $I_{Dir}^{n-1}(X_c)$ describes the direction of the referenced pixel X_c along 0° and 90° directions with quadrant. Then, the n^{th}-order tetra pattern of referenced pixel X_c, $LTrP_{P,R}^{n}(X_c)$, is encoded as

$$LTrP_{P,R}^{n}(X_c) = \left\{ f_3\left(I_{Dir}^{n-1}(X_{p,R}), I_{Dir}^{n-1}(X_c)\right) \right\}\Big|_{p=1,2,\cdots,P;R=1} \tag{20}$$

where $f_3(\cdot,\cdot)$ is the coding function which describes the referenced pixel X_c with four quadrants and be written as

$$f_3\left(I_{Dir}^{n-1}(X_{p,R}), I_{Dir}^{n-1}(X_c)\right) = \begin{cases} I_{Dir}^{n-1}(X_{p,R}), & if \ I_{Dir}^{n-1}(X_{p,R}) \neq I_{Dir}^{n-1}(X_c) \\ 0 & else \end{cases} \tag{21}$$

Figure 8 illustrates the coding scheme of Eq. (21), if the quadrant of the referenced pixel X_c is as same as its neighborhood, the corresponding bit of tetra pattern is assigned to be "0", otherwise, the bit is assigned to be the same as the neighborhood. Then, the tetra patterns are decomposed into three binary patterns as follows:

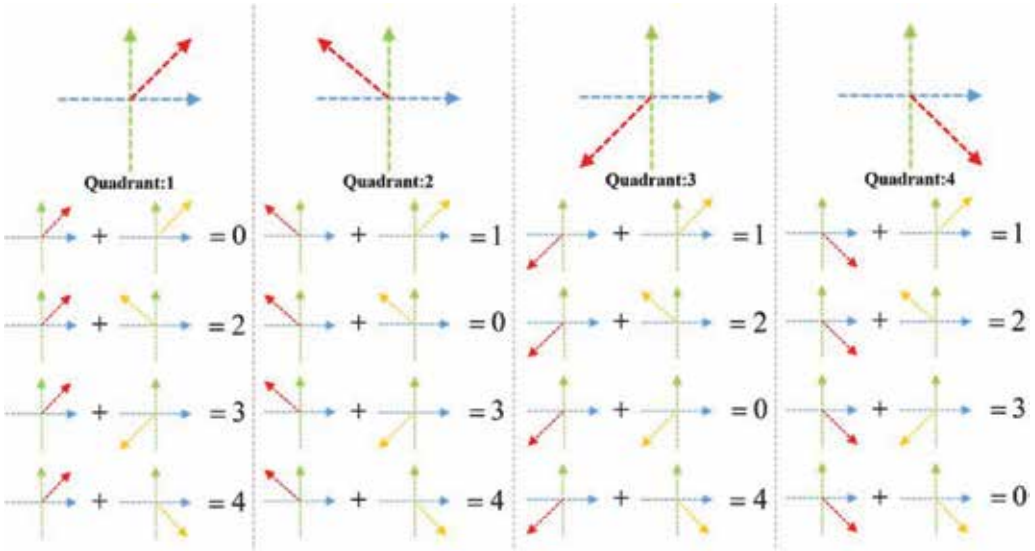


Figure 8. Illustration of coding LTrP micropattern.

$$LTrP^{n}_{P,R}\Big|_{\overline{Dir}} = f_4\left(LTrP^{n}_{P,R}(X_c)\right)\Big|_{\overline{Dir},\,p=1,2,\cdots P}$$
$$f_4\left(LTrP^{n}_{P,R}(X_c)\right)\Big|_{\overline{Dir}} = \begin{cases} 1, & if\ LTrP^{n}_{P,R}(X_c) = \overline{Dir} \\ 0, & else \end{cases} \tag{22}$$

where $\overline{Dir}$ contains four quadrants except the quadrant of the referenced pixel X_c and $f_4(\cdot,\cdot)$ is a coding function to generate the three binary patterns. Similarly, the three tetra patterns are encoded according to the abovementioned procedure for the rest directions of the referenced pixel. Therefore, the four tetra patterns with 12 8-bit binary patterns are generated. Moreover, the 13th 8-bit binary pattern is considered which is the magnitudes of horizontal and vertical first-order derivatives and be calculated by the following equation,

$$LTrP_{P,M} = f_5\left(M(X_p) - M(X_c)\right)\big|_{p=1,2,\cdots P} \tag{23}$$

$$M(X_i) = \sqrt{\left(I^1_{0^\circ}(X_p)\right)^2 + \left(I^1_{90^\circ}(X_p)\right)^2} \tag{24}$$

$$f_5\left(M(X_p) - M(X_c)\right) = \begin{cases} 1, & if\ M(X_p) - M(X_c) \geq 0 \\ 0, & else \end{cases} \tag{25}$$

where $M(X_p)$ is the magnitudes of horizontal and vertical first-order derivatives and $f_5(\cdot,\cdot)$ is a coding function to generate the binary patterns of the magnitude. **Figure 9** demonstrates an example of the second-order LTrP which takes a subregion as shown in **Figure 5(a)** as an example. The quadrant of referenced pixel X_c is 4, which is assigned by using Eq. (18) with

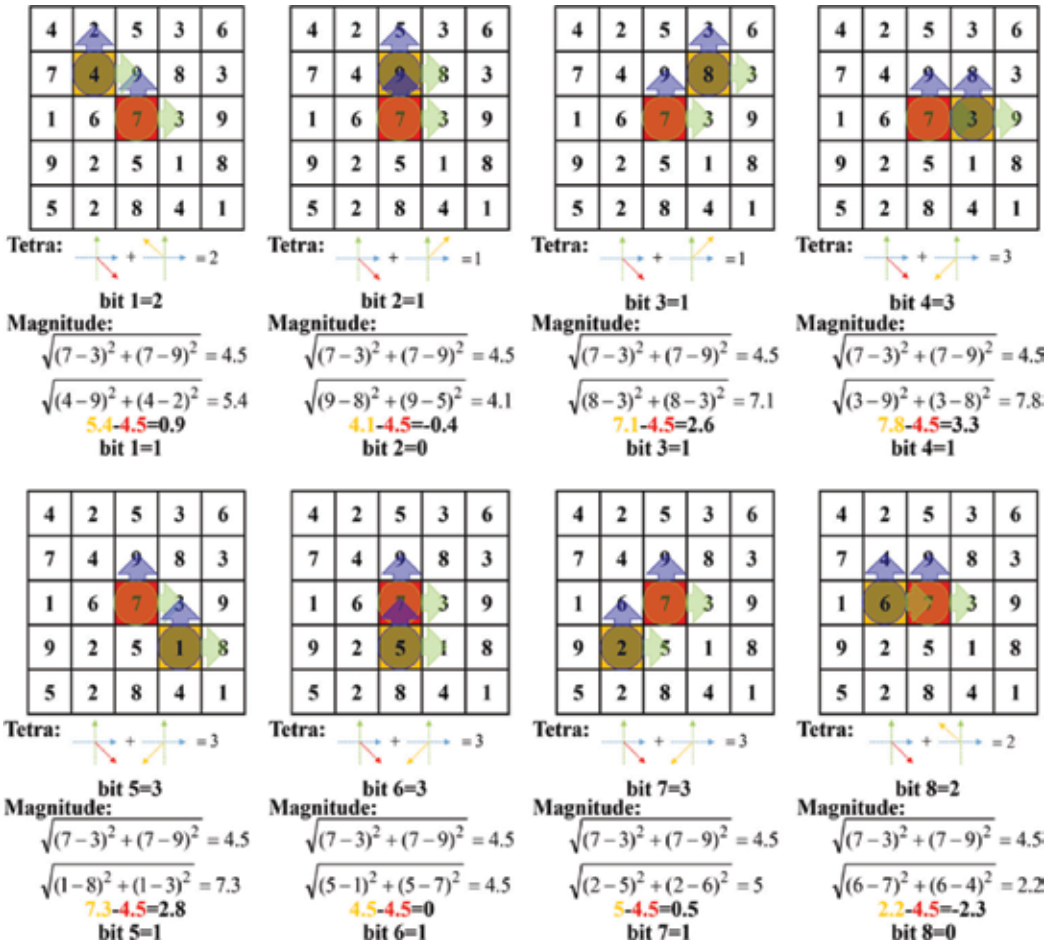


Figure 9. Example of coding second-order LTrP micropattern.

the first-order derivatives in 0° and 90° directions. Similarly, the quadrants of each neighborhood of referenced pixel X_c are 2, 1, 1, 3, 3, 3, 3, 2, respectively. We take the neighborhood pixel X_1 as an example, the quadrants of X_c and X_1 are 4 and 2, respectively, which is not the same. Thus, the corresponding bit of the LTrP is assigned to be "2" as shown in **Figure 9**. Similarly, the remaining bits of the LTrP are encoded by using the same procedure and the complete LTrP can be expressed as $LTrP^2_{P,R} = 21133332$. Then, the tetra pattern is decomposed into three 8-bit binary pattern according to Eq. (22). To generate the first 8-bit binary pattern, the tetra pattern with symbol "1" is set to be "1", and the rest symbols of tetra pattern are set to be "0". Then, we obtain the first 8-bit binary pattern "01100000". Repeatedly, we generate the other 8-bit binary patterns "10000001" and "00011110" by considering the tetra pattern values "2" and" 3", respectively. Finally, the 12 8-bit binary pattern is obtained by concatenating the rest tetra patterns with three directions (1, 2, and 3) of referenced pixel. The additional binary pattern is obtained from the magnitude and be encoded as "10111110".

2.4. Local vector pattern

Local vector pattern (LVP) [13] is inspired by local binary pattern (LBP) which is sample and intuitive. To compare with LBP and LDP, LVP further considers the neighborhood relationship with various distances from different directions and the relationship between various derivative directions.

LVP is a micropattern in high-order derivative space which considers the direction value in encoding procedure, as shown in **Figure 10**. The derivative direction vector of the referenced pixel X_c, $V_{\beta,D}(X_c)$, with various directions and distance are formulated as

$$V_{\beta,D}(X_c) = I\left(X_{\beta,D}\right) - I(X_c) \tag{26}$$

where I is a local subregion of an image, β is the index of angle (direction), and D is the distance between referenced pixel X_c and its neighbors. $V_{\beta,D}(X_c)$ is the derivative vector of the referenced pixel X_c along the β direction with D distance. **Figure 10** demonstrates the distance between X_c and its neighbors are 1, 2, and 3 and are marked with green, blue and yellow, respectively.

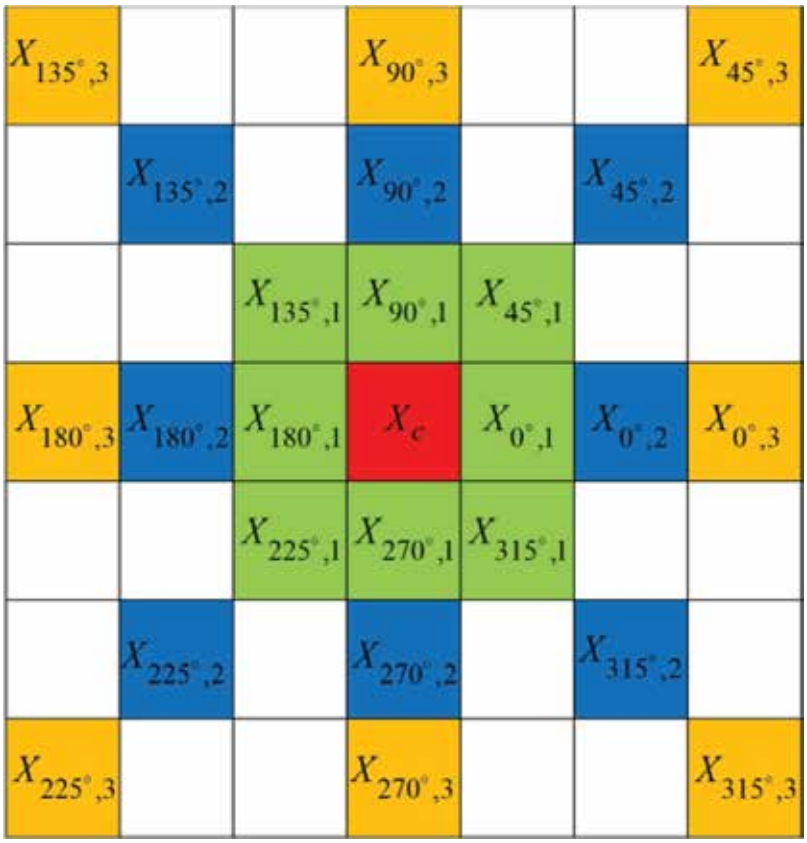


Figure 10. Neighborhoods pixels of $V_{\beta,D}(X_c)$ with various distance along different directions.

The LVP, $LVP_{\beta}(X_c)$, in β derivative direction at referenced pixel X_c is encoded as

$$LVP_{P,R,\beta}(X_c) = \left\{ f_5\left(V_{\beta,D}\left(G_{p,R}\right), V_{\beta+45^\circ,D}\left(G_{p,R}\right), V_{\beta,D}(G_c), V_{\beta+45^\circ,D}(G_c)\right)\right\}\Big|_{p=1,2,\cdots P,R=1} \tag{27}$$

where $f_5(\cdot,\cdot)$ is the coding function which can be formulated as

$$f_5\left(V_{\beta,D}\left(G_{p,R}\right), V_{\beta+45^\circ,D}\left(G_{p,R}\right), V_{\beta,D}(G_c), V_{\beta+45^\circ,D}(G_c)\right) = \begin{cases} 1, & if\ V_{\beta+45^\circ,D}\left(G_{p,R}\right) - \left(\dfrac{V_{\beta+45^\circ,D}(G_c)}{V_{\beta,D}(G_c)} \times V_{\beta,D}\left(G_{p,R}\right)\right) \geq 0 \\ 0, & else \end{cases} \tag{28}$$

Finally, the LVP of referenced pixel X_c is defined as the four 8-bit binary patterns, as shown in the following,

$$LVP_{P,R}(X_c) = \left\{ LVP_{P,R,\beta}(X_c) \middle| \beta = 0^\circ, 45^\circ, 90^\circ, 135^\circ \right\} \tag{29}$$

To extend the discriminative of 2D spatial structures, LVP integrates four pairwise directions $(0^\circ - 45^\circ, 45^\circ - 90^\circ, 90^\circ - 135^\circ, 135^\circ - 0^\circ)$ of vector to form a 32-bit binary pattern for each referenced pixel X_c.

The coding function of LVP is a weight vector of dynamic linear decision function which is a comparative space transform (CST) and addresses the two-class problem in pattern recognition. The dynamic linear decision function, $CST\left(X_{p,R}\right)$, can be formulated as

$$CST\left(X_{p,R}\right) = w(X_c)^T \cdot v\left(X_{p,R}\right) \tag{30}$$

where $w(X_c)^T$and $v\left(X_{p,R}\right)$ are the weight vector and pairwise direction value of the neighborhoods which are surrounded by referenced pixel X_c in two different directions. The formulations of $w(\cdot)$and $v(\cdot)$ can be expressed as,

$$w(X_c) = \left(1, \frac{V_{\beta+45^\circ,D}(X_c)}{V_{\beta,D}(X_c)}\right) \tag{31}$$

$$v\left(X_{p,R}\right) = \left(V_{\beta+45^\circ,D}\left(X_{p,R}\right), V_{\beta,D}\left(X_{p,R}\right)\right)^T \tag{32}$$

where the first term of $w(\cdot)$ is to describe the original value of neighborhood pixel $X_{p,R}$ at $(\beta + 45^\circ)$ direction and the second term is the transform ratio which compares the derivative value of the neighborhood $X_{p,R}$ in β direction to that of in $(\beta + 45^\circ)$ direction surrounds around the referenced pixel X_c. $v(\cdot)$ is the augmented pattern which presents the pairwise direction values of vector of neighborhood pixel $X_{p,R}$. Then, Eq. (30) can be rewritten as,

$$CST\left(X_{p,R}\right) = w(X_c)^T \cdot v\left(X_{p,R}\right) = V_{\beta+45^\circ,D}\left(X_{p,R}\right) - \frac{V_{\beta+45^\circ,D}(X_c)}{V_{\beta,D}(X_c)} \times V_{\beta,D}\left(X_{p,R}\right) \tag{33}$$

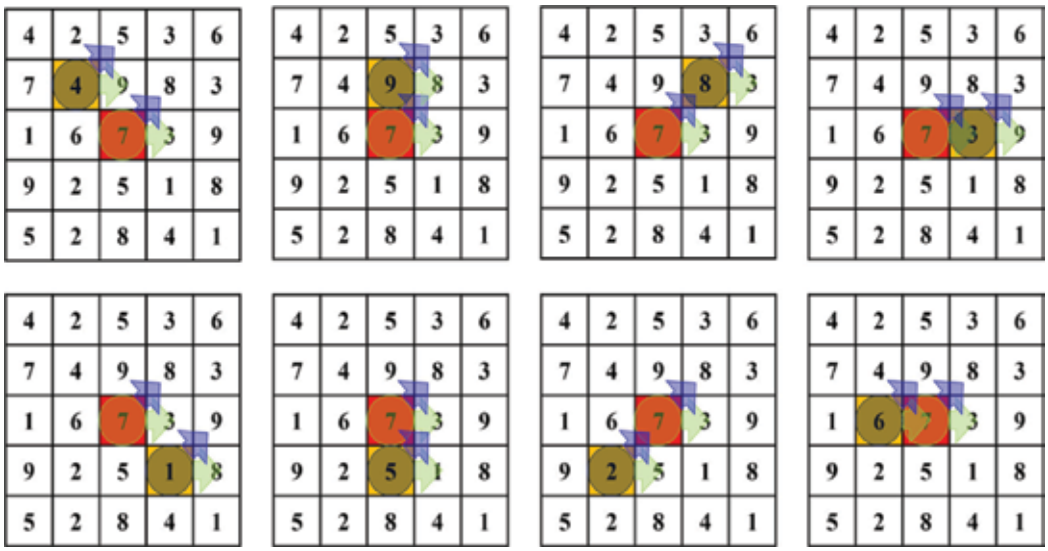


Figure 11. Example of first-order LVP in $\beta = 0^\circ$ direction.

We take the example of the local subregion of an image as shown in **Figure 5(a)** to illustrate the encoding process of generating first-order LVP, as shown in **Figures 11** and **12**. **Figure 11** illustrates the first-order LVP of the referenced pixel $X_c = 7$ in $\beta = 0^\circ$ direction. In **Figure 11**, we calculate the pairwise derivative direction vector of the referenced pixel X_c to form the 2D spatial structures, as shown in **Figure 12**. In **Figure 12**, the pairwise derivative direction vectors $V_{\beta,D}(X_c)$ and $V_{\beta+45^\circ,D}(X_c)$ are indicated as x- and y-axis, respectively, in which, $\beta = 0^\circ$ and $D = 1$. The first-order derivative direction value of referenced pixel X_c and its neighborhoods in directions $\beta = 0^\circ$ and $\beta + 45^\circ = 45^\circ$ are shown in **Figure 13**. Then, we calculate the transform ratio $\frac{V_{\beta+45^\circ,D}(X_c)}{V_{\beta,D}(X_c)} = \frac{-1}{4} = -0.25$ which is used to transform the β-direction value of the neighborhoods to comparative space $\beta + 45^\circ$-direction. The CST value of neighborhood pixel $X_1 = 4$ of referenced pixel $X_c = 7$ is evaluated according to Eq. (33) ($CST(X_{1,1}) = V_{45^\circ,1}(X_{1,1}) - \frac{V_{45^\circ,D}(X_c)}{V_{0^\circ,D}(X_c)} \times V_{0^\circ,1}(X_{1,1}) = -1 - \frac{-1}{4} \times -5 = -2.25$). Then, the first corresponding bit of the 8-bit binary codes of $LVP_{P,R,0^\circ}(X_c) = 01100100$ is encoded by using sign function. Similarly, the rest of LVPs with various pairwise directions are $LVP_{P,R,45^\circ}(X_c) = 10101011$,$LVP_{P,R,90^\circ}(X_c) = 11100001$, and $LVP_{P,R,135^\circ}(X_c) = 00101101$. The four binary pattern LVPs are concatenated to generate $LVP_{P,R}(X_c) = 011001001010101\ 11110000100101101$.

2.5. Local clustering pattern

Local clustering pattern (LCP) [14] is designed to solve the problems in face recognition: (1) to reduce feature length with low computational cost and (2) to enhance the accuracy for face recognition. To generate the local clustering pattern, four phases have to be considered: (1) to generate the local derivative variations with various directions; (2) to project the local derivative

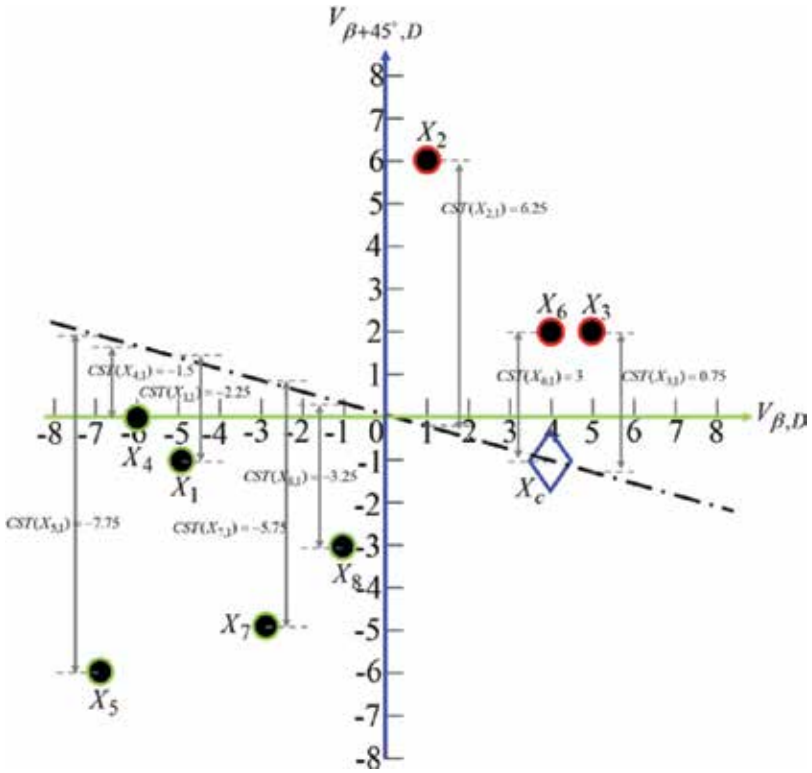


Figure 12. Comparative space transform (CST) in encoding first-order LVP in $\beta = 0^\circ$ direction.

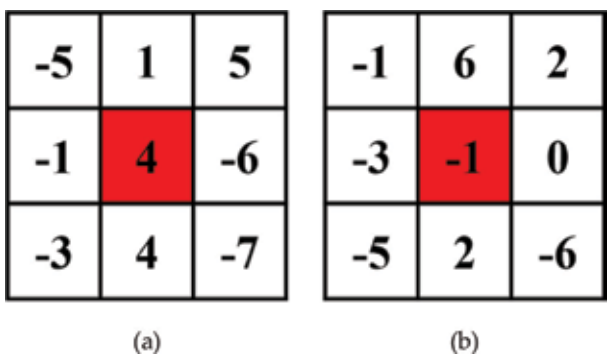


Figure 13. The first-order derivative direction value. (a) $\beta = 0^\circ$; (b) $\beta + 45^\circ = 45^\circ$.

variations with various directions on the pairwise combinatorial directions in the rectangular coordinate system; (3) to transform the coordinate from the rectangular coordinate system into the polar coordinate system; and (4) encoding the facial descriptor which is local clustering pattern, as a micropattern for each pixel by applying the clustering algorithm. The details are described in the following subsections: local clustering pattern (LCP) and coding scheme.

2.5.1. Local clustering pattern

Taken a subregion image $I(X)$ as an example, as shown in **Figure 1**, in which X_c is the referenced pixel and $X_p, p = 1, \cdots, 8$ are the adjacent pixels around X_c. LCP firstly generates the first-order derivatives of X_c, $I'_\alpha(X_c)$, in various directions and can be written as

$$I'_\alpha(X_c) = I_\alpha(X_p) - I_\alpha(X_c) \tag{34}$$

where α is the derivative direction including 0°, 45°, 90°, and 135° directions. Then, the LCP is generated by integrating the pairwise combinatorial directions of the derivative variations, $0^\circ - 45^\circ$, $45^\circ - 90^\circ$, $90^\circ - 135^\circ$ and $135^\circ - 0^\circ$, in polar coordinate system. The generation of LCP in pairwise combinatorial direction can be expressed as,

$$LCP_\alpha(X_c) = \sum_{n=1}^{N} f_{r,\theta}\left(I'_{\gamma,D}(X_p), I'_{\gamma,D}(X_c)\right) \times 2^{n-1}\big|_{\gamma \in \{\alpha, \alpha+45^\circ\}, N=8} \tag{35}$$

where $f_{r,\theta}(\cdot,\cdot)$ is the coding scheme and $D = 1, 2, 3$ is the distance between referenced pixel X_c and its adjacent pixels X_p, as shown in **Figure 10**. The coding scheme is executed in the polar coordinate system, and the formula can be formally defined as follows,

$$f_{r,\theta}\left(I'_{\gamma,D}(X_p), I'_{\gamma,D}(X_c)\right)\Big|_{\gamma \in \{\alpha, \alpha+45^\circ\}} = \begin{cases} 0, & \text{if } I'_{\gamma,D}(X_p) \text{ and } I'_{\gamma,D}(X_c) \in C_i \\ 1, & \text{else} \end{cases} \tag{36}$$

where C_i is the cluster center. Finally, the LCP at referenced pixel X_c, $LCP(X_c)$, is combinatorial of the four 8-bit binary patterns LCPs, and can be formally as

$$LCP(X_c) = \left\{LCP_\beta(X_c)\right\}\big|_{\beta = 0^\circ, 45^\circ, 90^\circ, 135^\circ}. \tag{37}$$

2.5.2. Coding scheme

In this subsection, we further discuss the coding scheme in LCP which is considered as the problem of classification. The coding scheme of LCP is executed in the polar coordinate system based on the characteristics of the derivative variations in the pairwise combinatorial directions.

First, four combinations of the derivative variations in the pairwise directions are utilized in LCP, including $0^\circ - 45^\circ$, $45^\circ - 90^\circ$, $90^\circ - 135^\circ$, and $135^\circ - 0^\circ$. The coordinate of the pairwise combinatorial directions of the derivative variations is in the rectangular coordinate system (RCS). To consider the magnitude and orientation between pairwise combinatorial directions, the coordinate is transformed from the rectangular coordinate system (RCS) into the polar coordinate system (PCS) by calculating the magnitude (m) and orientation (θ) for each pair directions of derivative variations. The magnitude (m) and orientation (θ) of X_p are calculated as

$$m_\gamma(X_p) = \sqrt{\left(I'_{\gamma,D}(X_p)\right)^2 + \left(I'_{\gamma+45^\circ,D}(X_p)\right)^2}\big|_{\gamma \in \alpha} \tag{38}$$

$$\theta_\gamma(X_p) = \arctan \frac{I'_{\gamma+45^\circ, D}(X_p)}{I'_{\gamma, D}(X_p)} \Big|_{\gamma \in \alpha} \tag{39}$$

where $\frac{-\pi}{2} < \theta_\gamma < \frac{\pi}{2}$ is normalized to 0°~360°.

The feature vectors $\mathbf{v}$ are m_γ and θ_γ coordinate in the polar coordinate system and can be written as

$$\mathbf{v} = \left[m_\gamma(X_n), \theta_\gamma(X_n)\right]^T \tag{40}$$

where $\gamma \in \alpha$ and $n = 1$~9 are the pixels in the subregion image $I(X)$ including the referenced pixels and its adjacent pixel in the polar coordinate system.

LCP is ensemble of several decisions from the results of clustering. Each clustering result is considered as a problem of a two-class case, whose center vector $\mathbf{C}$ is written as

$$\mathbf{C} = [C_1, C_2]^T \tag{41}$$

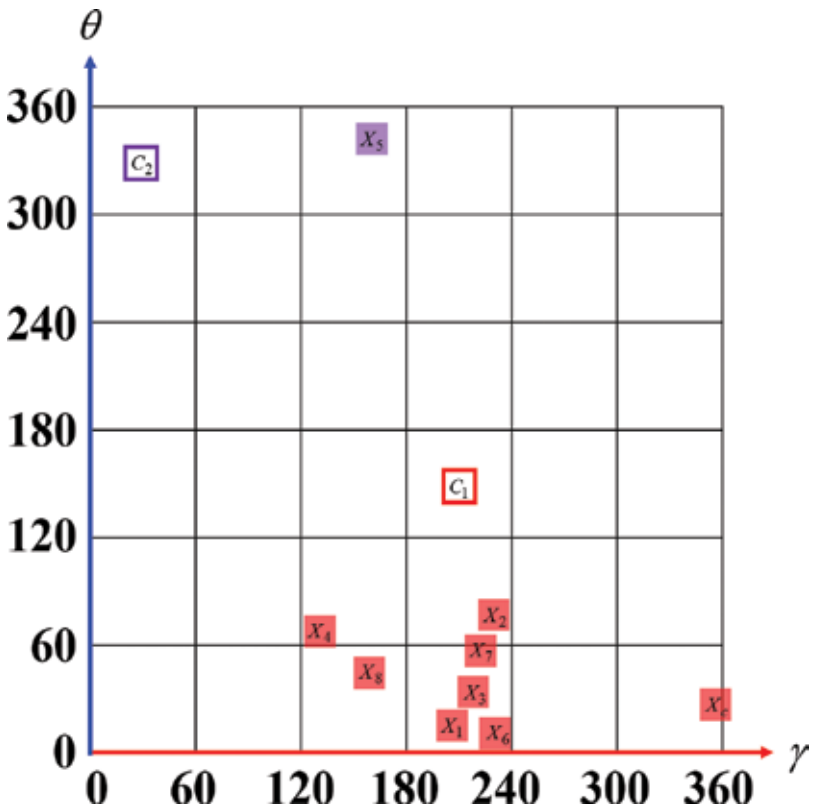


Figure 14. Example of the LCP takes **Figure 5(a)** as an example (the derivative variations along 0° and 45°).

where C_1 and C_2 are the two-class centers, in which C_1 is also the center of X_c. To classify the feature vectors **v** in sub-image I, we randomly initialize two-class centers **C** and adopt the k-means clustering algorithm for classification. The clustering procedure is repeated T times to find the cluster two-class centers $\mathbf{C_i}$ that have the highest probability $P(\mathbf{C_i}|\mathbf{v})$.

The adjacent pixels of the reference pixel X_c are encoded as the following equation,

$$C\left(m_\gamma(X_p), \theta_\gamma(X_p)\right)\Big|_{\gamma \in \alpha} = \begin{cases} 0, & if\ X_p \in C_1 \\ 1, & else \end{cases} \tag{42}$$

where C_1 is the cluster center which includes X_c.

2.5.3. Example

The local subregion of an image as shown in **Figure 5(a)** is taken as an example to illustrate the encoding process of generating first-order LCP, as shown in **Figure 14**. First, LCP calculates the first-order derivatives along 0° and 45° directions as shown in **Figure 13**. Then, the coordinates of referenced pixel X_c and its neighborhoods are translated from rectangular coordinate system (RCS) into polar coordinate system (PCS). The results of coordinate translation are shown in **Figure 14**. After that, the clustering technique is applied to find the centers of two clusters, as indicated as the hollow rectangles with red and purple colors, respectively. Only X_5 belongs to the second class, the rest pixels belong to the first class. Then, the corresponding bit of the 8-bit binary codes of $LCP_{P,R,0^\circ}(X_c) = 00001000$.

3. Comparison

In this section, we discuss the characteristics of the local patterns descriptors as mentioned. The local binary pattern (LBP) generates the local facial descriptor by comparing the gray value between referenced pixel and its adjacent pixels for each pixel in the face image. The texture information, such as spots, lines and corners, in the images is extracted. Although LBP considers the spatial information to generate the local facial descriptor, it omits the directional information and is sensitivity when light is slightly changed.

The local derivation pattern (LDP) analyzes the turnings between referenced pixel and its neighborhoods from the derivative values. The derivative values with four directions are considered to generate the local facial descriptor in the high-order derivative space. However, the turnings between referenced pixel and its neighbors are discussed in the same derivative direction.

The local tetra pattern (LTrP) utilized the two-dimensional distribution with derivative values in four quadrants to describe the texture information and that can extract more discriminative information. Although LTrP considers the derivative variations with two dimensions, there exist two problems: (1) the dimension of facial descriptor and (2) the sensitivity of the features. To compare with LBP and LDP, the dimension of facial descriptor of LTrP is high. The features

of LTrP in the four quadrants of the rectangular (or Cartesian) coordinate system are altered when illumination is changed.

The local vector pattern (LVP) designs the comparative space transform (CST) and that is associated with the pairwise directions of vector to encode the micropatterns. Comparing LVP with LBP, LDP, and LTrP, LVP not only successfully extracts distinctive information but also reduces the feature length. However, its computational cost is higher than LBP and LDP.

The local clustering pattern (LCP) derivatives the local variations with multidirections and that are integrated to form the pairwise combinatorial direction. To generate the discriminative local pattern, the features of local derivative variations are transformed into the polar coordinate system by generating the characteristics of magnitude (m) and orientation (θ). LCP generates the discriminative local clustering pattern with low-order derivative space and low computational cost which are stable in the process of face recognition. The summarization of each method is demonstrated in **Table 1**.

In **Table 1**, we analyze these methods with three indicators: (1) information used, (2) distribution of coding scheme, and (3) feature length. The indicator of the information used presents the information which is used in facial descriptor generation. LBP uses the original values such, as gray value; LDP considers the single high-order derivative values; LTrP uses both horizontal and vertical high-order derivative values; LVP uses the high-order derivative values and be described as the vector representation; the high-order derivative values are utilized in clustering process of LCP.

The distribution of coding scheme is to present how many directions of used information are considered in coding at each time. LBP and LDP generate the micropattern by considering a single direction at each time, for example, LDP generates the micropatterns of one direction at a time and then integrates the results of each direction to form the facial descriptor; LTrP considers two-direction information, horizontal and vertical, when coding; LVP and LCP use the pairwise combinatorial directions.

The feature length is to demonstrate the feature length of each micropattern. LBP considers eight neighborhoods and its feature length is 8; LDP further considers four directions including 0°, 45°, 90°, and 135°, its feature length is $8 \times 4 = 32$ bits, in which "8" is the number of neighborhood of referenced pixel and "4" is the number of derivative directions; the feature length of LTrP is $8 \times 13 = 8 \times (3 \times 4 + 1) = 104$ bits, where "8" is the number of neighborhood

Methods	Information used	Distribution of coding scheme	Feature Length
LBP	Original values	One dimensional	8
LDP	High-order derivative values	One dimensional	8×4
LTrP	High-order derivative values	Two dimensional	8×13
LVP	High-order derivative values	Two dimensional	8×4
LCP	High-order derivative values	Two dimensional	8×4

Table 1. Comparison of various methods.

of referenced pixel, "3" is the number of the binary patterns in a tetra pattern, "4" is the number of the tetra patterns, and "1" number of the binary pattern which is obtained from the magnitude; the feature length of LVP and LCP is $8 \times 4 = 32$ bits, where "8" is the number of neighborhood of referenced pixel, and "4" is the number of pairwise combinatorial directions.

4. Summary

The principal object of this chapter is to present the local pattern descriptors for understanding and accessing the facial descriptor in face recognition. The concept of local pattern is sample and intuitive, and the extended techniques of the basic local pattern are widely used in various areas. A partial listing of local pattern descriptors includes local binary pattern (LBP), local derivative pattern (LDP), local tetra patterns (LTrP), local vector pattern (LVP) and local clustering pattern (LCP) are widely applied to variety of image processing problems such as object detection, object recognition, image retrieval, fingerprint recognition, character recognition, face recognition, license plate recognition. Since it is impractical to cover all the approaches of local pattern descriptor in a single chapter, the basic and popular techniques included are chosen for their value in introducing and clarifying fundamental concepts in the field.

Author details

Chih-Wei Lin

Address all correspondence to: cwlin@fafu.edu.cn

College of Computer and Information Sciences, Fujian Agriculture and Forestry University, Fuzhou, China

References

[1] Moghaddam B, Pentland A. Probabilistic visual learning for object representation. IEEE Transactions on Pattern Analysis and Machine Intelligence. 1997;**19**(7):696-710. DOI: 10.1109/34.598227

[2] Turk M, Pentland A. Eigenfaces for recognition. Journal of Cognitive Neuroscience. 1991; **3**(1):71-86. DOI: 10.1162/jocn.1991.3.1.71

[3] Belhumeur PN, Hespanha JP, Kriegman DJ. Eigenfaces vs. fisherfaces: Recognition using class specific linear projection. IEEE Transactions on Pattern Analysis and Machine Intelligence. 1997;**19**(7):711-720. DOI: 10.1109/34.598228

[4] Swets DL, Weng JJ. Using discriminant eigenfeatures for image retrieval. IEEE Transactions on Pattern Analysis and Machine Intelligence. 1996;**18**(8):831-836. DOI: 10.1109/34.531802

[5] Ojala T, Pietikäinen M, Harwood D. A comparative study of texture measures with classification based on featured distributions. Pattern recognition. 1996;**29**(1):51-59. DOI: 10.1016/0031-3203(95)00067-4

[6] Nanni L, Alessandra L. Local binary patterns for a hybrid fingerprint matcher. Pattern recognition. 2008;**41**(11):3461-3466. DOI: 10.1016/j.patcog.2008.05.013

[7] Ahonen T, Abdenour H, Matti P. Face recognition with local binary patterns. In: Proceeding of the European conference on computer vision; Heidelberg. Berlin: Springer; 2004; pp. 469-481. DOI: 10.1007/978-3-540-24670-1_36

[8] Liu L, Zhang H, Feng A, Wan X, Guo J. Simplified Local Binary Pattern Descriptor for Character Recognition of Vehicle License Plate. In: Proceeding of the Seventh IEEE International Conference on Computer Graphics, Imaging and Visualization (CGIV). 2010; 157-161. DOI: 10.1109/CGIV.2010.32

[9] Ojala T, Pietikainen M, Maenpaa T. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. IEEE Transactions on Pattern Analysis and Machine Intelligence. 2002;**24**(7):971-987. DOI: 10.1109/TPAMI.2002.1017623

[10] Ahonen T, Hadid A, Pietikainen M. Face description with local binary patterns: Application to face recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence. 2006;**28**(12):2037-2041. DOI: 10.1109/TPAMI.2006.244

[11] Zhang B, Gao Y, Zhao S, Liu J. Local derivative pattern versus local binary pattern: Face recognition with high-order local pattern descriptor. IEEE Transactions on Image Processing. 2010;**19**(2):533-544. DOI: 10.1109/TIP.2009.2035882

[12] Murala S, Maheshwari RP, Balasubramanian R. Local tetra patterns: A new feature descriptor for content-based image retrieval. IEEE Transactions on Image Processing. 2012;**21**(5):2874-2886. DOI: 10.1109/TIP.2012.2188809

[13] Fan KC, Hung TY. A novel local pattern descriptor-local vector pattern in high-order derivative space for face recognition. IEEE transactions on image processing. 2014;**23**(7): 2877-2891. DOI: 10.1109/TIP.2014.2321495

[14] Lin CW, Lu KY. Local Clustering Patterns in Polar Coordinate for Face Recognition. Pacific-Rim Symposium on Image and Video Technology. Cham: Springer; 2015; pp. 656-666. DOI: 10.1007/978-3-319-29451-3_52

Chapter 6

Face Recognition Based on Texture Descriptors

Jesus Olivares-Mercado, Karina Toscano-Medina,
Gabriel Sanchez-Perez, Mariko Nakano Miyatake,
Hector Perez-Meana and Luis Carlos Castro-Madrid

Additional information is available at the end of the chapter

http://dx.doi.org/10.5772/intechopen.76722

Abstract

In this chapter, the performance of different texture descriptor algorithms used in face feature extraction tasks are analyzed. These commonly used algorithms to extract texture characteristics from images, with quite good results in this task, are also expected to provide fairly good results when used to characterize the face in an image. To perform the testing task, an AR face database, which is a standard database that contains images of 120 people, was used, including 70 images with different facial expressions and 30 with sunglasses, and all of them with different illumination intensity. To train the recognition system from one to seven images were used for each person. Different classifiers like Euclidean distance, cosine distance, and support vector machine (SVM) were also used, and the results obtained were higher than 98% for classification, achieving a good performance in verification task. This chapter was also compared with other schemes, showing the effectiveness of all of them.

Keywords: face recognition, face verification, texture, LBP, WBP, SVM

1. Introduction

Nowadays, face recognition is a non-intrusive biometric method in which the data acquisition is easy and can be carried out with or without the cooperation of the person under analysis. The face can be considered as the easiest way to recognize a person, increasing the acceptance of this kind of systems and their applications [1–3]. These systems consist of two tasks: identity verification, where the system verifies if the identity of the person is that which he/she claims it to be, and identification task, where the system determines the identity of the person among all the people in a database. Thus, the recognition task covers both tasks–identification and verification [4, 5].

Several problems must be considered in the development of a face recognition system such as illumination changes, facial expressions, and partial occlusions. It is because these kinds of changes can harm the accuracy of a face recognition system [6]. The changes in lighting conditions have received significant attention [6]. Because of that, a lot of systems were proposed in the last years, trying to reduce these problems [6]. Some systems proposed to this end are based on image processing techniques such as histogram equalization [6, 7] and contrast-limited adaptive histogram equalization (CLAHE) [8]. Another way to solve the problem of illumination changes is the development of different high-performance methods to solve these kinds of changes such as the eigenphases approach [7–11]. Also, are useful some methods based on frequency transforms like discrete cosine transform [12–14], discrete Gabor transform [15–17], discrete wavelet transforms [18–21], and discrete Haar transform [22]. Additional methods that could be applied are the eigenfaces [23, 24] which use the principal component analysis (PCA) [25, 26], modular PCA-based face recognition methods [27], the Fisherfaces approach [28], and the Laplacianfaces [29].

The local binary pattern (LBP) operator [30] has recently been proposed in several applications. The principal advantages of this algorithm are that it has a good computational performance and presents a good support when the images have gray-level changes. Because of that LBP can be applied for image characterization in several pattern recognition tasks [31]. This algorithm can be used for face characterization because the face images have a lot of little patterns which can be characterized using the LBP [30]. Several LBP variations have been proposed such as: the holistic LBP histogram (hLBPH) [30], the spatially enhanced LBP histogram (eLBPH) [32], holistic LBP Image algorithm (hLBPI) [32] and decimated image window binary pattern (WBP) [33]. All of these algorithms are based on the original LBP algorithm, but the computational complexity of the hLBPI and WBP are lower than the others providing also a good performance as shown in this chapter.

In recent years, the interest in the face recognition schemes has increased because of its potential implementation in mobile devices, which generally have a limited computational power. Hence, this chapter presents a comparison of the texture descriptors like hLBPI and WBP. Finally, some classification methods, like SVM, Euclidean distance, and cosine distance, are used to perform the recognition. In this chapter, these algorithms were evaluated with different illumination and facial expression changes.

The remainder of this chapter is organized as follows: Section 2 presents the description of the evaluated system. Section 3 presents the evaluation results. Finally, Section 4 provides the conclusions of this research.

2. Evaluated system

Figure 1 shows the block diagram of the evaluated face recognition system, where, firstly, the system receives the face image under analysis. It is then fed into an interpolation stage (Any other method can be applied.). Next, the texture descriptor algorithm is applied to characterize the image. Finally, the feature matrix is fed into the classification stage.

2.1. Texture descriptors

In this chapter were used two texture descriptors, the hLBPI and the WBP. The hLBPI algorithm, introduced by Ojala et al. [34], is based on the original LBP method. This algorithm uses masks of 3×3 pixels. There is a neighborhood, as is shown in **Figure 2a**, where all neighbors are compared with the central pixel where each of these pixels are labeled with a 0 if their values are smaller than the central pixel; otherwise, they are labeled as 1 (**Figure 2b**). Next, the label of each pixel is multiplied by 2^p, where p is the position of each pixel in the neighborhood from 0 to 7 (**Figure 2c**). Finally, all values are added to get the label that will be positioned in the place of the central pixel as shown in **Figure 2d**. This algorithm obtains 128 different values for the central pixel. These steps apply to an image to obtain a LBP matrix.

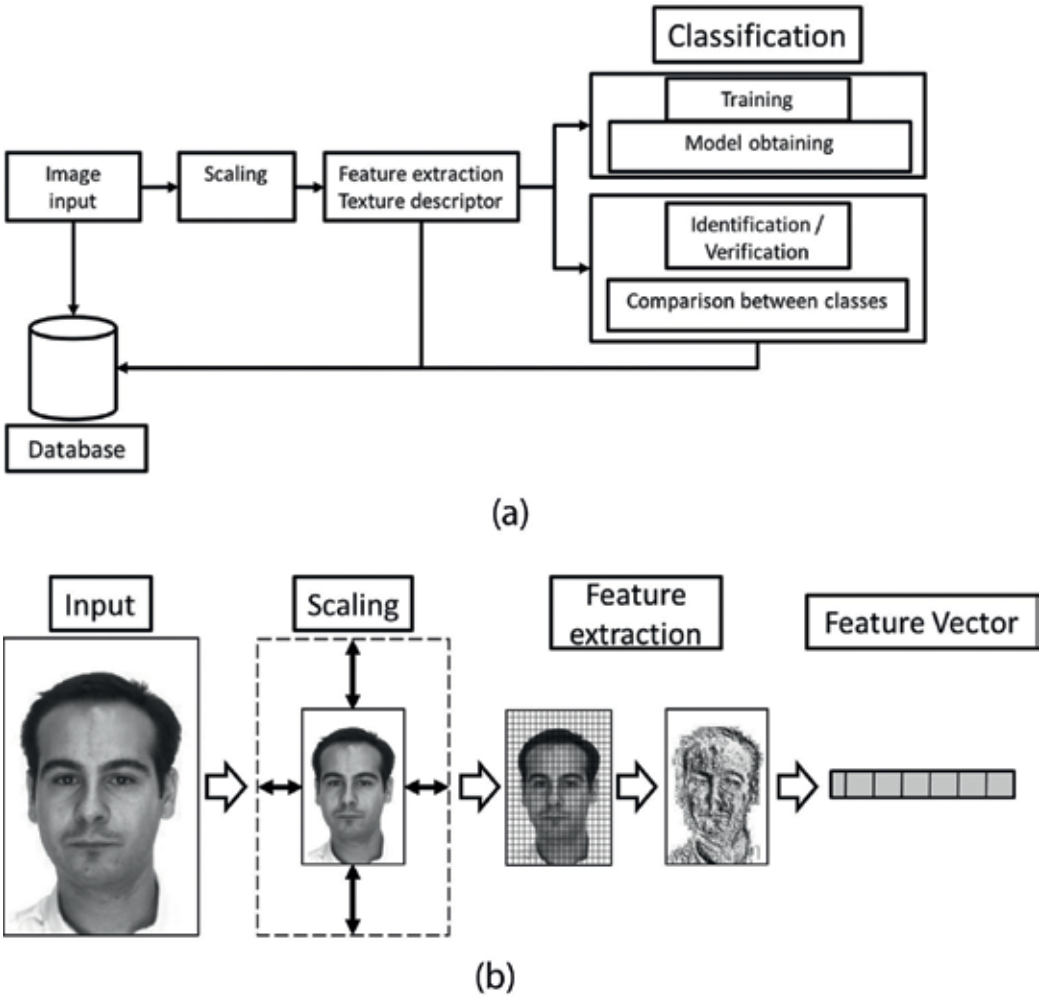


Figure 1. (a) Block diagram of the evaluated face recognition scheme, (b) illustration of the evaluated face recognition scheme.

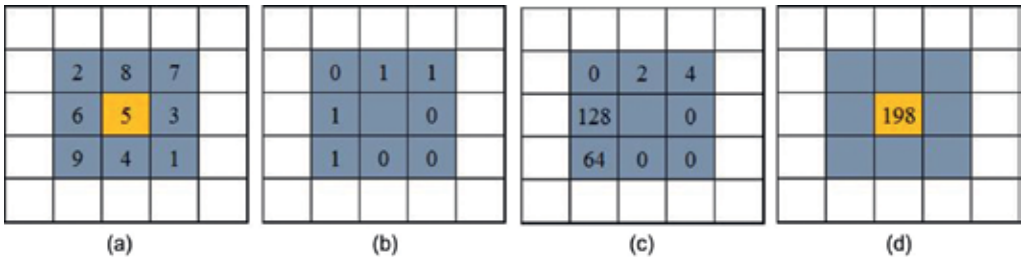


Figure 2. Example of implementation of LBP in a neighborhood of 3 × 3 pixels.

After obtaining the LBP matrix, can estimate an L-dimensional feature vector, where L is the total number of training images of $N \times M$ pixels, requires $8NM$ additions and $8NM$ comparisons for LBP estimation. The LBP matrix is then arranged in a column vector, X, of size NM which is multiplied by the matrix Φ, of size $L \times NM$ obtained from the PCA analysis. Thus, the estimation of the feature vector, given by $Y = \Phi X$, requires NML additions and NML multiplications. Then, the feature vector estimation requires $(24 + L)NM$ additions, $(16 + L)NM$ multiplications and $8NM$ comparisons. Thus, assuming that all operations represent the same computational cost, the hLBPI algorithm requires approximately $(48 + 2L)NM$ operations. On the other hand, the WBP is an algorithm that reduces the computational complexity of the original hLBPI without a big loss of accuracy in the recognition. Firstly, the image is reduced with bicubic interpolation using a factor of 9. Then the image is divided into lxm non-overlapping windows of 3 × 3 pixels such that the input size of the original image $(M \times N)$ could be represented as $3l \times 3m$. Then, the WBP is defined as follows:

$$WBP(j,k) = \sum_{p=0}^{P-1} s\left(I_{j,k}(x,y) - g_c\right)2^p \tag{1}$$

where $x = 3j + r$, $y = 3k + q$, $r = 1, 2, \ldots, 8$, $q = 1, 2, \ldots, 8$, $j = 1, 2, \ldots, M/3$, $k = 1, 2, \ldots, N/3$, $I_{j,k}(x,y)$; represents the (j,k)-th block of 3 × 3 pixels of the down sampled input image, g_c is the central pixel of the same block and $s\left(I_{j,k}(x,y) - g_c\right) = 0$ if $I_{j,k}(x,y) < g_c$ and 1 otherwise. Finally, the label of each pixel is multiplied by 2^p, where p is the position of each pixel in the neighborhood from 0 to 7. Next the feature matrix obtained from Eq. (1) is rearranged in a vector of size $MN/9$ which is fed into the classification stage. The main advantage of this algorithm is that the face image can be characterized by a small non-overlapping window of 3 × 3-pixel instead of the overlapped windows used by the original hLBPI.

A WBP example is shown in **Figure 3**. First, the original image is divided into windows of 3 × 3 pixels (**Figure 3a**). **Figure 3b** shows the result of the comparison of neighboring pixels and **Figure 3c** shows the result of its respective conversion to decimal values. The matrix resulting from the sum of the decimal values (WBP image) is shown in **Figure 3d**. The size is reduced to 1/9 of the original image. This matrix is then introduced in the classifier for training or recognition. The computational complexity of this algorithm includes the estimation of the LBP coefficients using non-overlapping blocks of 3 × 3 pixels, which require $8NM/81$ additions and $8NM/81$ comparisons. Thus, assuming that the three operations have similar complexity, the proposed algorithm requires $16NM/81$ operations.

A comparison of the computational complexity of other recently proposed methods for feature vector estimation during the testing operation is shown in **Figure 4**.

2.2. Classification stage

After obtaining the feature vectors, the next step is the classification stage, which involves performing the identification or verification task. In this chapter, in the training stage, the K-means was used to obtain a template by averaging the training images of each class. During the identification task, three classifiers are used. These are SVM, the Euclidean distance, and cosine distance, classifying the image under analysis as belonging to the N-class with the smaller distance or the largest probability in the SVM case. During the verification task, the system validates the identity of the person under analysis with a given threshold, and these

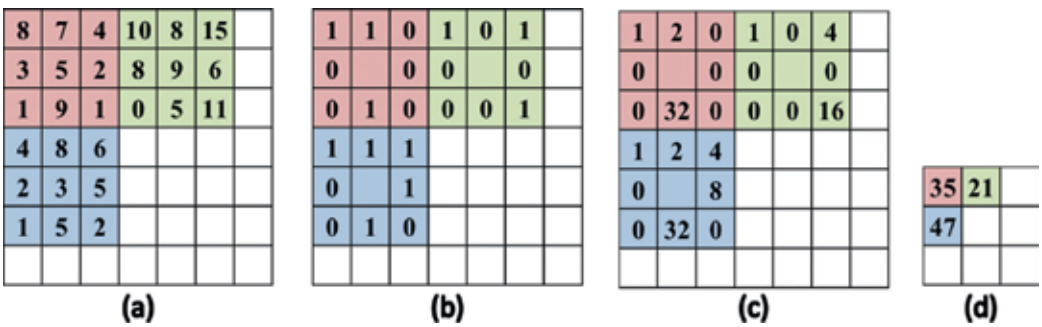


Figure 3. WBP example.

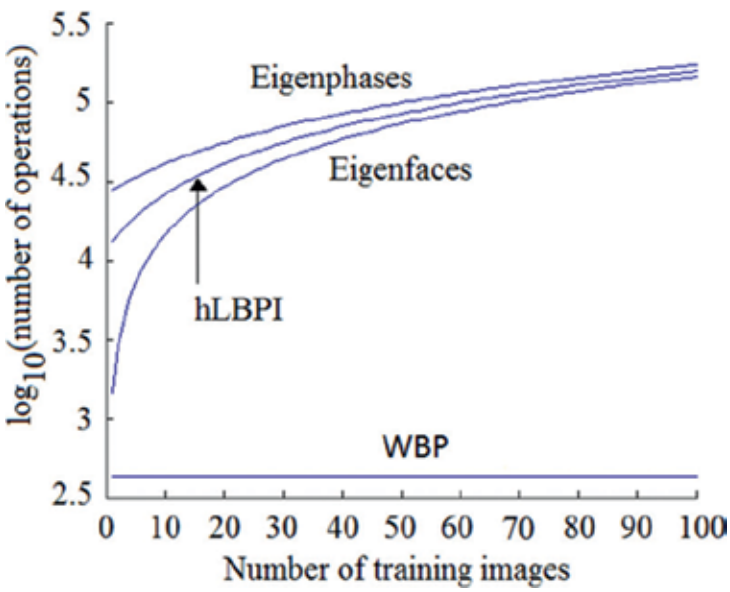


Figure 4. Computational complexity of different algorithms.

results are only obtained by the SVM. In this chapter, two different distances were evaluated, the Euclidean distance given by:

$$d_{st} = (x_s - y_t)(x_s - y_t)^T, \tag{2}$$

and the cosine distance is given by.

$$d_{st} = 1 - \frac{x_s y_t^T}{(x_s x_s^T)(y_t y_t^T)}, \tag{3}$$

where x_s is the estimated feature vector of the image under analysis and y_t is the center of the t-th class.

3. Evaluation results

In this chapter, the system was evaluated using the identification and verification tasks, wherein identification of the system is required to determine the identity of the person under analysis by comparing the facial characteristics stored in a database with the face characteristics extracted and the verification task, where the system must define if the identity corresponds with the person she/he claims to be [4]. In both tests, the results are compared with another algorithm like the eigenphase [11], Laplacianfaces [29], Fisherfaces, and eigenfaces [28], all of them used in the classification task with the SVM and k-means with Euclidean and cosine distance, the AR Face Database [35] was used for all tests.

The AR database was expanded with four additional images for each one in the original AR database. These images are shown in **Figure 5**, where **Figure 5a** is the original image and **Figure 5b–e** show the resulting images of the illumination variations.

After the expansion, the AR database has 12,000 face images in total, each person has 100 images and the database has 120 persons, where 65 are males and 55 females. The database was divided into two sets, the AR(A) which has 70 images with illumination and expression

Figure 5. Effects of illumination transformation applied to form the extended AR database.

changes and the AR(B) which has 30 images with partial occlusion using sunglasses and also illumination changes. **Figure 6** shows some examples of these images.

In real-world applications, the number of training images and recognition accuracy are strongly related; as more images are used for training, the recognition accuracy improves. However, in real applications the number of training images is limited. **Figure 7** shows the performance of hLBPI (**Figure 7a**) and WBP (**Figure 7b**) with different number of training images using three different classifiers.

3.1. Identification

Figure 8 shows the recognition performance of the texture descriptors hLBPI and WBP compared with the other classical methods, all of them using the set AR(A) and seven training images for each person.

An important evaluation to obtain also is the ranking of identification, where the ranking (n) denotes the probability that an image belongs to one of n classes with highest probability. That is, a ranking of 10 is the probability that the image belongs to one of the 10 most likely persons. **Figures 9** and **10** present the ranking evaluation with the set AR(A) and set AR(B).

In all cases, the training was done using seven images per person belonging to either the AR (A), while the recognition system was tested with images that were not used for training from the AR(A) and AR(B) sets respectively.

3.2. Verification

In the case of verification, the percentage of error is divided into two: false acceptance and false rejection. False acceptance occurs when an individual claims to be the person who is not and

Figure 6. Examples of (a) images from the AR(A) set. (b) Images from the AR(B) set.

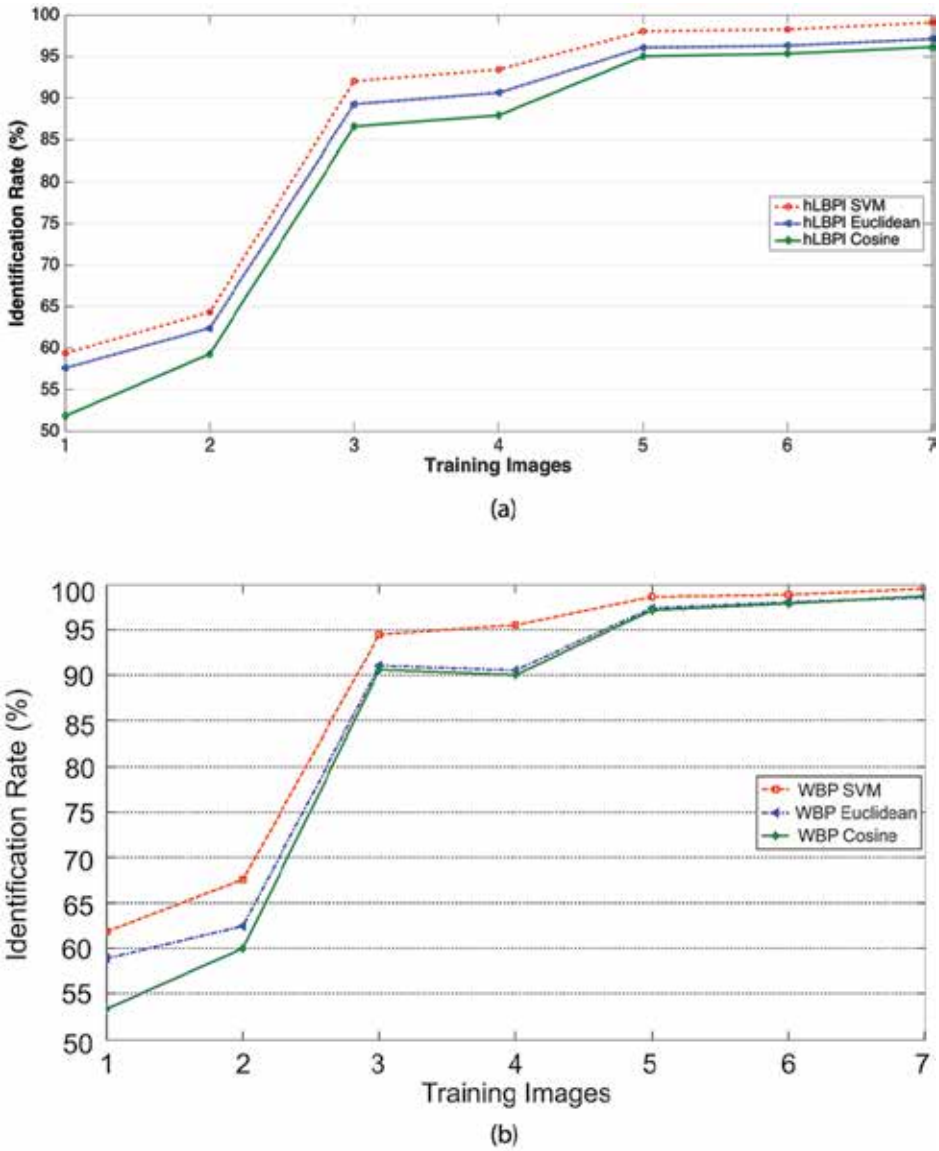


Figure 7. The identification rate obtained using different numbers of training images.

this is mistakenly accepted by the system. False rejection occurs when an individual provides their identity and the system erroneously rejects this statement.

Figure 11 shows the receiver operating characteristics (ROC) when the hLBPI and WBP algorithms are used for the verification task using the set AR(A). **Figure 11** shows that both algorithms have a similar performance with a false rejection and false acceptance, and they are very low.

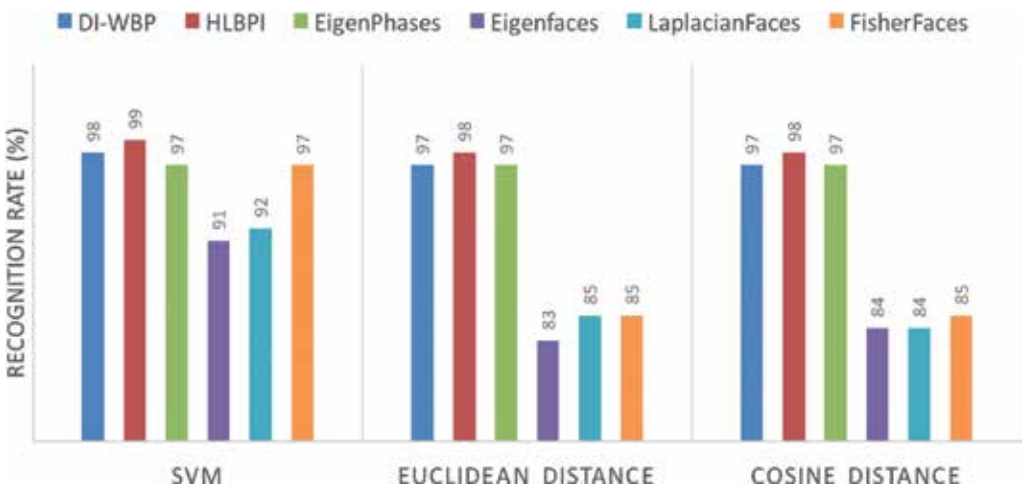


Figure 8. Recognition performance of the evaluating approach using: SVM, Euclidean distance, and cosine distance in the classification stage. The performance of hLBPI, eigenphases, eigenfaces, Laplacianfaces, and Fisherfaces are also shown for comparison.

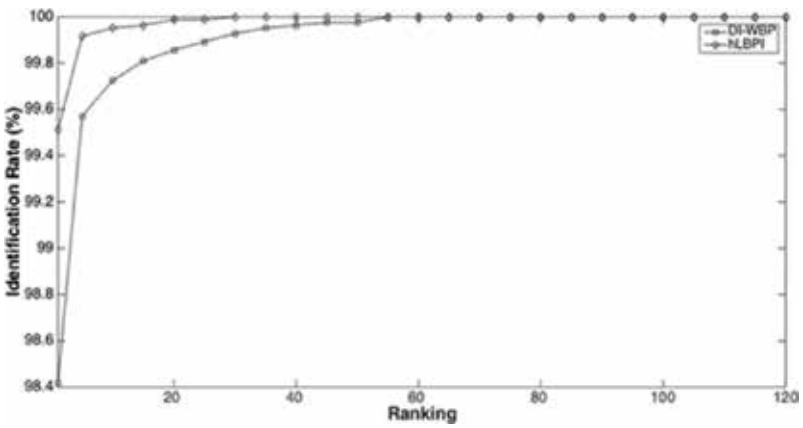


Figure 9. Ranking evaluation of image set AR(A).

The performance of most face verification systems depends on a suitable selection of the threshold value to diminish both the false acceptance and the false rejection. After many tests, it was concluded that this relationship, this threshold, is used to decide if the person is who she/he claims to be; next, it is explained, how the threshold can be obtained with an exponential function as shown in **Figure 12**, where you can select the point that has the lowest value of both false acceptance and false rejection. Thus, we can assume that the false acceptance probability is given by:

$$P_{fa}(Th) = e^{-\alpha Th} \tag{4}$$

and

$$P_{fr} = e^{-\beta P_{fa}} \tag{5}$$

Then from Eqs. (4) and (5) it follows that $\alpha = -Ln\big(P_{fa}(Th)\big)/Th$ where $P_{fa}(Th)$ is the false acceptance percentage, calculated from a threshold Th. Hence, for a desired false acceptance probability P_o the suitable value of Th is given.

$$Th = -Ln(P_o)/\alpha \tag{6}$$

Figure 13 shows the results of an exponential function and some experimental results of the relationship between the false acceptance rate and false rejection rate in this chapter.

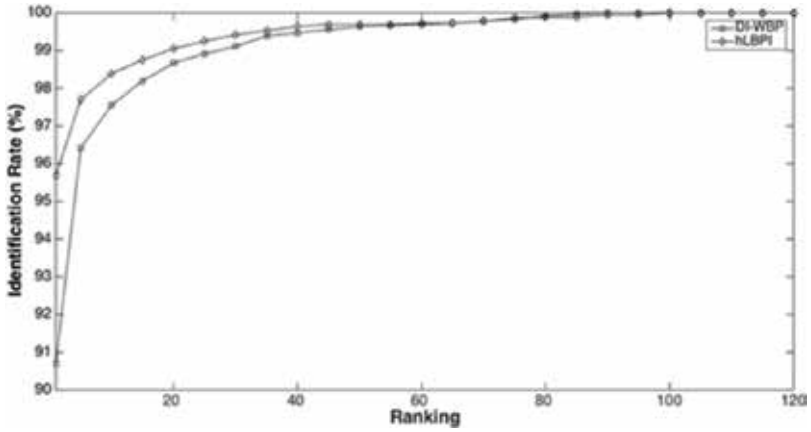


Figure 10. Ranking evaluation of image set AR(B).

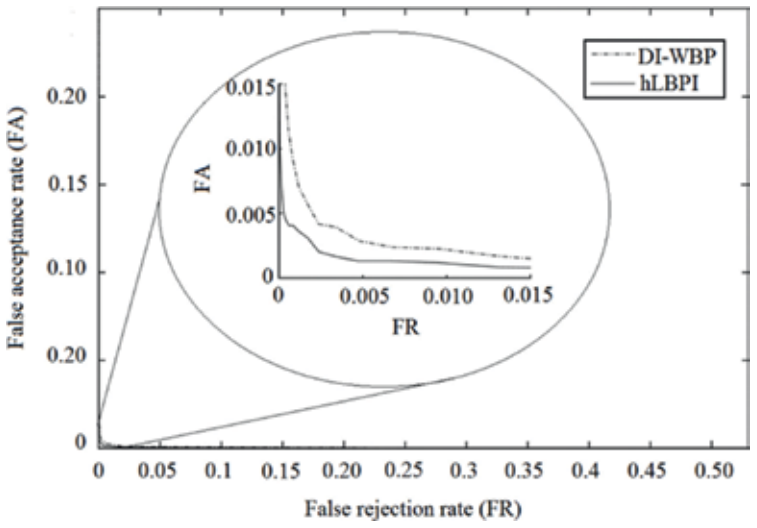


Figure 11. ROC of proposed and conventional hLBPI algorithms.

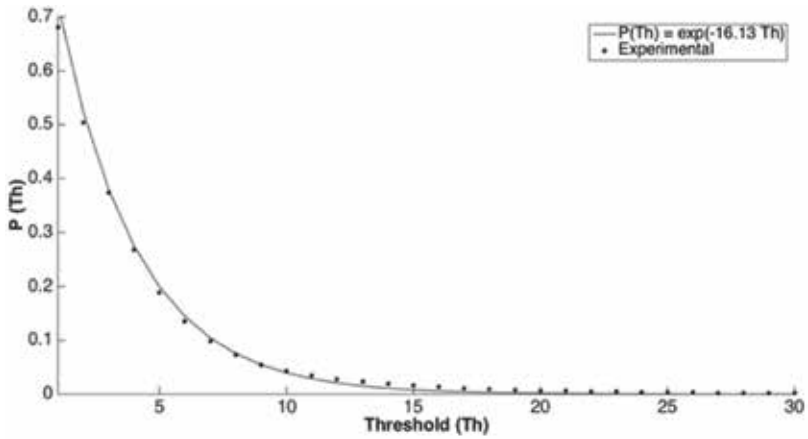


Figure 12. Relationship between the false acceptance rate and the threshold value.

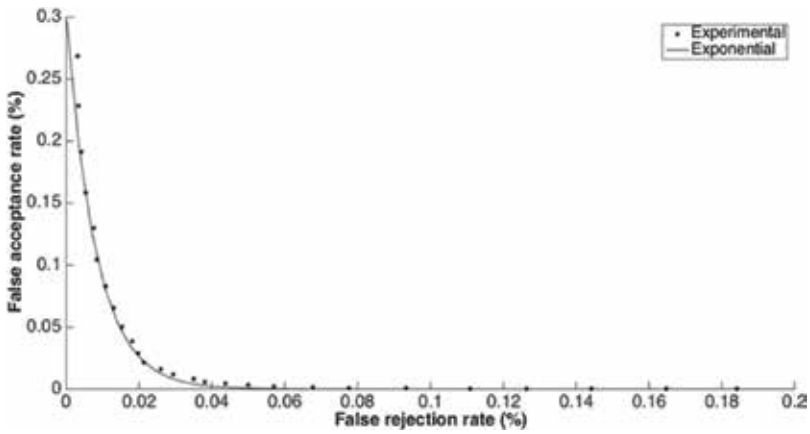


Figure 13. Relationship between the false acceptance rate and false rejection rate.

4. Conclusions

This chapter presented the application of different texture descriptors for tasks ranging from feature extraction to face recognition, which are based on the LBP algorithm, specifically the hLBPI and WBP. The evaluation results demonstrate that these algorithms provide good recognition rates. In most situations, the accuracy of recognition with the WBP is slightly lower than the accuracy with the hLBPI because the feature vector estimation of the WBP does not require PCA and uses non-overlapping blocks. This fact results in an important computational complexity reduction of approximately $2NML/9$ relative to hLBPI, where L is the feature vector size. If an L with a big value is used, it produces a more exact feature vector, although this causes the computational complexity to increase. Also, the evaluation results were compared with other

methods, such as the Eigenfaces, Laplacianfaces, and Fisherfaces. Another point that can be observed is that, as in all recognition systems, the accuracy percentage increases when a greater number of training images are used, so one could look for ways to generate training images based on an original image to increase the number of images, and then the system may be able to recognize face images in other types of environments, such as with a lot of lighting or with partial face occlusions. The results obtained with the set A shown in **Figure 8** where the system has a good performance both with images with facial expressions and with lighting, as well as when there are images with a partial occlusion of the face such as sunglasses as shown in **Figure 9**. In both cases, a recognition rate higher than 90% is obtained.

The evaluation results using the AR databases demonstrate that these algorithms provide good results also when it performs identity verification tasks, providing a theoretical criterion which allows selecting the threshold such that the system be able to provide a previously specified false acceptance or false rejection rate.

Acknowledgements

We thank the National Science and Technology Council of Mexico and to the Instituto Politecnico Nacional for the financial support during the realization of this chapter.

Author details

Jesus Olivares-Mercado*, Karina Toscano-Medina, Gabriel Sanchez-Perez,
Mariko Nakano Miyatake, Hector Perez-Meana and Luis Carlos Castro-Madrid

*Address all correspondence to: solidsnake98@gmail.com

Instituto Politecnico Nacional, CDMX, Mexico

References

[1] Kung SY, Mak M-W, Lin S-H. Biometric Authentication: A Machine Learning Approach. New York: Prentice Hall Professional Technical Reference; 2005

[2] El-Bakry HM, Mastorakis N. Personal identification through biometric technology. In: Proc. of the WSEAS International Conference on Applied Mathematics and Communications; 2009. pp. 325-340

[3] Li SZ, Jain AK. Handbook of Face Recognition. New York: Springer; 2011

[4] Chellappa R, Sinha P, Phillips PJ. Face recognition by computers and humans. Computer. 2010;**43**:46-55

[5] Gao Y, Leung MK. Face recognition using line edge map. IEEE Transactions on Pattern Analysis and Machine Intelligence. 2002;**24**:764-779

[6] Ruiz-del-Solar J, Quinteros J. Illumination compensation and normalization in eigenspace-based face recognition: A comparative study of different pre-processing approaches. Pattern Recognition Letters. 2008;**29**:1966-1979

[7] Ramirez-Gutierrez K, Cruz-Perez D, Olivares-Mercado J, Nakano-Miyatake M, Perez-Meana H. A face recognition algorithm using eigenphases and histogram equalization. International Journal of Computers. 2011;**5**:34-41

[8] Benitez-Garcia G, Olivares-Mercado J, Aguilar-Torres G, Sanchez-Perez G, Perez-Meana H. Face identification based on contrast limited adaptive histogram equalization (CLAHE). In: Proc. of International Conference on Image Processing, Computer Vision and Pattern Recognition; 2011

[9] Zaeri N. Eigenphases for corrupted images. In: Proc. of the International Conference on Advances in Computational Tools for Engineering Applications; 2009. pp. 537-540

[10] Olivares-Mercado J, Hotta K, Takahashi H, Nakano-Miyatake M, Toscano-Medina K. Improving the eigenphase method for face recognition. IEICE Electronics Express. 2009; **6**:1112-1117

[11] Benitez-Garcia G, Olivares-Mercado J, Sanchez-Perez G, Nakano-Miyatake M, Perez-Meana H. A sub-block-based eigenphases algorithm with optimum sub-block size. Knowledge-Based Systems. 2012;**37**:415-426

[12] Sharkas M. Application of DCT blocks with principal component analysis for face recognition; In: Proc. of the WSEAS International Conference on Signal, Speech and Image Processing; 2005. pp. 107-111

[13] Dabbaghchian S, Ghaemmaghami MP, Aghagolzadeh A. Feature extraction using discrete cosine transform and discrimination power analysis with a face recognition technology. Pattern Recognition. 2010;**43**:1431-1440

[14] Ajit Krisshna NL, Deepak VK, Manikantan K, Ramachandran S. Face recognition using transform domain feature extraction and PSO-based feature selection. Applied Soft Computing Journal. 2014;**22**:141-161

[15] Aguilar-Torres G, Toscano-Medina K, Sanchez-Perez G, Nakano-Miyatake M, Perez-Meana H. Eigenface-Gabor algorithm for feature extraction in face recognition. International Journal of Computers. 2009;**3**:20-30

[16] Owusu E, Zhan Y, Mao RQ. An SVM-AdaBoost facial expression recognition system. Applied Intelligence. 2014;**40**:536-454

[17] Qin H, Qin L, Xue L, Yu C. Gabor-based weighted region covariance matrix for face recognition. Electronics Letters. 2012;**48**:992-993

[18] Hu H. Variable lighting face recognition using discrete wavelet transform. Pattern Recognition Letters. 2011;**32**:1526-1534

[19] Dai D-Q, Yan H. Wavelets and face recognition. In: Delac K, Grgic M, editors. Face Recognition. Viena: I-Tech; 2007. pp. 59-74

[20] Eleyan A, Özkaramanli H, Demirel H. Complex wavelet transform-based face recognition. EURASIP Journal on Advances in Signal Processing. 2008;**2008**:195

[21] Delac K, Grgic M, Grgic S. Face recognition in JPEG and JPEG2000 compressed domain. Image and Vision Computing. 2009;**27**:1108-1120

[22] Gautam K, Quadri N, Pareek A, Choudhary S. A face recognition system based on back propagation neural network using Haar wavelet transform and morphology. Lecture Notes in Electrical Engineering. 2014;**298**:87-94

[23] Jirawatanakul J, Watanapa S. Thai face cartoon detection and recognition using eigenface model. Advances in Materials Research. 2014;**1341-1397**

[24] Hou YF, Pei W, Yan-Wen Chong Y, Chun-Hou ZC. Eigenface-based sparse representation for face recognition. In: Intelligent Computing Theories and Technology. Vol. 7096. Berlin Heidelberg: Springer; 2013. pp. 457-465

[25] Shlens J. A Tutorial on Principal Component Analysis. arXiv preprint arXiv:1404.1100, 2014

[26] Zhang YX. Artificial neural networks based on principal component analysis input selection for clinical pattern recognition analysis. Talanta. 2007;**73**:68-75

[27] Gottumukkal R, Asari VK. An improved face recognition technique based on modular PCA approach. Pattern Recognition Letters. 2004;**25**:429-436

[28] Belhumeur PN, Hespanha JP, Kriegman DJ. Eigenfaces vs. fisherfaces: Recognition using class specific linear projection. IEEE Transactions on Pattern Analysis and Machine Intelligence. 1997;**19**:711-720

[29] He X, Yan S, Hu Y, Niyogi P, Zhang H-J. Face recognition using laplacianfaces. IEEE Transactions on Pattern Analysis and Machine Intelligence. 2005;**27**:328-340

[30] Ahonen T, Hadid A, Pietikainen M. Face description with local binary patterns: Application to face recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence. 2006;**28**:2037-2041

[31] Xia W, Yin S, Ouyang P. A high precision feature based on LBP and Gabor theory for face recognition. Sensors. 2013;**13**:4499-4513

[32] Yang B, Chen S. A comparative study on local binary pattern (LBP) based face recognition: LBP histogram versus LBP image. Neurocomputing. 2013;**120**:365-379

[33] Benitez-Garcia G, Olivares-Mercado J, Toscano-Medina K, Sanchez-Perez G, Nakano-Miyatake M, Perez-Meana H. A low complexity face recognition scheme based on down

sampled local binary patterns. International Arab Journal of Information Technology, Accepted for publication. 2016

[34] Ojala T, Pietikainen M, Harwood D. Performance evaluation of texture measures with classification based on Kullback discrimination of distributions. In: Proc. of the IAPR Int. Conference on Computer Vision and Image Processing; Vol. 1. 1994. pp. 582-585

[35] Martinez AM. AR face database. CVC Technical Report 24, 1998

Section 4

Fuzzy Inference and Data Exploration

Chapter 7

Learning Algorithms for Fuzzy Inference Systems Using Vector Quantization

Hirofumi Miyajima, Noritaka Shigei and
Hiromi Miyajima

Additional information is available at the end of the chapter

http://dx.doi.org/10.5772/intechopen.79925

Abstract

Many studies on learning of fuzzy inference systems have been made. Specifically, it is known that learning methods using vector quantization (VQ) and steepest descent method (SDM) are superior to other methods. In their learning methods, VQ is used only in determination of the initial parameters for the antecedent part of fuzzy rules. In order to improve them, some methods determining the initial parameters for the consequent part by VQ are proposed. For example, learning method composed of three stages as VQ, generalized inverse matrix (GIM), and SDM was proposed in the previous paper. In this paper, we will propose improved methods for learning process of SDM for learning methods using VQ, GIM, and SDM and show that the methods are superior in the number of rules to the conventional methods in numerical simulations.

Keywords: fuzzy inference systems, vector quantization, neural gas, generalized inverse method

1. Introduction

There have been many studies on learning of fuzzy systems [1–8]. Their aim is to construct learning methods based on SDM. Some novel methods on them have been developed which (1) generate fuzzy rules one by one starting from any number of rules, or reduce fuzzy rules one by one starting from a sufficiently large number of rules [2]; (2) use genetic algorithm (GA) and particle swarm optimization (PSO) to determine fuzzy systems [3]; (3) use fuzzy inference systems composed of a small number of input rule modules, such as single input rule modules (SIRMs) and double input rule modules (DIRMs) methods [9, 10]; and (4) use a

self-organization or a vector quantization technique to determine the initial assignment of parameters [11–15, 19]. Specifically, it is known that learning methods using vector quantization (VQ) and steepest descent method (SDM) are superior in the number of rules (parameters) to other methods [16, 19]. So, why is it effective to combine VQ with SDM in fuzzy modeling? First, let us explain how to combine SDM with methods other than VQ. (1) Although the learning time is short, the generation method is known to have low test accuracy, while the reduction method has high test accuracy but takes long learning time [2]. (2) The method using GA and PSO shows high accuracy when the input dimension and the number of rules are small, but it is known that there is a problem of scalability [3]. (3) SIRM and DIRM methods are excellent in scalability, but the accuracy of learning is not always sufficient [9]. As described above, many methods are not necessarily effective models because of the difficulty of learning accompanying the increase of the input dimension and the number of rules and the low accuracy. On the other hand, the method combining VQ with SDM is possible to efficiently conduct learning of SDM by arranging suitably the initial parameters of fuzzy rules using VQ [1, 16]. However, since VQ is unsupervised learning, it is easy to reflect the input part of learning data, but how to capture output information in learning is difficult. With their studies, the first learning method is the one using VQ only in determining the initial parameters of the antecedent part of fuzzy rules using input part of learning data [1, 11–14]. The second method is the one determining the same parameter using input/output parts of learning data [15, 19]. Further, the third method is one iterating learning process of VQ and SDM for the second method. Kishida and Pedrycz proposed the method based on the third one [13, 15]. These methods are the ones determining only the antecedent parameters by VQ. Therefore, we introduced generalized inverse matrix (GIM) to determine the initial assignment of weight parameters for the consequent part of fuzzy rules as the fourth method and showed the effectiveness in the previous paper [16, 17]. In this paper, improved methods for learning process of SDM in learning methods using VQ, GIM, and SDM are introduced and show that the method is superior in the number of rules to other methods in numerical simulations.

2. Preliminaries

2.1. The conventional fuzzy inference model

The conventional fuzzy inference model using SDM is described [1]. Let $Z_j = \{1, \ldots, j\}$ and $Zj* = \{0, 1, \ldots, j\}$. Let R be the set of real numbers. Let $\boldsymbol{x} = (x_1, \ldots, x_m)$ and y be input and output variables, respectively, where $x_j \in R$ for $j \in Z_m$, and $y \in R$. Then, the rule of simplified fuzzy inference model is expressed as

$$R_i: \text{if } x_1 \text{ is } Mi_1 \text{ and } x_j \text{ is } M_{ij} \cdot \text{ and } x_m \text{ is } M_{im}, \text{ then } y \text{ is } w_i \tag{1}$$

where $j \in Z_m$ is a rule number, $i \in Z_n$ is a variable number, M_{ij} is a membership function of the antecedent part, and w_i is the weight of the consequent part.

http://dx.doi.org/10.5772/intechopen.79925

A membership value μ_i of the antecedent part for input $\boldsymbol{x}$ is expressed as

$$\mu_i = \prod_{j=1}^{m} M_{ij}(x_j) \tag{2}$$

Then, the output y^* of fuzzy inference method is obtained as

$$y^* = \frac{\sum_{i=1}^{n} \mu_i \cdot w_i}{\sum_{i=1}^{n} \mu_i} \tag{3}$$

If Gaussian membership function is used, then M_{ij} is expressed as

$$M_{ij}(x_j) = \exp\left(-\frac{1}{2}\left(\frac{x_j - c_{ij}}{b_{ij}}\right)^2\right) \tag{4}$$

where c_{ij} and b_{ij} denote the center and the width values of M_{ij}, respectively.

The objective function E is determined to evaluate the inference error between the desirable output y^r and the inference output y^*.

Let $D = \{(x^p, \ldots, x^p, y^r) | p \in Z_P\}$ and $D^* = \{(x^p, \ldots, x^p) | p \in Z_p\}$ be the set of learning data and the set of input part of D, respectively. The objective of learning is to minimize the following mean square error (MSE) as

$$E = \frac{1}{P}\sum_{p=1}^{P}\left(y_p^* - y_p^r\right)^2 \tag{5}$$

where $yp*$ and y^r mean inference and desired output for the pth input $\boldsymbol{x}^p$.

In order to minimize the objective function E, each parameter of $\boldsymbol{c}$, $\boldsymbol{b}$, and $\boldsymbol{w}$ is updated based on SDM using the following relation:

$$\frac{\partial E}{\partial w_i} = \frac{\mu_i}{\sum_{I=1}^{n} \mu_I} \cdot (y^* - y^r) \tag{6}$$

$$\frac{\partial E}{\partial c_{ij}} = \frac{\mu_i}{\sum_{I=1}^{n} \mu_I} \cdot (y^* - y^r) \cdot (w_i - y^*) \cdot \frac{x_j - c_{ij}}{b_{ij}^2} \tag{7}$$

$$\frac{\partial E}{\partial c_{ij}} = \frac{\mu_i}{\sum_{I=1}^{n} \mu_I} \cdot (y^* - y^r) \cdot (w_i - y^*) \cdot \frac{\left(x_j - c_{ij}\right)^2}{b_{ij}^3} \tag{8}$$

where t is iteration time and K_α is a learning constant [1].

The learning algorithm for the conventional fuzzy inference model is shown as follows:

Learning Algorithm A

Step A1: The threshold θ of inference error and the maximum number of learning time T_{max} are set. Let n_0 be the initial number of rules. Let $t = 1$.

Step A2: The parameters b_{ij}, c_{ij}, and w_i are set randomly.

Step A3: Let $p = 1$.

Step A4: A data $\left(x_1^p, \cdots, x_m^p, y_p^r\right) \in D$ is given.

Step A5: From Eqs. (2) and (3), μ_i and y^* are computed.

Step A6: Parameters w_i, c_{ij}, and b_{ij} are updated by Eqs. (6), (7), and (8).

Step A7: If $p = P$, then go to Step A8, and if $p < P$ then go to Step A4 with $p \leftarrow p + 1$.

Step A8: Let $E(t)$ be inference error at step t calculated by Eq. (5). If $E(t) > \theta$ and $t < T_{max}$, then go to Step A3 with $t \leftarrow t + 1$; else, if $E(t) \leq \theta$ and $t \leq T_{max}$, then the algorithm terminates.

Step A9: If $t > T_{max}$ and $E(t) > \theta$, then go to Step A2 with $n = n + 1$ and $t = 1$.

In particular, Algorithm SDM is defined as follows:

Algorithm SDM (*c*, *b*, *w*)

θ_1: inference error

T_{max1}: the maximum number of learning time

n: the number of rules

input: current parameters

output: parameters $\boldsymbol{c}$, $\boldsymbol{b}$, and $\boldsymbol{w}$ after learning

Steps A3 to A8 of Algorithm A are performed.

2.2. Neural gas method

Vector quantization techniques encode a data space $V \subseteq R^m$, utilizing only a finite set $C = \{\boldsymbol{c}_i \mid i \in Z_r\}$ of reference vectors [18].

Let the winner vector $\boldsymbol{c}_i(\boldsymbol{v})$ be defined for any vector $\boldsymbol{v} \in \mathbf{V}$ as

$$i(\boldsymbol{v}) = \arg\min_{i \in Z_r} \|\boldsymbol{v} - c_i\|. \tag{9}$$

By using the finite set C, the space V is partitioned as

$$V_i = \left\{\boldsymbol{v} \in V \mid \|\boldsymbol{v} - \boldsymbol{c}_i\| \leq \|\boldsymbol{v} - \boldsymbol{c}_j\| \text{ f or } j \in \mathrm{Z}_r\right\}, \tag{10}$$

where $V = \cup_{i \in Z_r} V_i$ and $V_i \cap V_j = \varphi$ for $i \neq j$.

http://dx.doi.org/10.5772/intechopen.79925

The evaluation function for the partition is defined by

$$E = \sum_{i=1}^{r} \sum_{v \in V_i} \frac{1}{n_i} \|v - c_i\|^2, \tag{11}$$

where $n_i = |V_i|$.

Let us introduce the neural gas method as follows [18]:

For any input data vector v, the neighborhood ranking $c_i k$ for $k \in Z_{r-1}^*$ is

determined, being the reference vector for which there are k vectors c_j with

$$\|v - c_j\| < \|v - c_i k\| \tag{12}$$

Let the number k associated with each vector c_i denoted by $k_i(v,c_i)$. Then, the adaption step for adjusting the parameters is given by

$$\triangle c_i = \varepsilon \cdot h_\lambda(k_i(v, c_i)) \cdot (v - c_i) \tag{13}$$

$$h_\lambda(k_i(v, c_i)) = \exp(-k_i(v, c_i)/\lambda) \tag{14}$$

where $\varepsilon \in [0, 1]$ and $\lambda > 0$.

Let the probability of v selected from V be denoted by $p(v)$.

The flowchart of the conventional neural gas algorithm is shown in **Figure 1** [18], where ε_{int}, ε_{fin}, and T_{max2} are learning constants and the maximum number of learning, respectively. The method is called learning algorithm NG.

Using the set D, a decision procedure for center and width parameters is given as follows:

Algorithm Center (c)

$D^* = \{(x^p, \ldots, x^p) \mid p \in Z_p\}$

$p(x)$: the probability of x selected for $x \in D^*$.

Step 1: By using $p(x)$ for $x \in D^*$, NG method of **Figure 1** [16, 18] is performed.

As a result, the set C of reference vectors for D^* is determined, where $C = n$.

Step 2: Each value for center parameters is assigned to a reference vector. Let

$$b_{ij} = \frac{1}{n_i} \sum_{x_k \in C_i} \left(c_{ij} - x_{kj}\right)^2 \tag{15}$$

where C_i and n_i are the set and the number of learning data belonging to the ith cluster C_i and $C = \cup_{i=1}^{r} C_i$ and $n = \sum_{i=1}^{r} n_i$.

As a result, center and width parameters are determined from algorithm center (c).

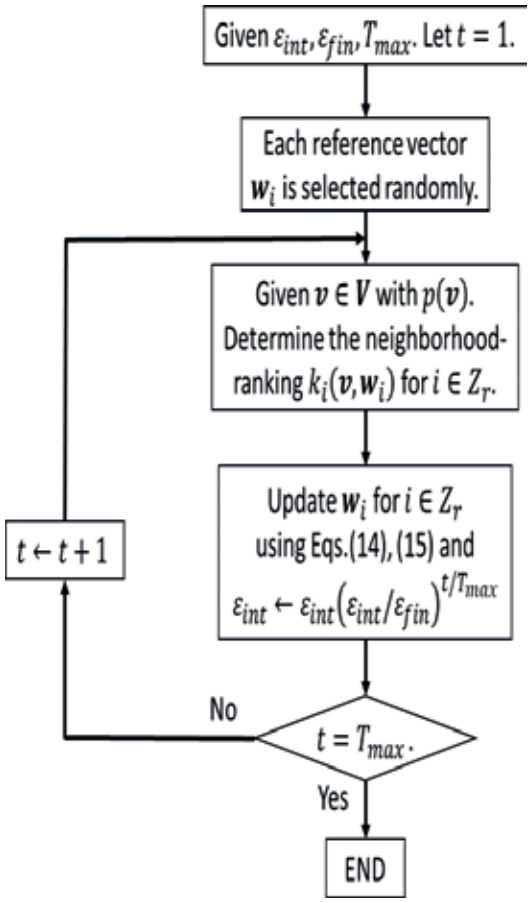


Figure 1. Neural gas method [18].

Learning Algorithm B using Algorithm Center (***c***) is introduced as follows [16, 17]:

Learning Algorithm B

θ: threshold of MSE

T_{max}^{0}: maximum number of learning time for NG

T_{max}: maximum number of learning time for SDM

M: the size of ranges

n: the number of rules

Step 1: Initialize()

Step 2: Center and width parameters are determined from Algorithm Center(P) and the set D^*.

Step 3: Parameters ***c***, ***b***, and ***w*** are updated using Algorithm SDM (***c***, ***b***, ***w***).

Step 4: If $E(t) \leq \theta$, then algorithm terminates else go to Step 3 with $n \leftarrow n + 1$ and $t \leftarrow t + 1$.

2.3. The probability distribution of input data based on the rate of change of output

It is known that many rules are needed at or near the places where output data change quickly in fuzzy modeling. Then, how can we find the rate of output change? The probability $p_M(\boldsymbol{x})$ is one method to perform it. As shown in Eqs. (16) and (17), any input data where output changes quickly is selected with the high probability, and any input data where output changes slowly is selected with the low probability, where M is the size of range considering output change.

Based on the literature [13], the probability (distribution) is defined as follows:

Algorithm Prob ($p_M(\boldsymbol{x})$)

Input: $D = \{(\boldsymbol{x}^p, y^r) \mid p \in Z_P\}$ and $D^* = \{(\boldsymbol{x}^p) \mid p \in Z_P\}$

Output: $p_M(\boldsymbol{x})$

Step 1: Give an input data $\boldsymbol{x}^i \in D^*$, we determine the neighborhood ranking ($\boldsymbol{x}^i 0$, $\boldsymbol{x}^i 1$, …, $\boldsymbol{x}^i k$,…, $\boldsymbol{x}^i P-1$) of the vector $\boldsymbol{x}^i$ with $\boldsymbol{x}^i 0 = \boldsymbol{x}^i$, $\boldsymbol{x}^i 1$ being closest to $\boldsymbol{x}^i$ and $\boldsymbol{x}^i k$ ($k = 0, \ldots, P-1$) being the vector $\boldsymbol{x}^i$ for which there are k vectors $\boldsymbol{x}^j$ with $\left\|\boldsymbol{x}^i - \boldsymbol{x}^j\right\| < \left\|\boldsymbol{x}^i - \boldsymbol{x}^{i_k}\right\|$.

Step 2: Determine $H(\boldsymbol{x}^i)$ which shows the rate of output change for input data $\boldsymbol{x}^i$, by the following equation:

$$H(\boldsymbol{x}^i) = \sum_{l=1}^{M} \frac{\left|y^i - y^{i_l}\right|}{\left\|\boldsymbol{x}^i - \boldsymbol{x}^{i_l}\right\|}, \tag{16}$$

where $\boldsymbol{x}^{i_l}$ for l Z_M means the lth neighborhood ranking of $\boldsymbol{x}^i$, $i \in Z_P$, and y^i and y^{i_l} are output for input $\boldsymbol{x}^i$ and $\boldsymbol{x}^{i_l}$, respectively. The number M means the range considering $H(\boldsymbol{x})$.

Step 3: Determine the probability $p_M(\boldsymbol{x}^i)$ for $\boldsymbol{x}^i$ by normalizing $H(\boldsymbol{x}^i)$ as follows:

$$p_M(\boldsymbol{x}^i) = \frac{H(\boldsymbol{x}^i)}{\sum_{j=1}^{P} H(\boldsymbol{x}^j)}, \tag{17}$$

where $\sum_{i=1}^{P} p_M(\boldsymbol{x}^i) = 1$.

See Ref. [19] for the detailed explanation using the example of $p_M(\boldsymbol{x})$. Using $p_M(\boldsymbol{x})$, Kishida has proposed the following learning algorithm [13]:

Learning Algorithm C

θ: threshold of MSE

T_{max}^0: maximum number of learning time for NG

T_{max}: maximum number of learning time for SDM

M: the size of ranges

n: the number of rules

Step 1: Initialize ( )

Step 2: The probability $p_M(\boldsymbol{x})$ is obtained from algorithm prob ($p_M(\boldsymbol{x})$).

Step 3: Center and width parameters are determined using $p_M(\boldsymbol{x})$ from Algorithm Center (P) and the data set D.

Step 4: Parameters $\boldsymbol{c}$, $\boldsymbol{b}$, and $\boldsymbol{w}$ are updated using Algorithm SDM ($\boldsymbol{c}$, $\boldsymbol{b}$, $\boldsymbol{w}$).

Step 5: If $E(t) \leq \theta$, then algorithm terminates else go to Step 3 with $n \leftarrow n + 1$ and t = 1.

2.4. Determination of weight parameters using the generalized inverse method

The optimum values of parameters $\boldsymbol{c}$ and $\boldsymbol{b}$ are determined by using $p_K(\boldsymbol{x})$. Then, how can we decide weight parameters $\boldsymbol{w}$? We can determine them as the interpolation problem for parameters $\boldsymbol{c}$, $\boldsymbol{b}$, and $\boldsymbol{w}$. That is, it is the method that membership values for antecedent part of rules are computed from $\boldsymbol{c}$ and $\boldsymbol{b}$ and weight parameters $\boldsymbol{w}$ are determined by solving the interpolation problem. So far, the method was used as a determination problem of weight parameters for RBF networks [1].

Let us explain fuzzy inference systems and interpolation problem using the generalized inverse method [1]. This problem can be stated mathematically as follows:

Given P points $\{\boldsymbol{x}^p \mid p \in Z_P\}$ and P real numbers $\{y_p^r \mid p \in Z_P\}$, find a function f: $R^m \rightarrow R$ such that the following conditions are satisfied:

$$f(\boldsymbol{x}^p) = y_p^r \tag{18}$$

In fuzzy modeling, this problem is solved as follows:

$$y_p = f(\boldsymbol{x}^p) = \sum_{i=1}^{n} w_i \varphi_{pi}(\|\boldsymbol{x}^p - \boldsymbol{c}_i\|) \tag{19}$$

$$\varphi_{pi}(\|\boldsymbol{x}^p - \boldsymbol{c}_i\|) = \frac{\mu_i}{\sum_{I=1}^{n} \mu_I}, \mu_i = \prod_{j=1}^{m} M_{ij}(x_j), \tag{20}$$

where μ_i and M_{ij} are defined as Eqs. (2) and (4).

That is,

$$\varphi\, \mathbf{w} = \mathbf{y}, \tag{21}$$

where $\varphi = (\varphi_{ij})$ ($i \in Z_P$ and $j \in Z_n$), $\boldsymbol{w} = (w_1, \ldots, w_n)^T$, and $\boldsymbol{y} = (y_1^r, \ldots, y_p^r)^T$.

Let $P = n$ and $\boldsymbol{x}^i = \boldsymbol{c}_i$. The width parameters are determined by Eq. (15). Then, if $\varphi_{ij}( )$ is suitably selected as Gaussian function, then the solution of weights $\boldsymbol{w}$ is obtained as

$$\boldsymbol{w} = \varphi^{-1}\boldsymbol{y} \tag{22}$$

Let us consider the case $n < P$. This is the realistic case. The optimum solution $\boldsymbol{w}^*$ that minimizes $E = ||y^r - \varphi \boldsymbol{w}||2$ can be obtained as follows:

$$\boldsymbol{w}^+ = \varphi^T \boldsymbol{y} \text{ and } E_{min} = \|(I - \Psi)\boldsymbol{y}\|^2, \tag{23}$$

where $\Phi^+ \triangleq [\Phi^T \Phi]^{-1}\Phi^T$, $\Psi \triangleq \Phi\Phi^T$, and I is identify matrix of $P \times P$.

The matrix Φ^+ is called the generalized inverse of φ. The method using Φ^+ to determine the weights is called the generalized inverse method (GIM).

Using GIM, a decision procedure for parameters is defined as follows:

Algorithm Weight(*c*, *b*)

Input: $D = \{(\boldsymbol{x}^p, y^r) | p \in Z_P\}$

Output: The weight parameters $\boldsymbol{w}$

Step 1: Calculate μ_i based on Eq. (2)

Step 2: Calculate the matrix Φ and Φ^+ using Eq. (20):

$$\varphi_{pi} = (\|x^p - c_i\|) = \frac{\mu_i^p}{\sum_{j=1}^n \mu_j^p}, \mu_i^p = \prod_{j=1}^m \exp\left(-\frac{1}{2}\left(\frac{x_j^p - c_{ij}}{b_{ij}}\right)^2\right)$$

Step 3: Determine the weight vectors $\boldsymbol{w}$ as follows:

$$\boldsymbol{w} = \Phi^+ \boldsymbol{y}^r \tag{24}$$

2.5. The relation between the proposed algorithm and related works

Let us explain the relation between the proposed method and related works using **Figure 2**.

1. The fundamental flow of algorithm A is shown in **Figure 2**(**a**). Initial parameters of $\boldsymbol{c}$, $\boldsymbol{b}$, and $\boldsymbol{w}$ are set randomly, and all parameters are updated using SDM until the inference error become sufficiently small (see **Figure 2**(**a**)) [1].

2. The first method using VQ is the one that both the initial assignment of parameters and the assignment of parameters in iterating step (see outer loop of **Figure 2**(**b**)) are also determined by NG using D*. That is, it is learning method composed of two stages. The center parameters $\boldsymbol{c}$ are determined using D^* by VQ, $\boldsymbol{b}$ is computed by Eq. (15) using the result of center parameters, and weight parameter $\boldsymbol{w}$ is set to the results of SDM, where the initial values of $\boldsymbol{w}$ are set randomly. Further, all parameters are updated using SDM for the definite number of learning time. In iterating processes, parameters of the result obtained by SDM are set as initial ones of the next process. Outer iterating process is repeated until the inference error become sufficiently small (see **Figure 2**(**b**)).

3. The second method using VQ is the one that is the same method as the first one except for selecting any learning data based on $p_M(\boldsymbol{x})$ (see **Figure 2**(**c**)). That is, center parameters $\boldsymbol{c}$ are determined by $p_M(\boldsymbol{x})$ using input and output learning data.

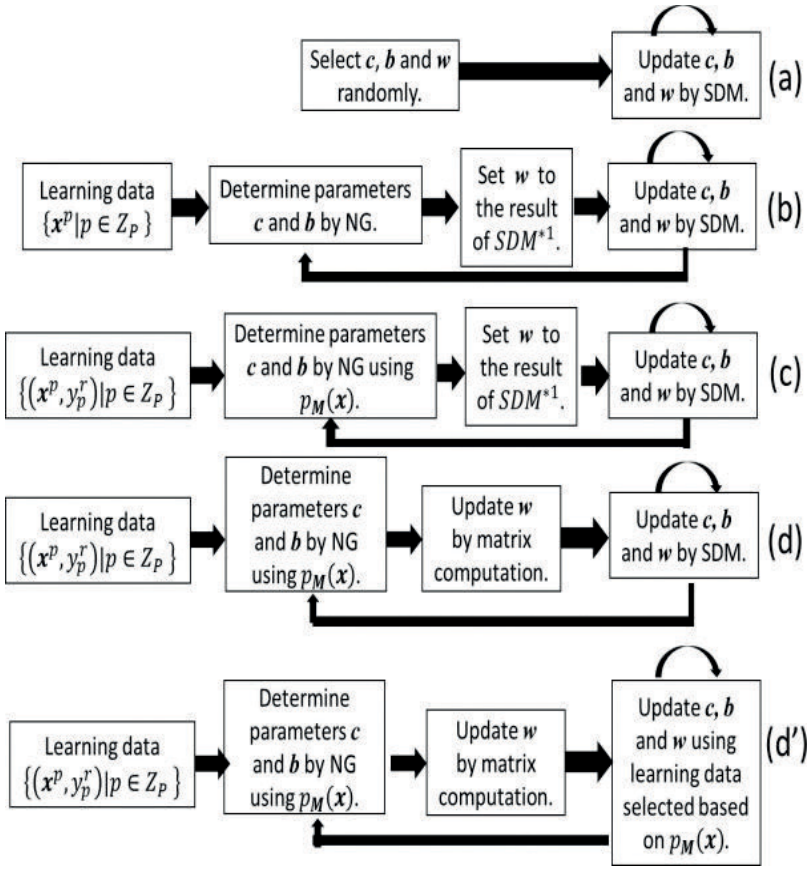


Figure 2. Concept of conventional and proposed algorithms: mark 1 means that initial values of w are selected randomly and parameters w are set to the result of SDM after the second step.

4. The third learning method using VQ is the one that parameters w are determined using GIM after parameters c and b are determined by VQ using $p_M(x)$ and all parameters are updated based on SDM. That is, it is learning method composed of three phases. In the first phase, the center parameters c are determined using the probability $p_M(x)$, and b is computed from the result of center parameters. In the second phase, weight parameters w are determined by solving the interpolation problem using GIM. In the third phase, all parameters are updated using SDM for the definite number of learning time. In iterating process, the result of SDM is set to initial ones of the next process based on hill climing. Outer process is repeated until the inference error becomes sufficiently small (see **Figure 2**(**d**)).

5. The fourth method is the same to the one as the third method except for using $p_M(x)$ in learning process of SDM (see **Figure 2**(**d'**)). This is a proposed method in this paper.

http://dx.doi.org/10.5772/intechopen.79925

3. The proposed learning method using VQ

Let us explain the detailed algorithm of **Figure 2**(**d′**). The method is called Learning Algorithm D′. It is composed of four techniques as follows:

1. Determine the initial assignment of $\boldsymbol{c}$ using the probability $p_K(\boldsymbol{x})$.
2. Determine the assignment of weight parameters $\boldsymbol{w}$ by solving the interpolation problem using GIM.
3. The processes (1) and (2) and learning steps of SDM using $p_M(\boldsymbol{x})$ are iterated.
4. The optimum value of M is determined by hill climing method [16].

The general scheme of the proposed method is shown as **Figure 3**, where $\boldsymbol{c}_{min}$, $\boldsymbol{b}_{min}$, and $\boldsymbol{w}_{min}$ are the optimal parameters for $\boldsymbol{c}$, $\boldsymbol{b}$, and $\boldsymbol{w}$.

T_{max1} and T_{max2}: The maximum numbers of learning time for NG and SDM.

θ and θ_1: Thresholds for MSE and SDM

M_0, M_{max}: The size of initial and final of ranges

$\triangle M$: The rate of change of the range

D and D^*: Learning data $D = \{(\boldsymbol{x}^i, y^r) \mid i \in Z_P\}$ and $D^* = \{\boldsymbol{x}^i \mid i \in Z_P\}$

n: The number of rules

$E(t)$: MSE of inference error at step t

E_{min}: The minimum MSE of E for the rule number

The proposed method of **Figure 3** consists of five phases: In the first phase, all values for algorithm are initialized. In the second phase, the probability $p_M(\boldsymbol{x})$ is determined for the size of range M. In the third phase, parameters $\boldsymbol{c}$ are determined by NG using $p_M(\boldsymbol{x})$, and parameters $\boldsymbol{b}$ are computed from parameters $\boldsymbol{c}$. In the forth phase, parameters $\boldsymbol{w}$ are determined from algorithm weight($\boldsymbol{c}$, $\boldsymbol{b}$). In the fifth phase, all parameters are updated using $p_M(\boldsymbol{x})$ by SDM. The optimum number n^* of rules and the optimum size M^* of range are determined in **Figure 4**. That is, the number M for the fixed number n is adjusted, and the optimum values of n^* and M^* with the minimum number for MSE are determined. Especially, Learning Algorithm D is same method as Learning Algorithm D′ except for the step with the symbol "*" in **Figure 3**. In learning steps of SDM for Learning Algorithm D, learning data is selected randomly (see **Figure 2**(**d**)).

Likewise, we also propose improved methods for **Figure 2**(**a**)–(**c**). In learning process of SDM for algorithm (a), (b), and (c), any learning data is selected randomly. In the proposed methods, any learning data is selected based on $p_M(\boldsymbol{x})$. These algorithms are defined as (a′), (b′), and (c′).

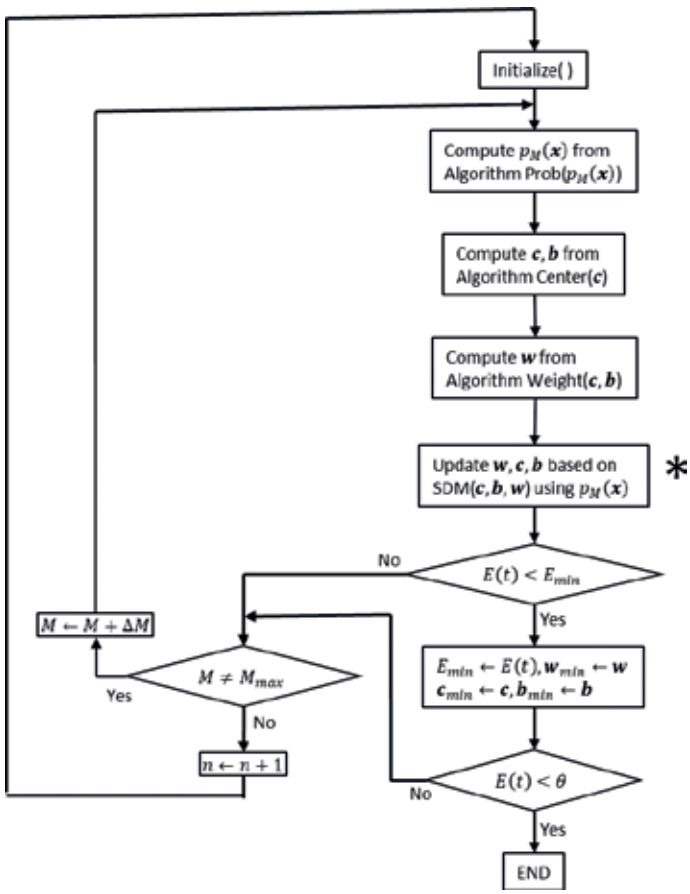


Figure 3. Flowchart of Learning Algorithm D′ corresponding to **Figure 2**(**d′**).

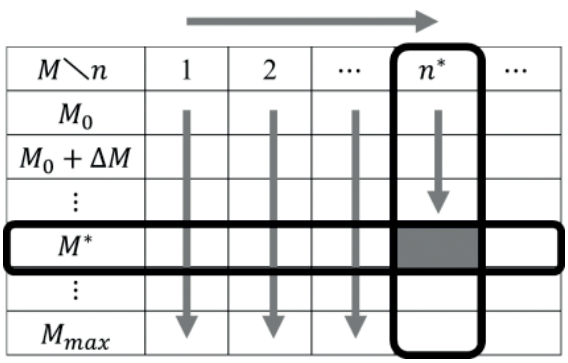


Figure 4. The optimum values M^* and n^* for M and n.

http://dx.doi.org/10.5772/intechopen.79925

4. Numerical simulations

In order to compare the ability of Learning Algorithms (a′), (b′), (c′), and (d′) with Learning Algorithms (a), (b), (c), and (d), numerical simulations for function approximation and pattern classification are performed.

4.1. Function approximation

The systems are identified by fuzzy inference systems. This simulation uses four systems specified by the following functions with two-dimensional input space $[0, 1]^2$ (Eqs. (25)–(28)) and one output with the range [0, 1];

$$y = \sin\left(\pi x_1^3\right) x_2 \tag{25}$$

$$y = \frac{\sin\left(2\pi x_1^3\right)\cos\left(\pi x_2\right) + 1}{2} \tag{26}$$

$$y = \frac{1.9\left(\left(1.35 + \exp\left(x_1\right)\right)\sin\left(13(x_1 - 0.6)^2\right)\exp\left(-x_2\right)\sin\left(7x_2\right)\right)}{2} \tag{27}$$

$$y = \frac{\sin\left(10(x_1 - 0.5)^2 + 10(x_2 - 0.5)^2\right) + 1}{2} \tag{28}$$

In this simulation, $T_{max1} = 100000$ and $T_{max2} = 50000$ for (a) and $T_{max1} = 10000$ and $T_{max2} = 5000$ for (b), (c), and (d) and $\theta = 1.0 \times 10^{-4}$, $K_0 = 100$, $K_{max} = 190$, $K = 10$, $K_c = 0.01$, $K_b = 0.01$, $K_c = 0.1$, the number of learning data is 200 and the number of test data is 2500.

Table 1 shows the results for the simulation. In **Table 1**, the number of rules, MSEs for learning and test, and learning time (second) are shown, where the number of rules means the one when the threshold θ of inference error is achieved in learning. The result of simulation is the average value from 20 trials. As a result, the results of (a′), (b′), (c′), and (d′) are almost same as the cases of (a), (b), (c), and (d) as shown in **Table 1**. It seems that there is no difference of the ability for the regression problem.

4.2. Classification problems for UCI database

Iris, Wine, Sonar, and BCW data from UCI database shown in **Table 2** are used as the second numerical simulation [20]. In this simulation, fivefold cross validation is used. As the initial conditions for classification problem, $K_c = 0.001$, $K_b = 0.001$, $K_w = 0.05$, $\varepsilon_{init} = 0.1$, $\varepsilon_{fin} = 0.01$, and $\lambda = 0.7$ are used. Further, $T_{max} = 50000$, $M = 100$, and $\theta = 1.0 \times 10^{-2}$ for iris and wine. $T_{max} = 50000$, $M = 200$, and $\theta = 2.0 \times 10^{-2}$ for BCW; and $T_{max} = 5000$, $M = 100$, and $\theta = 5.0 \times 10^{-2}$ for sonar are used.

Table 3 shows the result of classification problem. In **Table 3**, the number of rules, RMs for learning, and test data are shown, where RM means the rate of misclassification. As a result, the

		Eq. (25)	Eq. (26)	Eq. (27)	Eq. (28)
(a)	The number of rules	8.3	22.5	52.4	6.1
	MSE for learning($\times 10^{-4}$)	0.47	0.35	0.65	0.41
	MSE of test ($\times 10^{-4}$)	2.29	21.12	2.83	7.37
(b)	The number of rules	4.7	6.8	9.6	4.0
	MSE of learning ($\times 10^{-4}$)	0.44	0.38	0.84	0.35
	MSE of test ($\times 10^{-4}$)	0.70	2.96	2.34	0.48
(c)	The number of rules	5.4	7.4	11.1	3.5
	MSE of learning ($\times 10^{-4}$)	0.24	0.54	0.65	0.33
	MSE of test ($\times 10^{-4}$)	0.65	1.36	4.48	0.44
(d)	The number of rules	4.3	6.1	9.7	3.5
	MSE of learning ($\times 10^{-4}$)	0.28	0.39	0.69	0.29
	MSE of test ($\times 10^{-4}$)	0.57	1.93	1.78	0.36
(a′)	The number of rules	5.0	8.9	11.8	4.7
	MSE for learning ($\times 10^{-4}$)	0.37	0.41	0.52	0.45
	MSE of test ($\times 10^{-4}$)	1.55	9.56	2.8	1.06
(b′)	The number of rules	5.0	8.9	13.0	4.3
	MSE for learning ($\times 10^{-4}$)	0.42	0.38	0.65	0.39
	MSE of test ($\times 10^{-4}$)	1.41	9.66	4.12	2.38
(c′)	The number of rules	5.7	8.0	13.1	4.1
	MSE for learning ($\times 10^{-4}$)	0.40	0.23	0.57	0.35
	MSE of test ($\times 10^{-4}$)	1.70	1.28	3.90	1.10
(d′)	The number of rules	4.6	6.9	10.0	3.6
	MSE for learning ($\times 10^{-4}$)	0.39	0.49	0.62	0.35
	MSE of test ($\times 10^{-4}$)	1.43	2.58	1.89	0.42

Table 1. The results of simulations for function approximation.

	Iris	Wine	BCW	Sonar
The number of data	150	178	683	208
The number of input	4	13	9	60
The number of class	3	3	2	2

Table 2. The dataset for pattern classification.

results of (a′), (b′), (c′), and (d′) are superior in the number of rules to the cases of (a), (b), (c), and (d) as shown in **Table 3**. It seems that there is the difference of ability for pattern classification.

Let us consider the reason why we can get the good result by using the probability $p_M(\boldsymbol{x})$. In the conventional learning method, parameters are updated by any data selected randomly

		Iris	Wine	BCW	Sonar
(a)	The number of rules	3.4	7.8	14.4	11.0
	RM for learning (%)	3.0	1.4	1.6	5.3
	RM of test (%)	3.3	10.3	4.3	20.6
(b)	The number of rules	2.0	20.8	26.0	3.7
	RM of learning (%)	3.3	13.6	2.2	5.1
	RM of test (%)	3.3	16.6	3.5	18.2
(c)	The number of rules	2.0	3.2	4.8	4.0
	RM of learning (%)	3.3	1.5	1.6	5.1
	RM of test (%)	4.0	6.7	3.8	19.0
(d)	The number of rules	3.7	2.5	2.5	4.0
	RM of learning (%)	3.3	1.1	1.3	5.1
	RM of test (%)	3.8	6.5	2.1	18.3
(a′)	The number of rules	2.3	2.2	3.5	4.6
	RM for learning (%)	2.9	1.4	1.6	5.0
	RM of test (%)	3.5	8.5	3.9	20.0
(b′)	The number of rules	2.0	2.0	2.1	3.7
	RM for learning (%)	3.9	3.0	2.1	5.0
	RM of test (%)	4.9	9.2	3.9	19.0
(c′)	The number of rules	2.3	3.0	3.6	4.0
	RM for learning (%)	3.3	2.6	2.2	5.3
	RM of test (%)	4.0	7.2	3.5	19.4
(d′)	The number of rules	2.3	2.0	2.4	3.3
	RM for learning (%)	3.0	1.8	2.2	5.0
	RM of test (%)	3.5	7.6	3.7	19.1

Table 3. The result for pattern classification.

from the set of learning data. In the proposed method, parameters are updated by any data selected based on the probability $p_M(\boldsymbol{x})$. The probability $p_M(\boldsymbol{x})$ is determined based on output change for learning data, so many fuzzy rules are likely to generate at or near the places where output change is large for the set of learning data.

For example, if the number of learning time is 100 and $p_M(\boldsymbol{x}^0) = 0.5$, then learning data $\boldsymbol{x}^0$ is selected 50 times from the set of learning data in learning. As a result, membership functions are likely to generate at or near the places where output change is large for the set of learning data. The probability $p_M(\boldsymbol{x})$ is used in a method to improve the local search of SDM.

5. Conclusion

In this paper, we proposed the improved methods using VQ, GIM, and SDM. The features of the proposed methods are as follows:

1. In determining the initial assignment of parameters, both input and output parts of learning data are used.
2. The initial assignment of weight parameters is determined by GIM.
3. In order to determine the range of the rate of output change, hill climing is used.
4. Any learning data in SDM is selected based on the probability distribution

$p_M(\boldsymbol{x})$ considering both input and output of learning data.

As a result, it was shown that the proposed methods using the probability distribution considering both input and output parts of learning data were superior to other methods in numerical simulation of pattern classification.

In the future works, we will consider the new idea using VQ and apply the proposed method to control problem.

Author details

Hirofumi Miyajima[1], Noritaka Shigei[2] and Hiromi Miyajima[3]*

*Address all correspondence to: k2356323@kadai.jp

1 Faculty of Informatics, Okayama University of Science, Okayama, Japan

2 Kagoshima University, Kagoshima, Japan

3 Former Kagoshima University, Kagoshima, Japan

References

[1] Gupta MM et al. Static and Dynamic Neural Networks. John Wiley & Sons: IEEE Press; 2004

[2] Fukumoto S, Miyajima H, Kishida K, Nagasawa Y. A Destructive Learning Method of Fuzzy Inference Rules. Proc. of IEEE on Fuzzy Systems;1995. pp. 687-694

[3] Cordon O. A historical review of evolutionary learning methods for Mamdani-type fuzzy rule-based systems, designing interpretable genetic fuzzy systems. Journal of Approximate Reasoning. 2011;**52**:894-913

[4] Kosko B. Neural Networks and Fuzzy Systems, A Dynamical Systems Approach to Machine Intelligence. Englewood Cliffs, NJ: Prentice Hall; 1992

[5] Lin C, Lee C. Neural Fuzzy Systems. PTR: Prentice Hall; 1996

[6] Casillas J, Cordon O, Herrera F, Magdalena L. Accuracy Improvements in Linguistic Fuzzy Modeling, Studies in Fuzziness and Soft Computing. Vol. 129. Berlin Heidelberg: Springer-Verlag; 2003

[7] Liu B. Theory and Practice of Uncertain Programming, Studies in Fuzziness and Soft Computing. Vol. 239. Physica-Verlag Heidelberg: Springer; 2009

[8] Zhoua SM, Ganb JQ. Low-level interpretability and high-level interpretability: a unified view of data-driven interpretable fuzzy system modeling. Fuzzy Sets and Systems. 2008; **159**:3091-3131

[9] Miyajima H et al. SIRMs fuzzy inference model with linear transformation of input variables and universal approximation, advances in computational intelligence. Proc. 13th International Work Conference on Artificial Neural Networks, Part I. pp. 561-575, Spain; 2015

[10] Yubazaki N, Yi J, Hirota K. SIRMS (single input rule modules) connected fuzzy inference model. Journal Advanced Computational Intelligence. 1997;**1**(1):23-30

[11] Kishida K et al. A self-tuning method of fuzzy modeling using vector quantization. Proc. FUZZ-IEEE'97. pp. 397-402;1997

[12] Kishida K et al. Destructive fuzzy modeling using neural gas network. IEICE Trans. on Fundamentals. 1997;**E80-A**(9):1578-1584

[13] Kishida K et al. A learning method of fuzzy inference rules using vector quantization. Proceedings of the 21st International Conference on Artificial Neural Networks. 1998;**2**: 827-832

[14] Fukumoto S et al. A decision procedure of the initial values of fuzzy inference system using counterpropagation networks. Journal of Signal Processing. 2005;**9**(4):335-342

[15] Pedrycz W et al. Cluster-centric fuzzy modeling. IEEE Transactions on Fuzzy Systems. 2014;**22**(6):1585-1597

[16] Miyajima H et al. Fast learning algorithm for fuzzy inference systems using vector quantization. International MultiConference of Engineers and Computer Scientists. 2016;**I**:1-6

[17] Miyajima H et al. The ability of learning algorithms for fuzzy inference systems using vector quantization. ICONIP 2016, part IV, LNCS9950; 2016. pp. 479-488

[18] Martinetz TM et al. Neural gas network for vector quantization and its application to time-series prediction. IEEE Transaction on Neural Network. 1993;**4**:558-569

[19] Miyajima H et al. An improved learning algorithm of fuzzy inference systems using vector quantization. Advanced in Fuzzy Sets and Systems. 2016;**21**(1):59-77

[20] UCI Repository of Machine Learning Databases and Domain Theories. ftp://ftp.ics.uci.edu/pub/machinelearning-Databases

Chapter 8

Query Morphing: A Proximity-Based Approach for Data Exploration

Jay Patel and Vikram Singh

Additional information is available at the end of the chapter

http://dx.doi.org/10.5772/intechopen.77073

Abstract

We are living in age where large information in the form of structured and unstructured data is generated through social media, blogs, lab simulations, sensors etc. on daily basis. Due to this occurrences, acquisition of relevant information becomes a challenging task for humans. Fundamental understanding of complex schema and content is necessary for formulating data retrieval request. Therefore, instead of search, we need exploration in which a naïve user walks through the database and stops when satisfactory information is met. During this, a user iteratively transforms his search request in order to gain relevant information; morphing is an approach for generation of various transformation of input. We proposed 'Query morphing', an approach for query reformulation based on data exploration. Identified design concerns and implementation constraints are also discussed for the proposed approach.

Keywords: query transformation, query reformulation, proximity-based query processing, data exploration, exploratory search

1. Introduction

A fundament search activity begins with the formulation of search intension and mines meaningful information from available information space. This helps the user in gaining intellectual skills and cognitive understanding. Traditional search systems usually support lookup searching in that user has a proper wisdom of their information goal. This type of search relies on traditional 'Query-Result' paradigm in that user pose a query for the relevant document retrieval, browse through results and analyze them to fulfill his information need. This approach performs well in the case of short navigational information requests and fulfills

an information location need, but fails in information discovery need [39]. For discovery-oriented applications such as uncovering the information pattern from genomics, health care data, scientific data etc., additional assistance is required to formulate queries and navigation in data space to gain the desired information [16]. In such scenarios, the user usually uncertain about his information goals and/or less familiar with data semantics and context that makes the phrasing of information request [12] challenging. Also, initial search aims and intentions evolve as new information is encountered. Hence, the burden of analyzing, re-organizing and keeping track of the information gathered falls on the user alone [16, 17]. Exploratory search is one such emerging research area that realizes the importance of user's efforts in multiple phases of discovering, analyzing, and learning. Exploratory search systems can deliver pleasing quality information due to their recall-oriented reformulation from short typed ill-phrased query to precise query [23, 29, 37, 38].

User's search tasks can be categorized into three behaviors: Lookup, Learn and Investigate that is shown in **Figure 1**. The user may perform multiple types of search task in parallel, therefore searches are denoted by overlapping clouds. Generally, there is interplay between search tasks, for example lookup task interplay with investigate or learn. If we analyze the search behaviors, we can relate traditional search tasks with the lookup tasks in that carefully formulated queries yield precise result with the minimal relevance comparison. For exploratory search tasks, the system seeks more involvement beyond just a query specification and result presentation. A group of tasks allied with exploratory search is of type learn and investigate. Learning behaviour are aiming to knowledge acquisition in that user tries to develop addition, knowledge about the domain and better understand the problem context. It is an iterative process that simulates analogical thinking and relate users' experiences to return a set of data objects. Reformulating queries and comparing results take much time in learning

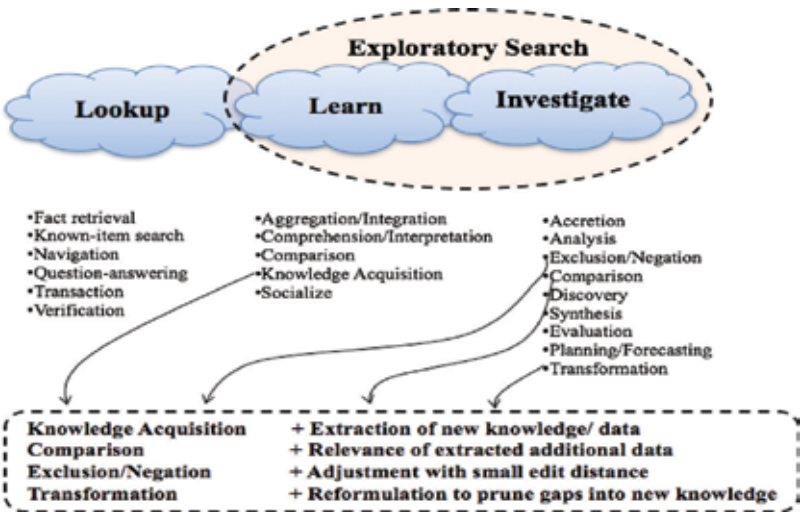


Figure 1. Exploratory search and sub-activities.

http://dx.doi.org/10.5772/intechopen.77073

tasks. Investigate behaviour prunes gap in knowledge and transform existing data into new knowledge.

Increase in several competing technologies leads to the generation of large structured and unstructured operational and transactional data. The key source, includes sensors, lab simulators, social media, web pages etc. In this setting, fundamental understanding of complex schema and content is necessary for formulating a data retrieval request otherwise user often stumbled upon empty or huge result set of his query. For such situations, we came up with an imitative towards 'Query Reformulation' as a vital task of Query Evaluation, named a 'Query morphing'. The proposal extract relevant and additional data objects from available data space and then recognize suggestions to acquire intermediate query reformulation.

Morphing refers to undergo a gradual process of transformation of input, e.g. for Image morphing [24, 10], Data Morphing [20]. Some traditional information retrieval techniques that transform initial query submitted by user are mapped in **Figure 2**. These transformations techniques aim to retrieve relevant information and improve system performance as well. Query reformulation techniques perform various transformations by applying user cognitive effort or system assistance and formulate semantically equivalent queries to reduce costs [26, 32, 40]. Pre classified data is required as database abstraction is performed for query reformulation. For successful reformulation it is better to understand the searchers intend and for that query rewriting can be a good option for query transformation. Query rewriting can be viewed as a generalization of query relaxation, query expansion [1, 40] and query substitution techniques [2, 40]. Query expansion techniques answer additional documents by evaluating inputs and expanding original user query through terms addition. Query relaxation techniques, conflict the expansion techniques [3]. Query relaxation is done to generalize query as sometimes ill-phrased query leads to fewer answer. Transformation process based on typical possible alternatives on original query is done in query substitution techniques [4]. An off-the-shelf dictionary/treasure is required for all these query transformation techniques [5].

Techniques grouped towards the left part in the **Figure 2** assist users for precise and unambiguous query formulation and execution. Various relevant query recommendations are generated and suggested that assist users in real-time query reformulation. Query suggestion techniques determines list of relevant queries that may help to achieve a user's search need [6, 18]. Query auto-completion techniques self-complete the formulation of queries have previously been observed in search logs. During a search, user often search is a sequence of queries of similar information need, query chain identifies this sequence. Query logs of earlier queries posed by global user are required to compute query suggestions list. Query recommendation

Figure 2. Query transformations and various equivalent techniques.

techniques track user's querying behaviour, identify the interested area from the available data space and recommends set of queries that retrieved relevant information. The query is steering [12, 26] is one process that navigates the user through complex data structures. For query recommendation and steering interactive query session is required to achieve ultimate search goal [12].

Due to the big data occurrences, traditional ways of query transformation repeatedly encountered challenges of relevance. To contrive such inherent challenges of transformation and relevance for exploration in large data a technique 'Query morphing' is designed. Our proposal suggests additional relevant data objects for the formulation of precise query by exploring available data space and leveraging use feedback. We concur that query morphing will also acquire the properties of traditional methodologies by observing that search query and respective results analogous to the history log.

1.1. Contribution and outline

The main contribution of this paper is an algorithm designed for query reformulation based on exploration technique. Algorithm named 'Query morphing' explores into the proximity of initial user query and extract additional relevant data objects. These retrieved data objects from the n-dimensional neighborhood assist user in his intermediate query reformulation. Proximate data objects are selected based on implicit and explicit relevance. We expect that our proposal, guides on exploration over several ample databases, such as Medical database, DNA database, social database, scientific database, etc. Finally, various existing reformulation techniques are revisited to establish the fact that how 'Query morphing' is different from traditional transformation techniques.

Next section listed some related research prospects and approaches. In Section 3 the proposed approach is conveyed, in which conceptual design is represented with algorithm and schematic diagram. Various design issues, analysis of implementations as well as intrinsic implementation complexity in proposed approach are recognized in Section 4. Lastly, the conclusion is presented.

2. Literature review

Many of today's query processing platforms carry a much profound repertoire and resilient querying techniques to regulate huge observational data in a limited resource environment. [12, 11]. Below some aspects are reviewed that helps user in searching relevant information from a large data set. We consider some prominent researches delivered for automatic exploration in data space, formulation of approximate quires and techniques that assist the user in the query formulation process to cover our aspects.

2.1. Automatic exploration

Analyzing vast amount of real time data can be an extremely complex task and required automation. In such scenario without any assistance, user ended up with ill-formulated query

that retrieves no result or huge result set. Traditional Database Management tools and systems are constructed by considering that database semantics is well understood by users [39]. Therefore, current applications with huge and complex database do not work well with these traditional Data Base Management techniques. Many interactive data exploration strategies are proposed and developed by researchers that extract and uncover great knowledge from complex data via highly ad-hoc interaction.

Automatic Interactive Data Exploration (AIDE) framework is well explained in [16] by authors. In that, the user is directed towards the data area of interest by deliberately incorporating relevance feedback. Various machine learning and data mining techniques can be integrated in that to achieve the best performance. Similarly, in [17] YAML framework is suggested and it uses attribute-value pair frequency to make exploration effective. Automatic exploration strategy performs formulation of user's queries and leads towards relevant information.

2.2. Query approximation

In exploratory query aspects where the user is satisfied with 'closed-enough' answer, approximation modules implemented in search system help to achieve shorter response time. This approximation module is built without changing underling database architecture. For example, Aqua approximate query answering system [4] rewrites queries using summery synopsis to provide approximate answers. Automatic Query Processing (AQP) widely uses statistical techniques based on the synopsis [14] to analyze large amount of data. Four main key synopsis are used by researchers for approximation which is random sample synopsis, histogram synopsis, wavelet and sketches synopsis.

Most fundamental and commonly used synopsis is a random sampling in that subset of data objects are fetched based on stochastic mechanism. It is easy to draw samples from a small available data, although to make the sampling process scalable, advance sampling techniques are required e.g. BlinkDB [6] architecture. In this architecture samples are selected based on accuracy of query and response time that device dynamic sampling strategy. A Histogram synopsis method group the data values into subset by summarizing the attribute frequency distribution or combined attribute frequency distribution. By using advance methods such as aggregation over joints are also used to approximate more general class of query. Another synopsis is wavelet synopsis which is identical with the above but the only variation is that it transforms and express most substantial data into the frequency domain. A faster response is one characteristic of approximate query processing. Speedup with accuracy is the key objective of AQP, therefore, returned results must be verified. Interactive approximate query processing performs error estimation [5] and error diagnosis via close forms or bootstrap that guarantees runtime efficiency and resource usage.

2.3. Assisted query formulation

Due to the big data contingency and complex schematic structure of data, sensible formalisms of query is required for complex information retrieval which is mastered by a small group of users usually. Most users in real life apply brute force approaches which manipulate data by hand as they have little knowledge regarding query formulation. Assisted query formulation

techniques are proposed to resolve these issues. These techniques assist the user by suggesting some query terms for subsequent formulation of incremental queries and reduction of irrelevant data retrieval. Fundamental operations such as equijoin and semijoin [11] are characterized for the formation of Boolean membership queries in polynomial time. A user membership driven learning algorithms [2] can also serves better formulation for simple Boolean queries. Many other formulation techniques for query construction such as locate minimal project join queries, discovering query approach [34] etc. are developed to answer query formulation similar to example tuples [27].

We termed our approach as 'Query morphing' because in literature a traditional method, morphing points transformation of inputs e.g. Data Morphing [20], Image morphing [10, 24]. Similarly, a small transformation of user queries are also carries out in our approach. We realized that the success of our approach is mainly rely on effective database exploration and user participation. The properties of traditional techniques are also incorporated in our query morphing approach as user's search request and retrieved outcomes analogous to the user history log.

3. Query morphing: a query reformulation approach

There are many tools available to extract knowledge from data, but they are inadequate in finding an appropriate subset of data. A deep analysis is needed to gain relevant knowledge [21] from available information space. Most of the tools follow the traditional lookup behavior that aims to retrieve the best literal match in a short time by assuming that the user is aware of 'what he is looking for'. That means systems are designed by considering that the user has a clear understanding about his search goals and familiar with database schema and context. It is observed that the success of the search process anticipates effective query articulation. Therefore, domain expert user successfully performs the search operation [33] and retrieve relevant results as he had formulated his query with appropriate terms [22, 23, 28]. But naïve user has to face challenges in the formation of his information seeking task due to less domain awareness. To resolve this he should be assisted through flexible query answering system [12] in query construction by delivering additional possible result sets along with original query results [25, 30]. The motive of such system is to reduce user's cognitive effort in subsequent queries [31, 8] by enhancing his knowledge.

Most systems support 'Query-Result' paradigm which is not sufficient as query formulation [12, 34] affects performance of the system. Instead 'Query-Result-Review-Query' paradigm can help as a user's search intention evolves with search progresses. The traditional methodologies retrieve results based on predefined relevant criteria and fails in identifying shift occurs in user's search intensions. Therefore a recall-oriented approach [19, 35] for query reformulation is designed. The idea behind it is as follows, the user poses an initial query Q and system yields effort T on finding the optimal results for Q. A small portion is set aside for the exploration. Various syntactically adjustments are performed with a small edit distance

from Q to create variations Q_i. A conceptual example is shown in **Figure 3**. The result set returned by user's original query request is painted by the small red circle. Possible additional relevant result sets of user's interest for queries are explored in the large data spectrum, providing that result set belongs to surrounding closed region of the original request. Orange elliptic represents the query results that correspond to variations of original query request. After analyzing results user may formulate another query to shift interest towards another query result as shown in the right portion of **Figure 3**. A new region of query result of the user's request includes both new and previous variations of the query. A data space expedition and feedback incorporation is observed for the query reformulation and additional relevant data suggestion in this approach. The additional relevant data objects are retrieved by performing exploration and exploitation in available database. The properties of traditional techniques are also incorporated in our query morphing approach as user's search request and retrieved outcomes analogous to the user history log.

Creating transformation of input like text, image, data etc. is a fundamental process in computer science called morphing [10]. We analogous our query with the morphing inputs and named our reformulation algorithm as 'Query morphing' [41]. Our algorithm helps user in formulation of intermediate queries by creating variants/transformation of the original search query. The assistance to the user will be based on the optimal query reformulations derived during exploration and exploitation of dataspace. The proposed algorithms are developed by considering 'Query-Result- Review-Query' paradigm of computing [7, 15, 39]. The design framework for the same is conveyed in following section and shown in **Figure 4**.

3.1. Proposed approach

Our query reformulation approach can be seen into two sub activities, one is tradition query processing the other one is generation of morphs that derive intermediate query reformulation. Initially query Q_i will be validated and processed by the query engine in the traditional query processing mechanism by the DBMS. Data objects retrieved after processing initial query Q_i are identified on d-dimensional space that is already created and partitioned into non-overlapping rectangular cells and exploited for subsequent interactions. If we say more specifically, $S = D_1 \times D_2 \times \ldots, \times D_d$ be a d-dimensional space where $D = \{D_1, D_2, \ldots, D_d\}$ be a

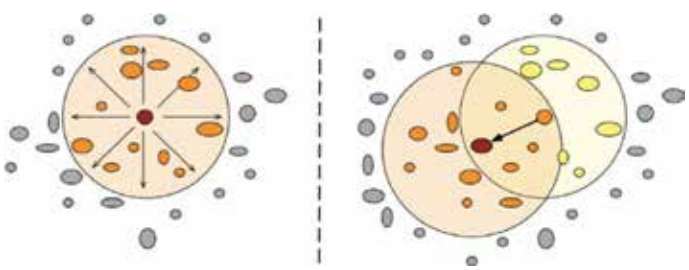


Figure 3. A conceptual example of query morphing.

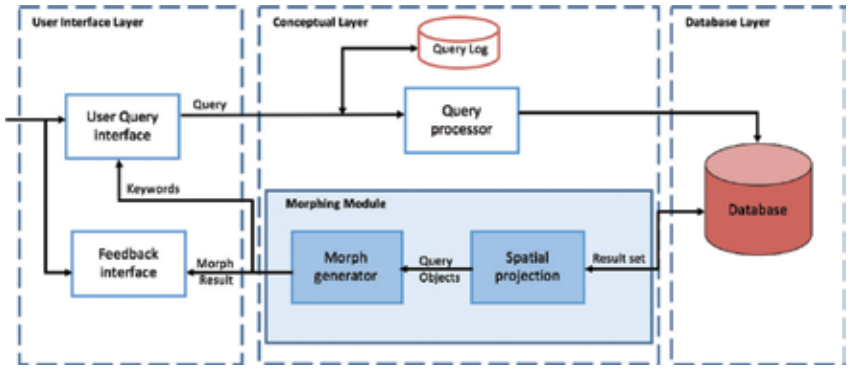


Figure 4. Query morphing and user's interactions.

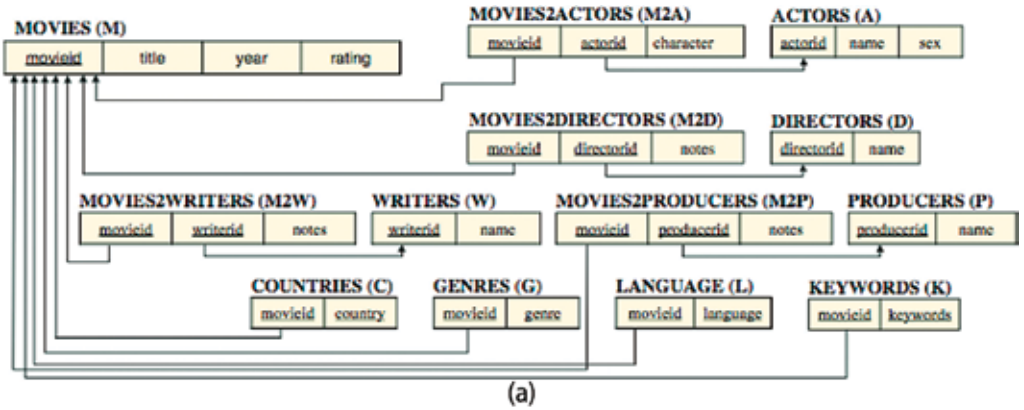
set of totally ordered and bounded domains (attributes). Divide S into m^d non-overlapping rectangular cells by partitioning every dimension D_i ($1 \leq i \leq d$) into m equal length intervals. A d-dimensional data points, p, is projected in a grid cell, u, if in each attribute the value of v, is less than the right boundary of that attribute in u and greater than or equal to the left boundary of that attribute in u. When we consider exploration in high dimension, it can be assumed that relevant data points would exist in the close neighborhood [13, 16, 36]. Thus, the futuristic query formulation is pivotal by neighborhood exploration of each objects from previous queries. Neighborhood of each data object is initialized as a cluster of most probable results and achieved through sub-space clustering technique. A modified 'cluster-clique' algorithm is proposed for cluster/morph generation.

We assume that pre-computed d-dimensional sub space of data point divided into hyper rectangular cells is available. The query result retrieved after processing initial query traditionally is projected over this d-dimensional spatial representation of data. Identify the initial data object in space and consider them as a different unique cluster. Exploration and exploitation is performed in the neighborhood of every data objects to form cluster covering maximal region. The selectivity of a cell containing data points is defined to be the fraction of total data points in the cell. Only cells whose selectivity are greater than the value of model parameter τ are considered as dense and preserved. So, a cell is said to be a dense cell, if the fraction of total data point in that cell excesses input model parameter τ. The computation of dense cells applies to all subspaces of d-dimensional space. Identify neighboring dense cells that form a cluster containing data points at lower dimension. Cluster-clique holds cluster of dense cells at k-dimension also acquire similar projection at *(k-1)* dimension. The projection of subspace is considered from the bottom up to identify subspaces that contain clusters and to identify the dense cells to retain. A cell for giving projection subspace $S_t = A_{t1} \times A_{t2} \times \ldots, \times A_{tk}$ where $k < d$ and $the < t_j$, if $I < j$ is the intersection of an interval in each dimension. The proposed algorithm employs a bottom up scheme by leveraging the *Apriori* algorithm because monotonicity holds: if a group of data points is a cluster in a k-dimensional space, then this group of data points is also part of a cluster in any *(k-1)*-dimensional projections of this space.

The proposed algorithm employs a bottom up scheme by leveraging the *Apriori* algorithm because monotonicity holds: if a group of data points is a cluster in a *k*-dimensional space, then this group of data points is also part of a cluster in any *(k-1)*-dimensional projections of this space. The recursive step from *(k-1)*-dimensional cells to *k*-dimensional cells involves a self-join of the *k-1* cells sharing first common *(k-2)*-dimensions. Cluster-clique thins the collection of candidates to reduce the time complexity of the *Apriori* process, and keep only the set of dense cells to form clusters in the next level. The portion of the database enclosed by the dense cells is called its coverage. All the subspaces are sorted according to their coverage and less covered subspaces are eliminated to perform thinning. The selection of cutting point between removed and taken subspaces is computed using MDL principle in information theory. Two connected k-dimensional cells u_1, u_2 have either common face in the subspaces or another *k*-dimensional unit u_s exist such that both the cells u_1 and u_2 is connected to u_s. A maximal set of connected dense units in k-dimensions form a cluster. Computing clusters is equivalent to computing connected components in the graph where the dense cells represent the vertices and cells sharing common face endures edges between them. This can be computed in quadratic time of the number of dense cells in worst situation. After the identification of all the clusters, a finite set of maximal segment or region is specified by applying a DNF expression whose union forms the cluster. Finding the minimal descriptions for the clusters is equivalent to finding optimal cover of the clusters. By examining all dense units, clusters are formed at higher dimensions and derived as query morphs. Top n keywords from the relevancy list are selected for suggestion to user for query reformulation.

Proposed system first process initial query of user Q_i in traditional way and return initial data result objects O $\{o_{i1}, o_{i2} \ldots o_{in}\}$. Returned data objects are projected on pre-computed *d*-dimensional sub space of data point divided into hyper rectangular cells. The algorithm will consider these projected data objects as independent clusters C $\{c_{i1}, c_{i2}, \ldots, c_{in}\}$. Next, cells in proximity (neighborhood) are explored and exploited to form a larger cluster. The cell is dense means cells containing at least τ data point are merged to form such clusters C $\{c_1, c_2, .., c_n\}$ at lower dimension. After identifying clusters at 1-dimentional subspace we subsequently move further in higher dimensions. As per the monotonicity interesting clusters and c_2 at 1-dimension exist, then at 2-dimensional subspace they can be form a unique cluster c_{12} if their intersection is dense enough and we can drop low dimension clusters c_1 and c_2 cluster from the cluster set. Similar computation is performed subsequently at 3rd, 4th, 5th and up to d^{th} dimension to form higher dimension clusters. We stop once we retrieve all the clusters. Now, we consider each cluster from the cluster set as independent morphs of initial user query (Q_i). Top N relevant morphs from the set is recommended to the user formulation of succeeding exploratory queries.

An example: refer schema of the movie database, initial query variant (Q_{i+1}) and corresponding result set shown in the **Figure 5**. {D.name = "D.Fincher"} is 1-dimension subspace cluster. {G.genre = "Drama", 1990 < M.year<2009} is 2-dimention subspace cluster. We are finding interesting fragments of data from the clusters: that values can occur on single or multiple attribute means on 1-dimensional or k-dimensional subspace cluster. We are looking for interesting




```
SELECT D.name, M.title, M.year, G.genre
FROMD,M2D,M,G
WHERE D.name = 'D. Fincher'
AND D.directorid = M2D.directorid
AND M2D.movieid = M.movieid
AND M.movieid = G.movieid;
```

(b)

D.name	M.title	M.year	G.genre
D. Fincher	The Aviator	2004	Biography
D. Fincher	Gangs of New York	2002	Drama
D. Fincher	Goodfellas	1990	Biography
D. Fincher	Casino	1995	Drama
D. Fincher	Shutter Island	2004	Thriller
D. Fincher	M. Jackson: Video Greatest Hits	1995	Drama
D. Fincher	The Last Waltz	1978	Documentary
D. Fincher	Raging Bull	1998	Drama

(c)

Figure 5. (a) imdb movie database schema (b) variant of initial query Q_{i+1} and (c) Result set of query Q_{i+1}.

pieces of information at the granularity of clusters: this may be value of a single attribute (1-dimensional cluster) or the value of k attributes (m-dimensional cluster). User want to retrieve all movies directed by 'D. Fincher'. For that refer example query shown in **Figure 5(b)**. From the retrieved results we can say that movies with genre "Drama" are frequently directed by "D. Fincher" so user possibly concerned in movies with {G.genre = "Drama"}. Similarlly, for {G.genre = "Drama",1992 < M.year<2009}. Moreover we intend to retrieval of potentially relevant data that may satisfies user's information but may not part of result set retrieved originally from the initial query. Consider following exploratory/variant of initial query (Q_{i+1}):

(Q_{i+1}): SELECT *D.name*

FROM *G, M2D, D, M*

WHERE D.name*! = 'D. Fincher'*

AND *G.genre = 'Drama'*

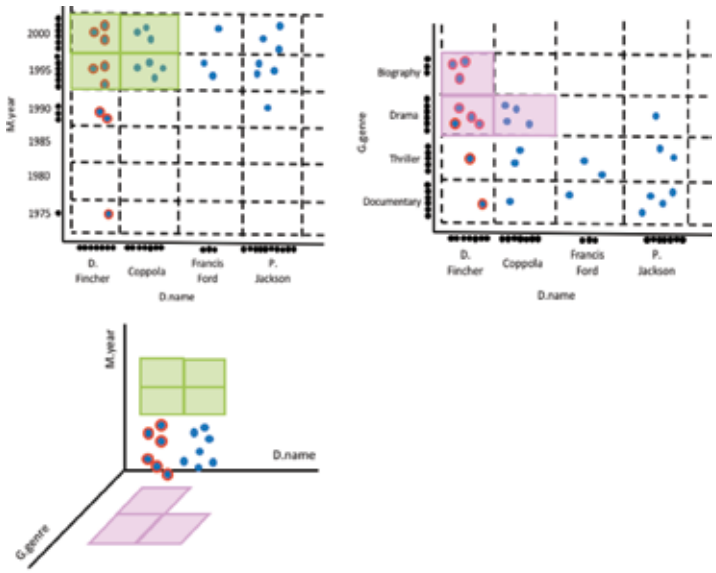
AND *D.directorid = M2D.directorid*

AND *D.directorid = M2D.directorid*

AND *M.movieid = G.movieid*

Retrieve movies of other directors who have directed *drama movies* too, by considering that results are of user's interest. In the disigned approach, subspace clustering is used to generate these query morphs/variants and interesting additional results from variant quires. The system will compute dataset (**Figure 5(c)**) of initial query shown in **Figure 5(b)** and project it on the space.

Initially, all data points are projected on the d-dimensional space and data points of initial query result are identified. These data points are treated as initial cluster and then the neighborhood is explored to retrieve the larger cluster. As shown in **Figure 6** algorithms perform exploration and form larger cluster by merging neighborhood cells who are dense enough. In movie database, axis-paralleled histograms are constructed for the year and genre at 1-dimention. After 1-dimention next is to steer towards higher dimensions, and at 2-dimension like {G.genre, D.name} and {G.year, D.name.} etc. as shown in **Figure 6(a)** and **(b)**. Neighborhood exploration is performed and clusters are constructed. After finding all the cluster a finite set



D.name	M.title	M.year	G.genre
D. Fincher	The Aviator	2004	Biography
D. Fincher	Gangs of New York	2002	Drama
D. Fincher	Goodfellas	1990	Biography
D. Fincher	Casino	1995	Drama
D. Fincher	M. Jackson: Video Greatest Hits	1995	Drama
D. Fincher	The Last Waltz	1978	Biography
Coppola	Youth without Youth	2004	Drama
Coppola	A Hollywood Love story	2001	Drama
Coppola	Supernova	2000	Thriller
Coppola	Lani Loa – The Passage	1998	Thriller
Coppola	The Rainmaker	1997	Drama
Coppola	Battle Beyond the Sun	1996	Drama
Coppola	Don Juan DeMarco	1994	Documentary

Figure 6. (a) Cluster formation on 2D space {D.name, M.year} (b) cluster formation on 2D space {D.name, G.genre} (c) cluster formation on 3D space {D.name, M.year, G.genre} (d) results set of generated morphs.

of maximal segment (*regions*) are computed using DNF expression whose union is a cluster shown in **Figure 6(c)** at higher dimension. Subsequently, move to 3rd, 4th, 5th …. dth dimension in search of relevant clusters. This consummates the exploration of each data subspace around the pertinent objects of anterior query. All the computed clusters are equivalent to the morph of initial/previous query. Morphs contains addition relevant results as well subset of originally retrieved results. The data items present in the morphs are dignified as relevant by standard measure to previous query and future probable search interest. Now based on implicit and explicit relevance top N morphs and set of relevant terms are suggested to the user. In our example, after computing relevance score using standard relevance measures we can say that query morphs containing movie genre 'Drama' and directed year >1995 scored higher number then morph with movie genre 'Thriller'. Hence, morphs with movie genre 'Drama' and year >1995 considered as high relevance. The system would also suggest top N terms from computed morphs like 'Coppola' based on relevance to the initial query as well result set. These terms help the user in formulating his next exploratory/variant query. As a next step, user may encounter shift towards different query results after reviewing result variants. The newly formulated query now surrounds both past and new variant of the user request.

4. Design issues and analysis

Many design issues are identified during the development of the propose solution which are as follow:

1. **Neighborhood selection and query morph generation**: The key challenge is identifying and defining the borderline for neighborhood of relevant data objects. Various researchers have addressed in their existing work. In Proposed algorithm, subspace clustering is used to define a non-overlapping boundary based on relevance of neighborhood objects. Each neighborhood region will be explored and exploited for extraction of keywords and phrases query reformulation. If the density of d-dimensional spatial cell is less than the threshold (τ), then cluster forming becomes a challenging task. Exploring cluster at higher dimension may also face issues like cluster overlapping, cluster size, number of clusters.
2. **Evaluation of relevant data objects and Top N morph suggestion**: Relevance is estimated to measure how closely data objects of different clusters are connected and also to define importance of the result items [39]. Identification of various information to define relevance criteria is one of the key challenges, as it influences overall system performance. In our approach, each cluster will be exploited as a region of user's interest and data objects will be extracted based on explicit and implicit relevance measure. Two key issues are identified during designing that are criteria selection for relevance and techniques for the computation of relevance score.
3. **Demonstration of additional information extracted from retrieved data objects through various visualization:** A visualization of entire result set with frequent terms is not a

feasible solution [9], for this various data summarization technique can be employed. For example, relevant terms from the morphs are suggested to user in a selective manner, so that user can use these keywords for intermediate query formation.

Fundamentally several adjustments can be made to perform the query reformulations, such as adding/removing predicates, changing constants, joining operation through foreign key relationships on auxiliary tables, etc. The kind of adjustment for creation of intermediate query may steer towards relevant result set in optimal processing cost. Query morphing technique is regulating proximity-based query reformulation due to neighborhood exploration characteristics. The ultimate goal is to morph the query that pulls user in a direction where information is available at low cost.

5. Conclusion

We proposed an algorithm for query reformulation using object's proximity, 'Query morphing' that mainly design to recommend additional relevant data objects from neighborhood of the user's query results. Each relevant data object of user query act as an exemplar query for generation of optimal intermediate reformulations. Multiple challenges are inferred during solution designing, includes: (i) neighborhood selection and Query morph generation (ii) Evaluation of relevant data objects and Top-K morph (iv) Evaluation of data object's relevance, (III). Demonstration of additional information extracted from retrieved data objects through various visualization. The discussed approach primarily based on proximity-based data exploration, and generalized approach of query creation with small edit distance. It could be realized with major adjustments to the query optimizer. The ultimate goal would be that morphing the query pulls towards the area where information is accessible at low cost.

Author details

Jay Patel and Vikram Singh*

*Address all correspondence to: viks@nitkkr.ac.in

Computer Engineering Department, National Institute of Technology, Kurukshetra, Haryana, India

References

[1] Andolina S, Klouche K, Cabral D, Ruotsalo T, Jacucci G. Inspiration wall: Supporting idea generation through automatic information exploration. In: Proceedings of the 2015 ACM SIGCHI Conference on Creativity and Cognition. ACM; 2015. pp. 103-106

[2] Abouzied A et al. Learning and verifying quantified boolean queries by example. In: Proceedings of the 32nd ACM SIGMOD-SIGACT-SIGAI Symposium on Principles of Database Systems. ACM; 2013. pp. 49-60

[3] Abouzied A, Hellerstein JM, Silberschatz A. Playful query specification with DataPlay. Proceedings of the VLDB Endowment. 2012;**5**(12):1938-1941

[4] Acharya S, Gibbons PB, Poosala V, Ramaswamy S. The aqua approximate query answering system. ACM SIGMOD Record. 1999;**28**(2):574-576. ACM

[5] Agarwal S et al. Knowing when you're wrong: Building fast and reliable approximate query processing systems. In: Proceedings of the 2014 ACM SIGMOD International Conference on Management of Data. ACM; 2014. pp. 481-492

[6] Agarwal S, Mozafari B, Panda A, Milner H, Madden S, Stoica I. BlinkDB: Queries with bounded errors and bounded response times on very large data. In: Proceedings of the 8th ACM European Conference on Computer Systems. ACM; 2013. pp. 29-42

[7] Ahn JW, Brusilovsky P. Adaptive visualization for exploratory information retrieval. Information Processing & Management. 2013;**49**(5):1139-1164

[8] Andolina S et al. Intentstreams: Smart parallel search streams for branching exploratory search. In: Proceedings of the 20th International Conference on Intelligent User Interfaces. ACM; 2015. pp. 300-305

[9] Author ZY, Gao K, Zhang B, Li P. Time Tree: A novel way to visualize and manage exploratory search process. In: International Conference on Human-Computer Interaction. Chicago: Springer International Publishing; 2016. pp. 313-319

[10] Beier T, Neely S. Feature-based image metamorphosis. ACM SIGGRAPH Computer Graphics. 1992;**26**(2):35-42. ACM

[11] Bonifati A, Ciucanu R, Staworko S. Interactive inference of join queries. In: Gestion de Données-Principes. Technologies et Applications (BDA); 2014

[12] Cetintemel U et al. Query steering for interactive data exploration. CIDR. 2013

[13] Chau DH, Kittur A, Hong JI, Faloutsos C. Apolo: Making sense of large network data by combining rich user interaction and machine learning. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems. ACM; 2011. pp. 167-176

[14] Cormode G, Garofalakis M, Haas PJ, Jermaine C. Synopses for massive data: Samples, histograms, wavelets, sketches. Foundations and Trends in Databases. 2012;**4**(1-3):1-294

[15] Dhankar A, Singh V. A scalable query materialization algorithm for interactive data exploration. In: Parallel, Distributed and Grid Computing (PDGC), 2016 Fourth International Conference on. IEEE; 2016. pp. 128-133

[16] Dimitriadou K, Olga P, Yanlei D. Explore-by-example: An automatic query steering framework for interactive data exploration. In: Proceedings of the 2014 ACM SIGMOD International Conference on Management of Data. ACM; 2014. pp. 517-528

[17] Drosou M, Evaggelia P. YmalDB: Exploring relational databases via result-driven recommendations. The VLDB. 2013;**22**(6):849-874

[18] Fan J, Li G, Zhou L. Interactive SQL query suggestion: Making databases user-friendly. In Data Engineering (ICDE), 2011 IEEE 27th International Conference on pp. 351-362. IEEE (2011)

[19] Glowacka D, Ruotsalo T, Konuyshkova K, Kaski S, Jacucci G. Directing exploratory search: Reinforcement learning from user interactions with keywords. In: Proceedings of the 2013 International Conference on Intelligent User Interfaces. ACM; 2013. pp. 117-128

[20] Hankins RA, Patel JM. Data morphing: An adaptive, cache-conscious storage technique. In: Proceedings of the 29th International Conference on Very Large Data Bases. Vol. 29. VLDB Endowment; 2003. pp. 417-428

[21] Hellerstein JM et al. Interactive data analysis: The control project. Computer. 1999; **32**(8):51-59

[22] Hellerstein JM, Haas PJ, Wang HJ. Online aggregation. In: Proceedings of the ACM SIGMOD Conference on Management of Data; 1997

[23] Idreos S, Papaemmanouil O, Chaudhuri S. Overview of data exploration techniques. In: Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data. ACM; 2015. pp. 277-281

[24] Kersten ML, Idreos S, Manegold S, Liarou E. The researcher's guide to the data deluge: Querying a scientific database in just a few seconds. PVLDB Challenges and Visions. 2011:3

[25] Klouche K et al. Designing for exploratory search on touch devices. In: Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems. ACM; 2015. pp. 4189-4198

[26] Li H, Chan CY, Maier D. Query from examples: An iterative, data-driven approach to query construction. Proceedings of the VLDB Endowment. 2015;**8**(13):2158-2169

[27] Psallidas F, Ding B, Chakrabarti K, Chaudhuri S. S4: Top-k spreadsheet-style search for query discovery. In: Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data. ACM; 2015. pp. 2001-2016

[28] Qarabaqi B, Riedewald M. User-driven refinement of imprecise queries. In: Proceedings of the International Conference on Data Endineering (ICDE); 2014

[29] Rocchio J. Relevance feedback in information retrieval. In: The Smart Retrieval System Experiments in Automatic Document Processing. Prentice-Hall Inc; 1971. pp. XXIII-1-XXIII-11

[30] Ruotsalo T, Jacucci G, Myllymäki P, Kaski S. Interactive intent modeling: Information discovery beyond search. Communications of the ACM. 2015;**58**(1):86-92

[31] Ruotsalo T et al. Directing exploratory search with interactive intent modeling. In: Proceedings of the 22nd ACM International Conference on Conference on Information & Knowledge Management. ACM; 2013. pp. 1759-1764

[32] Salton G, Buckley C. Improving retrieval performance by relevance feedback. Readings in Information Retrieval. 1997;**24**(5):355-363

[33] Sellam T, Kersten ML. Meet Charles, big data query advisor. Proceedings of the biennial Conference on Innovative Data Systems Research (CIDR). 2013;**13**:1-1

[34] Shen Y, Chakrabarti K, Chaudhuri S, Ding B, Novik L. Discovering queries based on example tuples. In: Proceedings of the 2014 ACM SIGMOD International Conference on Management of Data. ACM; 2014. pp. 493-504

[35] Singh V, Jain SK. A progressive query materialization for interactive data exploration. In: Proceeding of 1st International Workshop Social Data Analytics and Management (SoDAM'2016) Co-Located at 44thVLDB'2016. VLDB; 2016. pp. 1-10

[36] Stolte C, Tang D, Hanrahan P. Polaris: A system for query, analysis, and visualization of multidimensional relational databases. IEEE Transactions on Visualization and Computer Graphics. 2002;**8**(1):52-65

[37] White R, Muresan G, Marchionini G. Report on ACM SIGIR 2006 workshop on evaluating exploratory search systems. Acm Sigir Forum. 2006;**40**(2):52-60. ACM

[38] White R. Interactions with Search Systems. Cambridge University Press; 2016

[39] White RW, Roth RA. Exploratory search: Beyond the query-response paradigm. Synthesis Lectures on Information Concepts, Retrieval, and Services. 2009;**1**(1):1-98

[40] Yu JX, Qin L, Chang L, Ozsu MT. Keyword Search in Databases (Synthesis Lectures on Data Management). Morgan and Claypool Publishers; 2010

[41] Patel J, Singh V. Query morphing: A proximity-based approach for data exploration and query reformulation. In: International Conference on Mining Intelligence and Knowledge Exploration. Springer; 2017. pp. 261-273

Section 5

Cellular Automata Applications

Chapter 9

Cellular Automata and Randomization: A Structural Overview

Monica Dascălu

Additional information is available at the end of the chapter

http://dx.doi.org/10.5772/intechopen.79812

Abstract

The chapter overviews the methods, algorithms, and architectures for random number generators based on cellular automata, as presented in the scientific literature. The variations in linear and two-dimensional cellular automata model and their features are discussed in relation to their applications as randomizers. Additional memory layers, functional nonuniformity in space or time, and global feedback are examples of such variations. Successful applications of cellular automata random number/signal generators (both software and hardware) reported in the scientific literature are also reviewed. The chapter includes an introductory presentation of the mathematical (ideal) model of cellular automata and its implementation as a computing model, emphasizing some important theoretical debates regarding the complexity and universality of cellular automata.

Keywords: random number generator, complexity, universal computing model, cellular automata, self-programmable cellular automata, self-organization, synthesis of cellular automata, BIST

1. Introduction

Cellular automata (CA), as a massive parallel computing architecture of highest granularity, consist of a network of finite-state machines with only local interactions. The evolution of the system is based on the evolution of all its components. Starting from simple configurations and applying local (simple) rules, some cellular automata display a complex behavior.

The cellular automata model is connected to significant landmarks in the artificial intelligence domain, including the origins of the artificial life concept [1]. It is considered to be one of the main representatives of the so-called self-organizing artificial systems, together with neural

networks and genetic algorithms. These three models have also in common the natural inspiration, as each of them replicates some features or constructive principles of natural systems [2]. For the genetic algorithms, the natural analogy is the evolutionist idea of combining individuals with certain qualities to obtain a better individual, together with the survival of the fittest concept. For neural networks, it is the resemblance with the natural neural systems and the way biologic neurons transmit information combining the input stimuli. For cellular automata, it is the structural analogy—large systems consisting of simple elements having only local interactions, like a volume of gas particles.

A different manifestation of self-organization is associated with the three models [3]. For genetic algorithms, the self-organization is the process of solving an optimization problem, combining parts of solutions as in the analogy with the biological evolution. In the case of neural networks, the self-organization is associated with the learning algorithms (which may imply millions of coefficients for deep learning, convolutional neural networks). For cellular automata, the self-organization refers to the coherent global evolution, which sometimes displays patterns or regularities, in a system that does not have a central, global control.

One of the main features of cellular automata is considered to be their simplicity. However, the very first cellular automata, proposed by von Neumann and Ulam in the early 1950s had 29 states per cell. It was a mathematical model of an artificial system capable of self-reproduction [1] (able to construct a similar object). This concept of artificial self-reproduction opens the new horizon of the so-called artificial life, since reproduction is considered the main feature of the living beings.

Probably the most famous example of cellular automata is the Game of Life [4], invented by John Conway in 1970. This example of two-dimensional binary cellular automata is the best illustration for the main feature of cellular automata: a simple, regular structure displays a vast phenomenology, which may manifest a certain order starting from random states. In the Game of Life, the cells of two-dimensional cellular automata may have only two states (0 and 1, meaning “dead” or “alive”). In the next time step, the cells will live, die, or be born depending on the number of living neighbors in the present moment. The significance that Conway associated to the Game of Life evolution is only symbolic but became iconic for artificial life.

Like the whole domain of artificial intelligence, cellular automata have passed through periods of development and stagnation. Successful applications in modeling and simulation of complex processes, cryptography, image processing, etc., were developed and reported in the scientific literature (for a review of applications, see [5, 6]). Random number generation is one of the typical applications for cellular automata, with applications like built-in self-test (BIST) of integrated circuits and cryptography.

The rest of the chapter is organized as follows: Section 2 presents the mathematical (ideal) model of the cellular automata, the structure and phenomenology of the corresponding computing (real-life) model, and its applications. In order to study and use cellular automata, one should understand both the potential and limitations of the model—Section 3 discusses some of its fundamental issues (complexity vs. simplicity, universality vs. difficulty of synthesis, massive parallelism vs. sequential simulation).

http://dx.doi.org/10.5772/intechopen.79812

The following sections are dedicated to applications of cellular automata for random number generators and reviews the results reported in scientific literature. Section 4 gives some examples of cellular automata randomizers. Two trends have been identified: the use of the classical, regular model (often related to a theoretical approach in search of better statistical properties), and the alteration of the model, in order to obtain a more unpredictable evolution. Hence, an important topic is the modification of the model. Section 5 discusses the actual implementations of the cellular automata randomizers that were proposed by different authors and their performances. The chapter ends with a section of concluding remarks.

2. Cellular automata: the model

The presentation in this section is only a basic overview of concepts and features of cellular automata and is based on classical books on the topic like [2, 7, 8].

2.1. The model—ideal and real

The cellular automata model is defined as a combination of structure and functionality. The structure consists of finite automata and a network of interconnections between them. The functionality is completely defined by local rules that depend on local conditions, related to the type of interconnection network. Specifically, this means that there is no global control. The basic components (finite-state machines) are named cells. The next state of each cell is computed based on the present state of the cell and its neighbors (in the network). All cells are updated synchronously. **Figure 1** suggests how the next state is computed in linear cellular automata, for one cell. All cells' states are computed in a similar manner.

There are two assumptions for the ideal model: infinity and regularity. In reality, the architecture is finite and boundary conditions (functional and topologic) should be defined. Depending on these conditions, cellular automata may be also irregular, at least at the edges.

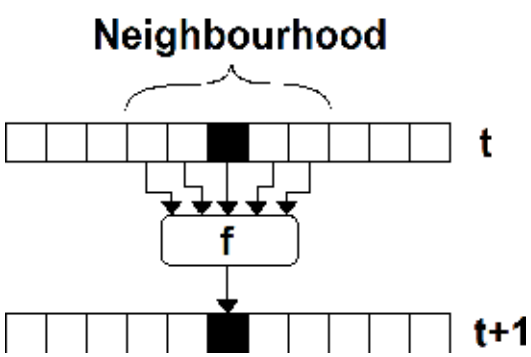


Figure 1. The computation of the next state for one cell, in linear cellular automata.

The most important features of the generic cellular automata computing model are: it is discrete in space and in time, it is finite and regular (except for the limit conditions), and it is parallel. The model can be regarded, in terms of computation theory, as an elementary single instruction multiple data (SIMD) architecture, since all cells perform identical operations [2]. The system is synchronous; therefore, in simulations, the algorithm that computes the next configuration can update cells' states in any order. The local rules, or functions, or laws, are deterministic, implying a global deterministic evolution (which is, however, often hard to predict).

The complexity of the resulting structure, as a digital system, is depending on the number of cells' states, the number of cells (the dimension), and the dimension of the neighborhood, related to the type of interconnection network [9].

2.2. Definitions of specific terms

Before going further, here is a short list of specific terms related to cellular automata that will be used in this chapter:

- cell—the elementary computing element, which is a finite-state machine (**Figure 2**),
- binary cellular automata—each cell has only two states, encoded on 1 bit,
- dimension of cellular automata—total number of cells,

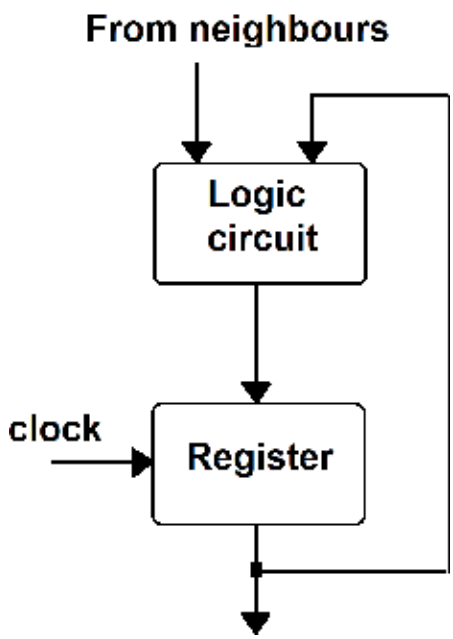


Figure 2. The cell as a finite-state machine connected to the cells in the neighborhood.

- neighborhood—the set of cells, which are involved in the computation of the next state (see **Figures 1** and **2**),
- neighborhood dimension—the number of cells involved locally in the computation of the next state value,
- von Neumann and Moore neighborhood—in two-dimensional rectangular topologies, these types of neighborhoods contain the closest four neighbors (van Neumann), or eight neighbors (Moore), plus the central cell
- limit conditions (or boundary conditions) refer to the interconnection of the cells at the edges
- cyclic boundary conditions—the opposite edges are adjacent,
- rule—the function performed by each cell,
- rule 30—one of the most studied evolution rules, introduced by Wolfram in [10],
- configuration—vector containing all cells' states,
- seed—the initial configuration or initial state,
- trajectory—evolution in the configurations' space,
- basin of attraction—graph representation that illustrates the cyclic trajectory in the configurations' space,
- programmable cellular automata—in the context of an implementation, it allows the modification of the rule or/and topology.

A frequently used denomination of local rules was introduced by Wolfram in [8]. In the context of linear binary cellular automata with neighborhood dimension of three, each rule (which is a Boolean function with three variables) is designated by a decimal number, equal to the binary number obtained from the look-up table of the function. See Appendix 1 for most frequently used rules.

In applications, the synthesis of cellular automata implies to define the particular topology (number of states per cell, number of cells, type of interconnection network, dimension of neighborhood, border conditions), the local rules, timing conditions, and the seed, in order to obtain the desired functionality. For instance, in image processing applications, the initial configuration is the image. The local rules are established in order to obtain the desired function (for instance, edge detection). The automata will run for a certain number of states or until it reaches a stable configuration.

In the previous example, the input data are the initial configuration, the output data are the final configuration, and the computation related to the image processing task is done by the evolution of the global state, through local changes. Computation with cellular automata may also be considered in terms of propagation and combination of patterns, in an analogy with propagation of signals and logical combination of inputs.

2.3. Taxonomy of cellular automata

The classification of cellular automata is based either on structure (topology) or behavior (function and/or phenomenology).

The topology refers mainly to the type of network and local connections (neighborhoods and boundary conditions). In linear cellular automata, the cells are connected in a row (vector) to their nearest neighbors. Further subdivision of linear cellular automata is based on the neighborhood dimension, which is one of the main factors that affect the complexity of the cell. In two-dimensional cellular automata models, the interconnection network is two dimensional, typically rectangular, but also hexagonal networks have been explored for specific applications. However, topologically any two-dimensional network can be transformed in a rectangular one, by choosing an appropriate neighborhood [5]. In typical rectangular connections, there are two most used neighborhoods: von Neumann neighborhood which contains the four adjacent cells on the vertical and horizontal lines, and the central cell itself; and Moore neighborhood that contains the lateral neighbors, the central cell, and the cells adjacent at corners (**Figure 3**).

The theoretical analysis of two-dimensional cellular automata is an open field of research, and most often, the results are extensions of the better-known case of linear automata. Two-dimensional cellular automata are very important in applications, as for instance in image processing, where the image corresponds directly to the configuration of the system. Most of modeling applications also involve two-dimensional extensions, therefore, use two-dimensional cellular automata [11].

2.4. Phenomenology of cellular automata—Wolfram's taxonomy

Probably the most important contribution in understanding cellular automata capabilities as a computer model is Wolfram's experimental approach [8]. Wolfram explored the behavior of linear cellular automata for all rules (with a neighborhood of dimension 3), starting from different "seeds" or initial states. Wolfram established a classification based on the statistical

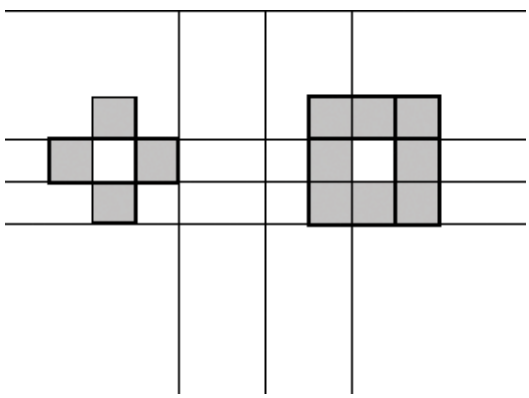


Figure 3. Neighborhoods in two-dimensional cellular automata: von Neumann (left) and Moore (right) neighborhoods.

properties of the linear cellular automata evolution, which is particularly important from the perspective of applications of cellular automata in generation of pseudorandom sequences.

Class 1. Cellular automata that evolve to a homogenous finite state. Automata belonging to this first class evolve from almost all initial states, after a finite number of steps, to a unique homogenous configuration, in which all cells have the same state or value. The evolution of class 1 cellular automata completely "destroys" any information on the initial state.

Class 2. Cellular automata that have a periodical behavior. The configurations are divided in basins of attraction, the periodical evolution of the global configuration depending on the initial state. The states' space being classified in attractor cycles; such cellular automata can serve as "filters."

Class 3. Cellular automata exhibiting chaotic or pseudorandom behavior. This class is particularly significant for ideal (infinite) cellular automata. Cellular automata belonging to the third class evolve, from almost all possible initial states, not periodically, leading to "chaotic" patterns. After an enough long evolution (number of steps), the statistical properties of these patterns are typically the same for all initial configurations, according to Wolfram. Although in practice cellular automata will always be limited to a finite number of cells, this class can be extrapolated maintaining the local rules. Such automata can be used for pseudorandom pattern generation [8, 10].

Class 4. Cellular automata having complicated localized and propagating structures. They can be viewed as computing systems in which data represented by the initial configuration are processed by time evolution, the result being contained in the final configuration of attractor cycle of configurations.

In conclusion, cellular automata are systems consisting of a regular network of identical simple cells that evolve synchronously according to local rules that depend on local conditions. Although the structure is very simple and regular, it produces a vast phenomenology, and therefore, cellular automata are often described as an "artificial universe" that has its own specific local laws [1].

3. Theoretical and practical issues

This section discusses specific issues of cellular automata that one should be aware before deciding to use or investigate this model.

3.1. Parallelism: hardware vs. simulation

Massive parallelism is one of the definitory features of cellular automata. However, simulations are often used instead of actual implementations, and for very good reasons: there are no commercially available hardware versions, hardware accelerators, or co-processors for the cellular automata computational model. The full performance of the massive parallel architecture will never be reached by simulation; in fact, the high granularity will make the simulations slow, even for small dimensions.

There are several hardware implementations for cellular automata reported in the scientific literature (for a recent review, see [12]). Most of them are particular implementations for specific applications. The so-called "cellular automata machines," including CAM (Cellular Automata Machine, project started in the 1980s at MIT [13]) and CEPRA (Cellular Processing Architectures, first prototype in the 1990s at the Darmstadt University, in Germany [14]) never reached industry. Both projects combine serial processing and pipeline techniques to emulate the parallelism of the cellular automata architecture.

Hence, the first paradox of cellular automata: a model of massive parallelism, reduced to sequential computation.

3.2. The problem of synthesis

The synthesis of cellular automata implies to establish the structure and functionality (including the states per cell, the dimension, the topology, and the local rule) that may solve a certain problem or perform a specific computation. This problem of synthesis is the root problem of massive parallel computing the decomposition of a computing problem in elementary tasks to be performed repeatedly by a large number of cells for a certain number of cycles. For cellular automata, the problem of synthesis was approached, for instance, from the perspective of evolutionary computing (similar with genetic algorithms) [15], but still remains the biggest challenge of cellular automata applications.

In terms of computation theory, Wolfram considers that the model of cellular automata is universal, which means that it can solve any computable problem (claiming the equivalence with the Turing machine [16]). There is no general acceptance of the computing universality of the model, but particular rules were proved to be computational universal [16–18], meaning that they can perform any function (which, again, does not mean that they can solve any computing problem). In spite of such theoretical results, a huge challenge still remains: there is no algorithm or method for synthesis of particular cellular automata.

This is the second paradox of cellular automata: its computing capability is not reflected in applications, as there is no synthesis method. As massive parallel computing systems, cellular automata seem ideal for hardware implementation. Theoretically, they can compute any function; however, the difficulty of synthesis was a major drawback in the development of applications.

3.3. Complexity and self-organization

In the introductory section, we have mentioned two features of the cellular automata model: the self-organization and the complexity. Depending on the context, different authors looked at cellular automata from both perspectives: as complex systems that, starting from a random initial state, manifest the self-organization property, or as simple, regular systems that exhibit a very complex, random behavior [1, 4, 7, 8].

In [19], a possible reconciliation of the two perspectives includes them both in the concept of "apparent complexity," understood as a "complex phenomenal appearance backed by a structural simple generative rule" (in order to be efficiently used, the term "apparent complexity"

should be further, more clearly, defined). The same evolution or pattern may appear complex or simple, depending on the perspective of the analysis. In the particular case of linear cellular automata, of one does not know the rule and the initial state, it is difficult to infer them by analyzing the behavior. We recognize here again the problem of synthesis, as this issue is directly related to the previous one.

This dichotomy complexity—self-organization is the third paradox of cellular automata.

3.4. Variations of the model

The last important issue is related to the fact that the actual field of cellular automata research, particularly in application development, has adopted a lot of variations of the ideal model (for which the theory was developed). We will mention the most important ones, those who are significant for the topic of this chapter. For a detailed overview, see [3].

Hybrid or inhomogeneous cellular automata: either the cellular space is inhomogeneous (different structures of neighborhoods and cell properties, topological variations), or local rules vary in the cellular space. Local rules may also be modified after a number of time steps, in order to obtain a specific processing of the global configuration. Block hybrid cellular automata are a particular case of inhomogeneous cellular automata. The cellular structure is here divided in homogenous subdomains or blocks.

Automata with structured states' space: the cell's state is considered to be the combination of some significant parameters or state values. As in the case of finite-state machine design, such structuration leads to a global simplification, mainly regarding the dimension of the local rules space.

Multilevel cellular automata: a more complex model, built as a hierarchical structure of interconnected cellular automata (the model is not simply a multidimensional structure).

Self-programmable cellular automata: the local rules change in time, depending on the evolution, and may be implemented as multilevel cellular automata.

Asynchronous automata: the cells' states are updated in a certain order, taking into account the new values obtained for the neighboring cells already updated.

Cellular automata with supplementary memory layer: the computation of the following states takes into account the "history" of the system, or a number of previous states of each cell. These previous states are loaded in the memory layer. This model is practically a network of elementary processors.

Nondeterministic and probabilistic cellular automata: the next state is established in a nondeterministic manner or according to a certain distribution of probability. Due to its versatility, this model can be successfully used in complicated modeling applications.

In composite systems, the basic cellular automata model is considered as a space that interferes with autonomous mobile structures or agents that evolve in the cellular space. This model simplifies some modeling effort for example in the case of particles diffusion.

The list above is not exhaustive but gives an image of the versatility of cellular automata developments, starting from the very restrictive, regular, infinite ideal model. On the other extreme, there are random Boolean networks, often called n-k networks that allow any inhomogeneous set of rules and also random connectivity.

A global theory of all models derived from cellular automata does not exist by now and probably will never exist. Because most of the applications in this field are derived empirically and also empirically tested, this is not truly a weak point, as it can be compensated by appropriate experiments.

4. Random number generation with cellular automata

Randomization is essential for several artificial intelligence applications: cryptography, data mining, games, integrated circuits verification, etc. Cellular automata have been proposed and used as pattern generators such as: universal pattern generators, pseudorandom signal generators, generation of numerical sequences with predefined statistic properties, BIST generators for integrated circuit's verification [6].

The block scheme of a pseudorandom signal generator is represented in **Figure 4**. The output is considered to be pseudorandom since it is obviously produced by an algorithm (transformation), starting from the initial state (seed). The output sequence appears to be random for someone who does not know the generation algorithm, meaning that the previous number in the sequence cannot be computed, nor the next numbers predicted, based on the present output.

The apparent complexity of cellular automata phenomenology makes them useful tools for pseudorandom number/signal generators (frequently, the word "pseudo" is skipped). In this section, we will present such some examples of cellular automata random number generators reported in scientific literature, while next section will discuss their properties, implementation, and applications.

4.1. Examples of cellular automata pseudorandom number generators

Based on the phenomenology of cellular automata introduced by Wolfram in [8], the class 3 linear, binary automata were proposed as pseudorandom generators [10]. There are basically

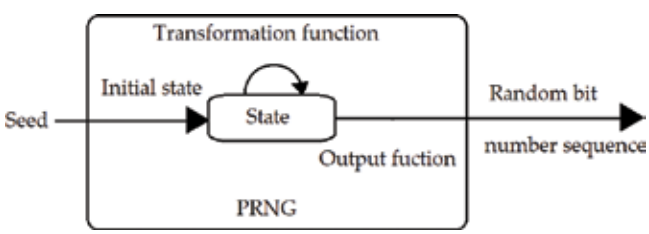


Figure 4. Block scheme of a pseudorandom signal/number generator (PRNG).

two ways to consider the output of such a generator, either as a one-bit digital signal (given by the evolution of the state of one particular cell), or an n-bit output (a subset of the global configuration). The statistical properties depend both on the rule and on the initial state, and the choice of the n-bit vector (because of intrinsic correlation of neighboring cells).

Note that all cellular automata will eventually enter into an attraction cycle—the length of the trajectory is related not only to the rule, the initial state but also to the dimension of the automata. This is why the "infinity" assumption often is mentioned in theoretical demonstrations regarding statistic properties of cellular automata randomizers. For better properties, larger automata will be used.

Wolfram analyzed the properties of several rules and proposed a generator based on the so-called "rule 30," for a one-bit signal generated by the central cell [10]. **Figure 5** presents the results of the simulation of a 64-cells linear cellular automata with rule 30 (Verilog HDL simulation), for 25 time steps.

Further research was oriented in discovering better rules and initial configurations (seeds) for improving the statistical properties of the sequence. A simple solution in this direction implies the use of two different rules, thus obtaining heterogenous cellular automata. Hortensius et al. proved in [20] that the results obtained with heterogenous rules (two different local rules) are better than in homogenous automata using either of those rules. Commonly used rules are listed in **Appendix 1**.

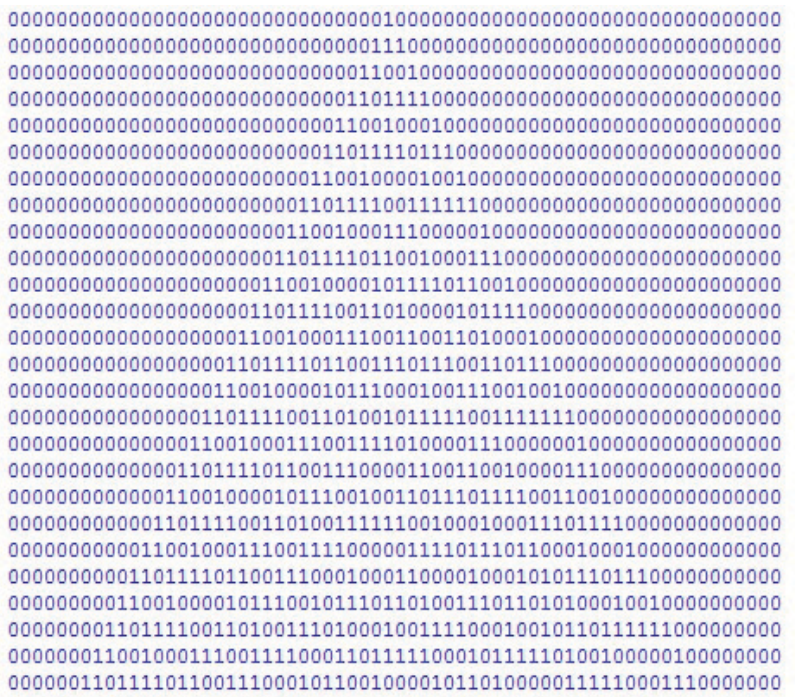


Figure 5. The output of a 64-cells linear cellular automata with rule 30 (Verilog HDL simulation).

Sipper and Tomassini [21] included two innovative improvements: the selection of rules based on an evolutionary algorithm (four rules were selected, namely 90, 105, 150, and 165) and the so-called "time spacing" of the sequence, meaning that the output sequence skips some time steps. This technique improves the quality of the sequence, according to the authors, as compared to classical random number generators.

In a later work, Tomassini and Sipper also proposed two-dimensional cellular automata [22] in order to improve the complexity of the evolution of the generator. Another two-dimensional cellular automata for pseudorandom sequence generation was presented in [23] and was applied by the authors in BIST and cryptography hardware implementations [6]. The difference is mainly quantitative in a two-dimensional topology, but also the neighborhood dimension is higher (and the rules more complex).

One-dimensional topology is simpler from the perspective of hardware implementation. Still theoretical research is done on linear cellular automata in order to discover binary functions or class of functions that may be used in random sequence generation. In [24], the authors prove that uniform linear cellular automata with nearest neighbor's connections can meet cryptographic requirements for random sequences (The selection of rules and seeds was done empirically.)

The examples given so far illustrate the theoretical and experimental effort to prove the properties of cellular automata and to discover significant combinations of rules and initial states that lead to "good" pseudorandom sequences. All these examples are based on the basic cellular automata model. The only modification of the model is, for some of these examples, the heterogeneity, but this is restricted to the implementation of two/four rules in a static manner to subsets of cells.

4.2. Examples based on variations of the cellular automata model

Another important trend of research is to improve the properties of the pseudorandom sequence introducing more substantial modifications to the cellular automata model, in order to obtain a more complex structure and expected behavior. A first example, presented in [9], uses a homogenous programmable cellular automata of 256 cells, with a global feedback loop. Each cell supports any of the 256 possible rules, but the actual rule is selected by the feedback (depending on the global state, with a simple transformation like module 2 sum). Such an approach is now called self-programmable, since the rule is not changed externally, but here the global reaction acts like a central control. According to the author, the initial state influences the properties of the generator.

Another structural modification of the cellular automata model is presented in [25]. Here, an additional memory layer was used to improve the complexity of the evolution, and the next state is computed based on the present state of the neighborhood and the previous state of the cell (or the entire neighborhood). The local rules are 4-variable functions, in the simplest form.

In physical implementations, the actual dimension of the cellular automata is reduced. Several authors proposed randomization schemes that improve the quality of the generators (to compensate the limited dimension). In [26], the output of the hybrid cellular automata is combined

with the output of a linear feedback shift register (LSFR) to produce a 1-bit random output using 32-bit cellular automata. A combination of rules 90 and 150 was applied in the cellular automata.

The technique of self-programming, meaning that the local rules are changed during the evolution, was presented in [27]. The local rules are changed based on local conditions (in [9], the global configuration determines the rule). The result is a hierarchical structure of cellular automata. Each cell can switch between two rules (90 and 165, 150, and 105 were used), depending on a "super rule" based on the state of a larger neighborhood (up to 7 cells). The idea of the "super rule" is to interrupt static behavior (the patterns that appear in low-quality pseudorandom signals).

The idea of self-programmable cellular automata, with a hierarchical structure, and a behavior based on local conditions, was developed in several ways since its introduction in [27]. For instance, [28] presents a combination of this technique in hybrid cellular automata, with a linear feedback shift register. The global feedback is used in [29] in the generation of the super rule with the additional selection of the cell whose rule is changed. In [30], the self-programmable cellular automata are optimized for field programable gate arrays (FPGA) implementation, using the splittable look-up table concept and embedded carry logic of the FPGA.

Other variation of the model is the asynchronous update of the cells' states. There are several examples in the scientific literature, of which we mention [31] because it is one of the few examples that use two-dimensional cellular automata (for 1-bit output) and also because it is designed as a randomization algorithm, not targeting the hardware implementation.

In conclusion, several variations of the cellular automata model were developed. The validation of the models is often experimental, in absence of theoretical developments.

5. Implementation and applications of cellular automata randomizers

In this section, we will make some qualitative comments regarding the implementation of cellular automata randomizers, their applications, and the evaluation of their performances.

Compared to other applications of cellular automata, random number generation eludes the problem of synthesis as explained in Section 3. Most of the research is done (and confirmed) empirically, including the selection of functions, seed, and other implementation details. In spite of the theoretical efforts to prove the properties of the classical cellular automata and the good results obtained by several researchers [10, 24], most of the reviewed papers include some variations of the basic model in order to obtain better statistic properties. One of the reasons may be the fact that the "infinite dimension" assumption is the guarantee of good random sequence generation with cellular automata. The performances depend both on the local rules and initial state. Particular cases when specific rules produced either excellent or poor, depending on the seed, are reported in the scientific literature.

Both linear and two-dimensional cellular automata were proposed in scientific literature for pseudorandom number generation. Linear topology is preferred for hardware implementation, although the two-dimensional topology would take the most advantage of the parallel computation.

As an alternative to the mathematical, ideal cellular automata model, several authors introduced variations. The most frequently used models include nonhomogenous cellular automata (with maximum four different rules performed by subsets of cells) and self-programmable cellular automata (that can be considered either as a hierarchical structure or as two parallel cellular automata). Other techniques that impact the apparent complexity use larger neighborhoods in the next state computation, or previous states.

The performances of a random number generator are determined by the statistical properties of the output, although there are other important properties like the dimension of sequence, speed, and costs of implementation. Since random number generators produce sequences that are not really random (being based on an algorithm or a process), their "randomness "is established by several statistical criteria that try to evaluate how difficult is to predict the evolution or compute the previous values.

One of the first systematic approaches to randomness evaluation was done by Knuth [32] who suggested a series of statistical tests (for instance, the frequency test, the autocorrelation or serial correlation test, the run test, etc.). Marsaglia [33] added several tests and launched a battery of 15 statistical tests in 1995. The new state of the art, since 2010, is established by the NIST test suite [34], specially designed for cryptographic applications.

There is no comparative study (and this is over the intention of this chapter) of the performances of all these alternatives presented by different authors. As a general remark, their evaluation is often done at the state-of-the-art level of the time when they were developed. Before the introduction of the Diehard test suite [33], the authors performed their own tests (see, for instance, [8, 20, 21]). Since the late 1990s, the Diehard tests were used to assess the performances of cellular automata randomizers (like in Refs [26–28]). Most of the recent work evaluates the performances based on NIST test suite [24, 30, 31].

Performances should be related to the application of the generators. Most common applications reported in scientific literature are integrated circuit verification (as in BIST pattern generators) and cryptography. Therefore, the performances should be evaluated according to the application. For BIST pattern generators, a comparison with LFSR is a common practice and is sufficient [6] (since LFSR is one of the consecrated methods for hardware random number generation). For cryptography, there are more strict requirements, both for hardware and software versions of random number generation. On the other hand, BIST applications are almost intended for hardware implementation [6, 26, 28].

Figure 6 presents a consecrated architecture for cryptography with cellular automata [25]. The initial message is combined with the cellular automata output to obtain the encrypted message (frequently used transformation is the exclusive or, XOR, function). In the decryption stage, the encrypted message is again combined with the output of identical cellular automata. The key is, in this scheme, the initial state (sometimes combined with the local rule, if several rules are available).

Comparing the hardware implementations of cellular automata reported in the scientific literature, one can notice the positive impact of the development of the FPGA technology,

http://dx.doi.org/10.5772/intechopen.79812

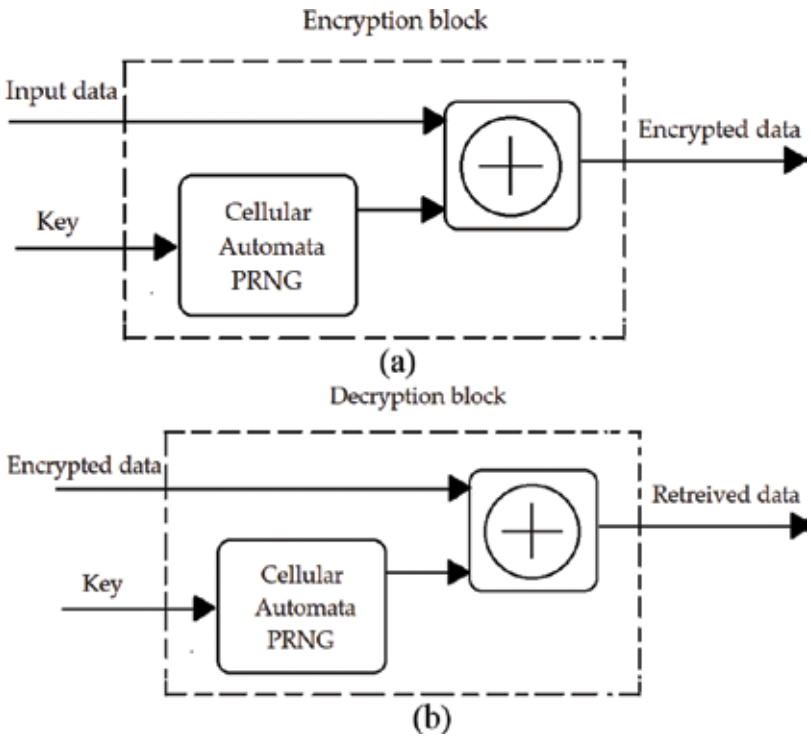


Figure 6. Encryption and decryption schemes in a cellular automata-based cryptographic system.

including for the hardware versions of cellular automata randomizers. Prior the FPGA implementations, the most common used rules were combinations of XOR functions, while the FPGA, look-up table (LUT)-based technology makes possible the implementation of all logic functions with same resources. On the other hand, some applications target application specific integrated circuits (ASIC) circuits (this is the case for BIST generators).

6. Conclusions

Cellular automata are one of the iconic models of massive parallelism, and their development is related to the artificial intelligence and artificial life domains. In absence of an effective hardware implementation, one can say that the model never reached its full potential as a massive parallel machine. Another drawback is the difficulty of synthesis of cellular automata that perform a specific global processing task, based on elementary computations at cell level. However, as dynamic systems with a vast phenomenology, they are an important direction of research in the theory of complexity, efficient modeling, and simulation tools and as the examples in this review have proved, good random number generators. The applications of

cellular automata in random number generation are based on their apparent complex behavior. Although the direct process is very simple, it is not reversible—the complexity of the inverse determination grows exponentially.

Random number generation is one of the successful applications of cellular automata, particularly for cryptography and verification of circuits (BIST generators). This chapter presents a selection of the results presented in the scientific literature. Two main directions of research were identified, based on these examples: the mathematical approach to demonstrate the properties of particular rules and configurations, and the engineering and empiric approach to develop new architectures. Some examples are oriented toward "classic," close to ideal model of cellular automata, while others include consistent modifications at structural and functional levels. Both hardware and software cellular automata randomizers are reported in the scientific literature. The development of the FPGA technology offers efficient solutions for hardware versions, when needed.

Acknowledgements

I want to thank the editor for the suggestions that very much helped to improve this chapter. Also, I would like to thank Eduard Franti for useful remarks and support.

Conflict of interest

The author declares no conflict of interest.

Appendix 1. Frequently used local rules

This appendix presents the nomenclature conventions (**A 1**) for local rules, frequently used local rules (**A 2**) and some rules invoked in the discussion of Turing equivalence of cellular automata.

The denomination of rules introduced by Wolfram [8] refers to 3-bit Boolean functions. Each function is designated by the configuration in the look-up table, read in reverse (hence, in this context, the look-up table is drawn starting with the last value). In **A 1**, the look-up table of the

Present state	111	110	101	100	011	010	001	000
Next state (of the central cell)	0	0	0	1	1	1	1	0

Here, rule 30.

A 1. Each 3-bit rule is designated by the value in the look-up table.

Rule	LUT configuration	Examples
30	00011110	[10]
90	01011010	[20, 21, 26, 27]
105	01101001	[22, 27]
150	10010110	[20, 21, 26, 27]
165	10100101	[22, 27]

A 2. Frequently used local rules for random number generation with linear cellular automata.

rule 30 is given. **Table 2** gives the specific configuration in the look-up table for some of frequently used rules cited in this paper, according to the references list.

In the basic rule in the Game of Life binary, two-dimensional cellular automata, the significance of 0 and 1 is that a cell is dead or alive. With a Moore neighborhood, the rule can be described as follows:

- if a cell is alive, it remains alive in the next step only if it has two of three living neighbors,
- if a cell is dead, it becomes alive only if it has three living neighbors,
- a living cell with less than two neighbors alive will die of loneliness,
- a living cell with more than three neighbors alive will die overcrowded.

In the Game of Life, there are several patterns that persist for ever, either in a dynamic or static manner. For instance, gliders are patterns that are moving. The demonstration of the Turing equivalence for the Game of Life involves flows of gliders that perform logic functions.

Rule 110, in linear cellular automata (binary code 01101110), is a class IV rule in Wolfram's taxonomy and, like Life, is Turing equivalent.

Author details

Monica Dascălu

Address all correspondence to: monica.dascalu@upb.ro

Research Institute for Artificial Intelligence, Politehnica University of Bucharest, Bucharest, Romania

References

[1] Coveney P, Highfield R. Frontiers of Complexity, the Search for Order in a Chaotic World. London: Faber and Faber; 1995

[2] Garzon M. Models of Massive Parallelism. Berlin: Springer-Verlag; 2012

[3] Dascalu M. Self-Orgnizing Systems. Bucharest: Politehnica Press; 2016. pp. 101-116

[4] Conway J. The Game of Life. Vol. 1970. Scientific American Magazine. pp. 120-123

[5] Dascalu M, Franti E. Automate Celulare – Modelare si Aplicatii. Bucharest: Editura Tehnica; 2009. pp. 119-125

[6] Chaudhuri P, Chowdhury D, Nandi S, Chattopadhyay S. Additive Cellular Automata: Theory and Applications. Vol. 1. Computer Society Press; 1997

[7] Gutowitz H. Cellular Automata and the Sciences of Complexity, I and II. Complexity; 1996;**5**:16-22, **6**:29-35

[8] Wolfram S. Statistical properties of cellular automata. Reviews of Modern Physics. 1983; **58**:601-644

[9] Stefan G. Looking for the lost noise. In: Proceedings of the Semiconductor Conference (CAS'98); 02-04 October 1998. Vol. 2. Sinaia: IEEE; 1998. pp. 579-582

[10] Wolfram S. Random sequence generation by cellular automata. Advances in Applied Mathematics. 1996:123-169

[11] Weimar J. Simulation with Cellular Automata. Berlin: Logos Verlag; 1999

[12] Dascalu M. Cellular automata hardware implementations - an overview. Romanian Journal of Information Science and Technology. 2016;**19**(4):360-368

[13] Toffoli T, Margolus N. Cellular Automata Machines: A New Environment for Modeling. Cambridge, USA: MIT Press; 1987

[14] Halbach M, Hoffmann R. FPGA Implementation of Cellular Automata Compared to Software Implementation. In: Proceedings of the Organic and Pervasive Computing Workshops (ARCS'04); 26 March 2004; Augsburg. Germany

[15] Mitchell M. Computation in cellular automata: A selected review. In: Scuster, Gramms, editors. Nonstandard Computation. Weinheim: VCH Verlagsgessellschaft; September 1996

[16] Wolfram S. Cellular automata as models of complexity. Nature. 1984;**311**:419-424. DOI: 10.1038/311419a0

[17] Sutner K. Universality and cellular automata. In: Margenstern M, editor. Machines, Computations, and Universality. MCU 2004. Lecture Notes in Computer Science. Springer-Verlag; 2004. p. 3354

[18] Lena Di P, Margara L. Computational complexity of dynamical systems: The case of cellular automata. Information and Computation. 2008;**206**(9–10):1104-1116

[19] Malita M, Stefan G. Real complexity vs. apparent complexity in Modeling social processes. Romanian Journal of Information Science and Technology. 2013;**16**(2–3):131-143

[20] Hortensius PD, McLeod RD, Card HC. Parallel random number generator for VLSI systems using cellular automata. IEEE Transactions on Computers. 1989;**38**(10):1466-1473

[21] Sipper M, Tomassini M. Co-evolving parallel random number generator. In: Voigt HM, Embeling W, Rechenberg I, Schwefel HP, editors. Parallel Problem Solving from Nature IV (PPSN IV); 1141. Lecture Notes in Computer Science. Heidelberg: Springer-Verlag; 1996

[22] Tomassini M, Sipper M, Perrenoud M. On the generation of high-quality random numbers by two-dimensional cellular automata. IEEE Transactions on Computers. 2000;**49**:1146-1151

[23] Chowdhury DR, Gupta IS, Chaudhuri PP. A class of two-dimensional cellular automata and applications in random pattern testing. Journal of Electrical Testing: Theory and Applications. 1994;**5**:65-80

[24] Ping P, Xu F, Wang ZJ. Generating high-quality random numbers by next nearest-neighbor cellular automata. In: Proceedings of the 2nd International Conference on Systems Engineering and Modeling (ICSEM-13). Paris, France: Atlantis Press; 2013

[25] Franti E, Dascalu M. More security and autonomy for users: Encryption system with evolutive key. In: Data, Text and Web Mining and their Business Applications. WIT Press; Cambridge Printing; 2007. pp. 335-344

[26] Tkacik TE. A hardware random number generator. Cryptographic Hardware and Embedded Systems. 2003;**2523**:450-453

[27] Guan S, Tan SK. Pseudorandom number generation with self-programmable cellular automata. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems. 2004;**23**(7):1095-1101

[28] Comer JM, Cerda JC, Martinez CD, Hoe DHK. Random number generators using cellular automata implemented on FPGAs. In: Proceedings of the 44th Southeastern Symposium on System Theory (SSST'12); Jacksonville; FL. 2012. pp. 67-72. DOI: 10.1109/SSST.2012.6195137

[29] Mocanu D, Gheolbanoiu A, Hobincu R, Petrica L. Global feedback self-programmable cellular automaton random number generator. Technical Journal of the Faculty of Engineering. 2016;**39**(1):1-9

[30] Petrica L. FPGA optimized cellular automaton random number generator. Journal of Parallel and Distributed Computing. 2018;**111**:251-259

[31] Bilan S, Bilan M, Bilan S. Research of the method of pseudo-random number generation based on asynchronous cellular automata with several active cells. In: Proceedings of the 21st International Conference on Circuits, Systems, Communications and Computers (CSCC 2017); 04 October 2017. 2017. DOI: 10.1051/matecconf/201712502018

[32] Knuth DE. The Art of Computer Programming, Vol. 2: Seminumerical Algorithms. 2nd ed. Boston: Addison-Wesley; 1981

[33] Marsaglia G. Diehard Test Suite. [Internet]. 1998. Available from: https://web.archive.org/web/20160125103112/http://stat.fsu.edu/pub/diehard/ [Accessed: 2018-05-19]

[34] Rukhin A, Soto J, Nechvatal J, Smid M, Barker E, Leigh S, Levenson M, Vangel M, Banks D, Heckert A, Dray J, Vo S. A Statistical Test Suite for Random and Pseudorandom Number Generators for Cryptographic Applications. [Internet]. 2010. Available from: https://nvlpubs.nist.gov/nistpubs/legacy/sp/nistspecialpublication800-22r1a.pdf [Accessed: 2018-05-19]

Chapter 10

Hard, firm, soft … Etherealware: Computing by Temporal Order of Clocking

Michael Vielhaber

Additional information is available at the end of the chapter

http://dx.doi.org/10.5772/intechopen.80432

Abstract

We define *Etherealware* as the concept of implementing the functionality of an algorithm by means of the clocking scheme of a cellular automaton (CA). We show, which functions can be implemented in this way, and by which CAs.

Keywords: cellular automaton, etherealware, asynchronous, update rule, universality, temporal order, clocking-computable

Your task: Compute a lot of different functions on n-bit inputs.

Your device: A (fixed!) cellular automaton (CA) on the (fixed!) ring or torus topology with n cells, is capable of holding one bit each.

You may not change the CA (its update rule) nor the topology. You may not enter additional information in the form of parameters (there would be no space to store them anyway) — and yet you are supposed to evaluate many different functions. The available degree of freedom is the clocking scheme of the cells, anything from synchronous to completely asynchronous is allowed.

Can you do it?

The perhaps surprising answer is: *yes!*

Every bijective function on the set $\{0, 1, \ldots, 2^n - 1\}$, which acts as an even permutation is clocking-computable, as well as many non-bijective functions.

1. Introduction

1.1. What is etherealware?

Computation takes place in dedicated *hard*ware or on general-purpose *hard*ware by dedicated *soft*ware. Different functionality requires either changing the *hard*ware (think ASIC, FPGA) or changing the *soft*ware running on it. *Firm*ware is an intermediate concept, where the *hard*ware is modified by microprogramming a CPU or personalizing an FPGA.

Etherealware is the first way to use fixed hardware (certain cellular automata (CA) in this case), run fixed software (the same update rule for all cells, for all time, for all purposes), and still deliver diversity in the resulting function: by changing *only* the clocking scheme, the order in which the CA's cells are updated.

1.2. State-of-the-art

The study of synchronous CAs starts with Wolfram [1]. We use asynchronous CAs as deterministic devices with a finite number of computation steps, which is a *new* point of view.

Previously, asynchronous CAs have been treated as dynamical systems, where infinite computations are considered, and the focus lies on concepts like orbits, fixed points, ergodicity, transients, cycles and their periods, and long-term behavior. Also, randomness can be introduced to average over many possible asynchronous schemes. Papers in this respect are:

Ingerson and Buvel [2], 1984, distinguish synchronous, random (completely asynchronous), and periodic clocking, which yield clearly distinguishable behavior.

Barrett et al. [3–6], 1999–2003, consider sequential dynamical systems (SDS), including CAs with arbitrary toplogy and neighborhoods. They cover random graphs as topology and dynamical systems topics such as fixed points and invertibility.

Siwak [7], 2002, gives an overview of simulating machines, including CAs and SDSs, and unifies them under the concept of "filtrons."

Lee *et al*. [8], 2003, give an asynchronous CA on the two-dimensional grid $\mathbb{Z} \times \mathbb{Z}$, which is Turing universal.

Laubenbacher and Pareigis [9], 2006, build upon [3–6] and observe that not all $n!$ permutations of the cells lead to different temporal rules. Their equivalence classes coincide—for our setting, CAs on the torus—with our result ([10], Thm. 1]).

Fatès et al. [11], 2006, consider ECAs with quiescent states ($000 \mapsto 0, 111 \mapsto 1$, i.e., with even Wolfram rule ≥ 128). They show that 9 ECAs diverge, while the other 55 converge to a random fixed point, in 4 clearly distinguishable time frames $\Theta(n \log(n)), \Theta(n^2), \Theta(n^3)$, or $\Theta(n2^n)$ with characteristic behavior per time frame.

Macauley, McCammond, and Mortveit [12, 13], 2007–2010, also treat SDSes, in particular ECAs. For each ECA, [13] gives the periodic states and the dynamics group. Conjecture 5.10 in [13] about Wolfram rule 57 coincides with our finding that ECA-57 generates the alternating groups

on patterns of n bits. They verified this claim for n up to 8, while in [14] this is extended to $n \leq 10$ and in this paper up to $n \leq 28$.

Dennunzio et al. [15, 16], 2012–2013, consider ACAs, every turing machine can be simulated by an ACA, with quadratic slowdown. Introducing a certain fairness measure, they show that injectivity and surjectivity are equivalent (μ-a.a.), and the existence of a diamond is equivalent to not μ-a.a. injectivity.

Salo [17], 2014, shows that nonuniform CA generates all SFTs (subshifts of finite type) and several non-SFT sofic shifts.

2. Notations: ECA and update rules

Here, continuing the work in [10, 14], we again employ CAs as computing devices, whose work comes to an end, when the pattern transformation or function evaluation has been obtained. Also, the clocking, the temporal update rule, is completely deterministic and replaces the usual ways of representing an algorithm, either in software (initial data) or in hardware (choice of ECA and connecting graph). Thus, the algorithm resides exclusively in the clocking scheme. We therefore call functions computable in this way as "clocking-computable functions." The main additional contribution of this paper is the introduction of *unfair clocking schemes*.

2.1. Cellular automata: Neighborhoods and local update rules

We consider cellular automata (CA) on a torus or ring of n cells, that is index set $\mathbb{Z}/n\mathbb{Z}$, over the binary alphabet $\{0,1\}$. Cell index wraparound, that is, $c_i = c_j$ for $i \equiv jmodn$, and the canonical cell names are $c_{n-1}, c_{n-2}, \ldots, c_1, c_0$. We deal with elementary CAs (ECA) with three input cells, where the middle one is also the output cell.

The neighborhood (c_{i+1}, c_i, c_{i-1}) can have eight different values from $\in \mathbb{F}_2^3$. Let $k := 4 \cdot c_{i+1} + 2 \cdot c_i + c_{i-1} \in \{0, \ldots, 7\}$. Then c_i is replaced by $c_i^+ := p_k \in \{0,1\}$, where $p_0, \ldots, p_7$ are defined via Wolfram's rule [1] $\sum_{k=0}^{7} 2^k p_k \in \{0, \ldots, 255\}$.

The $2^{2^3} = 256$ ECAs can be arranged into 88 equivalence classes under the symbolic symmetry 0/1 and the chiral symmetry left/right ($c_{i+1} \leftrightarrow c_{i-1}$); see ([14], Appendix A). It is sufficient to consider one member per class.

We considered quad CAs (QCAs) with four inputs and nonstandard neighborhoods in ([10], Section 1.2).

Local bijectivity requires that ECA $(a, 0, c) = 1-$ ECA $(a, 1, c)$. This is equivalent to requiring that the hexadecimal digits of the rule be from `3, 6, 9, C`.

Example: The behavior of the ECA with Wolfram rule $57 = \mathbf{00111001}_2 = 39_{16}$:

$111 \mapsto \mathbf{0}$, $110 \mapsto \mathbf{0}$, $101 \mapsto \mathbf{1}$, $100 \mapsto \mathbf{1}$, $011 \mapsto \mathbf{1}$, $010 \mapsto \mathbf{0}$, $001 \mapsto \mathbf{0}$, $000 \mapsto \mathbf{1}$, in other words $0c_i1 \mapsto c_i^+ = c_i$, for all other contexts we have $0c_i0, 1c_i0, 1c_i1 \mapsto c_i^+ = \bar{c}_i$.

2.2. ECA: Global update rules on the torus

2.2.1. Fair update schemes

We repeat the *definition* of asynchronicity rules from ([10], Section 2).

The set AS_n of asynchronicity rules over $\mathbb{Z}/n\mathbb{Z}$ consists in all words of length n over the alphabet $\{<, \equiv, >\}$ such that both "$<$" and "$>$" occur at least once. We also include the word "$\equiv \cdots \equiv$," the synchronous case, and have.

$AS_n = (\{<, \equiv, >\}^n \setminus (\{<, \equiv\}^n \cup \{\equiv, >\}^n)) \cup \{\equiv^n\}$ with $|AS_n| = 3^n - 2^{n+1} + 2$.

A rule $\mathrm{AS} = \mathrm{AS}_{n-1} \cdots \ \mathrm{AS}_0 \in AS_n$ defines the firing order as follows:

AS_i	**Meaning**
$<$	Cell c_i fires after cell c_{i-1}
$>$	Cell c_i fires before cell c_{i-1}
$\equiv$	Cell c_i fires simultaneously with cell c_{i-1}

To ensure bijectivity, we must first have a locally bijective CA, and, furthermore, no two adjacent cells may fire simultaneously. Why this is so will be dealt with in Chapter 5. There are exactly $2^n - 2$ bijective fair rules, those from $\{<, >\}^n \setminus \{<^n, >^n\}$; see also ([10], 4.1]).

A fair update step (bijective or not) can be decomposed into a sequence of elementary steps such that all cells fire exactly once during the execution of that sequence; see [3].

2.2.2. Unfair update schemes

We now include unfair updates, where some cells may fire less often than others (even not at all).

We start with elementary steps (μ steps in [10]). During one elementary step, any nonempty subset $I \subseteq \{n-1, n-2, \ldots, 1, 0\}$ of indices may define the active cells.

These cells fire simultaneously, hence $c_i^+ = \begin{cases} ECA(c_{i+1}, c_i, c_{i-1}), & i \in I, \\ c_i, & i \notin I. \end{cases}$

We define elementary steps as words $s = s_{n-1} \cdots s_1 s_0 \in \{0,1\}^n$, with the meaning $s_i = 1$, if cell c_i fires, and $s_i = 0$ if cell c_i is inactive in this step.

A sequence of such elementary steps is upper indexed by the time step (t).

A fair update rule consists in a number of elementary steps such that every cell fires exactly once. The fair rule $as =$ "$<><><>$" for $n = 6$ can be decomposed into ($s^{(1)} = 010101,\ s^{(2)} = 101010$), while the sequence ($s^{(1)} = 010001, s^{(2)} = 101010$) is unfair, since cell c_2 does not fire at all.

The number of bijective elementary steps is the number of words of length n over the alphabet $\{0,1\}$, such that no adjacent 1's occur to ensure bijectivity. For $n = 3, 4, 5, 6$ there are 3, 6, 10, 17 such steps, respectively. The sequence obeys the law $n_k = n_{k-1} + n_{k-2} + 1$. At least for $n = 3, \ldots, 7$, the sets are as follows: prepend a 0 to each pattern of length $k-1$, prepend a 10 (a 01) to each pattern of size $k-1$ terminating in 0 (in 1), and add the new pattern 10^{k-1}. The

sequence is used again in Section 5 and has been verified to coincide with OEIS A001610 up to $n = 24$.

We also can define unfair bijective rules (full steps), where we first fix some subset of size $1 \le k \le n-1$ of active cells by a word a from $\{0,1\}^n \setminus \{0^n, 1^n\}$, $a_i = 1$ meaning that cell c_i is active.

We next order adjacent active cells by the usual $<$, $>$ signs. Hence, a run of r consecutive 1's (with wraparound) has 2^{r-1} ways to fix the internal firing order. This order is independent of the other 1-runs, since the cells are separated by at least an inactive cell with $a_i = 0$. The number of bijective unfair rules on a torus of size n is

$$\sum_{p=1}^{2^n-1} \prod_{i=1}^{\#runs} 2^{r_i-1}$$

where the pattern p avoids 0^n and 1^n and then the 1-runs in the pattern have lengths $r_1, r_2, \ldots$, considering wraparound.

For $n = 3, \ldots, 7$, we have $9 = 3+6$, $30 = 4+10+16$, $90 = 5+15+30+40$, $257 = 6+21+50+84+96$, and $714 = 7+28+77+154+224+224$, such unfair rules, respectively. The terms count how many patterns with $k = 1, 2, \ldots, n-1$ active cells are feasible. These terms can be found in OEIS [18] as subsequences of A209697.

3. Patterns

We consider pattern conversions $\mathbb{F}_2^n \ni v \mapsto w \in \mathbb{F}_2^n$, where $\mathbb{F}_2^n$ can be identified with the set $\{0, 1, \ldots, 2^n - 1\}$. From Definition ([10], Def. 3), by $ECA_{\mathrm{AS}}(v) = w$, we mean that the elementary CA with rule ECA maps $v \in \{0,1\}^n$ to $w \in \{0,1\}^n$ via the asynchronicity scheme AS. We define $ECA^{\tau}_{\mathrm{AS}}(v) = ECA_{\mathrm{AS}}\big(\ ECA^{\tau-1}_{\mathrm{AS}}\ (v)\big)$ recursively for $\tau \in \mathbb{N}$, starting with $ECA^{1}_{\mathrm{AS}}\ (v) = ECA_{\mathrm{AS}}\ (v)$.

In [10, 14], we considered five universality properties (o) to (iv), where each $v \mapsto w$ makes use of a certain update rule AS applied several times. We only give a summary here. Property (iv) is ruled out for any $n \in \mathbb{N}$, while properties (o) to (iv) have only been verified experimentally, for $n \le 15$.

$$(o) \quad \exists v \in \mathbb{F}_2^n, \quad \forall w \in \mathbb{F}_2^n, \quad \exists \tau \in \mathbb{N}, \quad \exists AS \in AS_n : \quad ECA^{\tau}_{\mathrm{AS}}(v) = w.$$

Some v is mapped to every w by varying the rule AS and the required number of time steps. There are 44 ECAs doing this.

$$(i) \quad \forall v \in \mathbb{F}_2^n, \quad \forall w \in \mathbb{F}_2^n, \quad \exists \tau \in \mathbb{N}, \ \exists AS \in AS_n : \quad ECA^{\tau}_{\mathrm{AS}}(v) = w.$$

All v are mapped to all w by varying the rule AS and the required number of time steps. There are 6 ECAs (rules 19, 23 (for $n \not\equiv 0 \bmod 2$), 37 (for $n \not\equiv 0 \bmod 3$), 41, 57, 105 (for $n \not\equiv 0 \bmod 4$)) doing this (checked for $n \le 15$).

$$(ii) \quad \forall v \in \mathbb{F}_2^n, \quad \exists \tau \in \mathbb{F}, \quad \forall w \in \mathbb{N}_2^n, \quad \exists AS \quad \in AS_n : \quad ECA^{\tau}_{\mathrm{AS}}(v) = w.$$

All v are mapped to all w at the same time, which time may vary for v but not for w, for different rules AS .

ECA	$n=5$	6	7	8	9	10	11	12	13	14	15
57	445	—	70	242	35	13	13	13	13	13	10
105	—	—	570	—	14	—	6	—	6	—	8

Table 1. Minimum time τ required to satisfy (iii).

ECA-57 realizes this for $n = 5, \ldots, 15$ (no further n has been considered).

ECA-105 realizes this for odd $n = 7, 9, 11, 13, 15$ (no further n has been considered).

$$(iii) \quad \exists \tau \in \mathbb{N}, \quad \forall v \in \mathbb{F}_2^n, \quad \forall w \in \mathbb{F}_2^n, \quad \exists AS \quad \in AS_n : \quad ECA_{\mathrm{AS}}^{\tau}(v) = w.$$

All v are mapped to all w at the same time, varying the rule AS. This is actually possible for the two survivors of property (ii); see **Table 1**. The required time roughly decreases with growing n, since we have $3^n - 2^{n+1} + 1$ rules to choose from for 2^n patterns w. Thus for higher n the probability to meet the conversion early on increases.

$$(iv) \quad \exists \tau_0 \in \mathbb{N}, \quad \forall \tau \geq \tau_0, \quad \forall v, w \in \mathbb{F}_2^n, \quad \exists AS \quad \in AS_n : \quad ECA_{\mathrm{AS}}^{\tau}(v) = w.$$

Eventually, all conversions may happen at all times for some update scheme. This property cannot be satisfied, for no ECA; see ([14], Thm. 2).

For more details and results for QCAs consult Section 2 of [14] and Section 3 of [10].

4. Bijective functions

We first introduce the computer algebra system GAP and then give several examples.

4.1. GAP: Graphs, algorithms, programming

4.1.1. GAP and the alternating group A_{2^n}

GAP [19] is a system for computational discrete algebra, in particular computational group theory. We use GAP to decide, whether certain fair or unfair update rules generate the full symmetric or alternating group S_{2^n} or A_{2^n}, respectively.

Our results so far:

Theorem 1.

i. The fair update rules for ECA-57 generate the full symmetric group S_8 for $n = 3$.

ii. The fair update rules for ECA-57 generate the full alternating group A_{2^n} for $n = 4, \ldots, 11$.

http://dx.doi.org/10.5772/intechopen.80432

iii. The (unfair) elementary update rules with exactly one active cell for ECA-57 generate the full alternating group A_{2^n} for $n = 4, \ldots, 28$.

Proof.

(i) By exhaustive generation of all $8! = 40320$ permutations on $\{000, 001, \ldots, 111\}$.

(ii, iii) First observe that all these rules consist of an even number of transpositions. Hence, we will at most obtain the alternating group A_{2^n}. This group comprises those bijective functions on $\{0, 1, \ldots, 2^n - 1\}$ which have positive sign as a permutation.

We checked (ii) with GAP for certain sets of 5 fair rules for each of these sizes n, and GAP's function `IsNaturalAlternatingGroup(G) returned true`.

For (iii) we have the canonical set of n elementary unfair update rules, with exactly one cell active in each rule. This set generates all fair and unfair update rules and hence is sufficient to decide on the group generated by all rules.

Again, GAP shows that indeed the alternating group is generated, for $n = 4, \ldots, 28$. We used 144 GB of RAM, which was sufficient for $n = 28$ but not so for $n = 29$. □

We believe that, apart from the special case $n = 3$, we always obtain the alternating group.

Conjecture.

For every torus size $n \in \mathbb{N}, n \geq 4$, both the $2^n - 2$ fair update rules, as well as the n elementary unfair update rules with a single active cell, are a generating set for the full alternating group on 2^n elements, using ECA-57.

4.1.2. Example for GAP usage with $n = 4$

We replaced 0 by 2^n, since GAP only uses numbers from $\mathbb{N}$. P00 to P03 are the permutations generated by the elementary steps $s = 0001, 0010, 0100$, and 1000, respectively.

```
mjv@Panda ~/GAP $gap -b
gap> P00 := (16,1)(2,3)(4,5)(6,7)(10,11)(14,15);
(1,16)(2,3)(4,5)(6,7)(10,11)(14,15)
gap> P01 := (16,2)(4,6)(5,7)(8,10)(12,14)(13,15);
(2,16)(4,6)(5,7)(8,10)(12,14)(13,15)
gap> P02 := (16,4)(1,5)(8,12)(9,13)(10,14)(11,15);
(1,5)(4,16)(8,12)(9,13)(10,14)(11,15)
gap> P03 := (16,8)(1,9)(2,10)(3,11)(5,13)(7,15);
(1,9)(2,10)(3,11)(5,13)(7,15)(8,16)
gap> G04 := Group(P00,P01,P02,P03);;
gap> IsNaturalAlternatingGroup(G04);
true
gap> Size(G04);
10461394944000
```

4.1.3. Example for GAP usage with $n = 5$

```
mjv@Panda ~/GAP $cat G5
P00 := (32,1)(2,3)(4,5)(6,7)(8,9)(10,11)(12,13)(14,15)
(18,19)(22,23)(26,27)(30,31);
P01 := (32,2)(4,6)(5,7)(8,10)(12,14)(13,15)(16,18)
(20,22)(21,23)(24,26)(28,30)(29,31);
P02 := (32,4)(1,5)(8,12)(9,13)(10,14)(11,15)(16,20)
(17,21)(24,28)(25,29)(26,30)(27,31);
P03 := (32,8)(1,9)(2,10)(3,11)(16,24)
(17,25)(18,26)(19,27)(20,28)(21,29)(22,30)(23,31);
P04 := (32,16)(1,17)(2,18)(3,19)(4,20)(5,21)(6,22)
(7,23)(9,25)(11,27)(13,29)(15,31);
G05 := Group(P00,P01,P02,P03,P04);
```

```
mjv@Panda ~/GAP $gap -b
gap> Read("G5");
gap> IsNaturalAlternatingGroup(G05);
true
gap> Size(G05);
131565418466846765083609006080000000
gap> quit;
```

Here, $1.315\ldots \cdot 10^{35} = 32!/2$.

4.1.4. Example for GAP usage with $n = 28$

We define the permutations P00 to P27 via a C++ − program; see Appendix A. Using it like gap.out 28 > G28, its output, the file G28, is then read in by GAP.

```
mjv@turing:/var/GAP$ gap -b -m 140G
gap> Runtime();Read("G28");Runtime();IsAlternatingGroup(G28);Runtime();
IsNaturalAlternatingGroup(G28);Runtime();
5592
1639756
true
8915688
true
8915688
gap>
```

Times are in milliseconds. Therefore, reading in the 56 GB of permutations P00 to P27 generating the group G28 took 1640 seconds, or about half an hour, while actually checking the resulting group for $A_{2^{28}}$ took another 7280 seconds or 2 hours. The same procedure for G29 resulted in lack of memory; 140 GB of RAM are not sufficient.

http://dx.doi.org/10.5772/intechopen.80432

Recall that, according to Stirling's formula, $A_{2^{28}}$ has $2^{28}!/2 \approx \left(\frac{2^{28}}{e}\right)^{2^{28}} \approx 10^{10^9}$ elements, a number with about 1 billion (1000^3) digits. Amazingly, GAP gets it done!

4.2. Examples for bijective functions

4.2.1. Multiplication by 9 mod 16

We give two realizations of the function *byNine* : $x \mapsto 9x$ mod 16 in **Table 2**.

Observe that the first 23 steps only implement the permutation $(08)(2A)$, consisting of two transpositions. The final 24th step is almost identical to the whole function.

First realization in hexadecimal	1. Active cells	2. Active cells	Second realization in hexadecimal	
0xFEDCBA9876543210	0101	0100	0xFEDCBA9876543210	(Id)
0xAB98EFDC67012345	1010	1000	0xBA98FEDC76103254	
0x0312C57E4DA98B6F	0101	0001	0x32107E5CF698BAD4	
0x5243806B19FDCE7A	1000	1000	0x23016F4CE798ABD5	
0xDA4B08639175CEF2	0001	0001	0xAB89674CEF10235D	
0xDB5A18729064CFE3	0010	1010	0xBA89765CFE01324D	
0xFB781A509246EDC3	0100	0101	0x3021D4FE5CA9B867	
0xBF7C5E14D206A983	1000	0010	0x253491AB08FDEC76	
0x37FCDE945A86210B	0001	0001	0x0736918B2ADFCE54	
0x26ECDF954B87301A	1010	1010	0x1627908A3BDECF45	
0x84CE751F632DBA90	0101	0101	0x948D1A20B37CE56F	
0xC18B604A7239EFD5	1010	1010	0xD1C94F35E268B07A	
0xE9234A60D8B1C57F	0101	0101	0x79E165BFC8423AD0	
0xBD321F759CE4806A	1010	1010	0x6DB470EA8C132F95	
0x37B895DF1EC62A40	0100	0100	0x4736DAC02E9B851F	
0x37FCD19B5A862E04	1000	0010	0x07369E842ADFC15B	
0xBF7C5913D206AE84	0100	0101	0x25349CA608FDE17B	
0xFB781D539246EAC0	0010	1010	0x3021D8F75CA9B46E	
0xDB5A1F739064C8E2	0001	0001	0xBA89725DFE01364C	
0xDA4B0E629175C8F3	1000	1000	0xAB89634DEF10275C	
0x52438E6A19FDC07B	0101	0001	0x23016B45E798AFDC	
0x0312CB7F4DA9856E	1010	1000	0x32107A54F698BEDC	
0xAB98E3D567012F4C	0101	0100	0xBA98F2D476103E5C	
0xFEDCB29076543A18	1000	1000	0xFEDCB29076543A18	(08)(2A)
0x7E5C3A18F6D4B290			0x7E5C3A18F6D4B290	

Table 2. Multiplication $x \mapsto 9x$ in $\mathbb{Z}/16\mathbb{Z}$.

Active cells	FED...210 → 62C...931 some patterns in binary							All values F...0 in hexadecimal
0001	1111	1110	1101	…	0010	0001	0000	0xFEDCBA9876543210
1010	1110	1111	1101	…	0011	0000	0001	0xEFDCAB9867452301
0101	1100	0101	0111	…	1011	1010	1001	0xC57E03124D6F8BA9
1010	1000	0000	0110	…	1110	1111	1101	0x806B5243197ACEFD
0101	0010	1010	0100	…	1100	0101	0111	0x2A43F86B91D0EC57
1000	0011	1111	0001	…	1000	0000	0110	0x3F12AC7ED495B806
0100	1011	0111	1001	…	0000	1000	0110	0xB79A2CFE541D3086
1000	1111	0111	1101	…	0100	1100	0110	0xF7DE28BA105934C6
0001	0111	1111	0101	…	0100	1100	0110	0x7F5EA03298D1B4C6
1000	0110	1110	0100	…	0101	1100	0111	0x6E4FB12398D0A5C7
0100	0110	1110	0100	…	1101	1100	1111	0x6E4739AB10582DCF
1010	0110	1010	0000	…	1001	1000	1011	0x6A073DEF541C298B
0100	0100	0000	1010	…	0001	0010	0011	0x40ADB7C5F69E8123
1010	0000	0100	1110	…	0101	0010	0011	0x04E9F781B6DAC523
0001	1010	0110	1100	…	1111	1000	1011	0xA6C15D293470EF8B
1000	1011	0111	1100	…	1110	1000	1010	0xB7C04D392561FE8A
0101	0011	1111	1100	…	1110	0000	0010	0x3FC845B1AD697E02
1000	0010	1010	1000	…	1011	0101	0011	0x2A8C10E4F97D6B53
0100	1010	0010	0000	…	0011	1101	1011	0xA20C98E471F563DB
1000	1110	0010	0100	…	0011	1001	1111	0xE248DCA075B1639F
0100	1110	1010	0100	…	1011	0001	0111	0xEA405C28FD396B17
1010	1010	1110	0000	…	1111	0101	0111	0xAE04182CB93D6F57
0100	0000	1100	1010	…	0101	1111	1101	0x0CA6928E31B745FD
1010	0100	1000	1110	…	0001	1011	1001	0x48E6D2CA35F701B9
	0110	0010	1100	…	1001	0011	0001	0x62C478E0BF5DA931

Table 3. Exponentiation $x \mapsto 3^x$ in $\mathbb{F}_{17}$.

Hence *byNine* = `(19)(3B)(5D)(7F)` = `[(08)(2A)][(08)(19)(2A)(3B)(5D)(7F)]`, where the first bracket requires 23 steps, and the second is the elementary rule 1000 (only the leftmost cell is active). The difference between the two realizations (sequences $s^{(1,\ldots,24)}$ in hex), where the outer parentheses are inverses of each other and the inner part is self-inverse, and their difference:

```
1.: (5A581248)1(A5A5A)4(842185A5)8
Diff 124-81A——-A18-421-
2.: (48181A52)1(A5A5A)4(25A18184)8
```

4.2.2. Exponentiation and logarithm in $\mathbb{F}_{17}$

Identifying 16 with 0b0000, we can map $\mathbb{F}_{17}^*$ to $\{0,1\}^4$. Here is the exponentiation $x \mapsto 3^x$; see **Table 3**.

`FEDCBA9876543210` $\rightarrow$ …
$s^{(1,\ldots,24)}$ = [0001, 1010, 0101, 1010, 0101, 1000, 0100, 1000, 0001, 1000, 0100, 1010, 0100, 1010, 0001, 1000, 0101, 1000, 0100, 1000, 0100, 1010, 0100, 1010]
… $\rightarrow$`62C478E0BF5DA931`

The inverse function $x \mapsto \log_3(x)$ is therefore computed by the inverse update sequence $s^{(1,\ldots,24)}$ = [1010, 0100, 1010, 0100, 1000, 0100, 1000, 0101, 1000, 0001, 1010, 0100, 1010, 0100, 1000, 0001, 1000, 0100, 1000, 0101, 1010, 0101, 1010, 0001].

5. Non-bijective functions

5.1. Non-bijective global rules and in-degree distributions

In Section 2.2, we have said that, in order to ensure bijectivity, we must first have a locally bijective CA, and furthermore, no two adjacent cells may fire simultaneously. Here is why:

Example.

We start with the effect of AS_i = "$\equiv$" for the locally bijective ECA-57. We consider four adjacent cells and the effect of $\equiv$ between the two middle cells, which are thus updated simultaneously, $s = 0110$.

$v \mapsto$ **ECA-57**$_{0\equiv 00}(v)$											
0000	↦	0110	0001	↦	**0101**	1000	↦	1110	1001	↦	1101
0010	↦	0000	0011	↦	**0011**	1010	↦	1100	1011	↦	1111
0100	↦	0010	0101	↦	**0011**	1100	↦	1010	1101	↦	1011
0110	↦	0100	0111	↦	**0101**	1110	↦	1000	1111	↦	1001

We obtain the patterns 0011 and 0101 twice, while missing 0001 and 0111. Hence the image is smaller than the full 2^4 by 2 or by a factor 7/8 in general.

We have the following in-degree distribution (loss pattern 4 of [[10], Table 8]):

#(domain)	#(range)	In-degree	Distribution φ
12	12	1	$\varphi(1) = 12$
4	2	2	$\varphi(2) = 2$
0	2	0	$\varphi(0) = 2$

Twelve patterns map bijectively (1,1) to 12 patterns, 4 patterns map 2:1 to 2 patterns, while 2 patterns of the range are not met at all (0,1).

We have #(domain) = in-degree×#(range) for every in-degree. Also, the overall sum is

$$\sum \#(\text{domain}) = \sum \text{ in-degree} \times \#(\text{range}) = 2^n.$$

Any additional simultaneous firing may increase the losses. Additional in-degree distributions (loss patterns) for ECAs and QCAs may be found in ([10], Table 8).

We restate Theorem 3 (i) from [14]:

Theorem 2.

We assume an ECA that generates at least the alternating group A_{2^n}, when using temporal rules from $\{<,>\}^n \setminus \{<^n, >^n\}$. Let $f : \mathbb{F}_2^n \to \mathbb{F}_2^n$ be any non-bijective function on n symbols, where we require $n \geq 4$ for ECA.

Let $\#(w) = |\{v|f(v) = w\}|$ be the number of configurations v leading to configuration w. Then f is clocking-computable by an ECA with in-degree distribution $\varphi(0,1,2) = (12,2,2)$ or $(24,4,4)$, if and only if

$$\sum_{w \in \mathbb{F}_2^n} \lfloor \#(w)/2 \rfloor \geq \varphi(2) \cdot 2^n / \sum_{k=0}^{2} \varphi(k).$$

Proof. See [14], Theorem 3. □

5.2. Algorithm for non-bijective functions

The computation of a non-bijective function can be decomposed into 3 steps:

Step I. We start in the middle: Shrink the 2^n singletons of the domain to the desired distribution on the range. Find a sequence of elementary steps, necessarily including non-bijective ones, which generates the same distribution of counts in the image space as for the original function. Usually there are more than one of these sequences with the same complexity. Values/patterns as such do not yet play a role; therefore the requirements are usually easy to meet in a variety of ways, and the sequence of necessary elementary steps is short.

Step II. Bijectively map the input of the original function to the input of any of the results of Step I in such a way that the desired image counts are matched. Only the occurence counts must be observed, while, within this restriction, pattern values can be permuted.

Step III. Bijectively map the output from Step I to the output of the desired function, considering the flow within Steps II and I in that order. Now all values matter and cannot be permuted. However, this step takes place on a subset of size $|Im(f)|$ instead of 2^n.

In Steps II and III we use the meet-in-the-middle approach, applying update rules starting both from the input (identity) and the desired function values to take advantage of the birthday paradox: In this way, $2 \cdot \sqrt{\#}$ patterns are sufficient to generate $\sqrt{\#} \cdot \sqrt{\#} = \#$ potential matches in the middle (here $\# \approx |A_{2^n}|$), a considerable saving in both space and time.

http://dx.doi.org/10.5772/intechopen.80432

5.3. Example of a non-bijective function multiplication up to $3 \times 3 = 9$ on the torus of size $n = 4$

The 4-bit input is interpreted as a pair of numbers from the set $\{0, 1, 2, 3\}$, whose product, the output, lies within the set $\{0, 1, 2, 3, 4, 6, 9\}$.

In	Out	In	Out	In	Out	In	Out
00.00	0000	01.00	0000	10.00	0000	11.00	0000
00.01	0000	01.01	0001	10.01	0010	11.01	0011
00.10	0000	01.10	0010	10.10	0100	11.10	0110
00.11	0000	01.11	0011	10.11	0110	11.11	1001

5.3.1. Step I

The image consists of 7 patterns with occurence counts 7 (0000), 2 (0010,0011,0110), and 1 (0001,0100,1001).

In Step I, we thus have to shrink the set of patterns present from 16 to 7 in such a way that (any) 7 patterns are mapped to the same one, and additionally 3 sets of two patterns are joined within each set. Finally, the three remaining input patterns stay on their own.

An exhaustive search over sequences with 4 update rules, starting and ending with a non-bijective one, yields 72 such sequences with the desired shrinking factor, one example is the sequence $s^{(1,\ldots,4)} = 0011, 1101, 0110, 0111$ of active cells.

All patterns in hex	Active cells	Occurrence counts				
		1	2	3	5	7
0123.4567.89AB.CDEF	0011	16	0	0	0	0
3012.7654.A99A.EFDC	1101	14	2	0	0	0
ADCB.E781.7557.B218	0110	5	4	1	0	0
CBAF.85E5.5335.F05E	0111	5	3	0	1	0
AED8.E292.2222.8729		3	3	0	0	1

5.3.2. Step II

After Step I, the output pattern 2 has frequency 7. Therefore, 2 has to match 0 in Step III, while 0,1,2,3,4,8,C are matched to 5,7,8,9,A,B,E in any order in Step II. Thus, we already have 7! different possibilities for Step II.

Similarly, we can maps 0110 and 1001 to either $\{1, 4\}$, $\{3, C\}$, or $\{6, F\}$ and so forth. Let c run through the occurrence counts, here $c = 7$, 2 and 1 are actually taken, and let then $p(c)$ be the number of output patterns with in-degree c, here $p(7) = 1$, $p(2) = p(1) = 3$. We get

$$\prod_c c!^{p(c)} p(c)! = 7!^1 1! \cdot (2!)^3 3! \cdot (1!)^3 3! = 1451520$$

bijective functions consistent with the count distribution to choose from in Step II. One of these is given in the left column of **Table 4**.

5.3.3. Step III

The outcome of Steps I + II completely fixes the necessary permutation for Step III. However, we do not have to deal with 2^n values, but only $|\mathrm{Im}(f)|$ are relevant, in our example 7 instead of

Step II		**Step I**		**Step III**	
0xFEDCBA9876543210	0001	0x0123456789ABCDEF	0011	0x2AE879D	0001
0xEFDCAB9867452301	1010	0x30127654A99AEFDC	1101	0x3BF869D	0010
0xC57E03124D6F8BA9	0100	0xADCBE7817557B218	0110	0x3BDA49F	0100
0x817A4352096BCFED	0010	0xCBAF85E55335F05E	0111	0x3F9E0DB	1010
0xA1586370294BEDCF	0001	0xAED8E29222228729		0xB51CA73	0101
0xB0487261395AFDCE	1010			0xE048F62	0010
0x3A62D849B1F057EC	0100			0xC26AD40	0101
0x3E629C0DF5B417A8	0010			0x837F915	0010
0x3C409E2FD7B6158A	0100			0xA35D917	0101
0x3804DA2B97F651CE	1010			0xF209D46	1000
0xB2A670831D54F9EC	0001			0x7A81546	0100
0xA3B761820D45E9FC	1000			0x7EC5106	1010
0x2B3F690A854DE17C	0100			0xDCEF9A4	0101
0x2F3B6D4EC109A578				0x98BADF1	1000
				0x1032579	0001
				0x0123469	

Table 4. Steps II, I, and III for multiplication up to $3 \times 3 = 9$.

IN	**Step II**	**Step I**	**Step III**	**IN**	**Step II**	**Step I**	**Step III**
0010	5	2	0	0110	1	E	2
0001	7	2	0	1001	4	E	2
0000	8	2	0	1101	3	8	3
0100	9	2	0	0111	C	8	3
0011	A	2	0	1010	D	7	4
1100	B	2	0	1011	6	9	6
1000	E	2	0	1110	F	9	6
0101	0	A	1	1111	2	D	9

Table 5. Steps II, I, and III by output patterns.

http://dx.doi.org/10.5772/intechopen.80432

16. Again, meet-in-the-middle yields a match. Be aware that the necessary effort does not drop from 16! to 7! but only to $\frac{16!}{(16-7)!} \approx 11!$ or $\frac{2^n!}{(2^n-|Im(f)|)!}$ in general.

In total, 13 + 4 + 15 = 32 steps are required to compute this multiplication function (**Table 5**).

6. Efficiency

In the case of unfair bijective functions, any subset of cells may fire simultaneously, provided that no adjacent cells are contained in the set.

We define *local efficiency* of a bijective update sequence by two properties:

- No cell is active during two consecutive time steps (these two would cancel each other).
- No active cell can be moved to the previous time step.

Global efficiency—which shortest update sequence generates a certain function/permutation—is beyond the scope of this paper (it is dealt with implicitly by brute force in a breadth-first manner).

Lemma 3.

A sequence of rules is bijective and efficient, if $s_k^{(t)} = 1$ (cell c_k active in step t) implies that.

i. $s_{k-1}^{(t)} = 0$ and $s_{k+1}^{(t)} = 0$ (both inactive),

ii. $s_k^{(t-1)} = 0$ (inactive), and.

iii. at least one of $s_{k-1}^{(t-1)}$ and $s_{k+1}^{(t-1)}$ equals 1, active.

Proof.

(i) is required for and then ensures bijectivity.

If $s_k^{(t-1)}$ was active, both $s_k^{(t-1)}$ and $s_k^{(t)}$ could be removed, since each single-location action is an involution. Thus (ii) is required.

If $s_{k-1}^{(t-1)}$ and $s_{k+1}^{(t-1)}$ were both inactive, the active cell $s_k^{(t)}$ could be moved to $s_k^{(t-1)}$, maintaining bijectivity. Hence (iii) is required. □

Using this lemma, we now can compute an upper bound on the number of bijective functions realizable with t update steps: We define a matrix A with rows and columns indexed by the bijective elementary update rules, here named $s(1), s(2), \ldots$. Set $a_{ij} := 1$, if rule $s(i)$ followed by rule $s(j)$ is efficient, $a_{ij} := 0$ otherwise.

Let $\lambda(n)$ be the largest eigenvalue of A.

Example

	A_{ij} **for** $n = 5$**:**									
$s(i) \setminus s(j)$	$s(1)$	$s(2)$	$s(3)$	$s(4)$	$s(5)$	$s(6)$	$s(7)$	$s(8)$	$s(9)$	$s(10)$
$s(1)$ = 00001	0	1	0	0	0	0	0	1	1	0
$s(2)$ = 00010	1	0	1	1	0	0	0	0	0	0
$s(3)$ = 00100	0	1	0	0	1	0	1	0	0	0
$s(4)$ = 00101	0	1	0	0	1	0	1	1	1	0
$s(5)$ = 01000	0	0	1	0	0	0	0	1	0	1
$s(6)$ = 01001	0	1	1	0	0	0	0	1	1	1
$s(7)$ = 01010	1	0	1	1	0	0	0	1	0	1
$s(8)$ = 10,000	1	0	0	0	1	1	0	0	0	0
$s(9)$ = 10,010	1	0	1	1	1	1	0	0	0	0
$s(10)$ = 10,100	1	1	0	0	1	1	1	0	0	0

Observe that 10,100 followed by 00001 is efficient, while the other way round 00001 leaves room to move $s_2^{(2)}$ up to $s_2^{(1)}$. Hence, A is not symmetrical for $n \geq 5$.

In **Table 6**, we denote several figures describing the number of efficient bijective update schemes (including unfair ones). The number of rules is denoted by $\#S(n)$; it can be found as A001610 in OEIS [18]; see also Section 2.2.2. The number of nonzero entries in the matrix A is $\#A(n)$. The quotient $\#A(n)/\#S(n)$ is the average number of 1's in each row/column.

We denote by $\#_n(t)$ the number of effective bijective update schemes of length t on a torus of size n. Essentially, $\#_n(t) = \Theta\left(\lambda(n)^t\right)$.

Let $\rho(n) = \lim_{t\to\infty} \frac{\#_n(t)}{\lambda(n)^t}$. Then $\rho(n)$ gives the constant hidden in the Landau symbol Θ.

We have $\#_n(t)/\left(\lambda(n)^t \cdot \rho(n)\right) \to 1$.

Finally,

$$T(n) = \log_{\lambda(n)}(2^n!/(2 \cdot \rho(n)))$$

is the number of steps necessary such that $\#_n(T(n)) \approx |A_{2^n}| = 2^n!/2$. Invoking the birthday paradox by the meet-in-the-middle approach, we therefore expect to require $T(n)/2$ steps each, starting from the initial identity permutation and backwards from the desired function, to achieve a match yielding the function evaluation.

The asymptotic behavior, as far as we can infer from the range $n = 3, \ldots, 24$, is

$$\#S(n) = \Theta(\varphi^n), \varphi = 1.618\ldots$$

$$\lambda(n) = \Theta(1.3^n)$$

n	$\#S(n)$	$\#A(n)$	$\left\lfloor \frac{\#A(n)}{\#S(n)} \right\rfloor$	$\lambda(n)$	$\rho(n)$	$\lfloor \log(\lvert A_{2^n} \rvert) \rfloor$	$\lfloor T(n) \rfloor$
3	3	6	2	2.000	1.500	10	14
4	6	18	3	3.000	2.000	30	27
5	10	40	4	3.732	2.887	81	61
6	17	98	5	5.000	4.000	204	126
7	28	224	8	6.236	6.261	495	270
8	46	514	11	8.522	7.044	1166	544
9	75	1158	15	10.697	11.581	2685	1132
10	122	2602	21	14.599	11.142	6077	2266
11	198	5808	29	18.288	20.464	13,571	4668
12	321	12,930	40	24.941	17.141	29,978	9319
13	520	28,704	55	31.253	35.107	65,630	19,065
14	842	63,618	75	42.634	25.696	142,612	38,002
15	1363	140,806	103	53.426	58.716	307,933	77,402
16	2206	311,362	141	72.878	37.982	661,287	154,189
17	3570	688,024	192	91.321	96.473	1.413e + 06	313,092
18	5777	1,519,586	263	124.571	55.699	3.008e + 06	623,547
19	9348	3,354,944	358	156.098	156.314	6.380e + 06	1.26e + 06
20	15,126	7,405,058	489	212.932	81.313	1.348e + 07	2.51e + 06
21	24,475	16,341,254	667	266.822	250.502	2.842e + 07	5.08e + 06
22	39,602	36,056,154	910	363.969	118.336	5.976e + 07	1.01e + 07
23	64,078	79,547,616	1241	456.085	397.758	1.253e + 08	2.04e + 07
24	103,681	175,485,442	1692	622.138	171.942	2.623e + 08	4.07e + 07

Table 6. Efficiency related values for $n = 3, \ldots, 24$.

$$T(n) = 2.4 \cdot 2^n \cdot (1 + o(1)).$$

We can compare the asymptotic number $T(n)$ of elementary steps necessary for unfair update rules with the lower bound on the (full) steps for fair rules.

Lemma 4 ([10], Section 5, Lemma 1(i)).

There are clocking-computable bijective functions that require at least

$$\lceil \log(2^n!) / \log(2^n - 2) \rceil = 2^n + O(1)$$

steps or evaluations per cell.

Remarkably, as far as the experimental results for $n \leq 24$ indicate, the number $T(n)$ of elementary steps is only a constant factor (≈ 2.4) higher than the lower bound for full update steps.

If that result remains valid for all $n \in \mathbb{N}$, and the reported values strongly suggest this, this would mean that the typical full step can be replaced by no more than 2..3 elementary unfair steps, independent of n.

As can be appreciated, $n = 5$ might be feasible for an attack by meet-in-the-middle, but $n = 6$ and above is certainly no candidate for this brute-force approach. Dealing with these sizes will require a more intelligent approach, for instance using group-theoretic techniques like representation theory, applied to the alternating group.

7. Conclusion

We have seen that many functions are clocking-computable, namely, the even bijective ones and the non-bijective ones with enough "loss" in their image.

The elementary cellular automaton ECA-57 can be used to implement an etherealware computing device: The computed function is a result only of the clocking order.

Temporal order of activating the CA cells is thus a new way to encode algorithms, a "volatilization of information".

We increased the size for example programs from $n = 3$ in [10, 14] to $n = 4$ and are confident to also be able to tackle the case $n = 5$.

From $n = 6$ onwards, we shall need more mathematical concepts, e.g., from group theory.

Acknowledgements

My thanks go to Dr. Mónica del Pilar Canales Chacón for proofreading, commenting, and all the rest. Furthermore, the anonymous referee gave valuable comments concerning the first draft of the paper, pointing out a substantial error and several occasions for clarifying the intended meaning.

Appendix A. gap.cc

```
#include <stdlib.h>
#include <iostream>
#include <fstream>
#include <iomanip>

using namespace std;

int main(int argc, char** args) {

  long long N = atoi(args[1]); // size of torus

  const long long EN = 1LL<<N; // 2^N
```

http://dx.doi.org/10.5772/intechopen.80432

```
  long long ix;

  cout << "P00␣:=␣(" << (1LL<<N) << ",1)";
  ix = 0; // rightmost cell active, n=0
  for (long long i = 1; i < EN; i++) {
    if (((i ^ 1LL) > i) (((5LL<<(N-1LL))
        & (i ^ (i<<N))) != (1LL<<(N-1LL)))) {
      cout << "(" << i << "," << (i^1LL) << ")";
      ix++;
      if ((ix%6LL) == 0) {cout << endl ;}
    }
  }
cout << ";" << endl;

for (long long n = 1; n < N; n++) { // active cell/bit
  long long En = 1LL<<n;
  ix = 0;
  cout << "P" << (char) (0x30+((n/10)%10))
       << (char) (0x30+(n%10)) << "␣:=␣("
       << (1LL<<N) << "," << (1LL<<n) << ")";

  for (long long i = 1; i < EN; i++) // all 2^N patterns
    if (((i ^ En) > i) && (((0x5LL<<(n-1LL)) & (i ^ (i<<N)))
        != (1LL<<(n-1LL)))) {
      cout << "(" << i << "," << (i^En) << ")";
      ix++;
      if ((ix%6LL) == 0) {cout << endl ;}
    }
  }
  cout << ";" << endl;
} // n

cout << "G" << (char) (0x30+((N/10)%10))
     << (char) (0x30+(N%10)) << "␣:=␣Group(P00" ;

for (int n = 1; n < N; n++) {
  cout << ",P" << (char) (0x30+((n/10)%10)) << (char) (0x30+(n%10));
}
cout << ");" << endl;

//remainder, what to put interactively in GAP, no use as part of file
cout << "#IsNaturalAlternatingGroup(G"<< (char) (0x30+((N/10)
     << (char) (0x30+(N/10)%10)) << " ); " << endl ;
cout << "#IsNaturalSymmetricGroup(G"<< (char) (0x30+((N/10) %10))
     << (char) (0x30+(N%10)) << " ); " << endl ;
cout << "#Size(G"<< (char) (0x30+((N/10)%10))
     << (char) (0x30+(N%10)) << " ); " << endl ;
}
```

Author details

Michael Vielhaber[1,2*]

*Address all correspondence to: vielhaber@gmail.com

1 HS Bremerhaven, Bremerhaven, Germany

2 Universidad Austral de Chile, Instituto de Cs. Físicas y Matemáticas, Valdivia, Chile

References

[1] Wolfram S. Cellular automata as models of complexity. Nature. 1984;**311**:419-424

[2] Ingerson TE, Buvel RL. Structure in asynchronous cellular automata. Physica. 1984;**10D**: 59-68

[3] Barrett CL, Reidys CM. Elements of a theory of computer simulation I: Sequential CA over random graphs. Applied Mathematics and Computation. 1999;**98**:241-259

[4] Barrett CL, Mortveit HS, Reidys CM. Elements of a theory of simulation II: Sequential dynamical systems. Applied Mathematics and Computation. 2000;**107**:121-136

[5] Barrett CL, Mortveit HS, Reidys CM. Elements of a theory of simulation III: Equivalence of SDS. Applied Mathematics and Computation. 2001;**122**:325-340

[6] Barrett CL, Mortveit HS, Reidys CM. ETS IV: Sequential dynamical systems: Fixed points, invertibility and equivalence. Applied Mathematics and Computation. 2003;**134**:153-171

[7] Siwak P. Filtrons of automata. Proc. UMC 2002, unconventional models of computation. LNCS. 2002;**2509**:66-85

[8] Lee J, Peper F, Adachi S, Morita K, Mashiko S. Reversible computation in asynchronous cellular automata. Proceedings of Unconventional Models of Computation, LNCS. 2002; **2509**:20-229

[9] Laubenbacher P. Update schedules of sequential dynamical systems. Discrete Applied Mathematics. 2006;**154**:980-994

[10] Vielhaber M. Computing of functions on *n* bits by asynchronous clocking of cellular automata. Natural Computing. 2013;**12**:307-322. DOI: 10.1007/s11047-013-9376-7

[11] Fatès N, Thierry E, Morvan M, Schabanel N. Fully asynchronous behavior of double-quiescent elementary cellular automata. TCS. 2006;**362**:1-16

[12] Macauley M, McCammond J, Mortveit HS. Order Independence in Asynchronous Cellular Automata. 2007. arXiv:0707.2360v2 [math.DS]

[13] Macauley M, McCammond J, Mortveit HS. Dynamics groups of asynchronous cellular automata. Journal of Algebraic Combinatorics. 2010;**33**:31-55

[14] Vielhaber M. Computing by temporal order: Asynchronous cellular automata. Formenti E, editor. EPTCS 2012. Proceedings of Automata & JAC. Vol. 90; 2012. arXiv:1208.2762. DOI: 10.4204/EPTCS.90.14

[15] Dennunzio A, Formenti E, Manzoni L, Mauri G. Computing issues of asynchronous cellular automata. Fundamenta Informaticae. 2012;**120**:165-180

[16] Dennunzio A, Formenti E, Manzoni L, Mauri G. M-asynchronous cellular automata: From fairness to quasi-fairness. Natural Computing. 2013;**12**(4):561-572

[17] Salo V. Realization problems for nonuniform cellular automata. TCS. 2014;**559**. DOI: 10.1016/j.tcs.2014.07.031

[18] The On-Line Encyclopedia of Integer Sequences https://oeis.org

[19] The GAP Group, GAP—Groups, Algorithms, and Programming, Version 4.9.1; 2018. www.gap-system.org

Edited by Ricardo Lopez-Ruiz

Artificial intelligence (AI) is a field experiencing constant growth and change, with a long history. The challenge to reproduce human behavior in machines requires the interaction of many fields, from engineering to mathematics, from neurology to biology, from computer science to robotics, from web search to social networks, from machine learning to game theory, etc. Numerous applications and possibilities of AI are already a reality but other ones are still needed to reduce the human limitations and to expand the human capability to limits beyond our imagination. This book brings together researchers working on areas related to AI such as speech and face recognition, representation of learning and acoustic scenarios, fuzzy inference and data exploration, cellular automata applications with a special interest in the tools and algorithms that can be applied in these different branches of the AI discipline. The book provides a new reference to an audience interested in the development of this field.

Published in London, UK

IntechOpen

ISBN 978-1-83881-548-6