Speaker Verification Based on Speech Signal

Shariffah Fauziah Jap, Dr Ali Chekima, Mazlina Mamat, Wan Mahani Abdullah
School of Engineering and Information Technology
University Malaysia Sabah,
Locked Bag No. 2073,88999 Kota Kinabalu, Sabah
Email: chekima@ums.edu.my,
Telephone Number: 088 – 320000 (EXT 3065), Fax Number: 088- 320223

***Abstract*- This paper presents the initial effort to perform speaker verification by utilizing the speech signal characteristics found in individual's voice to recognize its speaker. A total of six speakers from different backgrounds were selected as sample and each of them is required to pronounces numbers zero to nine for 5 times. The recorded speech signal then undergoes a series of speech processing, which contains Pre-emphasis, Framing, Windowing and Endpoint Detection. To obtain the features of each speech signal, the Linear Prediction Coefficients (LPC) technique is used. The collection of LPC coefficients then were feed to the Multilayer Perceptron Neural Network trained by Back Propagation algorithm, which acts as a pattern matching algorithm. The results show that the speech signal has the potential to be used to verify its speaker in high accuracy.**

I. INTRODUCTION

Biometrics refers to the automatic identification of a person based on his/her physiological or behavioral characteristics. Voice is a part of human biometric. It is unique for each person. Voice biometrics provides three different services: identification, verification, and classification. Speaker verification authenticates a claim of identity, similar to matching a person's face to the photo on their badge. In the speaker-verification systems, the person is authenticated if he or she is the one who she or he claims to be. In this system only one to one data set is compared. This is different in speaker-identification systems, where we have to compare one to many data sets and find the one which matches. For example, speaker identification selects the identity of a speaker out of a group of possible candidates, similar to finding a person's face in a group photograph. On the other hand, speaker classification determines age, gender, and other characteristics of the speech.

Speaker verification based on speech signal uses the acoustic features of speech to differ between individuals. These acoustic patterns reflect both anatomy (e.g., size and shape of the throat and mouth) and learned behavioral patterns (e.g., voice pitch, speaking style)

II. SPEECH PROCESSING

A. Data Acquisition

The speech signals were recorded by using Decibel Frequency Analysis software in the Acoustic and Vibration lab in School of Engineering and Information Technology. The speech was captured in 20 kHz sampling rate, 16 bit, single channel and 1 second in length. A total of six speakers consisting of 3 males and 3 females were selected to record their voice and each speaker was asked to pronounce numbers zero to nine for eight times. This process contributes to 480 speech data where each of the data undergoes a series of speech processing step that is Pre-emphasis, Framing, Windowing and Endpoint Detection.

Pre-emphasis is the first process in the speech preprocessing where Pre-emphasis is a filtering process in which the frequency response of the filter has emphasis at a given frequency range. The speech signal was pre-emphasized using a pre-emphasis filter $H(z) = 1 - 0.9378z^{-1}$, where the value 0.9378 is the constant value for the filter equation.

The second step in the preprocessing stage is framing process where a speech signal is often separated into a number of segments called frames. Each frame will have the same number of samples and 50% of overlapping sections between frames. The next step in the preprocessing stage is to window each frame so as to minimize the signal discontinuities at the beginning and end of each frame. The concept is to minimize the spectral distortion by using the window to taper the signal to zero at the beginning and end of each frame.

Endpoint detection aims to distinguish the speech and non-speech segments of a digital speech signal. The algorithm for the endpoint detection in this project is based on the measurement of the short time energy. The speech energy is defined as the sum of magnitude of the 30ms of speech centered on the measurement interval. Then, the peak energy, maxi, and the silence energy, mini, are used to set two thresholds, that is the lower energy threshold and the upper energy threshold.

B. Linear Prediction Coefficients

The LPC coefficients for each frame in the speech signal were computed by using the build-in function in the Matlab software. These coefficients will be the input for the project's neural network. Fig. 1 and Fig. 2 show the LPC coefficient for the word '*zero*' for two speakers.

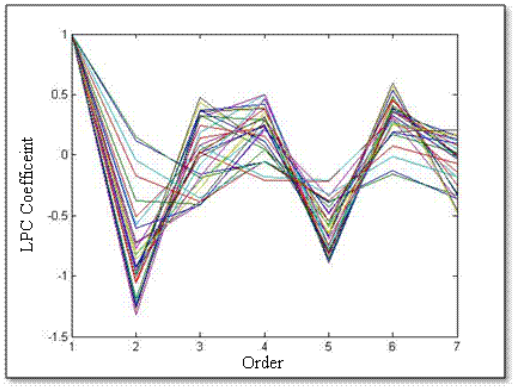


Fig. 1 The LPC coefficients for speaker 1

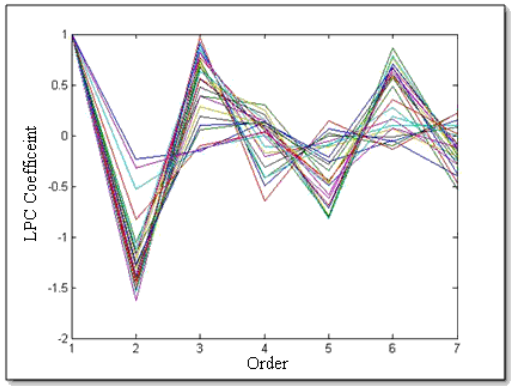


Fig. 2 The LPC coefficients for speaker 4

C. Pattern Matching Analysis

After the LPC coefficients calculated for each speaker, the coefficients will be saved in excel files. The LPC coefficients will be arranged randomly according to its speaker. This set of data will be the training and testing data to this project. The LPC coefficient will the input to the neural network specified in the Matlab software. The specified network will be train using the data provided.

The MLP networks in this project each comprising two layers with nodes ranging from 1 to 16. The number of outputs of each network was the number of speakers that the network is designed to distinguish between. We trained networks to distinguish between two, and four different speakers. The number of epochs that the network trained on is 50 epochs.

III. Speaker Verification With Artificial Neural Network

A. Training Data

After processing the speech signals for each speaker, only the speech signals for four speakers are acceptable for this project. This is due to noise affecting the speech signals.

Once these speech signals had been obtained and preprocessed, a Matlab program was written to extract the features as the inputs for the project network. The target values are set for each set of the inputs, where the value *'1'* to *'4'* is set to be the target values for each speaker. The set of inputs and their target values are then written into a data file. This data file that is written in Microsoft Office Excel software will be the input to the network. There are two set of data file created for this project, where each comprising a different combination of the speakers and the word uttered by the speaker. The data files were named 'TrainA.xls' and 'TrainB.xls'. The values were arranged so that each speaker had the same number of training sample. The training data also were randomized at each data file, so that the network would not train on a long sequence of any one of the speaker's speech signals consecutively. This also allows the network to be trained more evenly among speakers.

The first training file named *'TrainA.xls'*, contain one set of coefficients for each word. In this data file, each speaker will have 10 set of coefficient; where for one set of coefficient contains 390 values. This file also combines the set of coefficients for the four speakers. The second training file named *'TrainB.xls'*, only includes the coefficients for speaker 1 and speaker 2. This file also contains 40 set of coefficient compare to the first training file it only has two type of output that is the value '1' and '2'. Table I shows the two types of data files that were used for this project.

TABLE I
THE TWO TYPES OF DATA FILES

Data files	Types of data files
TrainA.xls	4 speakers (speaker 1, speaker 2, speaker 3, and speaker4); (4 outputs, 40 sets of coefficient)
TrainB.xls	2 speakers (only speaker 1 and speaker 2); (2 outputs, 40 sets of coefficients)

B. Network Structure

The network that was used to implement this project was a feedforward Multi-layer Perceptron (MLP) network. The neural network functions programmed in the Matlab software was used to construct the MLP networks to characterize the project data set. The MLP networks for this project contained only 2 layers with nodes ranging from 1 to 16. For each new training set, the network is initially trained with 1 node.

C. Testing Data

For the testing stage, two set of data file are created. This stage of the project can only be done after the network was properly trained with the training data. The two data files for the testing stage are named 'TestA.xls' and 'TestB.xls'. The first testing file named 'TestA.xls' is created to test the network that had been trained by the inputs data file named 'TrainA.xls'. This file combines 20 set of coefficients form all the four speakers. There are five sets of coefficients taken from each speaker. These sets of coefficients are the values

that had not been used as a training data. The second testing file named 'TestB.xls' is used to test the network that was train by the 'TrainB.xls' file.

The data for the testing file 'TestA.xls' can be classified as known data to the network where the network already been trained with the data from the speaker in the testing file. While for the testing file 'TestB.xls', it can be described as testing with unknown data. For the second testing, the network was only trained with the data for speaker 1 and speaker 2, where the testing file only contains data from the speaker 3 and speaker 4.

This testing methodology is conducted to observe the ability of neural network to recognize an identity of a speaker when the speakers are known or unknown to the network. Table II shows the two types of testing file used in this project.

TABLE II
THE TWO TYPES OF DATA FILES FOR TESTING STAGE

Data files	Types of data file
TestA.xls	4 speakers (speaker 1, speaker 2, speaker 3 and speaker 4); (4 outputs, 20 sets of coefficient)
TestB.xls	2 speakers (only speaker 3 and speaker 4); (2 outputs, 20 sets of coefficients)

IV. RESULTS

In order to illustrate the performance of the neural network that had been trained by the data, two set of data file were created for testing stage. There are two types of testing process conducted in this project; there are testing process with known data and testing process with unknown data.

A. Testing with Known Data

This testing process involving the data files 'TrainA.xls' and 'TestA.xls'. The testing stage begins after the network was properly trained by the data file 'TrainA.xls'. This process is conducted to test whether the trained network is able to recognize the speaker from a given data. The network in this stage was trained to recognize all 4 of the speaker, where the data file 'TrainA.xls' contains the coefficients for the 4 speaker. For the data file 'TestA.xls', it also contains the coefficients for the 4 speakers, therefore the network was tested if it is able to recognize the four speaker based on their coefficients.

The simulation of the trained network with the testing data begins with one node for the network. For each simulation the number of nodes will be increased by one, this is to observe the performance of the network in recognizing the speaker. Table III shows the MSE value for the simulation with nodes ranging from one to five nodes. From Table III, it can be observed that the value of MSE or known as the mean square error decreases when the number of the nodes increases. Based on this result, it can be suggested that increasing the number of nodes will decrease the MSE value. Fig. 3 shows the graph for MSE value in training phase with the nodes ranging from 1 to 16.

TABLE III
MSE VALUE FOR THE TRAINING PHASE WITH KNOWN DATA

Number of nodes	MSE value at the beginning of the process	MSE value at the end of the process
1 node	6.14626	1.19231
2 nodes	9.73982	0.818223
3 nodes	8.22509	0.303929
4 nodes	6.24237	0.11794
5 nodes	10.79	3.28146e-012

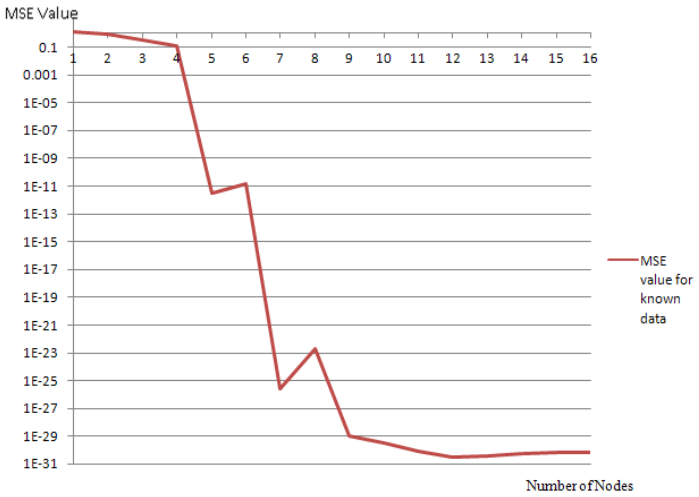
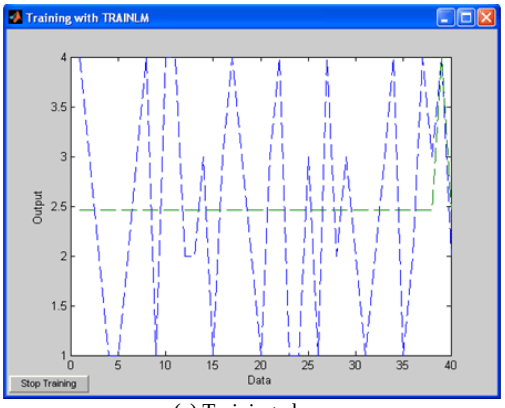
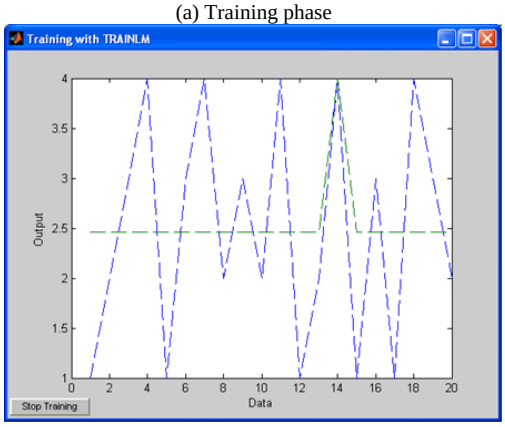


Fig. 3 MSE value for training phase with nodes ranging from 1 to 16

The results for simulation with one node are shown in Fig. 4 (a) and (b). In Fig. 4 and Fig. 5, the blue line indicates the target value while the green line indicates the simulation results. Fig. 4 (a) and (b) show that the network is not able recognize the speaker efficiently by using only one node. It shows that the simulation results are very different from the target value. At this simulation stage, it can be seen that the network can not be trained by only using one node. The efficiency of the network in recognizing the speaker will increase by using more than one node for the network. This can be seen in Fig. 5, where the network was able to differentiate the speaker more efficiently compared to the simulation results by using less than four nodes. In Fig. 5, it illustrates the simulation result after the network was trained by using 12 nodes.

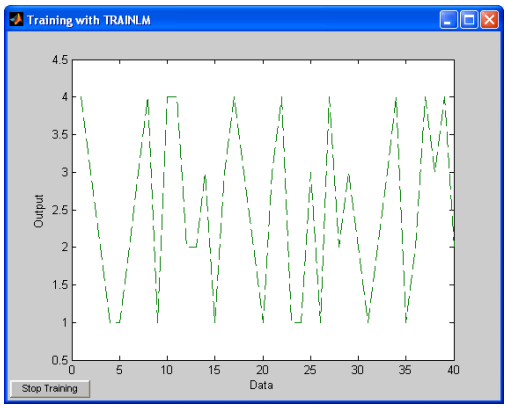


(a) Training phase

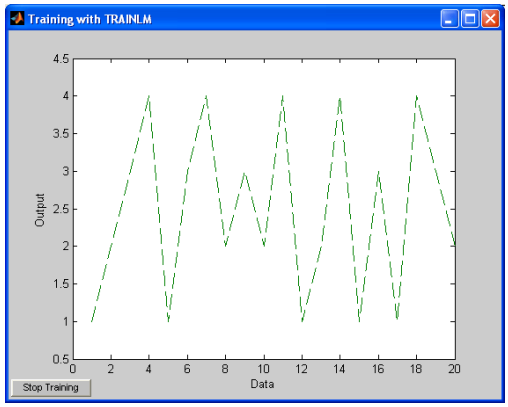


(b) Testing phase

Fig. 4 Results for the training and testing phase with known data by using one node



(a) Training phase



(b) Testing phase

Fig. 5 Results for the training and testing phase with known data by using 12 nodes

Fig. 6 shows the graph for the Coefficient of Determination, R^2, with the nodes ranging from 1 to 16. This value compares the correlation coefficient between the target data values and the outputs produced from the network. From Fig. 6, the value for R^2 varies from 0 to 1. When the value of R^2 is equal to 1, it shows that both of the value for the target data value and the output are matched.

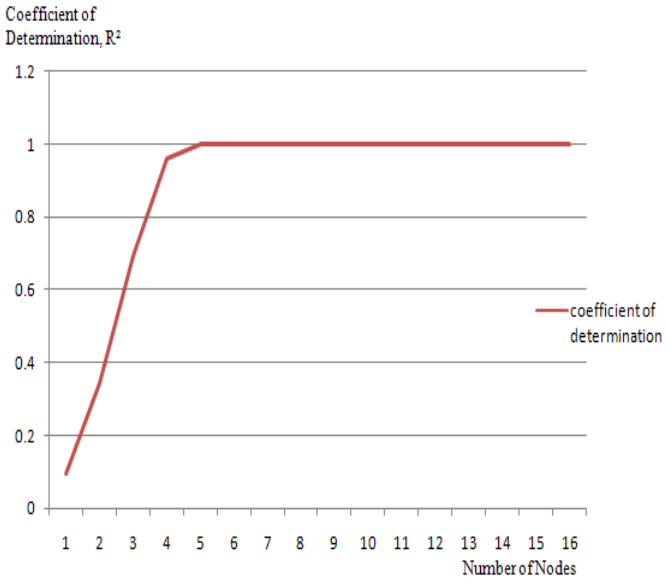


Fig. 6 The Coefficient of Determination, R^2, with the nodes ranging from 1 to 16 for testing with known data

B. Testing with Unknown Data

In the second type of testing, the data file 'TrainB.xls' and 'TestB.xls' were used to conduct the testing process. Mainly this process was conducted to test if the network was able to recognize speaker that its data are not trained to the network. In other word, this process was conducted to test the network with unknown data. Fig. 7 shows the MSE value for the second training phase.

Fig. 8 illustrates the simulation results with unknown data using 12 nodes. From this figure, it shows that only the training simulation results can be accepted depending on the

number of nodes specified in the network. The accuracy of recognizing the speaker in this training phase increased when the number of nodes increased. At the training phase of each simulation, the network is simulating with the same data from the training data. This would allow the network to recognize the speaker based on their data. Unlike the testing phase, the simulation results are completely different from the target values. In the testing phase, the network was only trained to recognize speaker 1 and speaker 2. While the data file used for testing phase are data involving only the speaker 3 and speaker4. Therefore the network won't be able to identify correctly the data for speaker 3 and speaker 4. This can be seen in the figure for testing phase, where the two lines never intersect. In Fig. 9, it shows the graph for the Coefficient of Determination, R^2, with the nodes ranging from 1 to 16 for testing with unknown data. The value for the coefficient of

determination should vary from -1 to +1. From Fig. 9, the value of R^2 varies from -11 to -24. This shows that the network won't be able to generate the correct output for the data provided in the testing phase.

TABLE IV
THE MSE VALUE FOR THE TRAINING PHASE WITH UNKNOWN DATA

Number of nodes	MSE value at the beginning of the process	MSE value at the end of the process
1	5.07447	0.155172
2	4.08986	0.144061
3	8.50888	0.08
4	5.09157	5.53001e-016
5	1.06229	7.07253e-022

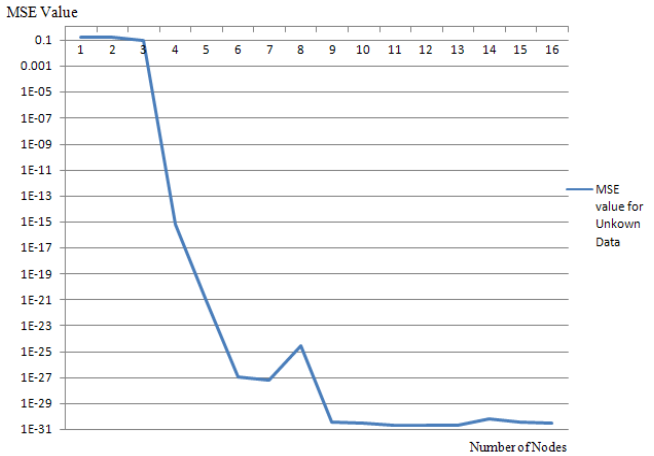
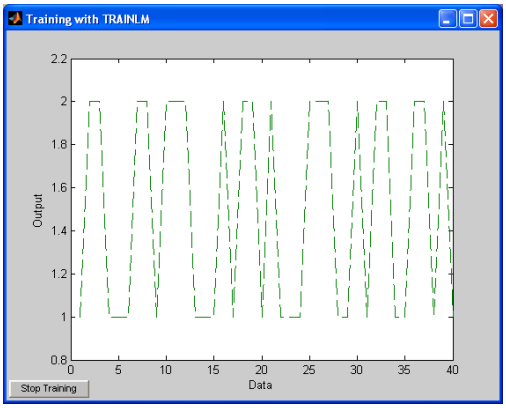
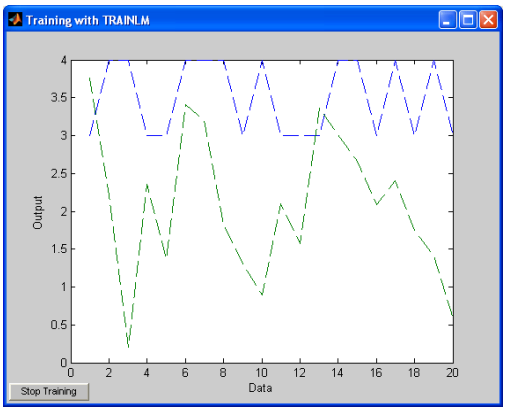


Fig. 7 The MSE value for training phase with nodes ranging from 1 to 16



(a)Training phase



(b)Testing phase

Fig. 8 Results for the training and testing phase with unknown data by using 12 nodes

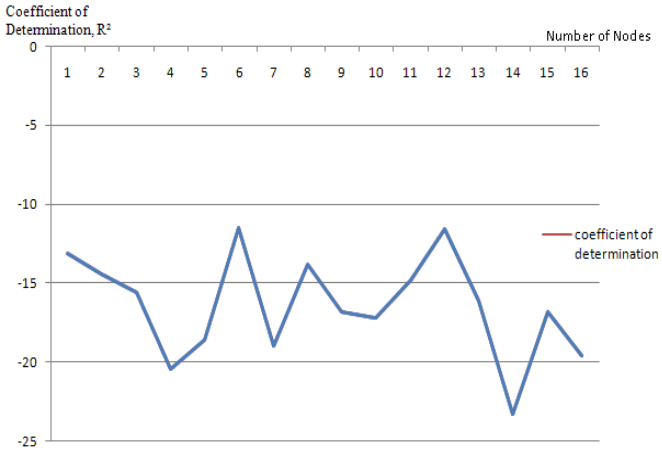


Fig. 9 The Coefficient of Determination, R^2, with the nodes ranging from 1 to 16 for testing with unknown data

V. CONCLUSION

In this paper, the main aim is on the speaker verification based on the speech signal characteristic. This can be achieved by extracting feature from the speech signal and also uses the neural network as a pattern matching tool. The speech signal

had to undergo a few preprocessing stages before its features were extracted using the Linear Prediction Coefficient. The preprocessing phase includes pre-emphasis, framing, windowing and also endpoint detection. The Linear Prediction Coefficient methodology provides the coefficient that can represent the speech signal of a speaker.

The feedforward Multi-layer Perceptron network was used as pattern matching devices in this project, where the network specified will be trained with a data file. The data files in this project were created based on the speaker's coefficient (LPCs). There were two types of testing method conducted in order to test the efficiency of the network specified in this project. Based on the result discussed in section 4, the efficiency of the network on recognizing the speaker will increase when the number of node are increased. Changing the number of epochs for the network also increases the efficiency of the network. After the testing, it also show that the network won't be able to recognize an unknown speaker, where this can be seen in the second type of testing conducted in this work.

Based on the results obtained from this work, it can be concluded that a speaker identity can be verified by using the characteristic of their own speech signal.

REFERENCES

[1] A. Syrdal, R. Bennett & S. Greenspan. 1995. Applied Speech Technology: CRC Press, Inc.
[2] Eric Keller. 1994. Fundametals of Speech Synthesis & Speech Recognition Basic Concepts, State-of-the-Art and Future Challenges: John Wiley & Sons Ltd.
[3] F.A. Westall, R.D. Johston & A.V. Lewis. 1998. Speech Technology for Telecommunications: Chapman & Hall.
[4] G. Chollet, M. Di Benedetto, A. Esposilo & M. Marinaro. 1999. Speech Processing, Recognition & Artificial Neural Network: Springer Verlag London Limited.
[5] Jan Pool. 2002. Investigation of the impact of High Frequency transmitted speech on Speaker Recognition. Electronic Engineering: University of Stellenbosch
[6] WB. Kleijn & K.K. Paliwal. 1995. Speech Coding & Synthesis: Elsevier Science B.V.
[7] Gordon E. Pelton. 1993.Voice Processing. McGraw Hill.
[8] D.G. Childers. 2000. Speech Processing and Synthesis Toolboxes. John Wiley & Sons.
[9] Thomas E. Quatieri. 2002. Discrete-Time Speech Signal Processing. Prentice Hall.
[10] Anil K. Jain. 2006. Biometric Personal Identification in Networked Society. Springer.
[11] James Wayman, Anil Jain, Davide Maltoni, Dario Maio. 2005. Biometric System.Springer.
[12] G. Chollet, M. Di Benedetto, A. Esposito, M.Marinaro. 1999. Speech Processing, Recognition and Artifical Neural Networks. Springer.