Biologically Inspired Speaker Verification

Tariq Tashan

A thesis submitted in partial fulfilment of the requirements of Nottingham Trent University for the degree of Doctor of Philosophy

November 2012

Acknowledgement

I would like to express my deep gratefulness to my director of study Dr. Tony Allen who enriched this research through his guidance and support, making this research journey full of experience and knowledge. I must also thank my second supervisor Dr. Lars Nolle for his support and advice throughout this research.

I would like to thank all my friends and colleagues in Nottingham Trent University for their encouragement and support for the last four years.

I gratefully acknowledge all the love and support that my brothers, my sweet sister, my brothers and sisters in law provide me with.

I am thankful to my soul mate, my lovely wife *Rokaia* for her love, encouragement, patience and support at all times and especially during the last four years.

I would like to thank my parents for all the good values they set in me. I wish my mother a happy long life. I pray for my father, may his soul be forgiven and rest in peace … wish you were here Dad.

Tariq Tashan

November 2012

Abstract

Speaker verification is an active research problem that has been addressed using a variety of different classification techniques. However, in general, methods inspired by the human auditory system tend to show better verification performance than other methods. In this thesis three biologically inspired speaker verification algorithms are presented.

The first is a vowel-dependent speaker verification method that uses a modified Self Organising Map (SOM) algorithm. For each speaker, a seeded SOM is trained to produce representative Discrete Fourier Transform (DFT) models of three vowels from a spoken input using positive samples only. This SOM training is performed both during a registration phase and during each subsequent verification attempt. Speaker verification is achieved by computing the Euclidean distance between the registration and verification SOM trained weight sets. An analysis of the comparative system performance when using DFT input vectors, as well as Linear Prediction Code (LPC) spectrum and Mel Frequency Cepstrum Coefficients (MFCC) alternative input features indicates that the DFT spectrum outperforms both MFCC and LPC features. The algorithm was evaluated using 50 speakers from the Centre for Spoken Language Understanding (CSLU2002) speaker verification database.

The second method consists of two neural network stages. The first stage is the modified SOM which now operates as a vowel clustering stage that filters the input speech data and separates it into three sets of vowel information. The second stage then contains three Multi Layer Perceptron (MLP) networks; each acting as a distinct vowel verifier. Adding this second stage allows the use of negative sample training. The input of each MLP network is the respective filtered output vowel data from the first stage. The DFT spectrum is again used as the input feature vector due to its optimal performance in the first algorithm. The overall system was evaluated using the same dataset as used in the first algorithm, showing improved verification performance when compared to the algorithm without using the MLP stage.

The third biologically plausible method is a speaker verification algorithm that uses a positive-sample-only trained self organising map composed of spiking neurons. The architecture of the system is inspired by the biomechanical mechanism of the human auditory system which converts speech into electrical spikes inside the cochlea. A spike-based rank order coding input feature vector is proposed that is designed to be representative of the real biological spike trains found within the human auditory nerve. The Spiking Self Organising Map (SSOM) updates its winner neuron only when its activity exceeds a specified threshold. The algorithm is evaluated using the same 50 speaker dataset from the CSLU2002 speaker verification database and the results indicate that the SSOM verification performance is comparable to the non-spike based SOM.

Finally, a new speech detection technique to detect speech activity within speech signals is also proposed. This novel technique uses the linear correlation coefficient (Parson Coefficient). The correlation is calculated in the frequency domain between neighbouring frames of DFT spectrum feature vectors. By summing the correlation coefficients within a sliding window over time, a correlation envelope is produced, which can be used to identify speech activity. The proposed technique is compared with a conventional energy frame analysis method and shows greater robustness against changes in speech volume level. A comparison of the two techniques, in terms of speaker verification application performance, is presented in Appendix A using 240 speech waveforms from the CSLU2002 speaker verification database.

Copyright Statement

Table of Content

List of Figures

List of Tables

List of Abbreviations

AANN	Auto Associative Neural Network
AGC	Automatic Gain Control
ANN	Artificial Neural Networks
CCE	Correlation Coefficient Envelope
CSLU	Centre for Spoken Language Understanding
DCT	Discrete Cosine Transform
DFT	Discrete Fourier Transform
ED	Euclidian Distance
EER	Equal Error Rate
EFA	Energy Frame Analysis
FAR	False Accept Rate
FFMLP	Feed Forward Multi Layer Perceptron
FNN	Fuzzy Neural Network
FRR	False Reject Rate
GMM	Gaussian Mixture Model
GNN	Generalised Neural Network
HMM	Hidden Markov Model
LCC	Linear Correlation Coefficient
LP	Linear Prediction
LPC	Linear Prediction Code

LPCC	Linear Predictive Cepstral Coefficients
LVQ	Learning Vector Quantisation
MAER	Minimum Average Error Rate
MFCC	Mel Frequency Cepstrum Coefficients
MLP	Multi Layer Perceptron
PNN	Probabilistic Neural Network
RBF	Radial Basis Function
SNN	Spiking Neural Network
SOM	Self Organising Map
SSOM	Spiking Self Organising Map
SVM	Support Vector Machine
ZCR	Zero Crossing Rate

List of Publications

Journal paper:

Tashan, T., T. Allen and L. Nolle (2012). "Speaker verification using heterogeneous neural network architecture with linear correlation speech activity detection." Accepted for publication in Expert Systems: *The Journal of Knowledge Engineering*.

Conference papers:

Tashan, T., T. Allen and L. Nolle (2011). Vowel based speaker verification using self organising map. The Eleventh IASTED International Conference on Artificial Intelligence and Applications (AIA 2011), Innsbruck, Austria, ACTA Press.

Tashan, T. and T. Allen (2011). Two stage speaker verification using Self Organising Map and Multilayer Perceptron Neural Network. Research and Development in Intelligent Systems XXVIII. M. Bramer, M. Petridis and L. Nolle, Springer London: 109-122.

Tashan, T., T. Allen and L. Nolle (2012). Biologically inspired speaker verification using Spiking Self-Organising Map. Research and Development in Intelligent Systems XXIX. M. Bramer and M. Petridis, Springer London.

Chapter One

Introduction

1. Introduction

Speaker recognition is the process of classifying individuals from their speech signals. This process can be sub-divided into two main application tasks; speaker identification and speaker verification. Speaker identification is when an unknown speech signal is identified as belonging to one speaker from a set of known speakers. Since no identity is claimed, the unknown speech signal must be compared to speech signals of all speakers in the known set. This type of problem is often called ‘closed-set’ due to the prior knowledge that the unknown speech signal belongs to one of the speakers in the set and the goal is to find the identity of the speaker. Speaker verification is when an unknown speech signal is classified as either belonging to or not belonging to a claimed known speaker. Here, the unknown speech signal either belongs to a claimed known speaker or belongs to an 'impostor'. In this case, the impostor cannot be known a priori as in speaker identification. Therefore, speaker verification is referred to as an 'open-set' problem.

When developing voice biometric authentication systems there are several design parameters that need consideration. The first is the type of classifier to use. Over the last two decades, speaker recognition (identification and verification) has been investigated using a wide range of methods. Some of the popular approaches being: probabilistic models such as Hidden Markov Model (HMM) and Gaussian Mixture Model (GMM) classifiers (Reynolds and Rose 1995), non-probabilistic binary linear models such as Support Vector Machine (SVM) classifier (Campbell *et al.* 2006) and non-linear statistical models i.e. Artificial Neural Networks (ANN) (Oglesby and Mason 1991; Farrell *et al.* 1994; Monte *et al.* 1996; Kishore and Yegnanarayana 2000; George *et al.* 2001; Kusumoputro *et al.* 2001; Mueen *et al.* 2002; Seddik *et al.* 2004a). A variety of different types of neural networks have been used to perform the speaker recognition task: Multi Layer Perceptron (MLP) (Seddik *et al.* 2004a), Radial Basis Function (Oglesby and Mason 1991), Neural Tree Network (Farrell *et al.* 1994), Auto Associative Neural Network (Kishore and Yegnanarayana 2000), Recurrent neural networks (Mueen *et al.* 2002), Probabilistic neural networks (Kusumoputro *et al.* 2001), Dynamic synapse based neural networks (George *et al.* 2001) and Self Organising Map (SOM) (Monte *et al.* 1996). Most modern voice biometric authentication systems employ GMM based methods in the verification engine; an offshoot of earlier research into the use of HMM algorithms for speech recognition systems. SOM based speaker recognition systems, on the other hand, are attractive alternatives to the conventional methods because they offer the potential of being able to do away with the need for the speech recognition front-end commonly included in speaker recognition systems (Ouzounov 1997).

Having decided on the classifier methodology to use, the next major decision is which morphological level is to be used to extract the features. Since speech signals contain both language information as well as speaker identity information, speaker recognition can be achieved by processing the speech signal at a variety of levels (sentence, word, syllable or phoneme). It has been shown that more information about the identity of the speaker can be obtained by processing the speech at the phoneme level (Han-Sheng and Mammone 1995a). However, the disadvantage of processing at this level is that an efficient speech recognition algorithm is required in order to locate the positions of the phonemes within the speech signal prior to the feature extraction stage. The penalty for using such speech recognition tools in speaker recognition systems is the need for substantial speech data in order to train the speech recognition engine. As a consequence, the speaker recognition performance of such systems has been shown to fall dramatically when only limited training data is available (Jayanna and Prasanna 2009). The limited data condition is when provided speech data is less than 15 seconds as defined in (Jayanna and Prasanna 2009) and (Angkititrakul and Hansen 2007). An alternative to this approach would be to detect the phoneme boundaries without using a speech recognition engine (Dong *et al.* 2002; Zhang *et al.* 2009).

In addition to the level at which the speech recognition is processed, there are three major formats in which the features can be extracted from the speech sample. A straightforward and simple representation of the speech signal in the frequency domain is the Discrete Fourier Transform (DFT) spectrum. This type of format is commonly used in speech and speaker recognition applications (Rabiner and Schafer 2010). The DFT spectrum can be obtained by calculating the magnitude of the DFT vector (Rabiner and Schafer 1978). Another feature format that preserves the speech signal characteristics are the Linear Prediction Coefficients (LPC); often used for speech compression tasks. The LPC spectrum is calculated by taking the magnitude value at the output of a filter whose coefficients are represented by the LPCs (Rabiner and Schafer 1978). Mathematically, the LPC spectrum represents a smoothed version of the DFT spectrum. In speech and speaker recognition applications the most widely used feature format are the Mel-Frequency Cepstrum Coefficients (MFCC). These features are calculated by firstly passing the DFT spectrum through a bank of triangular filters with Mel-frequency scale. The MFCC are then calculated by applying the Discrete Cosine Transform (DCT) to the logarithmic output of these filters (Davis and Mermelstein 1980). Although the MFCC are the most preferred input feature formats in the literature, there is evidence to suggest that these may not be optimal for neural network based systems (Sun *et al.* 1991).

The final design decision then is the sampling frequency that is to be used to produce the frames of data from which the features are to be extracted. Most modern systems are required to use 8 kHz in order to be used over standard telecommunication channels.

1.1 Research Motivation

The human auditory system is a sophisticated mechanism, which enables people to both understand the speech signal and recognise the speaker. This implies that the human brain, in combination with its auditory system, contains both speech recognition and speaker verification processing capability. It is known that the speech recognition system is usually developed within an average period of four years after birth (Ramscar and Gitcho 2007). However, there is evidence that a speaker verification capability is developed in very early stages of development; i.e. babies can recognise their mothers' voices well before they learn to understand even basic language (Mehler *et al.* 1978). This leads to the conclusion that the speaker verification system is effectively developed well before the speech recognition system, and that speaker verification in humans is thus language-independent and phoneme-based; in agreement with the study in (Han-Sheng and Mammone 1995a). Nevertheless, higher morphological level processing can provide extra information about the identity of the speaker through behavioural characteristics such as accent, rhythm, intonation style, pronunciation pattern and choice of vocabulary (Kinnunen and Li 2010). Inclusion of these parameters, within the mature human speaker verification system, potentially improves speaker verification accuracy but at the expense of adding complexity to the system. Another property of the human speaker verification processing capability is that it can make decisions using only very limited speech data for both training and testing phases. This aspect of human speech processing also supports the idea that it is the lower morphological levels, phonemes etc., which are responsible for such decisions.

Achieving speaker verification under limited speech data conditions is a commercially valuable requirement (Angkititrakul and Hansen 2007). However, it has been demonstrated that industrial speaker verification systems, which include a speech recognition engine, suffer from a substantial decrease in verification performance when implemented in limited speech data environments (Jayanna and Prasanna 2009).

1.2 Research Aim and Objectives

Taking all the above into consideration, the aim of this work is to develop a speaker verification method that is inspired by the mechanism of the human auditory system in order that it can operate without the need for a speech recognition front-end. This aim can be achieved through three main objectives:

- Phoneme-based SOM speaker verification system: This investigates the use of an SOM for speaker verification based on the similarity between the tonotopic nature of the auditory nerve response (Young 2008) and the topological nature of the SOM (Kohonen 1990). Using a phoneme-based SOM for speaker verification will allow phoneme classification without the need for building a complete speech recognition engine. An SOM solution is also trainable using only positive samples.

- Hybrid speaker verification using SOM + MLP: This investigates how much extra verification performance can be gained by using negative training samples, when employing the phoneme-based SOM as a first coarse speaker verification stage, followed by an MLP network as second fine speaker verification system.

- Spiking SOM speaker verification: This investigates a biologically plausible model that uses Spiking SOM for speaker verification with biologically inspired spike-based features.

1.3 Thesis Organisation

The rest of this thesis is structured as follows: Chapter 2 describes the experimental infrastructure including an introduction to the SOM and Learning Vector Quantisation (LVQ) algorithms in general and the speech database which is used in this research. Then an introduction of the physiology of hearing in the human auditory system is presented. Chapter 3 provides an overview of the speaker verification research, presented in the literature, using different types of neural networks. Chapter 4 then explains the different speech feature formats for speaker verification and how the sampled speech signal is pre-processed. Chapter 5 then proposes a modified SOM for speaker verification using only three vowels and DFT spectrum components as the input feature vector. The algorithm is evaluated using 50 speakers from the CSLU2002 speaker verification database. Chapter 5 also presents a two-stage speaker verification using the modified SOM, followed by an MLP neural network, trained with both positive and negative speech training samples, with evaluation of verification performance using CSLU2002 database. Chapter 6 then proposes a spike-based feature vector, which is inspired by the nature of transmitted spikes over the auditory nerve. The new feature vector is then used as input feature vector to a Spiking SOM. Finally the spiking SOM is evaluated using the same speakers set of the CSLU2002 speaker verification database. Chapter 7 discusses the findings of the different experiments suggested in this research with final conclusions and recommendations for future work in this research area.

Chapter Two

Experimental Infrastructure

2. Introduction

This chapter presents a brief background to the neural network methods used in this research, an introduction to the speech database used to evaluate the proposed algorithms, and an introduction to the physiology of hearing in the human auditory system. Section 2.1 provides theoretical background to the self organising map, Section 2.2 demonstrates the learning vector quantisation, Section 2.3 describes feed forward MLP neural networks, and Section 2.4 introduce the CSLU2002 speaker verification database, while Section 2.5 details the physiology of hearing in the human auditory system. Finally Section 2.6 provides a summary of this chapter.

2.1 Self Organising Map

The Self organising map or Kohonen map (Kohonen 1990) is a competitive neural network model that can classify input patterns into clusters without supervision. The SOM consists of an input layer that represents the input patterns and an output layer or (Map) that represents the possible clusters which can be classified. The SOM can be implemented in one or multidimensional architectures. Figure 2.1 shows an example of a two dimensional (5 x 7) SOM (Pandya and Macy 1995). The clustering algorithm can be described as follows:

1) Initialising the weight links between the input layer and the output layer.

2) A Euclidian distance is calculated between an input pattern in the input layer and all the weight vectors of the neurons in the output layer as shown below:

$$D(X,Y) = \sqrt{\sum_{i=1}^{N} (x_i - y_i)^2} \qquad (2.1)$$

where X=x_1, x_2, ..., x_N is the input vector, Y=y_1, y_2, ..., y_N is the output vector and N is the vector size.

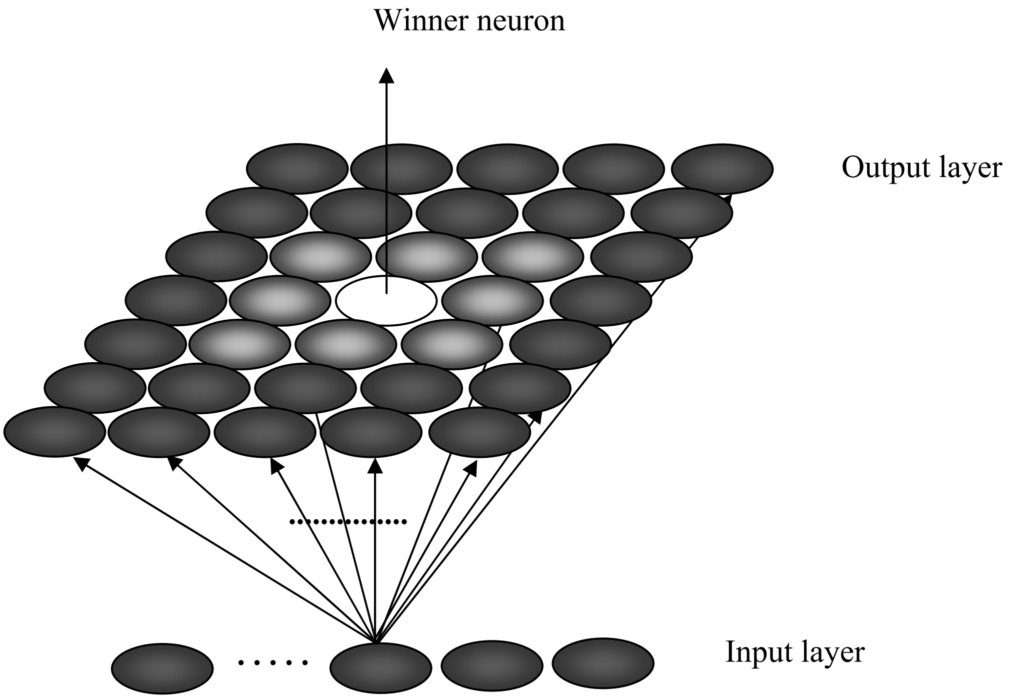

Figure 2.1 – An example of two dimensional (5 x 7) Self Organising Map.

3) The neuron with the minimum distance is then designated as the winner neuron.

4) After finding the winner neuron an update of the weights is applied for the winner neuron weight vectors and all neighbouring neuron weight vectors within a specific region *R(t)* using Equation 2.2.

$$W_{t+1} = \begin{cases} W_t + \beta(t)\,(X - W_t), & W_t \in R(t) \\ W_t & otherwise \end{cases} \tag{4.2}$$

where W_t is the old weight, W_{t+1} is the new weight, X is the input vector and $\beta(t)$ is the learning rate. The learning rate value starts with a value of less than 0.25 and decreases over iterations. The neighbourhood region function *R(t)* also starts with the whole size of the map and decreases gradually to end with updating only the winner neuron (Pandya and Macy 1995).

Steps 2 to 4 are applied for all input patterns for a specific number of iterations (epochs). After training each winning neuron is then a representative for the training patterns for which it is the designated winner.

2.2 Learning Vector Quantisation

The Learning Vector Quantisation (LVQ) is a self organising map with supervised learning. In the LVQ the map is split into groups of neurons, each group implementing one cluster. Figure 2.2 shows an example of an LVQ network.

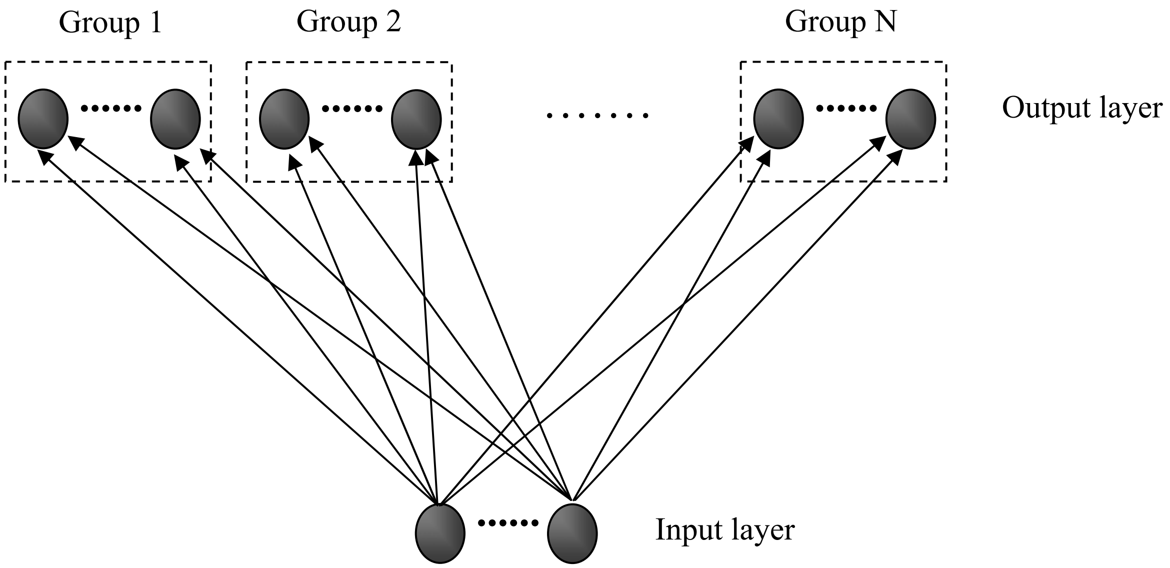

Figure 2.2 – An example of Learning Vector Quantisation network.

The clustering algorithm of the LVQ is similar to the clustering algorithm of the SOM except for step 4. After finding the winner neuron an update is applied for the winner neuron weight and all neurons weights within the same group using Equation 2.1. The weights for the other groups are updated using Equation 2.3.

$$W_{t+1} = \begin{cases} W_t + \beta(t)\,(X - W_t), & \textit{correct group} \\ W_t - \beta(t)\,(X - W_t), & \textit{incorrect group} \end{cases} \tag{2.3}$$

The role of Equation 2.3 is to update the weights in the other groups in the opposite direction to the winner neuron and its group. Other versions of the LVQ system can be obtained by modifying Equation 2.2 and Equation 2.3; more details on such modifications can be found in (Pandya and Macy 1995).

2.3 Feed Forward Multi Layer Perceptron

A Feed Forward Multi Layer Perceptron (FFMLP) or MLP is one of the most popular neural networks used for pattern classification. An MLP must contain an input layer and an output layer, and can contain one hidden layer or more. Each layer consists of a number of neurons. The number of neurons in the input and output layers are commonly determined by the problem whilst the number of hidden layer neurons is optimised for a specific classification task. Figure 2.3 shows an example of a three layer MLP network of (5 x 3 x 1) neurons.

After passing an input pattern vector to the input layer, the value of the vector at each node in the input layer is multiplied by weights corresponding to neurons in the next layer. Each neuron in the hidden layer and output layer operate by computing the sum of its weighted input, and passing the results into a nonlinear activation function.

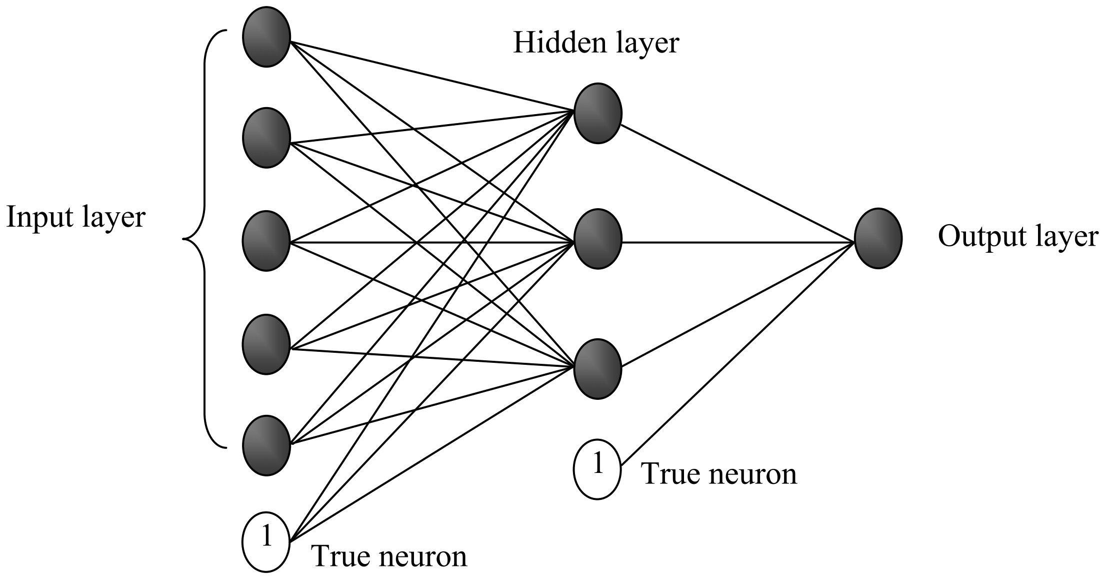

Figure 2.3 – An example of three layer Multi Layer Perceptron network of (5 x 3 x 1) neurons.

The mathematical representation of this neural process can be described as follows:

$$out_i = f(b_i + \sum_j W_{ij}\ out_j) \tag{2.4}$$

where out_i is the output of the i^{th} neuron in the considered layer, out_j is the output of the j^{th} neuron in the previous layer, W_{ij} is the weight connecting the two neurons and b_i is the bias weight of the i^{th} neuron-multiplied by a true neuron as shown in Figure 2.3. There are several types of nonlinear activation function *f*. One of the most frequently used is the sigmoid function:

$$f(x) = \frac{1}{1 + e^{\frac{-x}{Q}}} \tag{2.5}$$

where *x* is the input of the activation function calculated in Equation 2.4 and *Q* is the temperature of the neuron. The sigmoid function changes more gently with higher temperatures, whilst at very low temperatures the sigmoid function behaves as a step function. Figure 2.4 shows the sigmoid function at different temperatures.

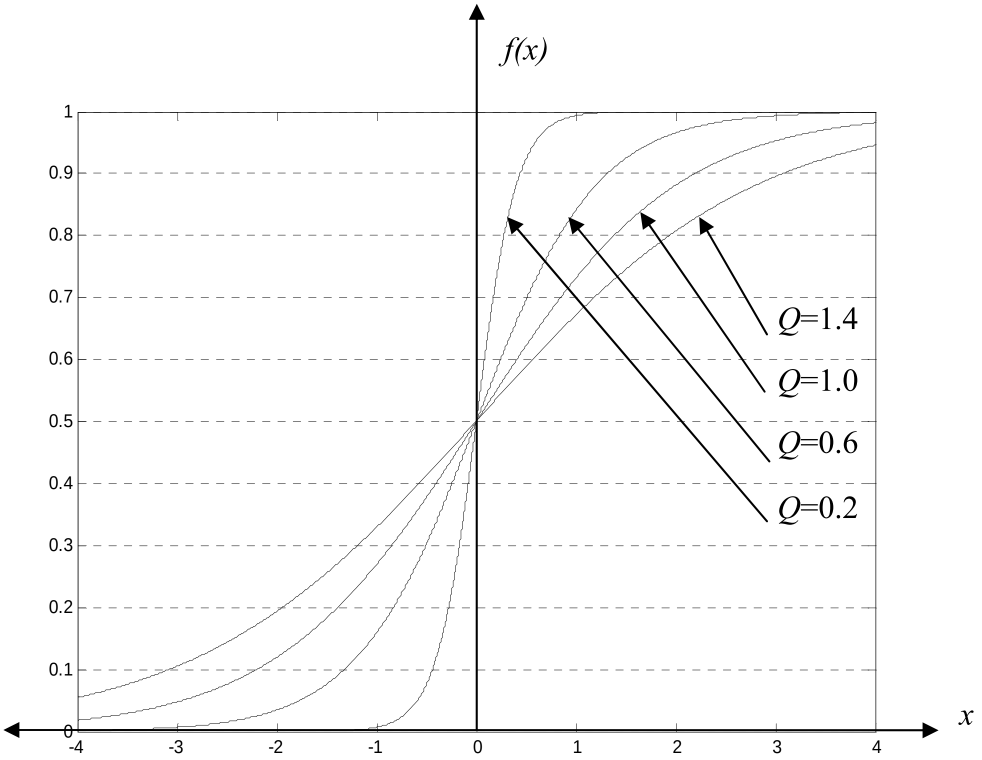

Figure 2.4 – The Sigmoid function at different temperatures.

The training of an MLP network is usually accomplished by using the Back Propagation algorithm. The back propagation training algorithm is employed through three main steps:

i) Feed forward an input training pattern through the network using Equation 2.4 at each neuron.

ii) Calculate the error at the output layer and error in previous layers in the back propagation path. For each individual output layer neuron the error is calculated as:

$$error_k = out_k (1 - out_k)\,(d_k - out_k) \qquad (2.6)$$

for the k^{th} neuron $error_k$ is the error, out_k is the output, d_k is the desired output. And the error at the hidden layer node is:

$$error_j = out_j\ (1 - out_j) \sum error_k\ W(t)_{jk} \qquad (2.7)$$

where $W(t)_{jk}$ is the weight connecting the j^{th} neuron in the considered layer to the k^{th} neuron the output layer.

iii) Update the weights connecting neurons across different layers as below:

$$W(t+1)_{jk} = W(t)_{jk} + \beta\ error_k\ out_j + \tau\,[W(t)_{jk} - W(t-1)_{jk}] \qquad (2.8)$$

$$W(t+1)_{ij} = W(t)_{ij} + \beta\ error_j\ out_i + \tau\,[W(t)_{ij} - W(t-1)_{ij}] \qquad (2.9)$$

where $W(t+1)_{jk}$ is the new weight value connecting the k^{th} neuron in considered layer with the j^{th} neuron from previous layer, $W(t)_{jk}$ is the present value of the same weight, $W(t-1)_{jk}$ is the old value of the same weight, τ is the momentum factor, β is the learning rate, $error_k$ is the error of the k^{th} neuron as calculated in step (ii) and out_j is the output of the j^{th} neuron from previous layer. The weights, error and outputs in Equation 2.9 similarly correspond to the neurons in the i^{th} and j^{th} layer respectively.

The three steps explained above represents one training step. A pass through all training patterns is one epoch. The training stops when a validation error (calculated using a separate validation dataset) starts to increase or remains constant over several epochs (Pandya and Macy 1995).

2.4 CSLU2002 Speaker Verification Database

Speaker verification problem has been investigated using different speech databases, such as the Texas Instrument and Massachusetts Institute of Technology (TIMIT) database, the National Institute of Standards and Technology (NIST) database, and the YOHO database. Speech databases usually updated over different releases. Some organisations release more than one speech database based on its application whether it is for speech recognition or speaker recognition. There are some key parameters that define the speaker recognition database, such as the sampling frequency which has been used to capture the speech signal, the recording channel, and the background environments. Due to availability, the proposed algorithms in this research were evaluated using the CSLU2002 speaker verification database. CSLU2002 is a commercially available speaker verification database from the Oregon Graduate Institute of Science and Technology Research Centre in Spoken Language Technologies. The database consists of 91 speakers, from which 50 speakers were arbitrarily selected (27 females and 23 males) for use in this work. Some speakers were not used due to missing recoding speech data over the two chosen sessions. The data were recorded over digital telephone lines with a sampling frequency of 8 kHz to produce 8-bit u-law files, which are then encoded into 8 kHz 16-bit wave format file. Two recording sessions (Session 1 for registration and Session 2 for testing) samples are used for evaluation, each session containing four samples for each speaker. Proposed algorithms that use only positive samples in this research were evaluated using the 50 speakers, while the algorithm that uses both positive and negative samples was evaluated using the first 30 speakers, where the rest of the speakers were saved to be used as unseen speaker speech data for testing (see Section 5.3.2). More information on the CSLU2002 database can be obtained on the website "http://www.cslu.ogi.edu/corpora/spkrec/index.html".

2.5 Physiology of Hearing

Sound waves are captured in the human auditory system over three stages: outer ear, middle ear and inner ear (cochlea). The outer ear consists of the pinna or the ‘concha’ which is the external visible part of the ear and the ear canal. The main two tasks of the concha are to collect the sound vibrations and introduce position information into the incoming sound. The ear canal works as a resonator with a peak frequency of 3 kHz (Møller 2006).

The middle ear consists of the tympanic membrane (the first part in the auditory system that converts sound vibrations into mechanical movement) connected to a combination of three small bones (ossicles): the malleus, the incus and the stapes. The tympanic membrane moves according to the sound vibrations received at the end of the ear canal, transferring the movement freely across the ossicles. This movement reaches the stapes which has a footplate end which is connected to the cochlea through an oval window. Figure 2.5 shows the structure of the ossicles. The ossicles, shown in Figure 2.5, are supported inside the middle ear cavity by several ligaments. Two important ligaments are the anterior malleal ligament and the posterior incudal ligament. These uphold the movement axis of the ossicles during the transmission of the sound vibration waves. It is obvious from Figure 2.5 that the tympanic membrane surface area is larger than the rounded end of the stapes (footplate), this difference causes an amplification in the sound vibration waves.

As well as the ligaments that support the ossicles, the tensor tympani muscle and the stapedius muscle also provide the malleus and the stapes with extra function. The tension of the tensor tympani muscle dampens the vibration of the tympanic membrane in a high intensity sound scenario (Møller 2006). The stapedius is the smallest muscle in the human body, its function is to protect the stapes from harsh movements and stabilise the amplitude of sound. Together the two muscles act as a first automatic gain controlling stage in the human auditory system; helping to moderate the received signal vibrations under high sound intensity conditions.

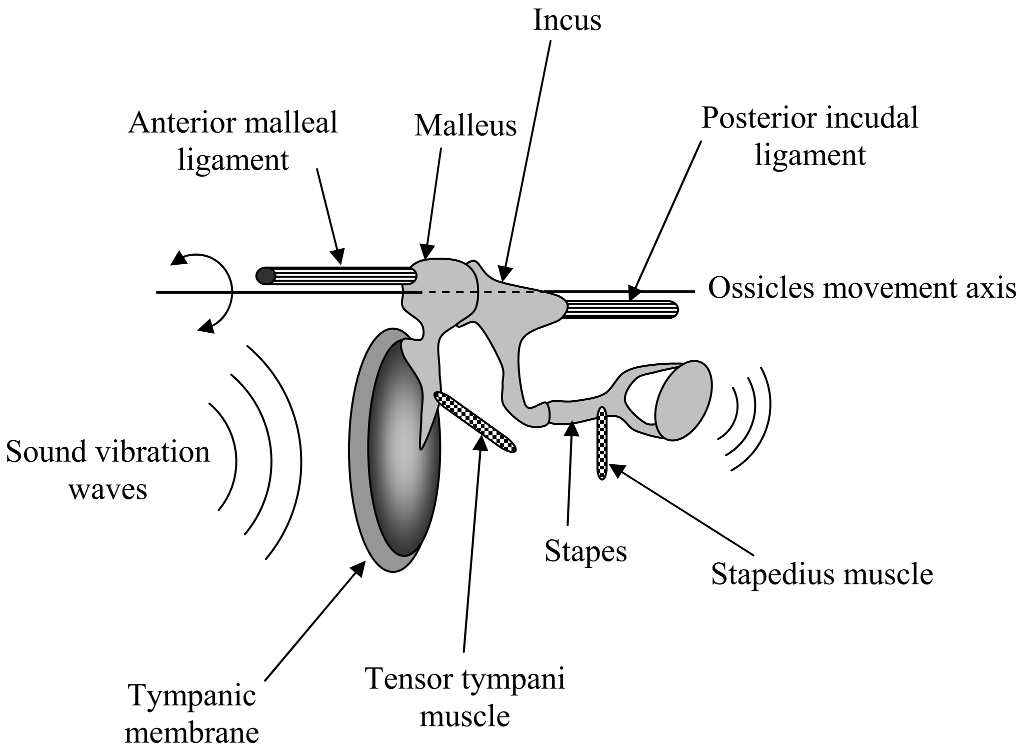

Figure 2.5 – The structure of the ossicles.

The cochlea is the auditory part of the inner ear. Its function is to convert the mechanical movement of the tympanic membrane and the ossicles into electrical signals within the auditory nerve. It has a snail-shaped bony structure of about two and a half turns with an uncoiled length of 3.1 to 3.3 cm. The hardness property of the cochlea makes it a perfect design for capturing sound vibrations without absorption. Along the fluid-filled tube inside the cochlea is the ‘basilar membrane’ (see Figure 2.6), which splits the tube into two parallel tubes connected at their end. The front-end of one tube is connected with the oval-shaped footplate of the stapes bone, via the tympanic membrane, receiving the vibrations in a piston like movement. This vibratory movement of the stapes is converted into fluid pressure waves that travel along the basilar membrane. The flexible membrane of the rounded window at the end of the cochlea duct allows the pressure wave to propagate easily, following the piston movement of the stapes. These pressure waves, in turn, displace the basilar membrane at specific positions corresponding to the frequency components of the received signal (Møller 2006).

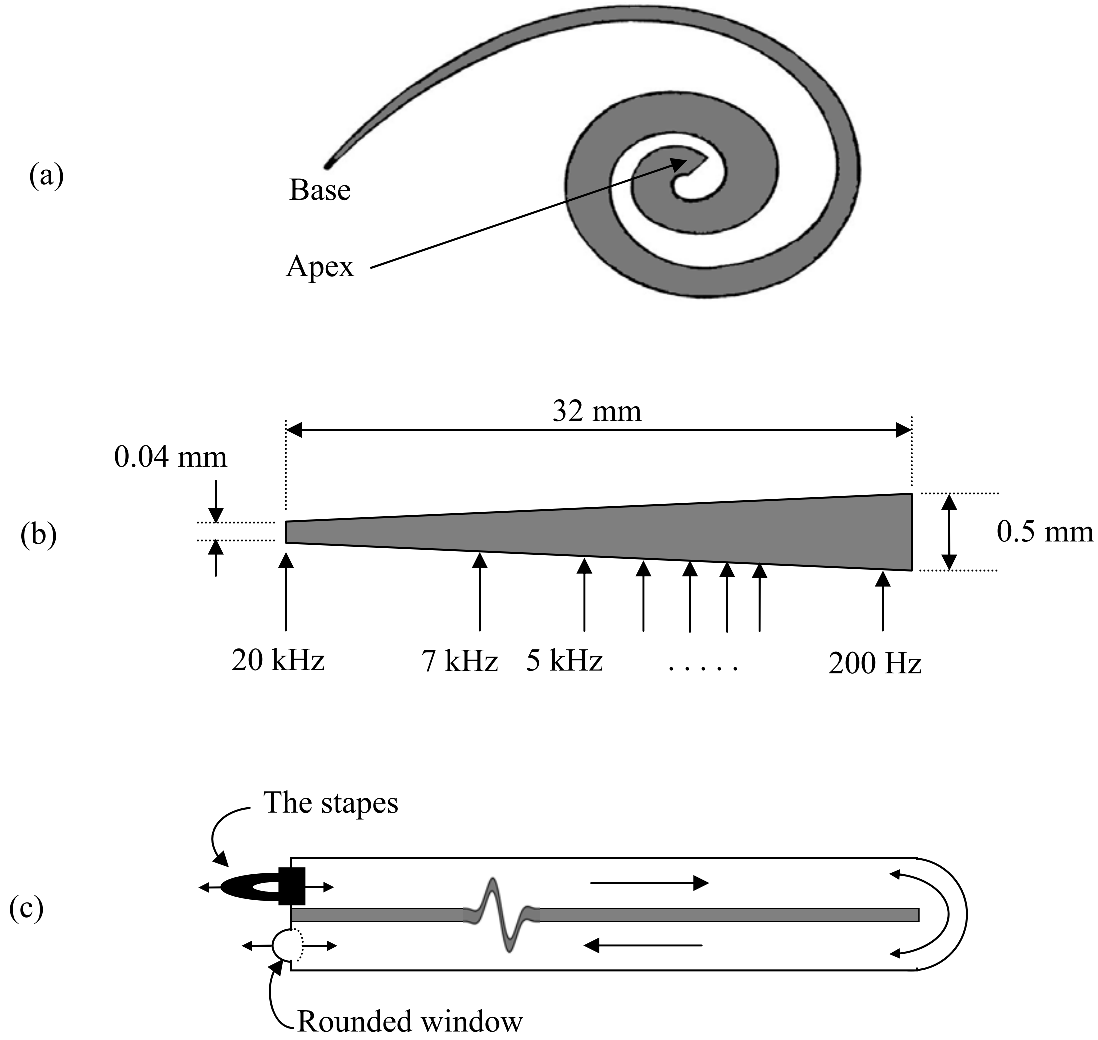

Figure 2.6 – Different views of the basilar membrane a) Spiralled top view b) Unfolded top view, showing dimensions and frequency sensation positions c) Side view, showing how the movement of the stapes is propagated as a pressure wave inside the cochlea duct.

Along the basilar membrane lies the sensory organ of hearing called the organ of Corti. The Corti contains the 'hair cells' which are connected directly to the auditory nerve. When the basilar membrane vibrates the hair cells at that position 'bend' converting the mechanical movement into electrical pulses through a bio-chemical process (Møller 2006). Figure 2.7 shows the organ of Corti in a cross-sectional view of the basilar membrane.

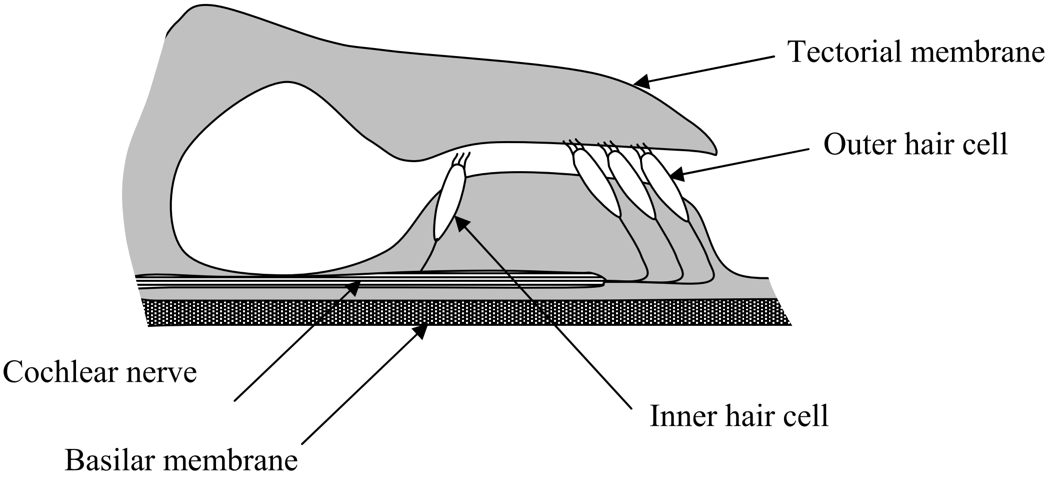

Figure 2.7 – The organ of Corti.

As shown above, the organ of Corti contains two types of hair cells, the inner hair cells and the outer hair cells. The outer hair cells are laid in 3-5 rows along the Corti, while the inner hair cells lie in one row. Both types are connected to the auditory nerve via the cochlear nerve. Each hair cell has a bundle of hairs (stereocilia) at its tip. The outer hair cells stereocilia are embedded in the Tectorial membrane, as shown in Figure 2.7, while the inner hair cell stereocilia are not.

The main difference between the two types of hair cells is their function. The inner hair cells are responsible for capturing the mechanical movement of the basilar membrane and converting it into a train of spikes. As illustrated in Figure 2.6c, when the stapes transfers sound vibration energy into the cochlea, the basilar membrane will move at a position dependant on the frequency of the vibration. This movement occurs as a vertical displacement in the basilar membrane causing horizontal shear motion in the Tectorial membrane. This latter motion is then transferred directly to the outer hair cells, and indirectly to the stereocilia of the inner hair cells through the Endolymph fluid. The inner hair stereocilia contain mechanically gated ion channels, which open when the stereocilia bend. This produces a membrane potential difference which, in turn, generates an electrical spike that travels through the efferent nerve fibers to the auditory nerve.

The outer hair cells react to the same movement in different way. They are connected to the auditory nerve via afferent nerve fibers which control the stiffness of the outer hair stereocilia. The outer hair stereocilia become stiffer when excessive movement appears (i.e. high sound amplitude). The role of the outer hair cells here is as a second automatic gain control stage, damping the high amplitudes of sound in order to protect the inner hair cells from excessive movement. Mathematically speaking, the out hair cells provide logarithmic scale to the intensity of the sound wave.

As each hair cell is connected to a unique auditory nerve, the position of the active inner hair cells on the basilar membrane provides frequency information to the cochlear nucleus section of the brain. At low sound levels, the intensity of frequency components are represented by the rate of spikes generated by the inner hair cells movement (Young 2008). Figure 2.8 shows how the captured spectrum by the basilar membrane is transformed into discharged spike trains travelling through the auditory nerve.

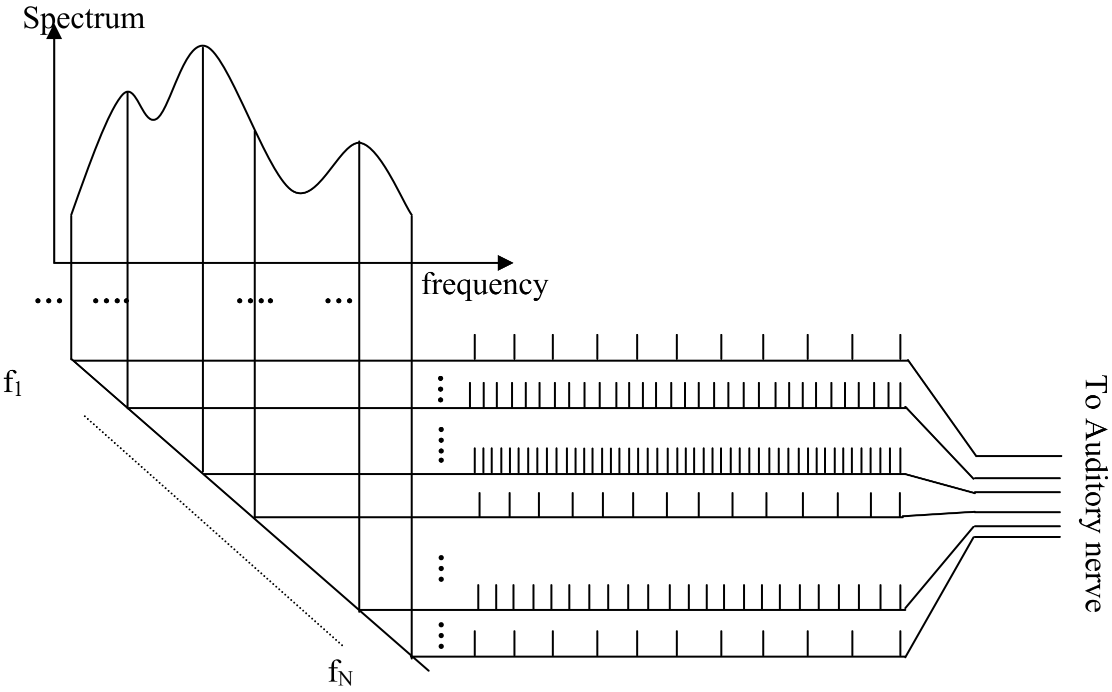

Figure 2.8 – The process of converting the captured spectrum into spike rates in the inner ear.

The human auditory system can capture understandable speech signals at variable intensities. This robustness is due to the Automatic Gain Control (AGC) mechanism, provided both by the muscles supporting the ossicles chain and the outer hair cells in the organ of Corti. In addition the spike discharge rate also plays a role in signal normalisation as a result of saturation in the nerve fibers. In Figure 2.8 each auditory nerve fibre is connected to one hair cell. During the resting position of the basilar membrane, the auditory nerve fibres have a 'resting' discharge spike rate. When the hairs (stereocilia) of that hair cell 'bend' due to basilar membrane movement, an increased rate of spike discharge is transmitted via the associated auditory nerve fibre. At low sound pressure levels, the rate of spikes increases relatively to sound intensity.

At normal conversational speech levels (60-80 dB) (Rabiner and Schafer 2010) the dominant hair cells spiking rates are saturated (Young 2008) and it is the number of phase locked saturated fibres that indicates the intensity of the central frequency (Greenberg *et al.* 2004). Figure 2.9 shows how a tonotopic representation can model the spectral envelope in terms of saturated nerve fibers.

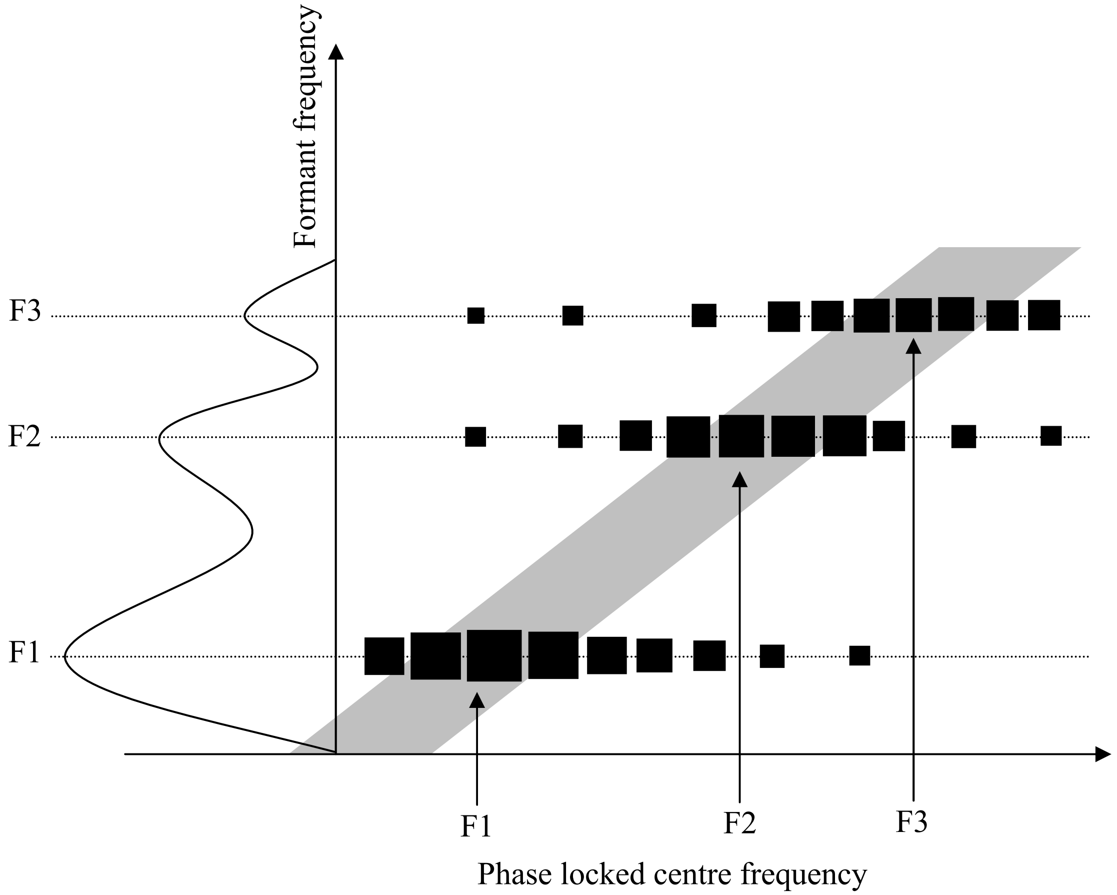

Figure 2.9 – Tonotopic representation of spectral envelope at normal conversation speech level (60-80 dB), black box size is related to the number of saturated nerve fibers.

In Figure 2.9 the vertical axis represents the active formant frequencies along the basilar membrane (i.e. the DFT spectrum), while the horizontal axis represents the response at the auditory nerve, where centre frequencies are locked according to formant frequencies' positions. Although, in Figure 2.9 only the formant frequencies are presented, each point along the spectrum envelope is presented in terms of the number of saturated nerve fibers. Here a question can be raised: what is the highest amplitude that can be represented in terms of saturated nerve fibers? The answer is the number of nerve fibers that can be saturated at a specific frequency, this implies that the DFT spectrum, which is captured by the basilar membrane, will be normalised over the maximum number of saturated nerve fibers. The resultant envelope in terms of number of nerve fibers is the gray shaded area.

However, not all nerve fibers are connected directly to the auditory lobe in the brain; many of them end at the Cochlear Nucleus. The auditory nerve connects the cochlea to the Cochlear Nucleus, which is the last section of the auditory system where the sound information is identifiably represented (Møller 2006). The neurons at the Cochlear Nucleus function as averaging nodes (Møller 2006) as shown in Figure 2.10.

As shown in Figure 2.10, the middle Cochlear Nucleus neuron, with fully saturated nerve fibers, produces a higher spike rate, than the two other neurons with less number of saturated nerve fibers. This produces a train of spikes that starts (onset point) before the spike trains of the other neurons. Cells inside the brain are known to respond quicker to onset point than to continuous spikes discharge (Gerstner and Kistler 2002; Greenberg *et al.* 2004; Møller 2006).

Beyond the cochlea, the exact form of signal processing employed by the human auditory system is currently uncertain. However, what is known is that the auditory cortex sections of the human brain are organised tonotopically (Young

2008) and that the phase locking of adjacent fibres is prominent within the auditory nerve (Greenberg *et al.* 2004).

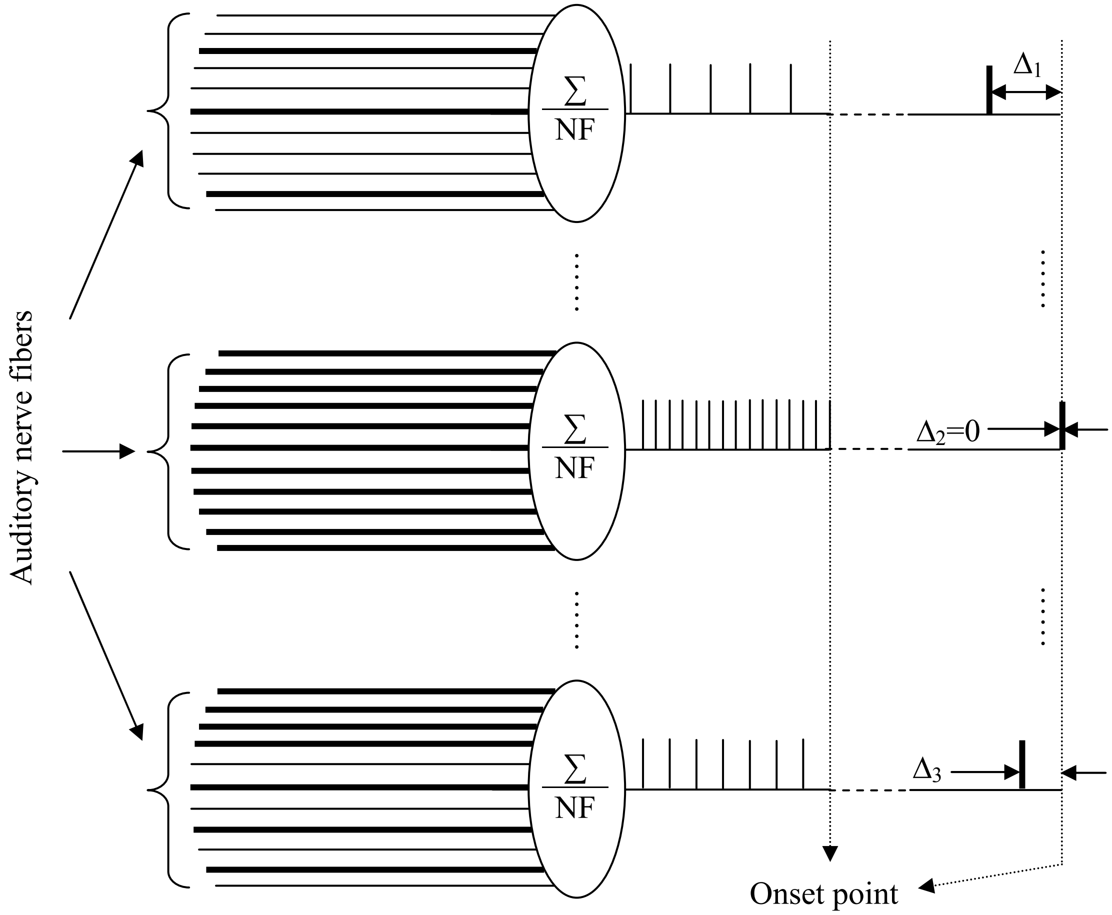

Figure 2.10 – Cochlear Nucleus neurons, bold lines represent saturated nerve fibers while thin lines are non-saturated nerve fibers. NF is the number of nerve fibers connected to each neuron.

Consequently, the known physiology of hearing leads us to the conclusion that the lower levels of the human auditory processing system can be approximated by an *N* component DFT spectrum vector. In the following section, a spike-based feature vector, derived from the DFT spectrum, is presented that is inspired by the physiology of the human audio processing system.

2.6 Summary

In this chapter the experimental infrastructure of this research is demonstrated. This includes theoretical background to the neural networks used in this research such as the self organising map, learning vector quantisation, and feed forward multi layer perceptron. An introduction to the speaker verification database is presented with details of the CSLU2002 database which adopted in this research. The chapter also describes the physiology of hearing in the human auditory system, and how the sound vibrations are converted into a spike-based signal through bio-chemical process.

Chapter Three

Literature Review

3. Introduction

Speaker recognition is a research problem that has been widely explored in the literature. Following the objectives of this thesis, the survey in this chapter focuses on research that employs different types of neural networks for speaker recognition tasks. Since speaker verification and identification share similar implementation methods, the terms speaker recognition, speaker verification, speaker identification and speaker authentication are used interchangeably throughout the following literature review.

3.1 Multi Layer Perceptron Classifier

The Multi Layer Perceptron (MLP) feed forward neural network is the most common type that has been adopted in the literature for speaker verification/identification. Different versions of speaker verification platforms can be obtained, as a result of using different types of MLP neural networks, or using different feature vectors. The next sections are described as follows: Section 3.1.1 addresses research that adopts phonemes/vowels as the feature vector in an MLP classifier system. Section 3.1.2 then focuses on studies that use different types of speech format as feature vector, whilst Section 3.1.3 cites studies that use common types of feature vectors in combination with different types of MLP neural networks. Section 3.1.4 addresses comparative studies, whilst Section 3.1.5 cites language-based MLP classifier studies.

3.1.1 Phonemes-Based Multi Layer Perceptron Classifier

A phoneme-based neural tree network for speaker verification is introduced in (Han-Sheng and Mammone 1995a). The paper uses HMM to segment the speech into phonemes, and uses a phonetic weighting scoring method to investigate the role of different phonemes in the speaker verification problem. The system is tested using 80 speakers from the YOHO database. The sampling frequency is 8 kHz with speech frame size of 25 msec. Twelve MFCC were used as feature vector inputs to the network. The system is claimed to overcome an equivalent HMM classifier system with Equal Error Rate (EER) of 0.13% over a 30 speaker test. Extended experimentation with the same algorithm is presented by the same authors on sub-word morphological level in (Han-Sheng and Mammone 1995b).

A speaker recognition system based on vowel spotting and neural networks is introduced in (Fakotakis and Sirigos 1996). The paper uses MLP networks as vowels spotters for each speaker in a speaker verification and identification problem. The proposed two hidden layer MLPs contain 15 input units, seven nodes in the first hidden layer, four nodes in the second hidden layer and one unit in the output layer. The speech was sampled at 16 kHz and segmented into frames of 30 msec. For each frame, 15 Perceptual Linear Predictive coefficients were used as the feature vector. The system was tested using 76 speakers from the TIMIT database with test utterances of less than 2.5 sec. The claimed verification accuracy is 97.69%. The results in the paper also conclude that verification accuracy significantly increases when the length of the test utterance is increased.

Specific phoneme MLP networks were investigated for the speaker verification task in (Delacretaz and Hennebert 1998). The paper uses HMM to provide the phoneme information from the speech data, and then each phoneme data is classified using an individual MLP network. Each MLP network contains 12 inputs, 20 hidden nodes and two outputs. Twelve LPC cepstrum coefficients were used as the feature vectors. The system was tested using 25 speakers from a Swiss German telephone database called HER. The paper implies that for speaker verification, nasals, fricatives and vowels provide better performance than plosives and liquids.

A neural network based on vowel phonemes is presented in (Badran and Selim 2000) for a speaker recognition task. First a vowel phoneme locating algorithm is introduced, then an MLP network classifier is suggested which contains 10 inputs, four hidden nodes and one output. Speech was sampled at 8 kHz and segmented into frames of 20 msec using a Hamming window. Ten Adaptive Weighted Cepstrum coefficients were used as the feature vector and the system was tested for speaker verification and identification using a self collected dataset of 10 speakers (3 females and 7 males). The best claimed text-dependent verification rate is 95.67% and 92.2% for text-independent. The paper recommends the use of vowel phonemes, diphthongs and semi vowels phonemes instead of using vowel phonemes only, to increase recognition accuracy.

The work in (Seddik *et al.* 2004a) presents a phoneme based MLP network for the speaker recognition task. The paper investigated the use of different numbers of phonemes (up to 48 phonemes). The network contains 12 inputs, 45 hidden nodes and one output. Twelve MFCC coefficients for each phoneme were used as feature vector. Each coefficient value in the feature vector represents the average value over a set of frames belonging to the phoneme. Speech was sampled at 16 kHz and segmented using a Hamming window. Phonemes positions were pre-extracted in the database. The system was tested using a dataset of 20 speakers from the TIMIT database. The paper conducts four experiments. The first three experiments use 5, 10 and 48 phonemes while the fourth experiment uses 11 vowel phonemes and 4 output nodes instead of one. The claimed recognition rates are 98.57%, 97.05 and 87.23% for the first three experiments and 77% for the fourth experiment. In the second experiment, when the network is tested with phonemes of the same kind as used during training the recognition rate increased to 100%. The paper addresses some key points regarding the use of phonemes in speaker recognition problems. For example; the use of phonemes which are similar in pronunciation in the training phase can confuse the network in the testing phase.

An MLP is proposed in (Tan and Ting 2011) for Malay speaker identification. The network contains 24 inputs, 20 hidden nodes and one output. LPC coefficients are used as input feature vector. An experiment is conducted using six vowels of 10 speakers self collected dataset with no gender information. Speech is sampled at 20 kHz and segmented using a Hamming window. A maximum identification of 93.33% is claimed when one frame of 35 msec is examined.

3.1.2 Multi Layer Perceptron Classifiers with Different Feature Formats

Some key factors associated with speaker recognition using neural networks are discussed in (Sun *et al.* 1991). The paper presents an optimised MLP with a single hidden layer of 14 nodes. The paper also investigates the use of different features extracted from the speech signal (power spectrum, Mel-scaled power spectrum, Reflection coefficients, LPC, Autocorrelation coefficients, Cepstrum, Mel-scaled Cepstrum) and refers to the power spectrum as the most useful for neural network classification in a speaker recognition system. Two datasets of 6 and 9 male speakers were used to test the system. The speech was sampled at 10 kHz and the claimed verification and identification rate is 99% over 7 digits.

An MLP is presented in (Seddik *et al.* 2004b) for speaker recognition task. The paper uses one network to classify speakers according to their first three fundamental frequencies positions and another network to classify the incorrectly classified cases from the first network by using the pitch feature. The network consists of input layer, two hidden layers with 12 nodes for each layer and output node. The paper compares the first frame of the speech waveform to a reference speech signal in the classification phase, instead of comparing the whole sentence or word. This is claimed to save time in the testing phase. Speech was sampled at 16 kHz and segmented into 256 samples (16 msec) frames. The formant features were extracted using two methods. The system was tested using a dataset of 40 speakers from the TIMIT database. The best claimed recognition rate is 95% when the proposed network structure is used.

A Radial basis function neural network is suggested in (Lacerda *et al.* 2010) as a classifier for speaker verification task. Twenty one energy coefficients of 8th level Discrete Wavelet-Packet Transform were used as input features. Each coefficient is fed into one network (with no structure information). Speech is sampled at 16 kHz with no segmentation information. The algorithm is evaluated using self collected speech data of 40 speakers (20 females and 20 males) speaking a phrase in the Portuguese language. The paper claimed 10% FRR and 5% FAR.

Another MLP neural network that uses Wavelet-based features is proposed in (Daqrouq 2011) for the speaker identification task. The network contains 35 inputs, two hidden layers of 20 nodes each and 5 outputs. Thirty five Wavelet Packet entropies were used as the input feature vector. Speech is sampled at 16 kHz. The algorithm is evaluated using self collected speech data of 29 speakers (10 females and 19 males), recorded in a normal office environments. The claimed average identification performance is 91.09%.

The author in (Pandiaraj *et al.* 2011) presents an auto associative neural network for a speaker identification task. The network consists of an input linear layer of 40 nodes, three non-linear hidden layers of 80, 20 and 80 nodes and an output linear layer of 40 nodes. The paper uses 40 coefficients which represents a "Pyknogram" (a time-frequency representation equivalent to the spectrogram) of the speech signal. No segmentation or sampling information is mentioned in the paper. Thirty six speakers from the CHAINS database were used to evaluate the algorithm without mentioning gender details. The best claimed identification rate is 92.1% for females and 89.9% for males.

A generalised regression neural network is proposed in (Wu and Tsai 2011) for the speaker identification task. Empirical decomposition features were used as an input with no information about the input vector size. Speech is sampled at 16 kHz with no segmentation information. The algorithm is evaluated using a self collected database of 36 speakers (18 females and 18 males) uttering Chinese text. Claimed identification performance is 89%.

3.1.3 Other Multi Layer Perceptron Classifiers

Adaptively boosted MLP networks were introduced in (Say Wei and Eng Guan 2001). The boosted MLP can be described as traditional MLP with added weight parameter when calculating the error at the output of the network during the training. This weight amplifies the error values when misclassification occurs. The system was compared with traditional MLP network. Sampling frequency was not mentioned. Speech is segmented into 32 msec. Twenty components containing 10

Linear Predictive Cepstral Coefficients (LPCC) and 10 first derivatives of LPCC were used as the feature vector. The system was tested using 20 speakers from the YOHO database. The system was tested for verification and identification. For verification the same impostors used during the training were used in testing. The best claimed verification results were 0.75% False Reject Rate (FRR) and 0.079% False Accept Rate (FAR) when the proposed system is adopted while 4.75% FRR and 0.5% FAR when a traditional MLP is used. For identification the best claimed performance is 99.25% when the proposed system is used while 95.25% when a traditional MLP is used.

An MLP is presented in (GuoBin *et al.* 2005) as a speaker identification classifier. Two MLP networks were used and designed according to the feature vector parameters. The first network is designed to use 13 MFCC coefficients as feature vector while the second network is designed to use 70 video image features of the lips. The first network contains 13 inputs, 30 hidden nodes and 20 outputs. The second network contains 70 inputs, 30 hidden nodes and 20 outputs. A combined network with 83 inputs, 40 hidden nodes and 20 outputs is also presented. Sampling frequency was not mentioned as well as speech pre-processing parameters. The system was tested using a self collected dataset of 20 speakers with no gender information. Claimed performance of the speech network is 75.38% with text dependent test and 66.67% with text independent test. The claimed identification accuracy increased to 100% when the combined network of speech and lip image data was used.

An MLP is employed for speaker identification to enhance the security of voice over internet protocol in (Ibrahim and Abdulghani 2012). The network contains 14 inputs, a hidden layer of 40 neurons and 4 outputs. Fourteen LPC coefficients were used as input feature vector. Speech is segmented using a Hamming window into frames of 30 msec with 10 msec shift. No information is mentioned regarding the sampling frequency. The algorithm is evaluated using a self collected dataset of four speakers (2 females and 2 males). Speaker identification performance is claimed to be 99.8%.

3.1.3.1 Genetically Optimised Multi Layer Perceptron Classifiers

A genetically optimised MLP network is presented in (Price *et al.* 2000) for speaker identification. The Genetic Algorithm is used to optimise the structure and parameters of the MLP network. Twenty cepstral coefficients were used as feature vector. No speech pre-processing parameters were mentioned. The system was tested on 21 speakers from the NIST96 database. The claimed EER is 5% when the same recording microphone is used to train and test, and 20% when using different microphones. The results shows comparable performance to the GMM based system with matched recording device and lower performance when different devices were used.

Another genetically optimised radial basis function neural network is proposed in (Yan and Yunian 2010) for the speaker recognition (identification) task. The Genetic algorithm is used to optimise the weights and the structure of the network. After optimisation the network contained an input layer of 15 nodes, one hidden layer of 18 nodes and an output layer of 15 nodes. Twelve MFCC coefficients are used as an input feature vector. Fifteen speakers from the TIMIT database were used to evaluate the network. The claimed results show improvement over the traditional radial basis function neural network, with fast learning generalisation capability and a claimed performance of 81.44% when 5 sec training speech data used, increasing to 95.01% when the training speech data is 20 sec.

3.1.3.2 Auto Associative Multi Layer Perceptron Classifiers

The work presented in (Kishore and Yegnanarayana 2000) suggests that auto-associative neural network models can be used to minimise the channel effects in a speaker verification application. The auto-associative network contains 19 linear input layer nodes and 19 linear output layer nodes. The number of nonlinear hidden nodes is investigated to be less than input nodes. The output layer is designed to follow the input layer, while the role of the hidden layer is to compress the dimension of the feature vector. Sampling frequency was not mentioned. Nineteen cepstral coefficients were extracted from a 27.5 msec frame to form the feature vector. The paper experiments with 14 and 10 hidden node models as well as individual and universal background speaker models. More robustness is claimed against the channel effect when the hidden layer contains 10 hidden nodes. Speech data of 230 male speakers from the NIST-99 database is used to experiment the algorithm. The paper recommends the use of individual background models over a universal background model with equal error rate reduction of 23.4%.

An auto associative neural network is used in (Sri Rama Murty *et al.* 2004) to capture residual phase information in a speaker identification task. The network contains 40 linear input nodes, three nonlinear hidden layers of 48, 12 and 48 nodes respectively and 40 linear output nodes. Forty LP samples were used as the feature vector. Speech was sampled at 8 kHz. Segmentation size was not mentioned. The system was tested using two datasets of 38 speakers and 76 speakers from the TIMIT database. The paper claims that voiced speech segment regions neighbouring the glottal closure instant are more speaker specific than other regions. The best claimed performance is 87% for the first dataset and 76% for the second dataset.

The work in (Kodukula *et al.* 2005) is an extension work of the system presented by the same author in (Sri Rama Murty *et al.* 2004). The paper repeats the previous experiment using 149 speakers from the NIST 2003 database and claims EER of 22%. Another experiment using 19 LPCC coefficients feature vector with a network of 19 linear inputs, three nonlinear hidden layers of 38, 9 and 38 nodes respectively and 19 linear outputs. Speech was sampled at 8 kHz and segmented into 20 msec segment. The author demonstrates how residual phase information contains complementary speaker specific information. Claimed EER is 15.5% and 13.5% when the scores of the two experiments are combined.

An auto associative neural network is presented in (Yegnanarayana *et al.* 2005) for the speaker verification task. The author proposed previous experiments using the same network in (Yegnanarayana *et al.* 2001) and as co-author in (Kodukula *et al.* 2005), (Kishore and Yegnanarayana 2000) and (Kishore *et al.*

2001). The network contains 40 linear inputs, three nonlinear hidden layers of 48, 12 and 48 nodes respectively and 40 linear outputs. Forty samples of Linear Prediction (LP) residual were used as feature vector. The paper highlights the use of features at supra-segmental level such as pitch and duration. The network is tested using a self collected dataset of 30 speakers (9 females and 21 males). The paper compares the results of the network with other types of features: vowel onset point spectral features (25 components as 20 weighted LPCC plus 5 delta weighted LPCC), duration and pitch. The comparison was not made using the same network, but by using a dynamic time warping method and other methods. The paper finally assembles the scores of the four methods in different combinations to train an MLP network. Best claimed EER is obtained when the four methods scores are combined. In the MLP testing phase the impostors are unseen impostors in the training phase.

The author in (Jothilakshmi *et al.* 2009) employed an auto associative neural network to capture speaker specific information in a speaker diarisation task. The network contains a linear input layer of 19 nodes, three non-linear hidden layers of 38, 5 and 38 nodes respectively and an output layer of 19 linear nodes. Nineteen MFCC were adopted as the input feature vector. The network is used first to detect speaker change in conversations, by training the network with features from one frame and testing the network with the next frame, and the difference in the output is then used to model a confidence score. Speaker change points can be detected using the confidence score. After segmenting the conversation into different durations, the network is secondly used to clustering these segments into speakers' classes. Speech was sampled at 8 kHz and segmented into frames of 16 msec with 50% overlap between adjacent frames. The system is evaluated using the NIST-RT'03S database, six broadcast news shows of about 10 min each were used as development dataset for training the system and tuning the parameters, the testing dataset consists of three 30 min shows. The claimed diarisation error measure is 12.01%.

An auto associative neural network is used in (Rao *et al.* 2010) for speaker recognition (identification) in mobile devices. The paper suggests multi-SNR multi-environments speaker models to improve the robustness against background and channel effects. The network contains a linear input layer of 39 nodes, three hidden non-linear layers of 60, 20, and 60 nodes respectively and linear output layer of 39 nodes. Thirty nine LP coefficients are used as the input feature vector. Fifty speakers from the TIMIT database were used to evaluate the algorithm. NOISEX data were used to generate the noisy TIMIT data. Best claimed identification performance is 98% using TIMIT clean speech data.

An auto associative neural network is used for speaker identification task in (Mubeen *et al.* 2012). The network consists of 19 linear input nodes, three non-linear hidden layers of 38, 4 and 38 nodes and an output linear layer of 19 nodes. Nineteen linearly weighted LPCC coefficients were used as input feature vector. The network is evaluated using a self collected database of 36 speakers with no gender details. Speech is captured using two types of microphones, a normal microphone and a Throat microphone. With no sampling frequency mentioned, speech is segmented using Hamming window into frames of 20 msec with a step of 5 msec.

the claimed performance over separated sessions is 84.9% for the normal microphone, 80.2% for the Throat microphone, and 88.7% when combining scores from both microphones. The paper addresses the difference in the spectral characteristics of the same speech signal between the two types of microphones, implying that it is due to the multi-capturing sensory conducted by the two devices.

3.1.4 Comparative Research

A comparative study between a Continuous Hidden Markov Model and MLP network is presented in (Kasuriya *et al.* 2001) for a speaker identification task. The MLP contains 60 input neurons, 20 neurons in the first hidden layer, 20 neurons in the second hidden layer and two output nodes. Speech was sampled at 11.025 kHz and segmented into 20 msec frames using a Hamming window. Fifteen MFCC from four frames were used to form a 60 coefficient feature vector. The two systems were evaluated using two self collected datasets of 50 speakers in office and telephone environment. The office dataset consists of 20 females and 30 males, whilst the telephone dataset consists of 25 females and 25 males. The paper considers using different recording sessions between training and testing modes as well as different recording environments (office and telephone). For the two environments condition the identification rate of the MLP network is claimed to outperform the continuous HMM method by 97.3% and 96.3% respectively.

An experimental comparison of different modelling techniques for speaker recognition under limited data conditions is presented in (Jayanna and Prasanna 2009). The paper shows that under limited data conditions (defined as 3 sec for training and 3 sec for testing) the performance of many speaker recognition models will decrease. The paper compares the following methods: Crisp vector quantisation, Fuzzy vector quantisation, Self Organising Map, Learning vector quantisation and Gaussian mixture model. The methods were tested using two datasets. The first dataset consists of 138 speakers from the YOHO database while the second dataset consists of 168 speakers from the TIMIT database. The best claimed recognition rate is 80% when the LVQ technique was used and 86.67% when a hybrid system of LVQ-GMM with universal background model is considered.

A comparison between the MLP neural network and the Radial basis function neural network in speaker identification scenario is presented in (Hmich *et al.* 2011). Twelve LPC coefficients are used as input feature vector for both networks. Two speech datasets were used to conduct the comparison. The first is a self collected speech data of nine males uttering two Japanese vowels, and speech in this dataset is sampled at 10 kHz with frame length of 25.6 msec and 6.4 msec step overlap. The second is ten speakers of the Numenta speech database with speech sampled at 16 kHz. The paper compares only learning time and claims that the Radial basis function neural network outperforms the MLP network, especially when the number of hidden nodes is increased.

3.1.5 Language-Based Multi Layer Perceptron Classifier

A Probabilistic neural network is suggested for speaker recognition in (Ye and Yabin 2009). The experiment conducted in this paper is seen to serve as an identification problem. The network consists of an input layer of 15 nodes, samples layer of 15 nerve cells, accumulated layer and an output layer. Fifteen MFCC coefficients were used as input feature vector. Speech is sampled at 8 kHz with no segmentation information. The method is evaluated using self collected speech data from 20 speakers (10 females and 10 males) with Chinese spoken digits from 0 to 9. The claimed classification accuracy is 97.5%. The paper recommends the probabilistic neural network for short duration testing environment and low mixture degree speaker identification.

A Multilingual speaker identification is presented in (Ranjan *et al.* 2010). The MLP network has up to 360 inputs, two hidden layers of 42 and 38 nodes respectively and 20 outputs to identify 20 speakers. Different features were adopted in the input feature vector such as LPC coefficients, Reflection coefficients, Number of zero crossings, Average power spectral density and the first three formant frequencies. No details were mentioned regarding the order of LPC or the reflection coefficients. Speech is sampled at 44.1 kHz. A self collected speech data of 20 speakers (10 females and 10 males) is used to evaluate the algorithm. One sentence is recorded in four different Indian languages. An average identification performance of 83.89% is claimed and with improved performance of 92.78% when a clustering algorithm is used.

An Arabic speaker verification problem in mobile devices using MLP is investigated in (Alarifi *et al.* 2011). The paper experiments one and two hidden layers network with different numbers of hidden nodes. MFCC coefficients are used as feature vector with no details about the order. Self collected speech data of 15 speakers is used to evaluate the network. The best claimed network structure that results higher number of trials with 100% accuracy is when using one hidden layer.

A Fuzzy min-max neural network is proposed in (Jawarkar *et al.* 2011) for the speaker identification task. This network utilises fuzzy sets as pattern classes. The network contains three layers, the hidden layer is growing adaptively to meet the problem demand. Eighteen MFCC coefficients are used as input feature vector. Speech is sampled at 22.05 kHz and segmented into frames of 23.33 msec with 50% overlap, then windowed using a Hamming window. The network is evaluated using self collected speech data of 50 speakers in one of Indian languages. The paper claims an identification accuracy of 99.99% when 15 sec testing speech data is used for experimentation.

The work in (Ke and Salman 2011) proposes a Deep Neural Architecture to learn speaker-specific characteristics in speaker verification and speaker segmentation environment. The network consists of two identical subnets. Each network is a feed forward MLP network. One network is designed to capture dominant information for recognition, while the other network is designed to capture non-dominant information. The number of hidden layers and hidden nodes are optimised empirically into four layers of 100, 100, 100, and 200 nodes. Fifteen of nineteen MFCC coefficients were selected for the input feature vector. Speech is segmented using a Hamming window into frames of 20 msec with overlap of 10 msec. The algorithm is evaluated using a total of 70 speakers from six databases TIMIT, NTMIT, KING, NKING, Chinese and Russian, with training speech data of

132 sec. The algorithm is compared with MFCC based GMM model and claimed to show better performance.

3.2 Self Organising Map Classifier

An SOM is proposed in (Monte *et al.* 1996) for the speaker identification problem. The paper uses a 25 x 25 Kohonen SOM to identify speakers based on comparing the histogram occupancy of each speaker's SOM with other speakers in the database. LPC and MFCC coefficient were investigated as feature vectors and the speech was segmented into 30 msec frames. The sampling frequency was not mentioned. The system was tested using 100 speakers from the TI database under different signal to noise ratio levels. The proposed system was compared with Arithmetic-Harmonic Sphere Measure and the best claimed results for clean speech was 100% when MFCC vectors were used with the SOM.

A two level classifier for speaker identification is presented in (Hadjitodorov *et al.* 1997). The paper investigates different versions of an SOM as a first stage classifier to obtain a prototype distribution map, which is then used to feed a second stage classifier of MLP network. The SOM is 15 x 15 in size, while the MLP contains two hidden layers. The first layer has 64 neurons and four neurons in the second layer, with one output neuron. Speech was sampled at 10.24 kHz, although framing information was not mentioned. Fifteen LPCC coefficients were used as the feature vector. The system was tested using two self collected datasets in the Bulgarian language; the first being clean speech data of 68 speakers (33 females and 35 males), while the second dataset consists of 92 speakers (44 females and 48 males) speech data recorded over telephone lines. The best claimed error rate with first dataset is 1.47% and 2.17% with the second dataset.

The work in (Voitovetsky *et al.* 1997) introduces a 6 x 10 SOM algorithm for speaker classification. Twelve cepstral coefficients were used as the feature vector. Two self collected datasets in the Hebrew language were used to test the algorithm. The first is high quality speech data of five speakers talking in different dialogue recordings, speech being sampled at 16 kHz. The second dataset is telephone quality type with 24 speakers participating in the dialogues. Speech was sampled at 8 kHz. Total classification error claimed using the first dataset is 5.6% and 6.2% for the second dataset.

A hybrid system based on SOM and MLP is presented in (Ouzounov 1997) for a speaker identification task. The SOM is used to generate a statistical histogram, the histogram features then being used to feed the hidden layer of the MLP network. The SOM size was optimised into 3 x 3 to give best results. Speech was sampled at 8 kHz and framed into 30 msec using a Hamming window. Twelve LPC derived cepstral coefficients were used as the feature vector. The best claimed identification error rate is 4%.

An SOM and associative memory hybrid model is presented in (Inal and Fatihoglu 2002) in a speaker recognition application (the paper claims that speaker identification and verification experiments were conducted although only identification results were illustrated). In this paper an SOM followed by associative memory neural network forms a speaker classifier model. The paper investigates a 10 x 10 SOM for text dependent speaker identification and 20 x 20 SOM for text independent speaker identification. Twelve MFCC coefficients were used as the feature vector. For the text dependent experiment the system was tested using a dataset of 10 speakers while for text independent experiment the system was tested using 38 speakers from the TIMIT database (results show performance for up to 20 speakers only). Sampling frequency was not mentioned. Speech was framed using a 660 points Hamming window. The claimed performance for the first experiment is 97.455% and 96.3% for the second experiment using 20 speakers of the TIMIT subset.

An unsupervised speaker recognition system using an SOM is presented in (Lapidot *et al.* 2002). The paper investigates the use of different SOM network sizes to recognise speakers from conversations. Speech was sampled at 16 kHz and segmented into 15 msec frames using a Hamming window. Twelve LPCC coefficients plus 12 ΔLPCC coefficients were used as the feature vector. The system was tested using two types of self recorded conversation in the Hebrew language. Conversations of ten speakers (one female and nine males) were recorded over a high quality channel, and 12 conversations of 24 male speakers were recorded over a telephone quality channel. The sampling frequency was 8 kHz for the telephone quality channel. The optimised size of the SOM is 6 x 10. A comparison with a time-series clustering approach is made and the claimed accuracy is over 80% using the proposed SOM.

An SOM is presented in (Mafra and Simoes 2004) for speaker identification. The paper investigates different SOM sizes. Speech was sampled at 22.05 kHz and segmented into 32.22 msec frames using a Hamming window. Fourteen MFCC coefficients plus 14 ΔMFCC coefficients were used to provide a 28 component feature vector. The system was tested using a self collected dataset of 14 Brazilian speakers (8 females and 6 males). The best claimed identification rate is more than 99% when the 16 x 16 SOM is used; requiring 17.5 sec training speech data and more than 2.8 sec testing speech data.

3.3 Spiking Neural Networks

A dynamic synapse neural network is presented in (George *et al.* 2001) in a speaker recognition application. The neurons in the dynamic synapse network transform a train of action potentials into another train of discrete release events. The network was trained using genetic algorithms. Gender classification is applied first using a rule based method. Two networks were designed for each gender; each network having an input layer of 16 nodes and an output layer of two nodes. The 16 input potential actions are obtained by passing the speech into four filter banks. From the output of each filter, four wavelet features are calculated to form a 16 coefficient feature vector. Speech was sampled at 12.5 kHz. The two networks were tested using 8 male speakers and 8 female speakers from the TI-26 database. The best claimed performance for the two networks is 100% and 67% for the female and male target speakers respectively and 87% and 84% for female and male non-target speakers.

A nonlinear dynamic neural network is presented in (Bing *et al.* 2006) for speaker identification. The paper uses a higher order synapse model for data transfer through the neurons. The network contains two dimensions. Each dimension represents a space of features. The main concept of the network is to capture the distinctive feature components and magnify their effect. Speech was segmented into frames of 10 msec. The network contains 20 neurons in each space and the memory size is 40 frames. Twenty MFCC coefficients were used as the feature vector. The network was trained using the Nelder Mead algorithm. The maximal log-likelihood is applied for testing. The system was tested using 40 speakers from the TIMIT database with a claimed identification rate of 92% to 97.5%.

Speaker identification using pulse coupled and MLP neural networks is presented in (Timoszczuk and Cabral 2007). The paper proposes two layers of pulse coupled neural networks for feature extraction followed by an MLP network for classification. Pulse neurons are represented using a Spike response model. The first layer converts the inputs into a pulse modulated sequence while the second layer extracts the features for the MLP network. The first layer contains 16 pulse neurons fed by 16 MFCC coefficients. The second layer is a ring of SOMs of 100 pulse neurons, trained with the standard concept of SOM training. The MLP network has 100 inputs, 300 hidden nodes and 10 outputs representing the ten speakers to be identified. Speech was sampled at 8 kHz and segmented into frames of 32 msec. The system was tested using 10 speakers from the CSLU v1.0 speaker recognition database. The claimed identification rate is 82%.

A spiking neural network is presented in (Wysoski *et al.* 2007) for speaker authentication. Each component of a 19 MFCC coefficients vector is encoded into train of spikes using Rank order coding. Speech was sampled at 16 kHz. The paper investigates two and three layer networks. Each neuron in the spiking neural network is an integrate-and-fire neuron. The first network contains 19 input neurons, two maps of 80 neurons in the first layer and two outputs neurons in the second layer. The two maps and the two output neurons represent the speaker model and the background model respectively. In the second network an additional layer is added to provide normalisation for the score similarity. The system was tested using 35 speakers from the VidTimit database. Eight other speakers were saved as unseen impostors for testing. In the testing phase, 8 unseen impostors plus training impostors were used. The paper claims that the results are comparable to the performance of a vector quantisation system under the same conditions.

An extended version of the spiking neural network presented in (Wysoski *et al.* 2007) is proposed in (Wysoski *et al.* 2010) as the auditory part in an audiovisual authentication process. The author argues that MFCC has been successfully used in speaker authentication, but that they may imprison other features which can uniquely describe a speaker. Therefore, the paper recommends other frequency domain features such as short time Fourier transform or Wavelets. However, for comparison with the previous work in (Wysoski *et al.* 2007) the author adopt MFCC coefficients in rank order coding format. Speech is sampled at 16 kHz with no segmentation information. Nineteen MFCC coefficients are converted into rank order coding features and used as input feature vector. The input feature vector is fed into a first layer of two maps, representing the speaker model and background model. Each map consists of 80 neurons. The output layer contains two neurons which are fully connected to the two maps in the first layer. The network is evaluated using 35 speakers from the VidTimit database. The minimum total error rate (FRR+FAR) claimed is 31.1%.

3.4 Summary

In this chapter many different neural network methods applied to speaker verification have been reviewed. The literature review here focuses mainly on three main approaches which are adopted in this research: MLP neural networks, SOM and Spiking neural networks. Figures 3.1 and Figure 3.2 show the distribution of the research cited for a population of 43 studies in the literature according to the type of the neural network and the feature vector used in these studies.

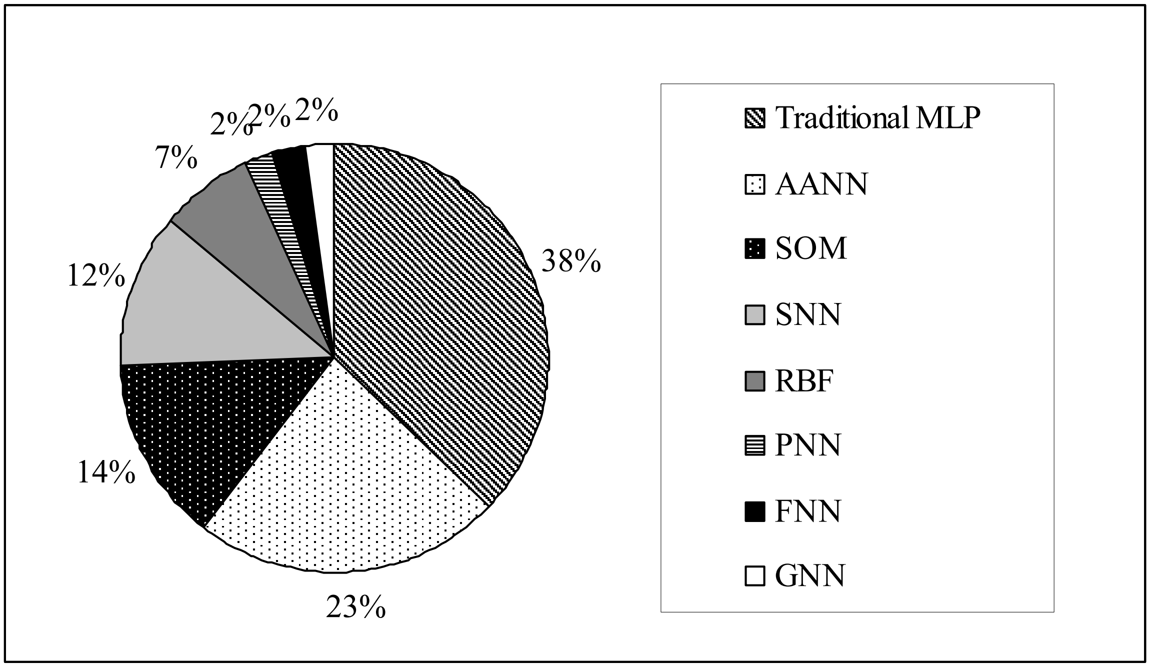

Figure 3.1 – Distribution of neural networks methods used in 43 studies in the literature.

As shown in Figure 3.1 the MLP neural networks are extensively represented in the literature due to its easy implementation and code availability over different platforms, while SOM is less used as a speaker recognition platform. Spiking neural networks, on the other hand, are rarely used in speaker recognition applications. Despite the fact that the SOM is represented in the literature less than MLP systems, this research focuses on the use of SOM for speaker verification due to the correlation between the topological nature of the SOM (Kohonen 1990) and the tonotopic nature of the auditory nerve response (Young 2008).

Another key parameter in any pattern recognition problem is the choice of the input feature vector. Two types of feature formats can be extracted from the speech signal: time domain and frequency domain features. Since the speech signal is analytically more intelligible in the frequency domain (Rabiner and Schafer 1978) than in time domain, frequency domain features dominate in the literature. Figure 3.2 shows the feature type's distribution cited in 43 studies in the literature.

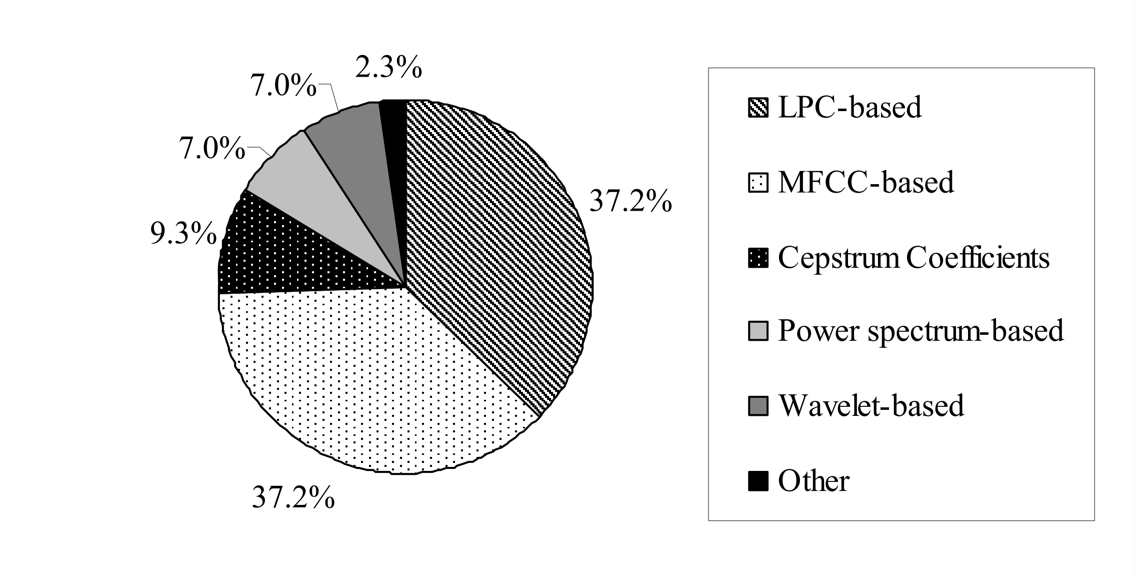

Figure 3.2 – Distribution of feature vector types used in 43 studies in the literature.

Several feature formats have been adopted in the literature: MFCC, LPCC, Wavelet transform and other frequency domain features. The most commonly used feature formats used in literature are the MFCC-based and LPCC-based features as shown in Figure 3.2. These have been used successfully in speech recognition applications and then employed in speaker recognition scenarios. However, a recent publication (Wysoski *et al.* 2010) argues that the use of MFCC could occlude important information about speaker identity. For comparison purposes MFCC, LPC and DFT spectrum feature vectors are investigated as input to a speaker verification platform in Chapter 5 of this research.

The speech signal can be captured with different audio qualities depending on the sampling frequency. Sampling frequencies ranging from 8 kHz to 44.1 kHz have been used. Considering 8 kHz is more demanding than using higher values, since less information is captured in the frequency domain. However, this work uses 8 kHz sampled speech due to database availability.

Several speech databases have been used to evaluate the performance of speaker verification or identification systems. Using a standard speech database to evaluate any speaker verification method is a key factor to determine its effectiveness. Using a small number of speakers to test a speaker verification method leads to lower confidence in the evaluation results. It is noticed that the evaluation of some studies in literature was based self collected speech datasets with a small number of speakers.

For many of the speaker verification algorithms reviewed, substantial speech data is required to provide training phase (Mafra and Simoes 2004; Ke and Salman 2011) and testing phase (Jawarkar *et al.* 2011), for both real speaker and impostors. Few papers have investigated limited speech data for the testing phase, with less investigating limited speech data condition for the training phase as well (Jayanna and Prasanna 2009). The importance of developing speaker verification systems in a limited speech data environment is that in commercial speaker verification applications, it is not preferable to collect substantial speech data from the client in order to enrol as a claimed speaker.

One of the main current challenges in the speaker verification process is the difference between the captured speech signal during enrolment and testing. This difference is expected due to the variability in the true speaker voice, but in many cases the difference occurs due other factors such as: channel effect (i.e. enrolling and testing using different devices) and environmental effect. These occur when enrolling and testing in different places with different background noise characteristics. Some of the cited references in the literature explored these conditions such as the multi environments scenario in mobile device channel (Rao *et al.* 2010), office/telephone environments (Kasuriya *et al.* 2001), different types of microphone (Price *et al.* 2000; Mubeen *et al.* 2012). In this research the difference between the recording environments over two different sessions is considered with in the speech database. However, the research does not explicitly investigate the channel effect (i.e. using different capturing device in enrolling and testing).

Chapter Four

Speech Features and Novel Speech Activity Detection

4. Introduction

The speech signal can be processed at several different morphological levels: sentences, words, syllables or phonemes. The lowest is the phonemes level, which represents the smallest part in any language structure. In speaker verification the choice of the morphological level is essential, since speaker identity information is not embedded equally over these levels (Han-Sheng and Mammone 1995a). On the other hand, selecting the type of feature vector to be used to present the speaker identity information is also important. A wide range of frequency domain features have been used in the literature. In this chapter a demonstration of the general characteristics of phonemes and vowels is first presented in Section 4.1. Different types of feature vectors for speaker verification are then illustrated in Section 4.2.

Pre-processing techniques are explained in Section 4.3 with details of three techniques for speech/vowel detection: energy frame analysis as presented in Section 4.3.1, zero crossing rate technique as detailed in Section 4.3.2, and a new proposed technique called linear correlation technique described in Section 4.3.3. The new technique is compared to energy frame analysis technique in Section 4.3.4 and compared to time domain correlation technique in Section 4.3.5.

4.1 Phonemes and Vowels

Phonemes are distinctive sounds that can be used to characterise most languages including English. Different languages may contain different phonemes, but many share the majority of them. Phonemes in American English for example, are classified mainly into consonants, vowels, semivowels and diphthongs (Rabiner and Schafer 2010). From among all phonemes, vowels are perhaps the more interesting patterns to classify sounds due to their distinctive spectral characteristics (Rabiner and Juang 1993). Although they are not vital to represent and classify written text, their role in speech/speaker recognition systems is very important.

Vowels are produced by quasi-periodic pulses of air caused by excited vocal cord vibration with fixed vocal tract (Rabiner and Schafer 1978). Their consistent characteristics in the frequency domain make them valuable for speech applications in general and particularly valuable for speaker recognition. Figure 4.1 shows the spectrogram of the ten vowels in American English language.

[redacted]

Figure 4.1 – Spectrogram of ten vowels of American English (Rabiner and Schafer 2010).

Figure 4.1 presents a full scale frequency description of the ten vowels, with the darker areas representing higher energy. It can be noticed that some of these vowels are more distinctive than others. For example the vowels /æ/ as in (b<u>a</u>t), /i/ as in (b<u>ee</u>t) and /u/ as in (b<u>oo</u>t) are the most the distinctive vowels, with vowel /æ/ containing the highest energy across the majority of the frequency range over time. Vowel /i/ contains two well separated energy regions (0-500 Hz) and (2000-4000 Hz), and vowel /u/ contains only one well recognised energy region (0-1000 Hz). Figure 4.2 plots the same ten vowels for a wide range of speakers in term of the first and second formant frequencies.

Although the rest of vowels in Figure 4.1 have their unique spectral representation, they can be confused with the three mentioned vowels. This can be easily spotted from Figure 4.2 where the overlapping regions show how vowels share common characteristics in term of the first and second formant frequencies.

Figure 4.2 – Vowels distribution of wide range of speakers in term of first and second formant frequencies (Peterson and Barney 1952).

Vowels with overlapping regions are very likely to share similar pronunciation. A cited study in Chapter 3 (Seddik *et al.* 2004a) argues that the use of phonemes with similar pronunciation (i.e. similar spectral characteristics) in the training phase of phoneme-based neural network, can confuse the network in the testing phase, thereby reducing the recognition performance. This is not the case with the three mentioned vowels, since they are statistically well separated with no overlapping area.

Figure 4.3 shows how the DFT spectrum can be used to differentiate between different speakers speaking the same vowel. Figure 4.3a shows an example of the

DFT spectrum of one vowel spoken by two different speakers whilst Figure 4.3b shows an example of the DFT spectrum of the same vowel spoken twice by one speaker.

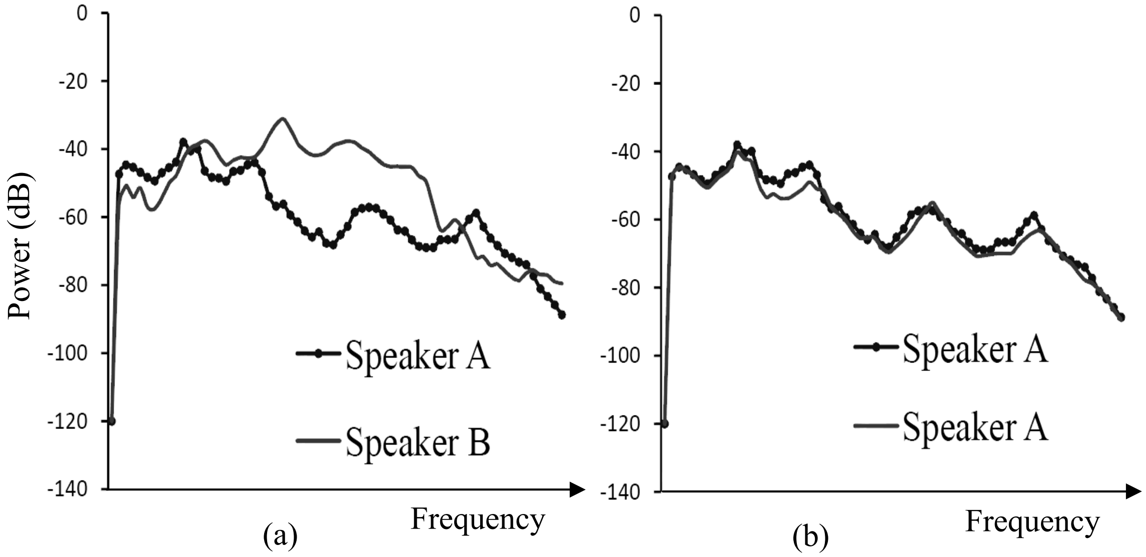

Figure 4.3 – Discrete Fourier Transform spectrum for vowel /æ/ spoken by a) different speakers b) same speaker twice.

It is clear from Figure 4.3 that vowel DFT spectrum for the two vowel utterances is far more similar when spoken by the same speaker than when spoken by different speakers. In this work the three common vowels (/æ/ as in five, /u/ as in two, and /i/ as in eight) were chosen due to their intra-speaker and inter-speaker discrimination property (Rabiner and Schafer 1978). Figure 4.4 clearly shows that the frequency spectrums of these three vowels are distinct for a given speaker as the maximum frequency spectrum amplitude (dark gray) occurs in different frequency regions for each of the three vowels.

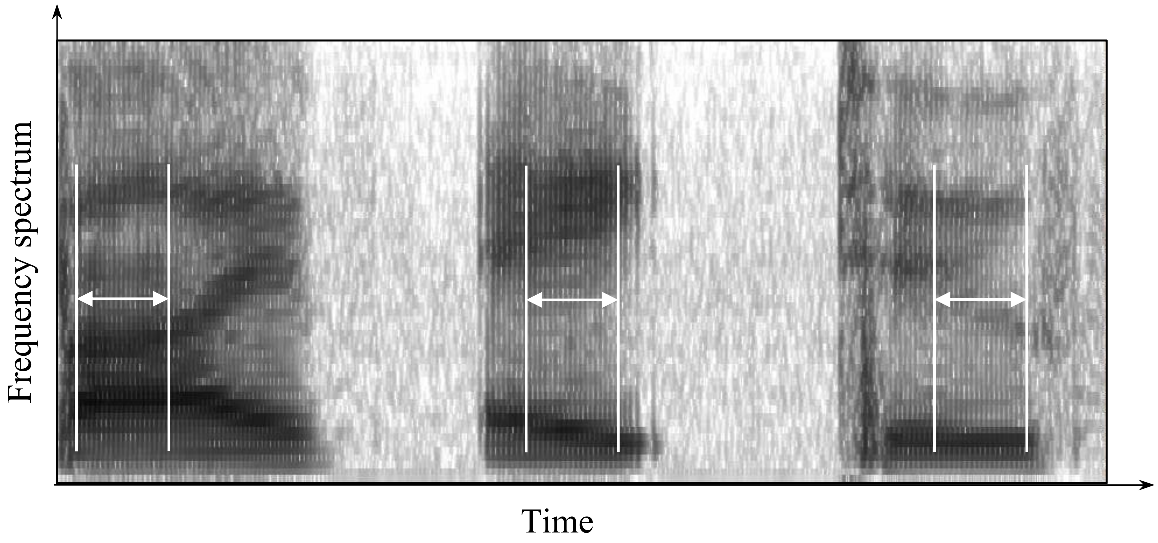

Figure 4.4 – Discrete Fourier Transform spectrum for the words (five, eight, and two).The frequency spectrum for each of three vowel segments indicated are clearly distinct.

Results presented in (Han-Sheng and Mammone 1995a) also show that these vowels contain more speaker identity information than other vowels and non-vowel phonemes. In addition, although other vowels are distinct and do contain identity information for a given speaker, they are not used here because they are shorter in duration than the vowels chosen in this work.

4.2 Feature Vectors for Speaker Verification

The choice of the format of the feature vector to be used in any speaker verification process has a significant impact on the performance of that process (Kinnunen and Li 2010). As shown in Chapter 3, many different feature formats have been adopted in the literature to conduct speaker verification/identification. The speech signal is basically generated by a combination of vibrations travelling through the vocal tract, throat and mouth. The signal is captured in the time domain as a sampled waveform in term of samples. However, this time domain representation is rarely used as a feature vector in speaker verification and other speech applications. The frequency domain representation of the speech signal is more intelligible and offers a more meaningful picture than the time domain (Rabiner and Schafer 2010). The Discrete Fourier Transform DFT spectrum, sometimes called power spectrum or spectrogram when plotted over time axis, is the raw representation format of the speech signal in the frequency domain and the majority of the frequency feature formats are derived from the DFT spectrum with further transformations. In this section three main feature formats will be described: DFT spectrum, LPC analysis/spectrum and MFCC coefficients.

4.2.1 Discrete Fourier Transform Spectrum

One of the most popular representations of speech signals, which gives a full description of speech in the frequency domain is the DFT spectrum. The DFT vector for a frame of speech with *N* samples is shown in Equation 4.1.

$$X[k] = \sum_{n=0}^{N-1} x[n]\, e^{-j(\frac{2\pi k}{N})n} \qquad 0 \le k \le N-1 \tag{4.1}$$

where *X*[*k*] is the DFT vector and *x*[*n*] is the windowed speech signal. The DFT spectrum is the magnitude of the DFT vector (Rabiner and Schafer 1978). This is usually presented in logarithmic scale format.

Figure 4.5 shows a frame of a vowel speech signal before and after the multiplication by a Hamming window. The Hamming window is commonly used in speech processing to minimise the discontinuity that occurs due to the segmentation in the speech waveform.

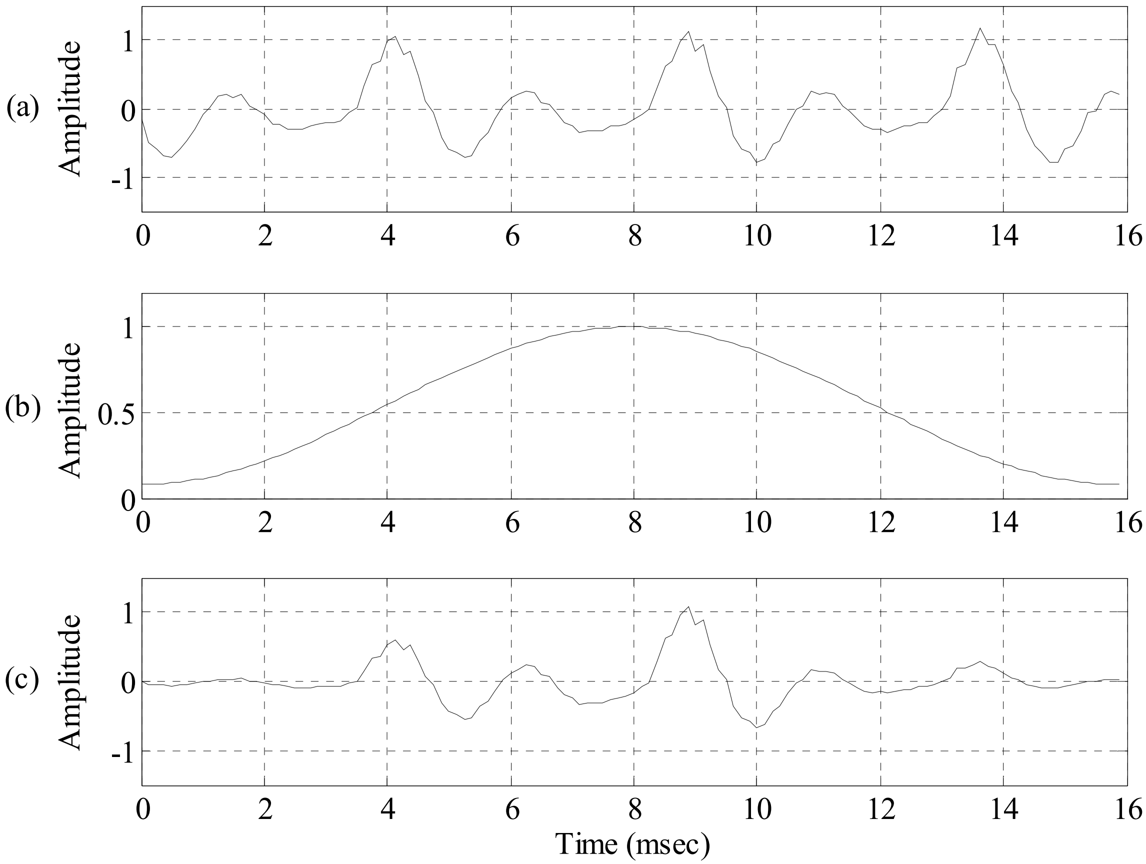

Figure 4.5 – Speech frame windowing a) frame of vowel speech signal b) Hamming window c) windowed speech signal.

The size of the DFT spectrum vector produced from Equation 4.1 is the same size as the number of samples (*N*). However, the resolution of the DFT spectrum can be controlled. To produce a DFT spectrum with more points than the size of the speech frame, the speech signal can be padded with zero samples. On the other hand if the size of the DFT spectrum vector is less than the size of the speech frame, the speech signal is truncated.

According to the derivation of the DFT spectrum, the size of the vector should always be two to the power of an integer number. The number of samples in

Figure 4.5 is 128 samples. Figure 4.6 shows different resolutions of the DFT spectrum of the windowed speech signal shown in Figure 4.5c.

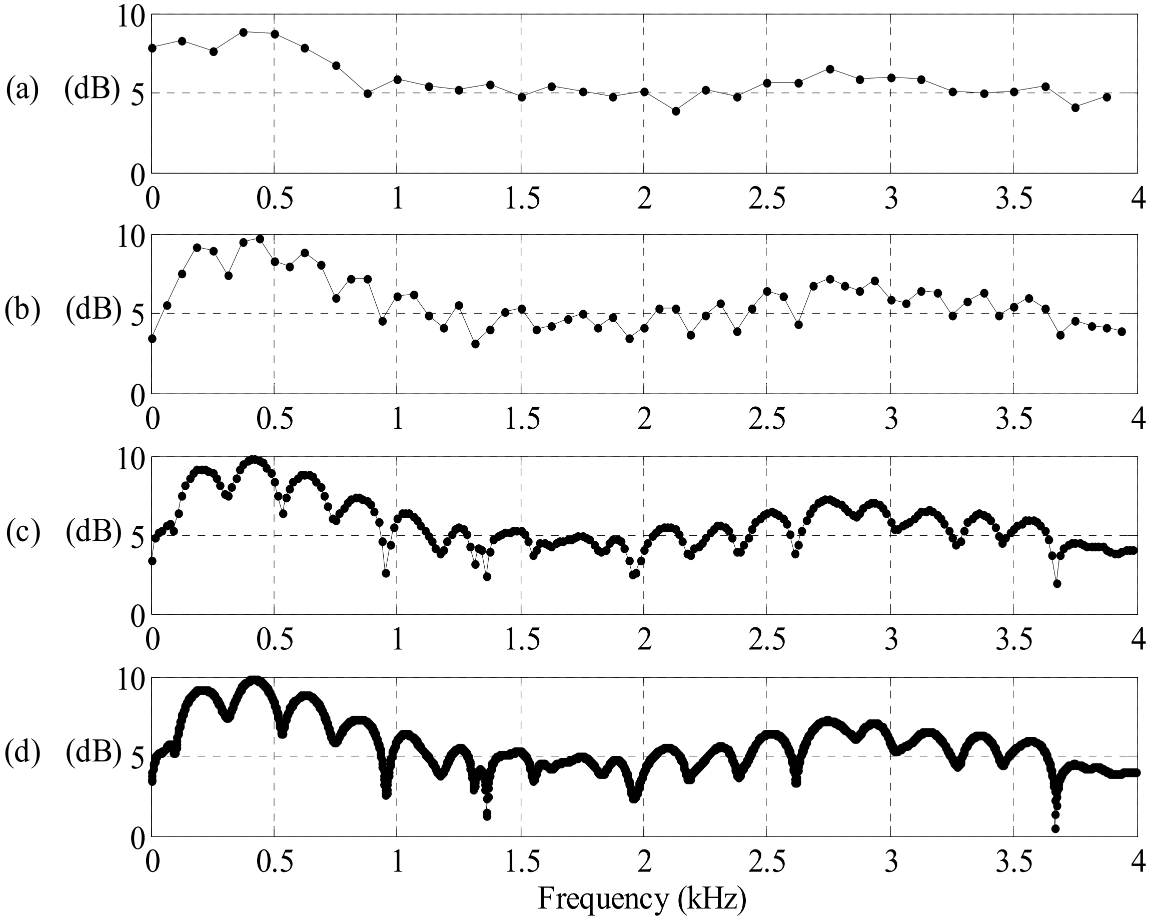

Figure 4.6 – Different resolutions of Discrete Fourier Transform spectrum a) 64-point b) 128-point c) 512-point, and d) 4096-point.

The DFT spectra shown in Figure 4.6 illustrate scenarios where the size of the DFT spectrum vector is less, equal to or more than the size of the speech frame. It is clear that the more points the DFT spectrum contains, the more resolution is obtained. However, the higher resolution is not necessarily adding extra information i.e. compare Figure 4.6c and Figure 4.6d to Figure 4.6b. Meanwhile decreasing the resolution effectively subtracts significant information since the original signal is truncated as shown in Figure 4.6a. In this research different DFT spectrum vector sizes were investigated, and it was found that increasing the resolution of the DFT spectrum vector size does not improve the verification accuracy. Decreasing the resolution on the other hand significantly worsens the verification performance. Therefore a DFT spectrum size of 128 points (which is equal to the size of speech frame of 16 msec) is used throughout the rest of this thesis.

4.2.2 Linear Prediction Coefficients Analysis/Spectrum

Another popular feature vector format is the Linear Prediction Coefficients (LPC). Their significant use appears in speech compression applications due to the high compression ratio that can be obtained when representing a speech signal. The LPC spectrum can be obtained by taking the magnitude at the output of the transfer function of a filter whose coefficients are represented by the LPC coefficients as shown below:

$$LPC\ spectrum = \left|H[z]\right|^2 = \left|\frac{G}{1+\sum_{i=1}^{M} a_i\ z^{-i}}\right|^2 \tag{4.2}$$

where $H[z]$ is the transfer function of the filter, G is the gain and a_i (i=1,…, M) are the LPC coefficients of the M order (Rabiner and Schafer 1978). Mathematically, the LPC spectrum represents a smoothed version of the DFT spectrum for low LPC orders. Figure 4.7 shows LPC spectrum obtained using different M values 10, 40 and 128 coefficients as well as the 128-point DFT spectrum vector.

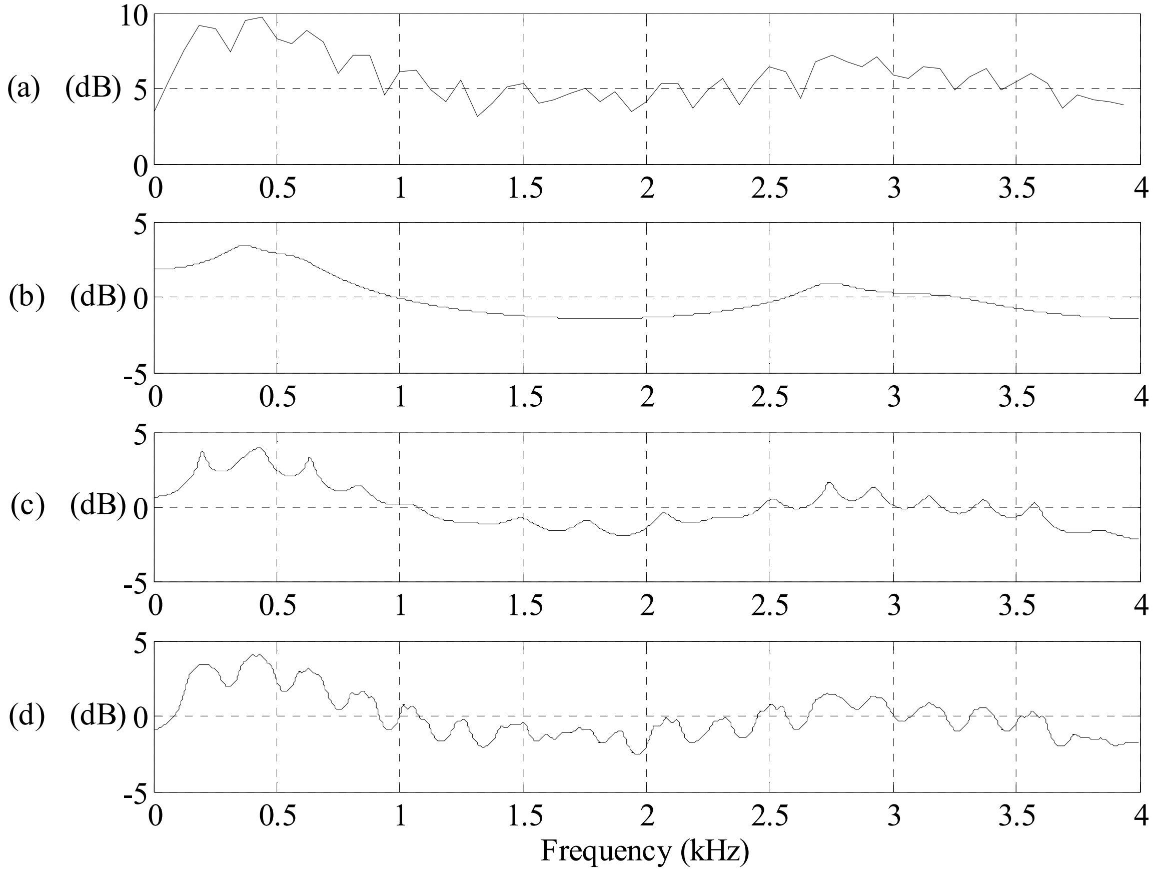

Figure 4.7 – Different resolutions of Linear Prediction Coefficients spectrum a) 128-point Discrete Fourier Transform spectrum b) 10th order Linear Prediction Coefficients spectrum c) 40th order Linear Prediction Coefficients spectrum, and d) 128th order Linear Prediction Coefficients spectrum.

It is noticed that the 10th order LPC spectrum in Figure 4.7b follows the main trend of the data when compared to the 128-point DFT spectrum in Figure 4.7a. By increasing the LPC order to 40 more detail appears that better follows the fine envelope of the formants in the DFT spectrum. Finally the 128th order LPC spectrum in Figure 4.7d shows a very similar image to the DFT spectrum in Figure 4.7a. This is expected to happen since no compression is applied, and the 128 speech samples are fully described as LPC coefficients.

4.2.3 Mel Frequency Cepstrum Coefficients

Mel Frequency Cepstrum Coefficients, or MFCC, are one of the most used feature vector formats in the literature for both speech and speaker recognition, as illustrated in Section 3.4. After their successful usage in speech recognition, MFCC were extensively used in speaker verification and identification. As described in (Davis and Mermelstein 1980) the MFCC are calculated by applying a Mel-frequency bank of triangular filters on the DFT spectrum, then a Discrete Cosine Transform (DCT) is applied on the logarithmic output of the filters to obtain the MFCC. Figure 4.8 shows the scheme diagram of the MFCC extraction process.

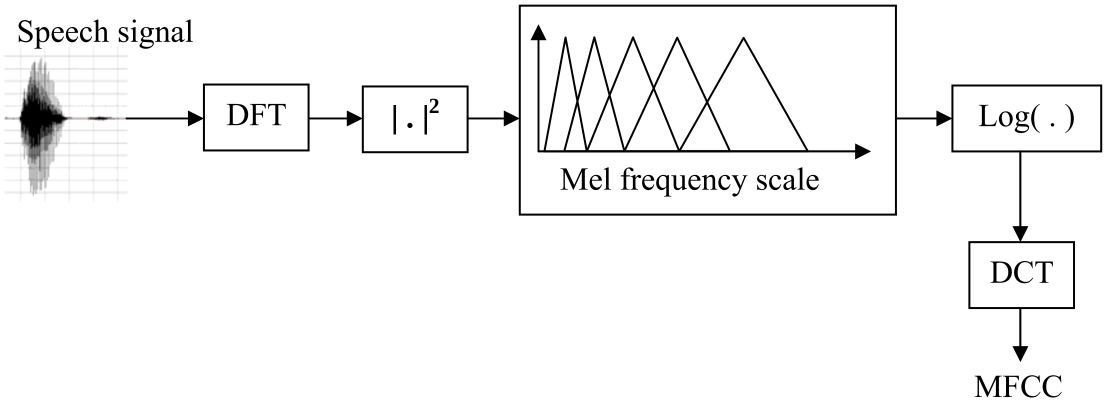

Figure 4.8 – Mel Frequency Cepstrum Coefficients extraction process.

As shown in Figure 4.8 the DFT spectrum is passed through a bank of triangular filters, the centres of these filters follow a mel-scale frequency which is described in (Memon *et al.* 2009) as:

$$f_{mel} = 2595 \log_{10}\left(1 + \frac{f}{700}\right) \qquad (4.3)$$

where f_{mel} is the mel-scale frequency and f is the linear frequency. After multiplying the DFT spectrum by the bank of triangular filters, the DCT is applied to the logarithmic values of the filters output energies to produce the MFCC as follows:

$$\left.\begin{aligned} MFCC_k &= w[k]\sum_{i=0}^{K-1} E[i]\, Cos\left[\frac{\pi\,(2i+1)\,k}{2K}\right] \\ w[k] &= \begin{cases} 1/\sqrt{K} & k=0 \\ \sqrt{2/K} & 1 \le k \le K-1 \end{cases} \end{aligned}\right\} \tag{4.4}$$

where $MFCC_k$ is the k^{th} MFCC coefficient with k=0, 1, 2, …, K-1 and $E[i]$ is the logarithmic value of the output energy of the i^{th} filter bank. Usually, for normalisation reasons, $MFCC_0$ is excluded from the feature vector since it represents the energy within the speech frame (Molla and Hirose 2004). Similar to an LPC analysis, the MFCC describes a compressed version of the DFT spectrum. By increasing the order of the cepstrum coefficients K, the MFCC vector converges to the DFT spectrum vector.

4.3 Pre-Processing Techniques

Speech segmentation is an essential tool in many speech applications. For example, a speaker verification system that uses phoneme/vowel information to perform the verification operation, will need an accurate speech segmentation technique in order to detect the phoneme/vowel boundaries correctly and precisely.

In traditional speech applications, Energy Frame Analysis (EFA) is commonly used to detect voiced regions in the speech signal (Dong *et al.* 2002; Qi *et al.* 2002). The Zero Crossing Rate (ZCR) of the speech signal is another technique that is usually used in combination with energy frame analysis to locate unvoiced speech in the time domain (Rabiner and Schafer 1978). A time-domain based correlation function also had been used to detect speech activity; either on its own (Ta-Hsin and Gibson 1996; Zhang *et al.* 2009) or in combination with the ZCR technique (Shen and Chen 2011). However, one of the problems when using time-domain speech detection techniques such as EFA and ZCR is how to set the respective energy frame and zero crossing rate thresholds. The energy threshold is impacted by the volume of the spoken words, whilst the zero crossing rate threshold is speaker dependent (Rabiner and Schafer 1978). To address this potential limitation, other speech detection techniques have been suggested that are based on frequency domain analysis features such as cepstral features (Haigh and Mason 1993). The technique presented here extends this work.

In the next three sections the following pre-processing techniques are demonstrated, Section 4.3.1 and Section 4.3.2 describe the EFA technique and the ZCR technique respectively, while Section 4.3.3 proposes a new DFT based pre-processing technique using the linear correlation technique. Section 4.3.4 shows a comparison between the proposed technique and the EFA technique, whilst Section 4.3.5 provides another comparison with a time-domain based correlation technique.

4.3.1 Energy Frame Analysis

Energy gives a good indication for voiced speech activity. To obtain the energy envelope of a speech signal the energy must be calculated within frames of up to 30 ms in order to ensure that the stationary assumption of the speech signal is valid (Rabiner and Schafer 1978). Equation 4.5 shows the energy calculation for a frame of *N* samples in speech signal *x*[*n*].

$$E_i = \sum_{n=1}^{N} x[n]^2 \tag{4.5}$$

where E_i is the energy of the i^{th} frame of speech signal and *x*[*n*] is the amplitude of the speech signal in the time domain. After calculating the energy envelope a threshold can be set to decide whether the frame represents voiced or unvoiced speech.

The setting of the threshold value is highly affected by two parameters; the volume of the spoken speech and the background noise level. Speech waveforms of high volume need high threshold values in order to avoid detecting unvoiced speech segments. Meanwhile speech waveforms of low volume require lower threshold values in order to avoid miss-detecting voiced speech segments. This volume dependency means that a general threshold cannot be set for all speech waveforms. For example, a threshold value that would successfully detect the speech activity regions in high volume waveform may not be able to detect the similar activity in a lower volume waveform. The common way to address this is by re-tuning the threshold values according to the average volume of the speech waveform at a word or phrase level.

One other limitation of using EFA based threshold values for detecting speech activity is the effect of the background noise level. In general the speech activity thresholds need to be set such that they exceed the energy of the background noise. Unfortunately, such a solution means that in noisy environments, low volume speech activity is difficult to segment from the background noise.

4.3.2 Zero Crossing Rate

Another speech signal property that can be extracted in the time domain is the zero crossing rate. When the ZCR is calculated for a speech waveform, it can be used to easily discriminate between low frequency voiced speech segments (especially vowels) and higher frequency unvoiced speech segments. The ZCR can be computed by counting the change in the sign of the speech samples within one frame as shown in Equation 4.6.

$$ZCR_i = \sum_{n=1}^{N-1} \left\{ \begin{matrix} 1 & x[n]\,x[n+1] < 0 \\ 0 & x[n]\,x[n+1] \geq 0 \end{matrix} \right\} \tag{4.6}$$

where ZCR_i is the zero crossing rate of the i^{th} frame of speech signal and $x[n]$ is the amplitude of the speech signal in the time domain. By calculating the ZCR for each frame in the speech waveform a ZCR envelope is obtained.

A threshold value can then be used to locate high frequency speech regions such as the /t/ at the end of the word (eigh<u>t</u>) and the beginning of the word (<u>t</u>wo). Although ZCR is a useful tool for isolating high frequency unvoiced speech (fricatives), it is still necessary to retune the threshold value for each new speaker, as the ZCR is dependent on the fundamental frequency of each speaker (Rabiner and Schafer 1978).

ZCR is usually used in combination with EFA to obtain an improved technique for voiced speech detection. In this combination, ZCR is used to detect the low energy high frequency fricative speech segments that normally cannot be detected using EFA.

4.3.3 Linear Correlation

The Linear Correlation Coefficient (LCC) or Parson product-moment correlation coefficient can compute the correlation between any two vectors (Rodgers and Nicewander 1988). The correlation coefficient is obtained by dividing the covariance of the two vectors by the product of their standard deviation as shown in Equation 4.7.

$$LCC(x,y)=\frac{N\left(\sum_{i=1}^{N}x_i\ y_i\right)-\left(\sum_{i=1}^{N}x_i\right)\left(\sum_{i=1}^{N}y_i\right)}{\sqrt{N\left(\sum_{i=1}^{N}x_i^2\right)-\left(\sum_{i=1}^{N}x_i\right)^2}\sqrt{N\left(\sum_{i=1}^{N}y_i^2\right)-\left(\sum_{i=1}^{N}y_i\right)^2}} \tag{4.7}$$

where x_i and y_i are two vectors of N samples. A value of 1 refers to fully correlated vectors and -1 refers to fully uncorrelated vectors. In this work it is suggested that the LCC in Equation 4.7 can be used to determine the correlation between DFT spectrum vectors in order to detect speech activity.

Speech signals are easier to investigate in the frequency domain than in the time domain (Rabiner and Schafer 1978). This is because the DFT spectrum of a speech segment gives a distinctive image of the speech signal which can be used to provide sufficient information about different voice characteristics in the frequency domain. Figure 4.9 shows the DFT spectrum of the spoken digits (five/eight/two) from the CSLU2002 database. By computing the LCC between each DFT spectrum vector and all other vectors, a two dimensional matrix will be obtained as shown in Figure 4.10.

In Figure 4.10 the highly correlated speech frames are represented as dark regions (close to a value of 1), while the lighter regions represent the highly uncorrelated speech frames (close to a value of -1). Although Figure 4.10 shows the full cross-correlation map of the speech waveform, which could be useful in speech recognition applications, it is not necessary to use all of the elements of the two dimensional matrix in order to locate the speech segments in the waveform.

To obtain detection ability, it is suggested here to slide a two dimensional window along the diagonal of the matrix, then sufficient information can be collected to locate the highly correlated speech regions precisely.

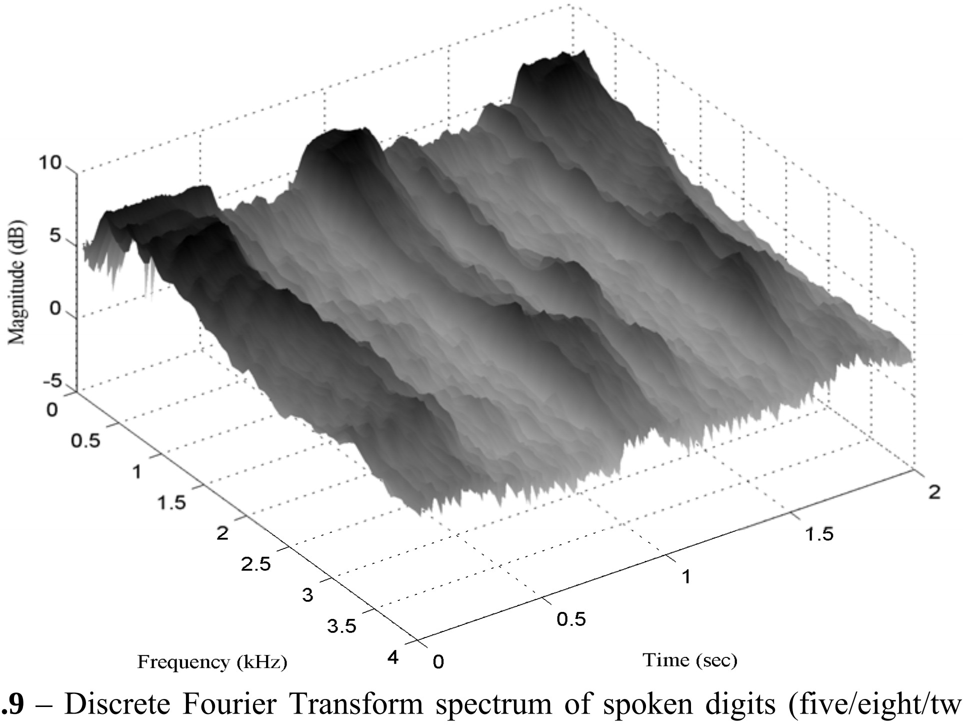

Figure 4.9 – Discrete Fourier Transform spectrum of spoken digits (five/eight/two) from the CSLU2002 database.

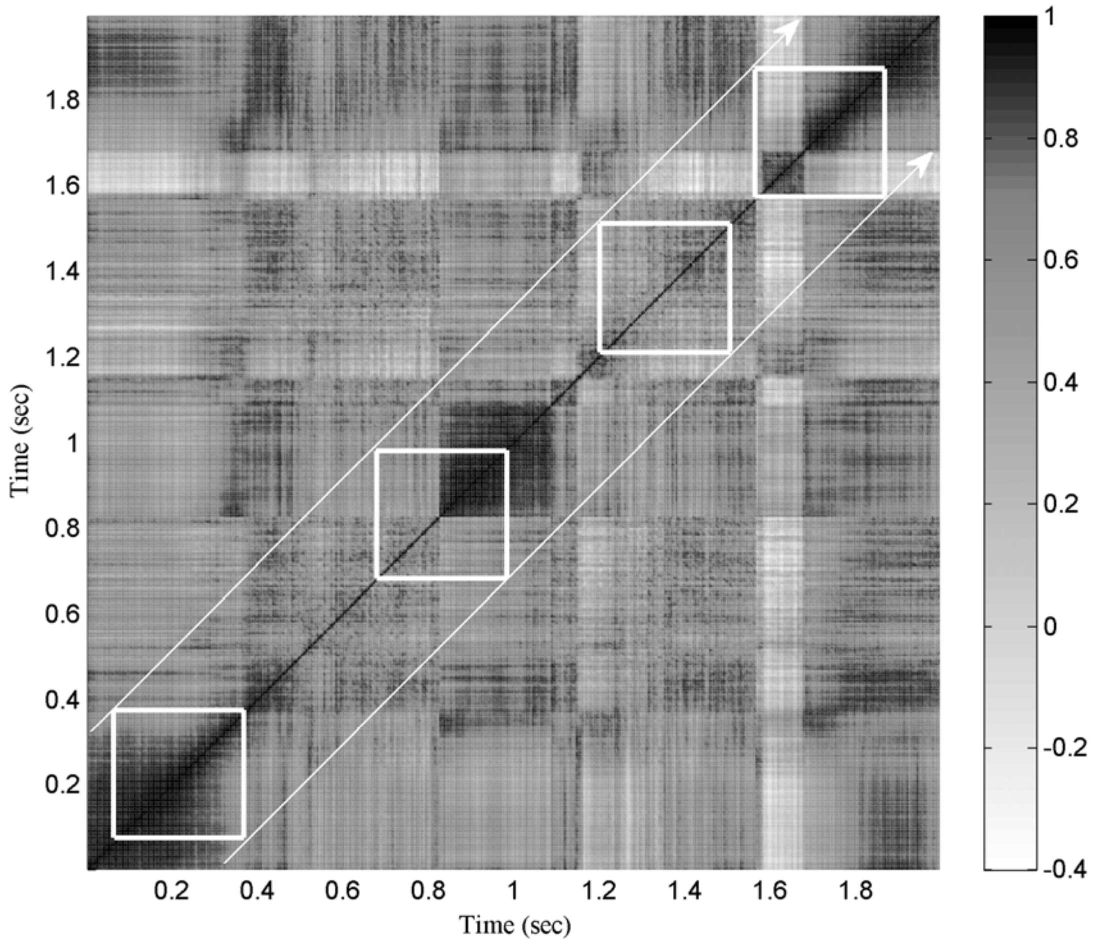

Figure 4.10 – Linear Correlation Coefficient values map for illustrated speech waveform.

It is noticed that the summation of the LCC's within the square window gives significant indication about the coherency between the speech frames inside the window. The proposed i^{th} Correlation Coefficient Envelope (CCE) is suggested to be as follow:

$$CCE_i = \frac{\sum_{n=1}^{D}\sum_{m=1}^{D} LCC(X_n, X_m)}{D^2} \times 100\% \tag{4.8}$$

where D is the number of frames considered in the square time window, X_n and X_m are the DFT spectrum vectors along both dimensions of the time window. Equation 4.8 produces a percentage scale that describes the LCC characteristic for the assessed waveform. For fully correlated speech frames of LCC value of 1 inside the square window, the resultant CCE is 100%, whilst for fully uncorrelated speech frames of LCC value of -1, the resultant CCE is -100%. The CCE for the same speech waveform illustrated previously is shown in Figure 4.11.

It is clear from Figure 4.11 that the highly correlated frames (>80%) correspond to the speech regions of the displayed waveform. Moreover, a threshold of >90% indicates the vowel regions of the three spoken words.

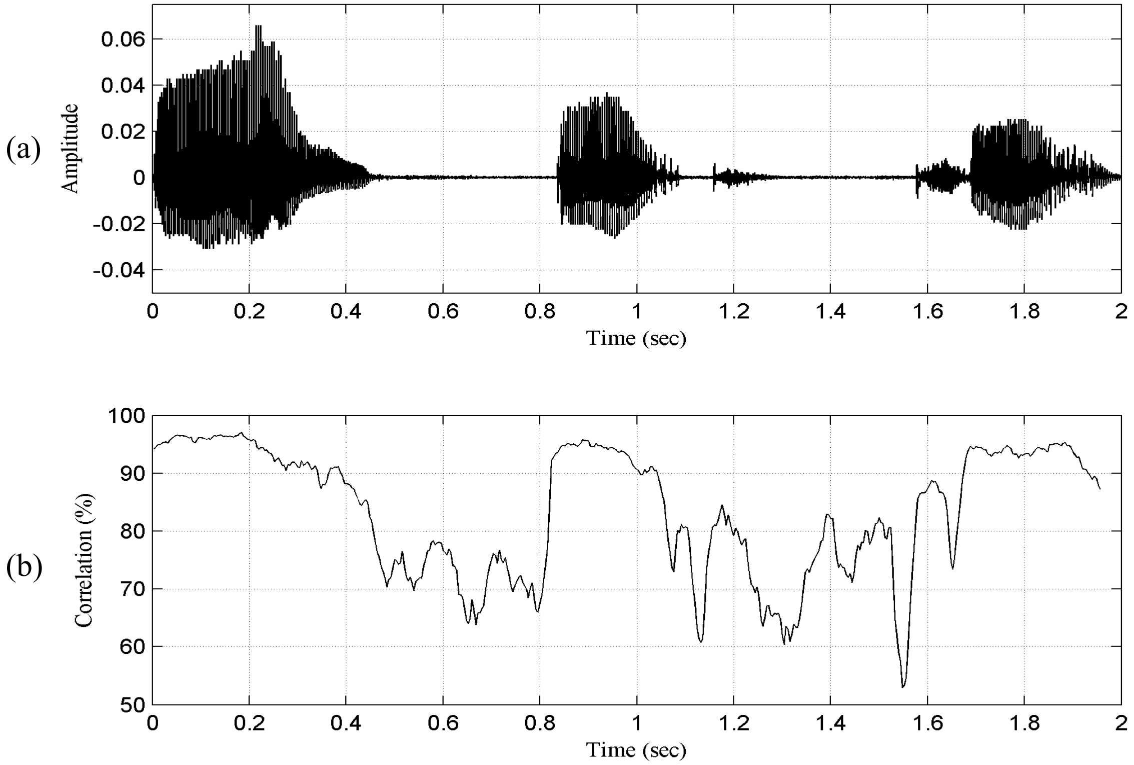

Figure 4.11 – Correlation Coefficient Envelope of spoken digits (five/eight/two) from CSLU2002 database a) time domain speech signal b) Correlation Coefficient Envelope.

4.3.4 Comparison between Linear Correlation and Energy Frame Analysis

Although the EFA algorithm is known to be highly biased by the energy of the signal, it can be shown that the LCC algorithm is robust to changes in energy.

Figure 4.12 shows the performance of the EFA and LCC under different volume levels.

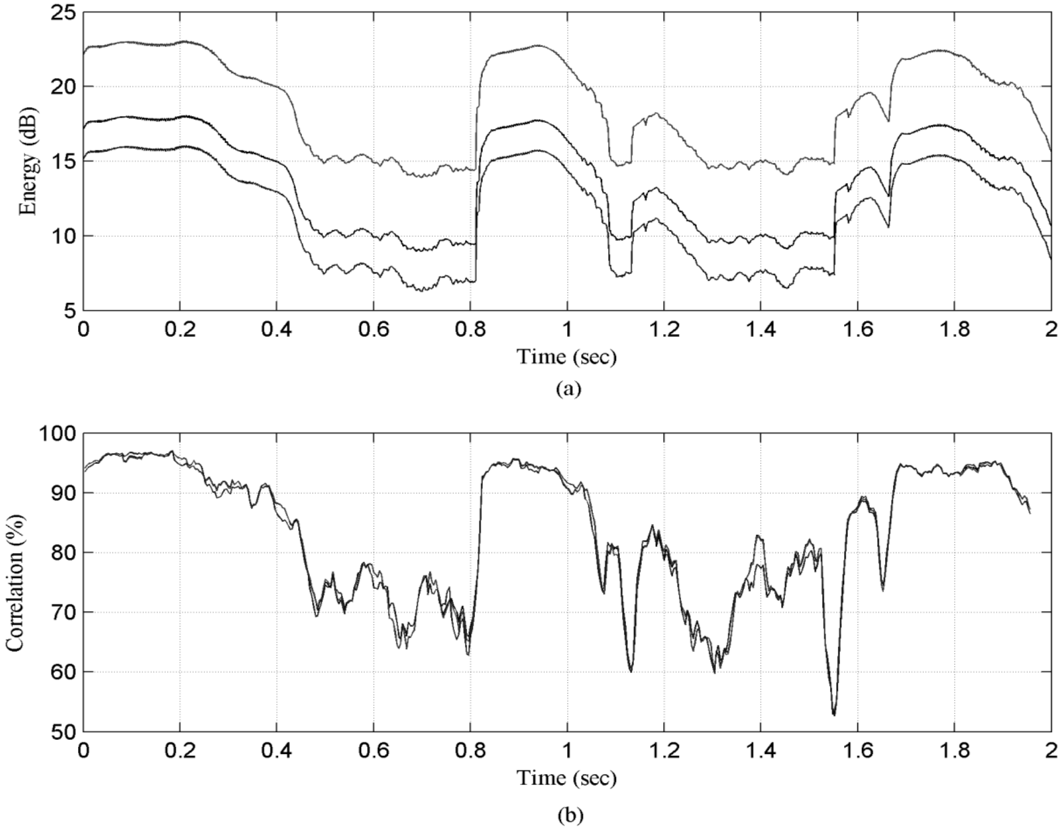

Figure 4.12 – Speech waveforms in different volume levels represented using a) Energy Frame Analysis and b) Correlation Coefficient Envelope.

It is clear from Figure 4.12a that when EFA is used a retuning is needed for the threshold that detects speech activity due to the shift in the average amplitude of the signal. However, when LCC is used there is no need to retune the threshold that detects speech activity because the LCC algorithm produces the same envelope irrespective of the volume of the spoken phrase as shown in Figure 4.12b (i.e. the three correlation envelopes overlay each other).

The advantage of the LCC algorithm over the EFA algorithm is even more obvious when volume variation within a phrase is considered. In speech a speaker often starts speaking with high volume, then allows the volume to decrease over time. Under such conditions, it is more difficult to set a phrase threshold value for the EFA algorithm, something that is not the problem when the LCC algorithm is used. Figure 4.13 shows the EFA and LCC performance against volume degradation within a speech waveform.

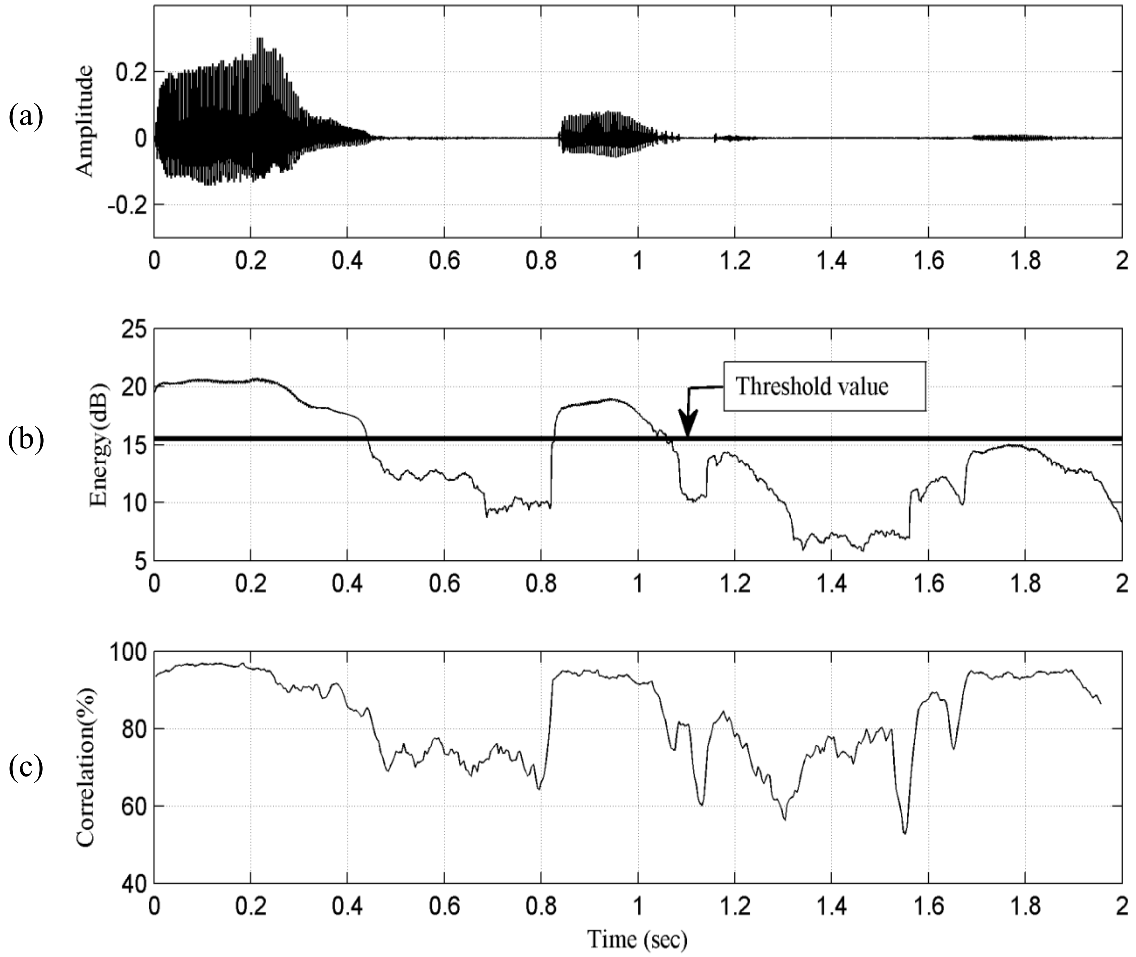

Figure 4.13 – Volume degradation over time in speech waveform represented using a) time domain speech signal b) Energy Frame Analysis envelope and c) correlation Coefficient Envelope.

In Figure 4.13 the EFA plot shows how a typical phrase threshold value can easily miss the third word, while the CCE plot shows the advantage of using the LCC algorithm to detect speech activity – it is not affected by the change in volume even within the same speech waveform.

4.3.5 Comparison between Linear Correlation and a Correlation Function in the Time Domain

The auto correlation function can also be used to detect speech activity in the time domain. The system in (Zhang *et al.* 2009) uses the time correlation function to discriminate between speech and noise, as shown in Equation 4.9.

$$Corr(i) = \frac{1}{4N(N+1)} \sum_{m=1}^{2N} \sum_{\substack{n=-N \\ n\neq 0}}^{N} x[n]\, x[n+m] \tag{4.9}$$

where *Corr(i)* is calculated per *i*th speech frame and *N* is the length of speech frame. The exclusion of (*n=0*) from Equation 4.9 is claimed to eliminate the effect of the embedded energy within the frame. In that research the correlation calculation is modified by dividing each *Corr(i)* value by the calculated energy for that frame. Figure 4.14 shows the correlation envelope obtained from Equation 4.9 and the proposed frequency domain correlation envelope for the same speech waveform used in this research.

It can be noticed from Figure 4.14 that the time-domain based correlation function does not provide the same level of coherency for the speech activity regions in different words (i.e. the peaks have different amplitudes). The frequency domain based correlation technique suggested in this thesis, on the other hand, does give a constant level of coherency for speech activity across different words. Consequently, with time domain based correlation there is still a need to set more than one threshold experimentally according to the speech waveform; unlike the case in the frequency domain based correlation where a global speech activity detection threshold can be used.

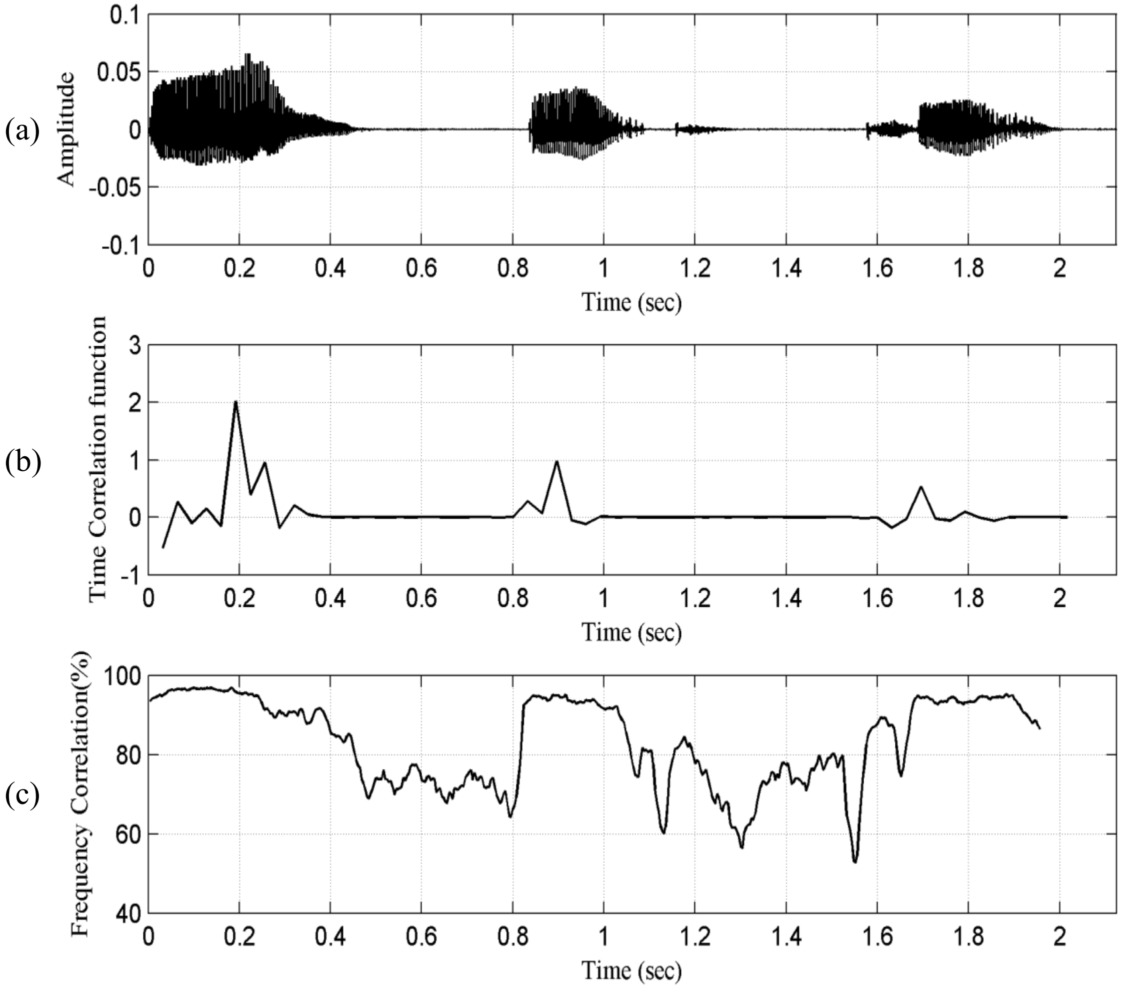

Figure 4.14 – Comparison between time domain correlation envelope and frequency domain correlation envelope a) Time domain speech signal of spoken digits (five/eight/two) from CSLU2002 database b) Time domain correlation using Equation 4.9 c) Correlation Coefficient Envelope.

4.4 Summary

This chapter details the general characteristics of phonemes, highlighting vowels and their impact in speaker verification process. A brief presentation of commonly used feature vectors is also presented. Following this a new technique for speech activity detection is then proposed. The technique employs a linear correlation coefficient algorithm between DFT spectrum feature vectors of overlapped frames of the speech signal. The key point of using this technique is that it requires minimal parameter setting. Two comparisons have been made with traditional speech detection techniques. The first is against the EFA algorithm and the second is against a time-domain based correlation function technique. The proposed LCC technique shows significant increase in robustness over EFA technique when the dynamic range of the volume of the speech signal increases. The LCC technique also shows a steady level of coherency when compared to time-domain based correlation technique.

Chapter Five

Self Organising Map Based Speaker Verification

5. Introduction

This chapter presents two proposed algorithms for speaker verification. The first is an SOM based algorithm, which employs a modified version of the SOM. The second algorithm then investigates the use of the modified SOM as a coarse verification stage, followed by a conventional MLP neural network as a fine second stage verifier. Section 5.1 demonstrates the two proposed algorithms with experimental results in Section 5.2 and Section 5.3, while Section 5.4 provides final conclusions from the two experiments.

5.1 Proposed Algorithms

In this chapter two vowel-based speaker verification algorithms are proposed. The first algorithm uses the outputs of a modified SOM for vowel clustering followed by a rule based Euclidian scoring method, while the second algorithm uses the modified SOM combined with MLP networks in order to benefit from the use of negative impostors' training samples; thereby improving verification performance.

5.2 Speaker Verification Using Modified Self Organising Map

Self organising maps are an intelligent clustering technique that is based on biological principles. One of the most popular SOM for speech applications is Kohonen's self Organising map since it clusters speech into a full scale of phonemes (Kohonen 1990). The authors in (Homayounpour and Chollet 1995) use the SOM in two stages to create a target speaker model and a general background speaker model; their paper uses LPCC as the feature vector parameters. The authors in (Ig-Tae *et al.* 2000) use an SOM to extract MFCC features in order to generate input to an MLP for speaker verification. The authors in (Mafra and Simoes 2004) use an SOM to create a speaker model, for each individual speaker in a database, for speaker identification purposes. Their paper also uses MFCC's as the input feature vector parameters. Finally Kinnunen *et al* in (Kinnunen *et al.* 2000) use an SOM as a clustering technique to obtain an MFCC vector based quantization codebook for an identification system.

Kohonen also made modifications to the original SOM to present a Learning Vector Quantization (LVQ) system (Pandya and Macy 1995). The main difference between the original SOM and the LVQ system is that LVQ has a specific number of categories in its output, each category represents a cluster which consists of a group of neurons, whilst in the SOM the number of the clusters is found in an unsupervised manner. More explanation on the LVQ with different versions LVQ1, LVQ2, and LVQ3 can be found in (Pandya and Macy 1995).

In this research an SOM is modified to have a specific number of categories in its output as in the LVQ system, but it differs from the LVQ in that each category consists of only one neuron instead of a group of neurons in order to simplify the verification process. In addition, a winner only update criterion is employed with a specific distance threshold in order to automatically remove silence and non-voiced frames.

The proposed algorithm will be described here using the DFT spectrum as the preferred input vector format. However, as described later, the system can easily be modified to use the LPC spectrum or MFCC as input vectors. The basic concept of the algorithm is based on the differences between the DFT spectrums of the same vowels for different speakers (Rabiner and Schafer 1978) as described in Chapter 4 (Section 4.1). Figure 5.1 shows the scheme diagram of the proposed algorithm. In Figure 5.1, both the registration SOM and the verification SOM are trained each time a speech sample occurs at their inputs. Section 5.2.1, Section 5.2.2, and Section 5.2.3 describe each stage of the proposed algorithm.

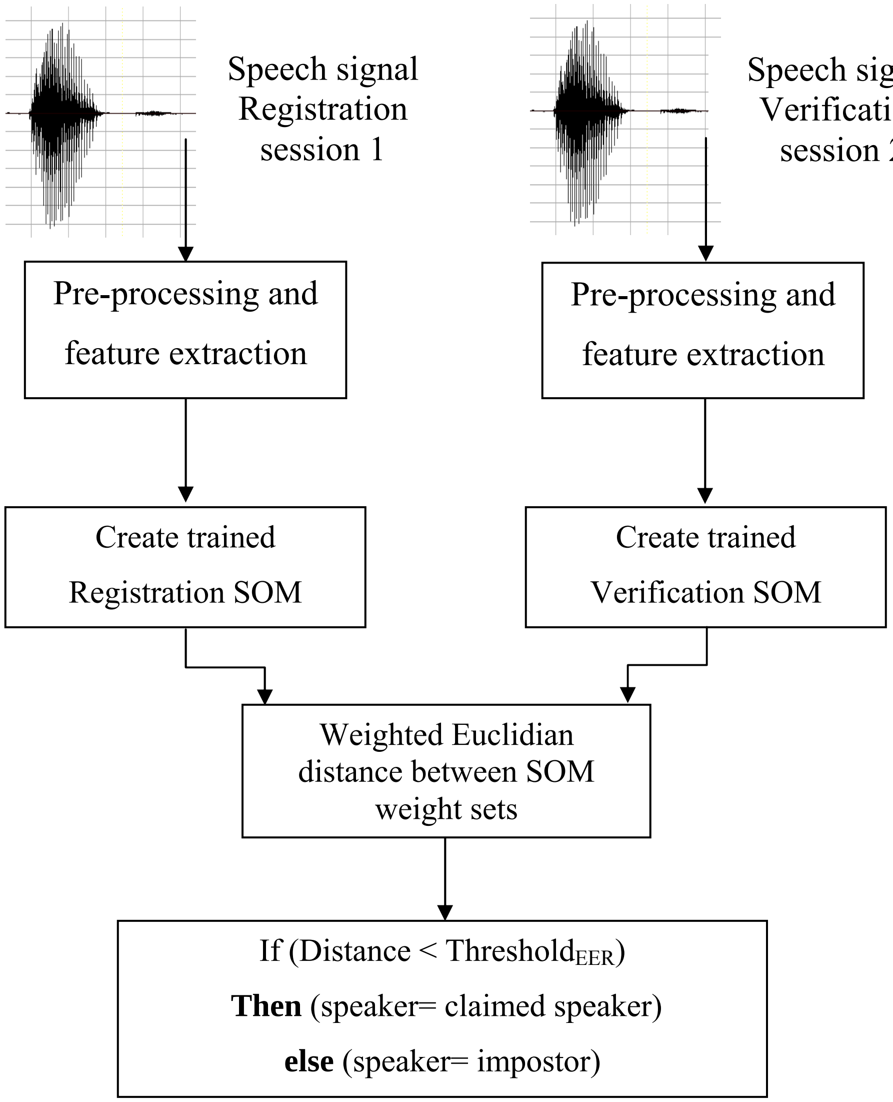

Figure 5.1 – Scheme diagram of the proposed algorithm.

5.2.1 Pre-Processing and Feature Extraction

The speech is segmented into frames of 16 msec. This frame size was chosen after studying different frame sizes from 5 to 32 msec. Frame sizes of less than 16 msec produce a lower resolution in the frequency domain (resulting in poor clustering results), whilst frame sizes of more than 16 msec may contain transition between phonemes. An overlapping frame analysis was also used with a 4 msec step. This functions as a smoothing tool for the DFT spectrum over successive frames. The contents of each frame are multiplied by a Hamming window, as in Equation 5.1, to reduce the distortion in the signal caused by the framing process.

$$\left.\begin{aligned} y(n) &= x(n)\,.\,w(n) & 0 \le n \le N-1 \\ w(n) &= 0.54 - 0.46\cos\left(\frac{2\pi n}{N-1}\right) & 0 \le n \le N-1 \end{aligned}\right\} \tag{5.1}$$

N is the number of samples in one frame and *w(n)* is the Hamming window.

Applying Equation 4.1 in Chapter 4 to each 16 msec frame of speech data produces 128 point DFT components. Due to the symmetry in the DFT spectrum only one half-side of the spectrum (64 components) is used. Three points smoothing, with average energy subtraction, is then applied to produce the input features vectors used in the clustering process. To determine the word boundaries, the energy frame analysis has been used.

5.2.2 Self Organising Map Registration and Verification Training

A one dimensional SOM of 64 input (DFT spectrum) and three output neurons (each output represents one vowel of size 64) is trained to produce weight vectors for the three output nodes that are representative of a given speaker. The three neurons are initially seeded with typical vowel samples taken from the three words (five, eight, and two) of the CSLU2002 database.

It is worth repeating that each time the SOM is used, either during registration or verification, the SOM is trained using its respective input speech sample. During its training phase the SOM is designed to update the winner neuron only if the input pattern lies within a specific distance region of the winner's current weight vector. Figure 5.2 illustrates the training process of the three output neurons in two-dimensional weight space.

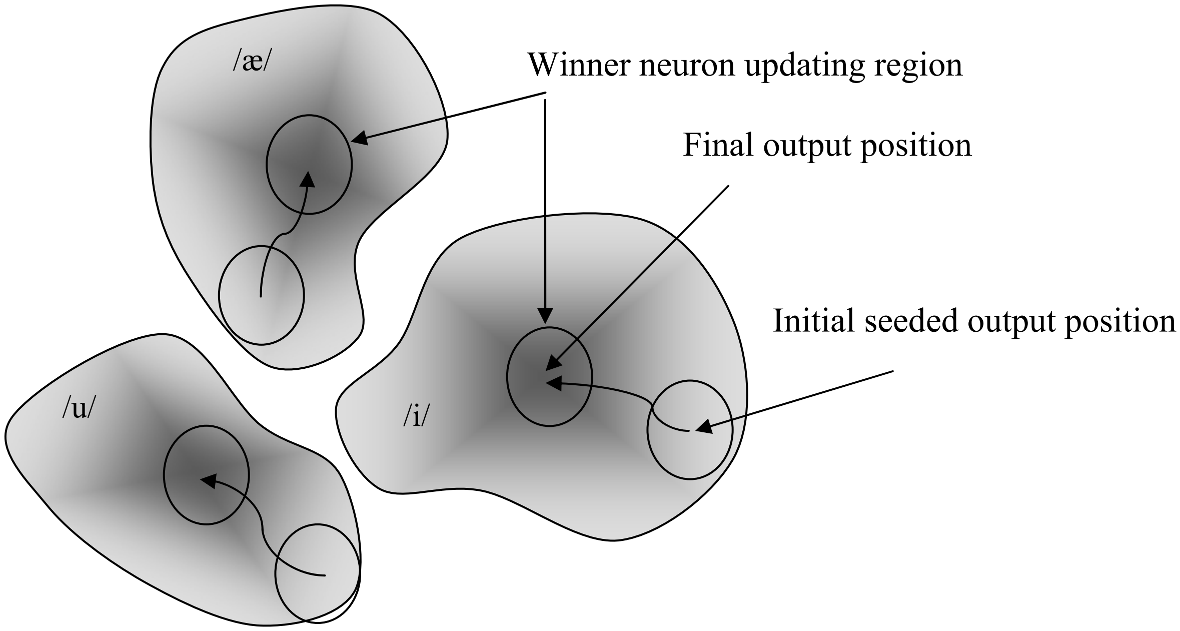

Figure 5.2 – Self Organising Map training process.

As shown in Figure 5.2, at the onset of training, the weight vectors of the three output nodes of the SOM are first seeded with initial vowel information from predefined positions within the speech signal. As training progresses, the weight vectors of the output nodes respectively move through the weight space to a position representing the greatest density of input vectors for each vowel as exemplified by the darkest point in each vowel area in Figure 5.2. At the end of training, the SOM thus represents a statistical three vowel voice model of the training speaker.

The update distance threshold 3.435 was optimised experimentally to achieve clustering of the vowel’s components whilst not clustering silence and non-vowel components as well so as to obtain the best verification accuracy within the speaker samples in the registration session.

After experimenting with different numbers of epochs, 100 epochs with an initial learning rate of 0.1 (decreasing linearly to zero over time) was found to be sufficient to ensure a successful clustering results. At the end of the clustering process each output neuron represents a unique vowel model for a specific speaker. The structure of the SOM is shown in Figure 5.3.

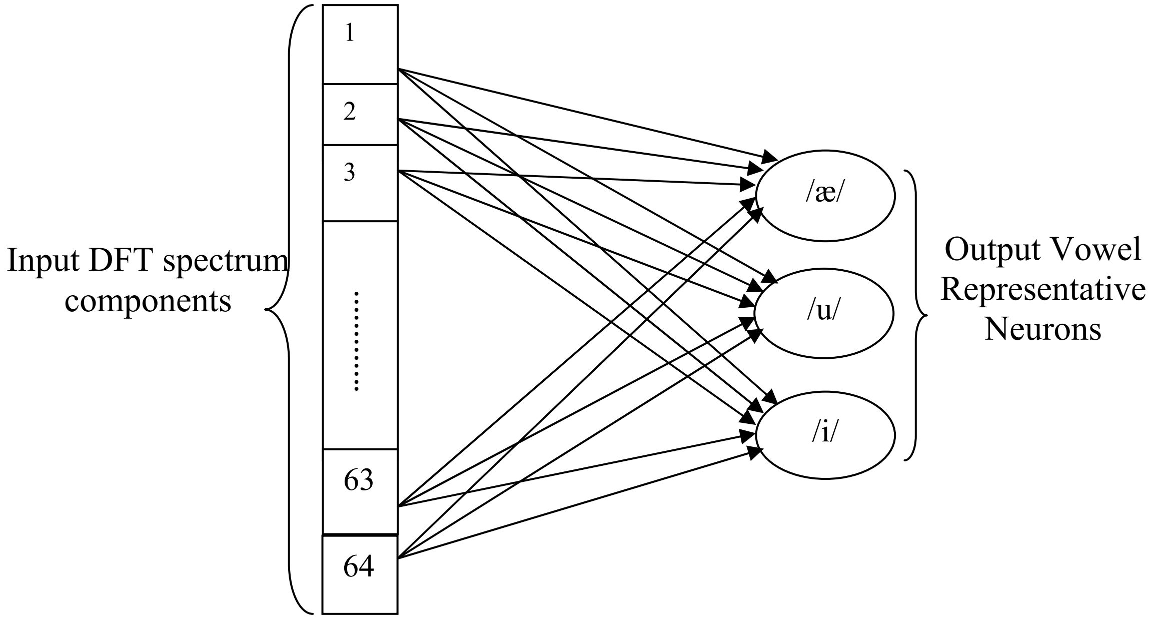
1
2
3
63
64
Input DFT spectrum components
/æ/
/u/
/i/
Output Vowel Representative Neurons

Figure 5.3 – Self Organising Map structure for the proposed algorithm.

When using alternative feature formats such as the LPC spectrum or MFCC, a couple of the SOM parameters need to be changed. For example if the LPC spectrum is used as the feature vector the update threshold optimised value would be 2.189. While if MFCC features are used the update threshold optimised value would be 1.044 and the SOM would be re-designed to have 19 input features; i.e. the first 20 MFCC with coefficient $MFCC_0$ (frame energy) excluded. The rest of the system parameters are the same.

5.2.3 Weighted Euclidian Distance between Self Organising Map Weight Set

The standard Euclidian Distance (ED) measure determines the distance between any two vectors in multi-dimension space. The components within each vector are weighted equally as shown below:

$$D(R,V)=\sqrt{\sum_{i=1}^{64}(r_i-v_i)^2} \tag{5.2}$$

where R=r_1, r_2, r_3, …, r_{64} is the claimed user registration trained weight set for one neuron and V=v_1, v_2, v_3, …, v_{64} is the verification trained weight set for the same neuron. To overcome speaker variability, Equation 5.2 is modified to form a weighted Euclidian distance. A weighting vector can be added to the Euclidian distance calculation for each vowel as shown in Equation 5.3

$$D(R,V)=\sqrt{\sum_{i=1}^{64}\alpha_i\,(r_i-v_i)^2} \tag{5.3}$$

where α= α_1, α_2, α_3, …, α_{64} is the weighting vector set. The role of α_i is to give higher weight for DFT spectrum components with low variability for the claimed speaker and lower weight for DFT spectrum components with high variability for the claimed speaker. A suggested model for α is shown in Equation 5.4

$$\alpha_i=\frac{1}{\Delta_i} \qquad 1\le i\le 64 \tag{5.4}$$

where Δ_i is the average absolute difference of the i^{th} component over the four different registration SOM trained weight sets that belong to the claimed speaker in the registration session of the CSLU2002 database.

The final Euclidian distance score between the verification sample SOM and the claimed speaker registration sample SOM is then the averaged sum of the three Euclidian distances. Using the distance values obtained from the Euclidian distance a decision can be made based on the use of an individual speaker model threshold derived from the Equal Error Rate (EER) position for the claimed speaker against the other speakers in the database.

5.2.4 Results

From the CSLU2002 database the words chosen for testing were (five/two/eight). Since the verification algorithm is a vowel based algorithm, these words were chosen as they include the desired vowels. Using the samples from the Session 1, the update distance threshold values and α vector were optimised based on the type of the feature vector input of the SOM.

The testing phase is accomplished by using the claimed speaker and impostor samples from Session 2 as verification samples. Table 5.1 presents the average verification accuracy of 50 speakers within the CSLU2002 database at their respective EER threshold positions.

Table 5.1 shows clearly that the DFT spectrum and the MFCC provide almost the same verification accuracy; followed by the LPC spectrum. Taking into consideration the extra calculations required for the MFCC and LPC spectrum, the DFT spectrum represents the optimum input feature vector choice for the proposed system as it facilitates faster real-time training of the verification SOMs.

Table 5.1 – Verification accuracy.

Type of input for SOM	Verification accuracy (%)
DFT spectrum	92.47
LPC spectrum	91.79
MFCC	92.32

The evaluation results presented here also show better performance when compared with the GMM based system described in (Reynolds and Rose 1995); specifically when testing with speech of (~1 sec) duration, where GMM classifier performance decreases to 80%. A similar comparison with a traditional SOM based system (Mafra and Simoes 2004) is also favourable when taking into consideration the fact that the SOM presented there needs substantial 17.5 sec speech data for training; whilst the proposed algorithm uses only four samples of three words taken from one registration session (~4 sec).

The proposed algorithm in this work also represents a limited data condition scenario as it only uses short speech segments for training and testing. The experimental results in (Jayanna and Prasanna 2009) show that the performance of traditional speaker verification algorithms (including conventional SOM) falls significantly when limited 3 sec speech data is used for training and testing. Thus under the similar limited data condition tested here the presented algorithm shows significant improvement to those presented in (Jayanna and Prasanna 2009).

To improve the verification performance, the negative impostors' samples can be used to provide better discrimination between the speaker model and the impostors' model. In the next section, an MLP is combined with the SOM to produce a two-stage speaker verification algorithm.

5.3 Speaker Verification Using Modified Self Organising Map and Multi Layer Perceptron

The speaker verification algorithm presented in this section consists of two stages. The first stage is a frame filtering stage that uses the modified SOM, presented in Section 5.2.2, as a claimed user voice model for the three vowels considered in the experiment. The second stage then consists of three MLP networks; each of which has been trained to function as a claimed user vowel verifier. The main structure of the SOM+MLP system is shown in Figure 5.4.

Both stages of the proposed algorithm in Figure 5.4 are trained first using Session 1 samples in a training phase. The testing phase is then applied after both SOM and MLP are trained. The two phases are described as follows:

Training phase: each individual speaker in the training set has four speech samples. From each sample a single SOM is extracted as explained in the previous section producing four SOMs per speaker. The four speech samples are then filtered using the same four SOMs to select only those speech frames that lie within the empirically optimised distance threshold. The resultant speech frame sets N_1, N_2 and

N_3 represent the vowel information that can be used to train the three MLP networks.

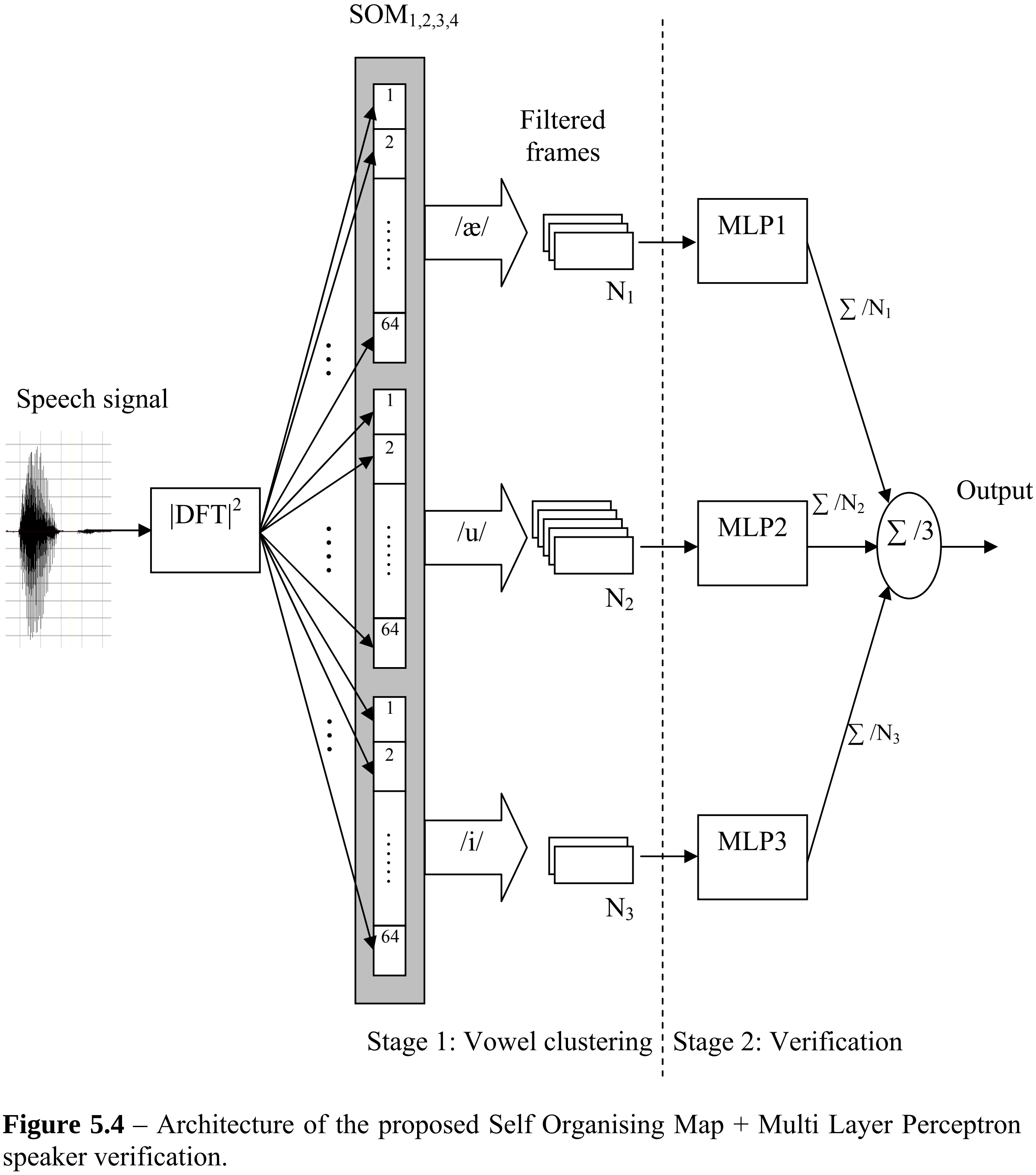

Figure 5.4 – Architecture of the proposed Self Organising Map + Multi Layer Perceptron speaker verification.

Testing phase: to test a new speech sample, the sample is passed to the four registration SOMs and any speech frame that is within the distance threshold of any of the four SOMs is passed through for testing in the verification stage. At the second stage the respective MLP networks are tested individually using the filtered frames of the test sample. Since each filtered frame represents an input pattern, the output of the MLP network is averaged over the number of filtered frames for that vowel to obtain a single output for each vowel. Finally the three averaged outputs of the three MLP networks are also averaged to achieve one output score for one test sample.

5.3.1 Multi Layer Perceptron Verifier

The second stage consists of three MLP networks. Each MLP is trained individually by using the filtered frames from the first stage. The MLP network suggested for each vowel consists of two layers, an input layer of 64 nodes, representing the DFT spectrum vector of each frame successfully filtered by any of the four registration SOMs for that vowel, and an output layer of one neuron with supervised binary output of 1 when the input vowel frame information belongs to the claimed speaker and output of 0 when the input vowel frame information belongs to an impostor. The structure of the MLP is shown in Figure 5.5.

A simple MLP network architecture is possible here because the SOM filtering stage removes noise, non-vowel data and undesired other vowels data. Each MLP network was trained using the standard back-propagation learning algorithm with a learning rate of 0.1 and a sigmoid activation function with a temperature of 1.0. To train and test the two stage speaker verification algorithm the same two sessions data from the CSLU2002 database, as used in the previous work, were used and divided as shown in the next sections.

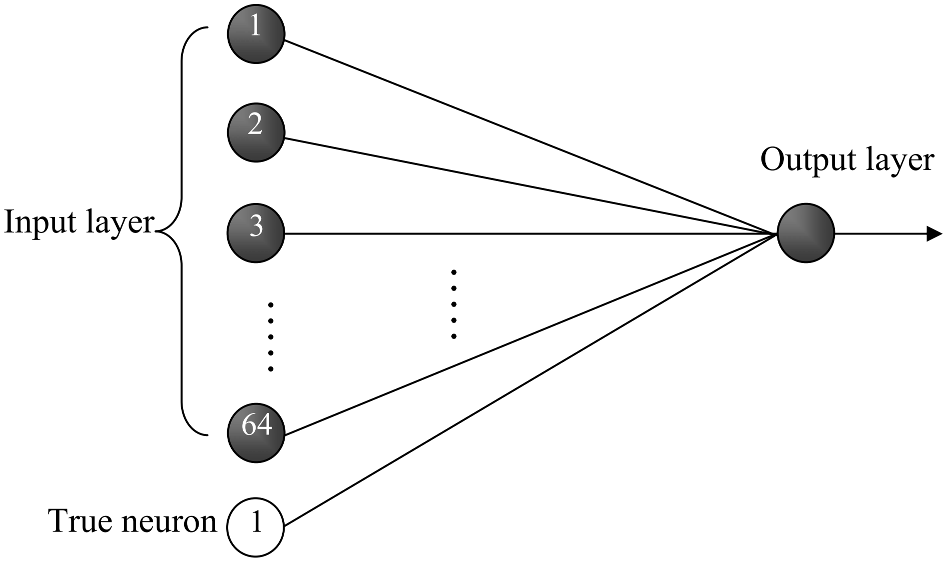

Figure 5.5 – Multi Layer Perceptron network structure.

5.3.2 Testing and Results

To train and test each MLP, two types of speech data are needed, claimed speaker speech data and impostor speech data. Each type is then divided into three parts, training, validation, and testing. Data from Session 1 are used only for training and validation while data from Session 2 are used only for testing. The first 30 speakers were used to evaluate the performance of the algorithm. The remaining speakers were kept aside to provide validation and testing data for impostors. Figure 5.6 explains how the data was divided to implement the algorithm for the first speaker. Note this data represents the filtered speech data (N1, N2 and N3). As shown in Figure 5.6 the speaker data was split to provide training data for both claimed speaker and impostors, as well as to reserve unseen data for validation in Session 1 and unseen testing data in Session 2.

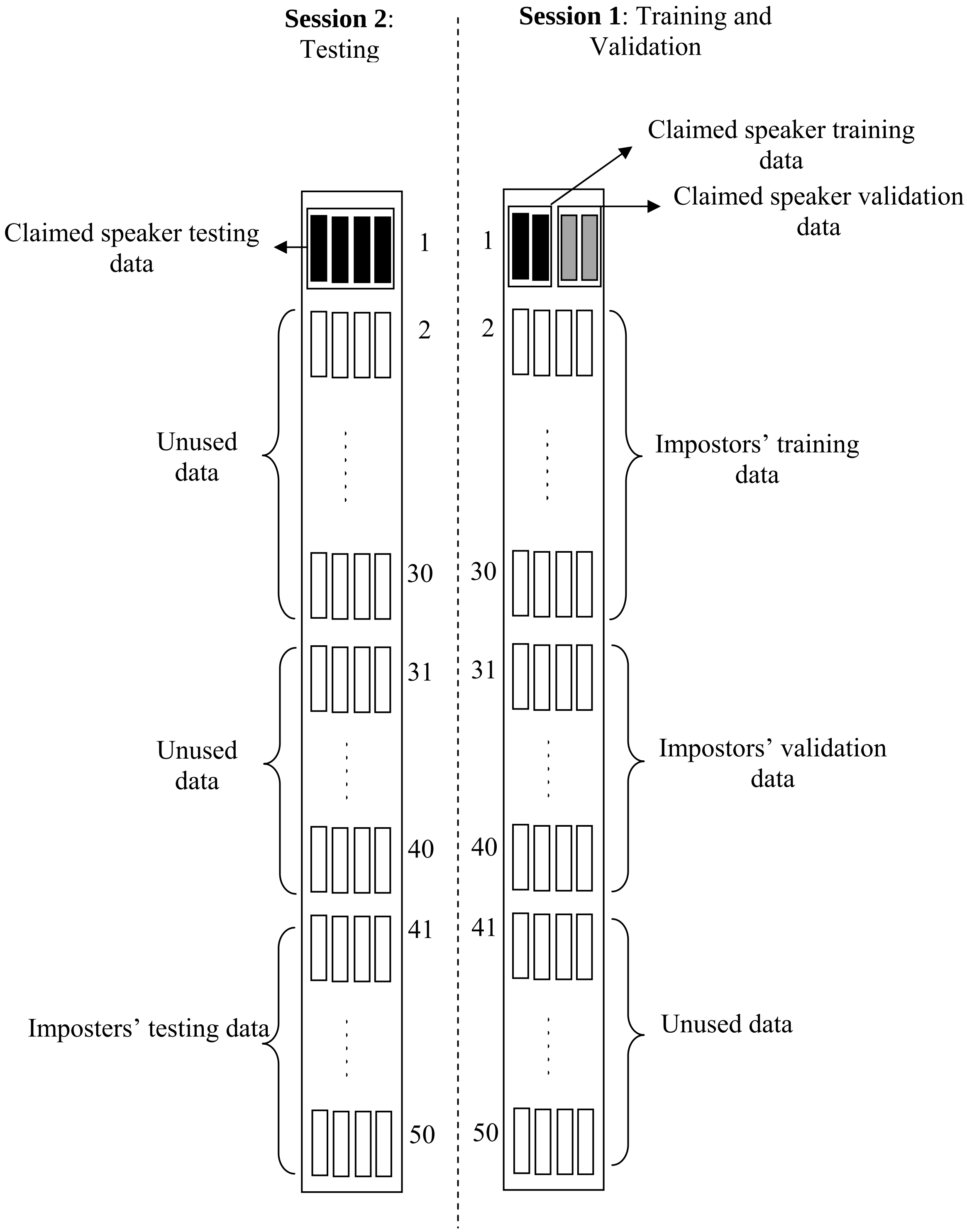

Figure 5.6 – Speech data division for the proposed algorithm.

For each individual vowel MLP verifier, the network was trained to give an output of 1 for filtered frames corresponding to the claimed speaker training data, and an output of 0 for filtered frames corresponding to the impostors training data.

At the end of each training epoch a validation error was calculated using the filtered validation data of the claimed speaker and impostors as shown in Equation 5.5. The network stops the training, if the validation error increases.

$$E_{Validation} = \frac{1}{2}\left[\frac{1}{M_1}\sum_{i}^{M_1}|T_i - A_i| + \frac{1}{M_2}\sum_{i}^{M_2}|T_i - A_i|\right] \tag{5.5}$$

M_1 and M_2 are the numbers of filtered validation data frames for the claimed speaker and impostors respectively, T_i is the target output which is equal to 1 in the first term and equal to 0 in the second term and A_i is the actual output. In Equation 5.5 the validation error is calculated individually for the claimed speaker and impostors to eliminate the effect of the unbalance between M_1 and M_2.

After training, the two stage speaker recognition system was tested using the unseen Session 2 data samples of the claimed speaker and impostors. Each frame of a test sample was presented sequentially to the trained system to produce a final output, representing the average of the three MLP averaged outputs over the whole sample, as a number between 0 and 1. Only filtered frames that are passed forward from the SOM stage are processed by the MLP stage, thus frames that were not passed forward by the first SOM stage do not contribute to the final output value. By applying speaker dependent variable thresholds to these values, the FRR and FAR can be calculated. Using the Minimum Average Error Rate (MAER) = min{(FRR+FAR)/2} the performance of the verification algorithm can be obtained as follows:

$$Performance\ (\%) = 100 - MAER \tag{5.6}$$

For direct comparison purposes, the same 30 speaker set were enrolled using the SOM+ weighted ED scoring system presented in Section 5.2.2. In addition, in order to gain an understanding of the results possible using an SOM only solution, the same data set was used to evaluate the performance of the SOM system in Section 5.2.2 with a conventional ED scoring mechanism. Figure 5.7 shows the performance of the first 30 speakers using:

1. The SOM with ED scoring based system (SOM).
2. The SOM with weighted ED scoring based system (SOM+ weighted ED).
3. The proposed SOM+MLP algorithm (SOM+MLP).

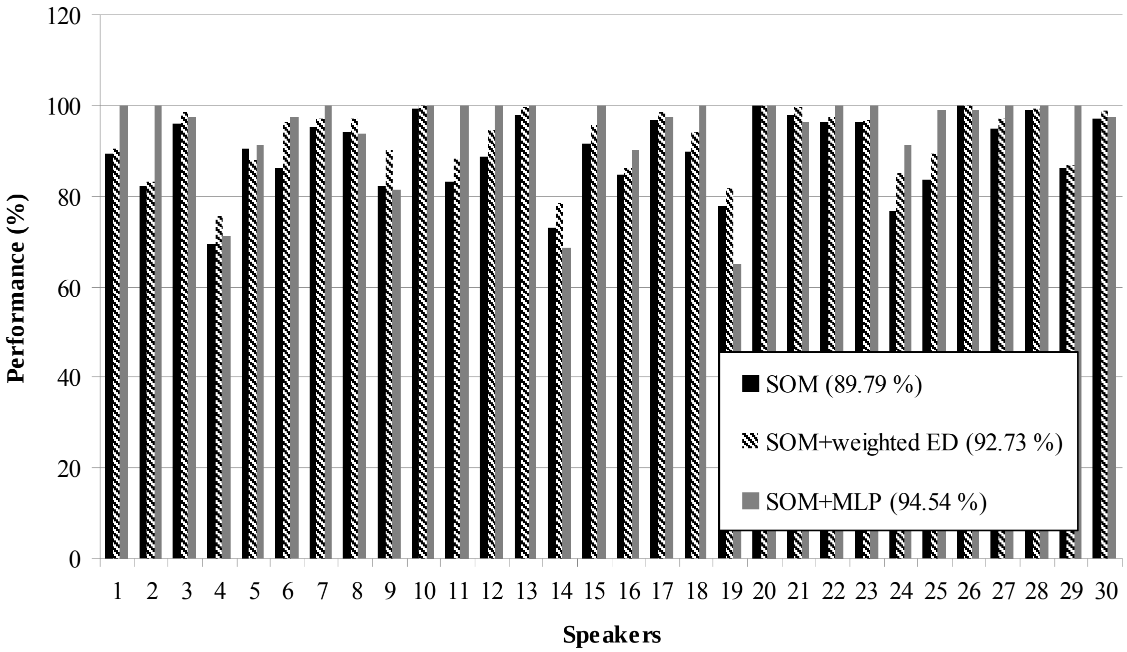

Figure 5.7 – Performance of using: SOM+ED, SOM+ weighted ED and SOM+MLP.

From Figure 5.7, it is clear that the three investigated methods have the same behaviour towards many speakers in the dataset. Upon further investigation it was found that the speakers 4, 14 and 19 with the lowest performance in the SOM+MLP curve showed high variability across the registration and verification sessions. The lowest performance occurs with speaker 19 when the SOM+MLP system was trained with two low variability samples from Session 1, i.e. the MLP networks have lost some of their robustness against speaker variability. The average performance of the three algorithms is shown in Table 5.2.

Table 5.2 – Speaker verification performance.

Method	Performance (%)
SOM+ED	89.79
SOM+ weighted ED	92.73
SOM+MLP	94.54

From Table 5.2 it is clear that the SOM+MLP system has the best average performance rate. This is particularly impressive given that the SOM+ weighted ED system saw four real-user samples during the training whereas the SOM+MLP system saw only two real-user samples during the training. In addition, as the SOM+MLP system is a more biologically plausible solution than the hybrid SOM+rule based weighted ED scoring method it can form the basis of further work investigating the use of spiking neural networks for speaker recognition.

5.4 Summary

Two speaker verification experiments have been performed. The first experiment uses a modified version of the original SOM and LVQ systems. The SOM+ weighted ED results (Table 5.1 in Section 5.2.4) show 92.47% verification on 50 speakers of the CSLU2002 speaker verification database. These results show that a seeded SOM using a threshold distance criterion to update the winner neuron obviates the need to remove the silence and other phonemes from the input speech. They also show that the DFT spectrum alone contains sufficient features to achieve a plausible level of speaker verification performance. Using the simply calculated DFT spectrum of the input speech as an input to the SOM, rather than MFCC's or LPC spectrum, as well as only clustering on three vowels considerably reduces the training time of the SOM such that a system can be trained in real-time each time the user performs a verification attempt. The average time required to train the SOM as used here was 0.26 sec using a Core 2 Duo processor of 2.4 GHz.

The second experiment presents a novel two stage speaker verification system. The first stage employs a modified SOM to filter the input speech data into frames of three vowels information. The filtered frames are related to the claimed speaker since the SOM is designed to extract only claimed speaker vowel data frames. The second stage consists of three MLP networks, these networks act as fine-grained speaker verifiers, since they are trained with pure vowels information to accept the claimed speaker data frames and reject impostor data frames. The DFT spectrum was adopted as the input feature vector. Fifty speakers from the CSLU2002 speaker recognition database were used to evaluate the algorithm. Three experiments were conducted. The first experiment used an SOM and ED to compare the SOM weight sets. The second experiment used the SOM and weighted ED as described in Section 5.2.2. The third experiment was applied using the proposed SOM and MLP system. The first experiment shows a performance of 89.7% while the second and the third experiments show performances of 92.7% and 94.54% respectively. In spite of being trained with 50% less speech data compared to the SOM+ weighted ED scoring based system, the proposed SOM+MLP algorithm gives the best average performance over the 30 enrolled speakers.

In addition, since short speech data duration is used during training and testing in this work, the experiment presented here can be considered as a limited data condition case. In a recent comparative study (Jayanna and Prasanna 2009), different speaker recognition systems were investigated under limited data conditions. The study included popular speaker recognition systems such as GMM with universal background model, Learning vector quantisation, Fuzzy vector quantisation and SOM. It was shown there that the performance of these systems decreases dramatically when limited speech data 3 sec is used for training and testing. It can be inferred from that study that any other popular speaker recognition technique, which is normally trained using substantial speech data, may suffer from similar performance degradation when trained and tested using limited speech data. Thus the proposed system presented in this section shows better limited data condition performance than all the traditional methods described in (Jayanna and Prasanna 2009).

Chapter Six

Speaker Verification Using Spiking Self Organising Map

6. Introduction

In this research the highest verification performance is not the ultimate goal. The potential target is to imitate the mechanism of the human auditory system. This is based on the evidence that babies can recognise their mothers' voices (Mehler *et al.* 1978) before they can develop speech recognition capability (Ramscar and Gitcho 2007); thereby implying that the human auditory system can provide speaker verification functionality without the need for speech recognition process. Section 6.1 presents a delayed rank order coding scheme as a suggested biologically plausible feature vector. Section 6.2 describes the theoretical background of spiking neural networks. Section 6.3 describes a proposed spiking SOM algorithm for speaker verification with evaluation and comparison to the non-spiking SOM based algorithm presented in Section 6.4. Finally Section 6.5 provides final conclusions with recommendations for future work.

6.1 Delayed Rank Order Coding

Rank order coding (Thorpe and Gautrais 1998) is a common coding technique that has been used to encode spike-based signals in spiking neural networks for speech recognition (Loiselle *et al.* 2005) and speaker authentication (Wysoski *et al.* 2007) purposes. One major disadvantage of using rank order coding is that it only takes into account the order of components as a feature vector and ignores the relative timing information among components. In this research, timing information is considered as well as the order of the components in a 'delayed' rank order coding feature vector. Taking the DFT spectrum, shown in Figure 2.8 in Chapter 2, as an example, a spike representing the frequency component with the largest amplitude will be generated with zero delay time (Δ_3) at a given onset point. A spike representing the second highest frequency component will be generated with a delay from this onset point (Δ_2). This delay is equivalent to the difference between the intensities of the two frequency maxima. Here the delay is approximated by the difference between the spectrum components since it is proportional to the number of saturated fibres phase locked to the frequency maxima. Figure 6.1 shows the delayed rank order coding derived from the DFT spectrum.

To explain how the delay rank order coding feature vector is calculated from a DFT spectrum feature vector, as shown in Figure 6.1; lets assume that for a given frame of speech the DFT Spectrum vector is [5, 12, 15, 8, 9, …, 6]. This represents the amplitudes of frequency components f_1 through to f_N.

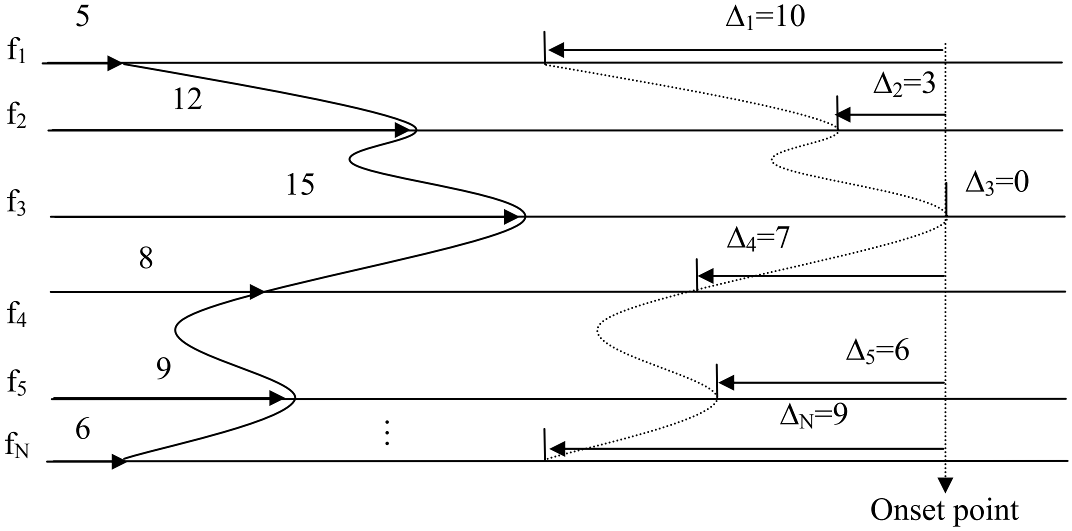

Figure 6.1 – Delayed rank order coding extracted from Discrete Fourier Transform spectrum, f_1, f_2, …, f_N are frequency positions along the basilar membrane. The envelope on the left is the DFT spectrum values while the spikes on the right forms the delayed rank order coding feature vector.

To convert the DFT spectrum vector into a delayed rank order coding vector, the largest frequency component amplitude value is represented by a zero delay, as shown in Figure 6.1 (left hand side), with the rest of the components being represented by delay differences that are proportional to the difference between their amplitudes and the largest component value the Delayed rank order coding vector is [10, 3, 0, 7, 6, …, 9].

where the components of the delayed rank order coding vector are the delay values of frequency components f_1 to f_N as shown in the right hand side of Figure 6.1.

One interesting aspect of the delayed rank order coding scheme is that it provides vector normalisation over the dynamic range of components values. In the standard DFT example the dynamic range is 15-5=10. The delayed rank order coding vector also has a dynamic range of 10, but this is normalised between 0 and Δ_{max} .This process compensates for any DC offset change in the DFT spectrum (i.e. no volume normalisation pre-processing is required).

6.2 Spiking Neural Networks

An ideal spiking neuron is similar in structure to other types of neurons, with three main parts, dendrites, soma and axon. The dendrites act as an input stage transferring the received inputs into the soma. A non-linear process is then applied inside the soma to produce an output when the summation of the inputs exceeds a threshold. The axon transfers the resultant output to other neurons. The main difference between a spiking neuron and other types of neurons is that it is spike-based operating neuron, where inputs and outputs are spikes rather than numeric values. In reality, in one cubic millimetre, there are approximately 10^4 cortical neurons with connection lengths of several kilometres (Gerstner and Kistler 2002).

6.3 Spiking Self Organising Map

Spiking neural networks for speaker recognition have been investigated in the literature using a variety of structures (George *et al.* 2003; Bing *et al.* 2006; Timoszczuk and Cabral 2007; Wysoski *et al.* 2007; Wysoski *et al.* 2010). A Spiking Self Organising Map (SSOM) is suggested in this thesis as a speaker verification platform. The one-dimensional SSOM contains three spiking neurons, each working under an integrate-and-fire mechanism. The SSOM has an input of 64 inputs that represents the delayed rank order coding of the DFT spectrum of one speech frame, as explained in Section 6.1.

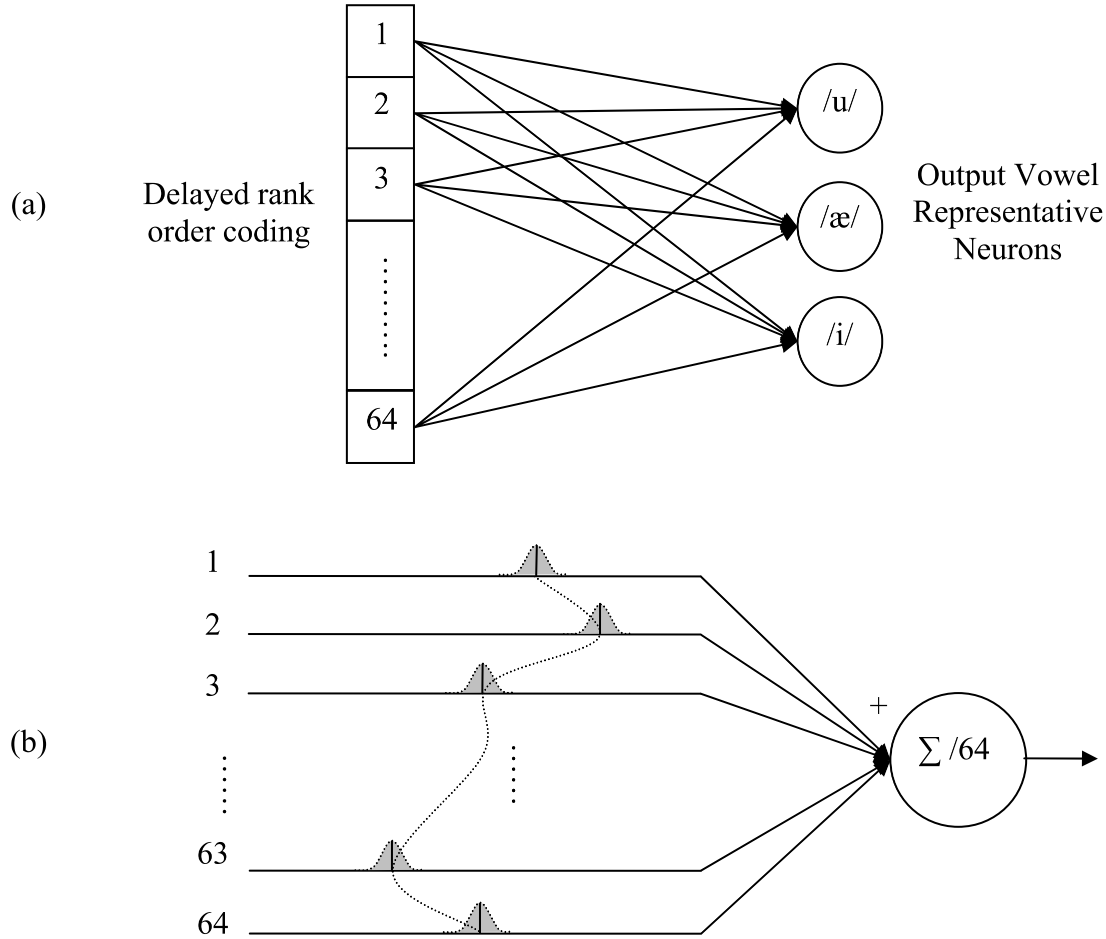
(a)
Delayed rank order coding
1
2
3
64
/u/
/æ/
/i/
Output Vowel Representative Neurons
(b)
1
2
3
63
64
+
∑ /64

Figure 6.2 – Proposed Spiking Self Organising Map algorithm a) Proposed Spiking Self Organising Map structure b) Spiking neuron showing a fully synchronised input vector.

The structure of the SSOM (shown in Figure 6.2a) is the same as the modified SOM that presented in Section 5.2.2 except that it uses the delayed rank order coding input vectors rather than the raw DFT spectrum vector. The choice of a 64 DFT spectrum component vector, rather than the 3600 component spike vector produced by the hair cells connected to the basilar membrane, is designed to approximate the frequency resolution down-scaling that is believed to occur as the signals move up through the various layers of the human auditory system (Møller 2006). Using a 64 DFT component input vector also allows a direct comparison to be made between the results produced by the SSOM experiments presented here and those obtained in Chapter 5.

Each spiking neuron in Figure 6.2a is initially seeded with a target vector. This vector is the delayed rank order coding of a selected speech frame from each of the three vowels (/u/, /æ/ and /i/). As before, the three vowels are contained in the words (two, five and eight) of the CSLU2002 database. The position of the target vector for each neuron is chosen after locating the vowel region within each word in the enrolment speech sample using the pre-processing linear correlation technique presented in Chapter 4. Once seeded, subsequent input vectors are then compared to the seed vector. When the spike timings of an input vector are fully synchronised with the spike timing of the target (seed) vector, each input synapse connected to the spiking neuron will respond with a maximum value of 1, resulting in an output of 1 as shown in Figure 6.2b. If a spike is off-synchronised with respect to its corresponding spike timing in the target vector, the response of the synapse will decrease according to a Gaussian distribution function (shaded area in Figure 6.2b); as in a biological auditory nerve response (Greenberg *et al.* 2004) and (Panchev and Wermter 2004). In other words, the more off-synchronised the spikes in the input vector are, the lower will be the output response produced by the spiking neuron. Based on this configuration, the output response of the spiking neuron to an input vector ranges between 1 (fully synchronised with target vector) and 0 (fully off-synchronised with target vector).

In the training phase, one SSOM is created for each enrolment sample. From that sample, the delayed rank order feature vector is calculated for each frame as explained previously and three target vectors (one for each vowel) are selected and used to seed the three output neurons of the SSOM. All other enrolment sample frames are then presented repeatedly to the SSOM in order to optimise the target delay vectors. During this training of the SSOM, the winner spiking neuron is updated only when the response at its output exceeds a specific threshold of 0.7, this threshold being optimised empirically to ensure correct clustering and include only pure vowel information. The threshold criterion also prevents the SSOM from clustering silence and non-vowel information. The update formula of the spiking neuron is similar to the standard SOM formula with weights replaced by delays as follows:

$$\Delta_{new} = \Delta_{old} + \alpha \left(\Delta_{old} - \Delta_{input}\right) \tag{6.1}$$

where Δ_{old} is the old delay value of the synapse, Δ_{new} is the new modified delay value, Δ_{input} is the delayed rank order component of the current input vector corresponding to the same synapse and α is the learning rate. The SSOM training parameters are similar to the modified SOM presented in Chapter 5 with 100 epochs and a learning rate of 0.1 which decreases linearly to zero over time. At the end of the training phase, each spiking neuron output of the trained SSOM represents a typical delayed rank order vowel model for the claimed speaker.

In the testing phase, the SSOM is used to verify a test sample. For each speech frame input, the spike timing input vector is compared to the spike timing of the target vector. By summing the synchronised spikes, a spike is generated at the output of the neuron only if the normalised summation exceeds an empirically optimised threshold value of 0.5. Each spiking neuron in the SSOM is more active when its related vowel information appears at the input. This is expected to be maximised when the test sample belongs to the claimed speaker. A lower activity output would be produced when the test sample belongs to an impostor.

To calculate a score for each vowel, the total number of spikes generated at the output over the number of frames presented, is then normalised over the duration of the vowel region within each word containing that vowel as follows:

$$S_i = \frac{number\ of\ spikes\ generated\ at\ neuron\ \ i}{number\ of\ \ frames\ within\ ith\ vowel\ region} \tag{6.2}$$

and:

$$S = (S_1 + S_2 + S_3)/3 \tag{6.3}$$

where S_i is the score of the i^{th} vowel and S is the final verification score averaged over the three individual output neuron scores.

6.4 Results

The proposed algorithm was evaluated using speech samples of 50 speakers (27 females and 23 males), where the speakers were arbitrarily selected from the 91 speaker CSLU2002 speaker verification database. The speech samples in this database were recorded over digital telephone lines with a sampling frequency of 8 kHz to produce 8-bit u-law files, which are then encoded into 8 kHz 16-bit wave format file. Two recording sessions samples are used for evaluation purposes, each session containing four samples for each speaker. Each speech sample contains the words (two, five & eight).

Each sample in Session 1 can be used as an enrolment sample to create one SSOM. The network is then tested against Session 2 samples for both the claimed speaker and impostors (the remaining 49 speakers in the dataset). By applying speaker dependent variable thresholds to the different scores values, the FRR and the FAR can be calculated for each speaker. The verification performance is obtained as follows:

$$Performance(\%) = 100 - MAER \tag{6.4}$$

where Minimum Average Error Rate (MAER) =min{(FRR+FAR)/2}. Figure 6.3 shows the results for the proposed delayed rank order based SSOM together with results from the DFT spectrum based SOM presented in Chapter 5.

It is clear from Figure 6.3 that both algorithms have similar performance, with the DFT spectrum-based SOM outperforming on average. However, the delayed rank order coding SSOM improved the results for speaker 4 significantly. Speaker 4 is noticed to have low performance when previous algorithms have been used, due to high speaker variability over sessions. The average performance of the 50 speakers for both algorithms is shown in Table 6.1.

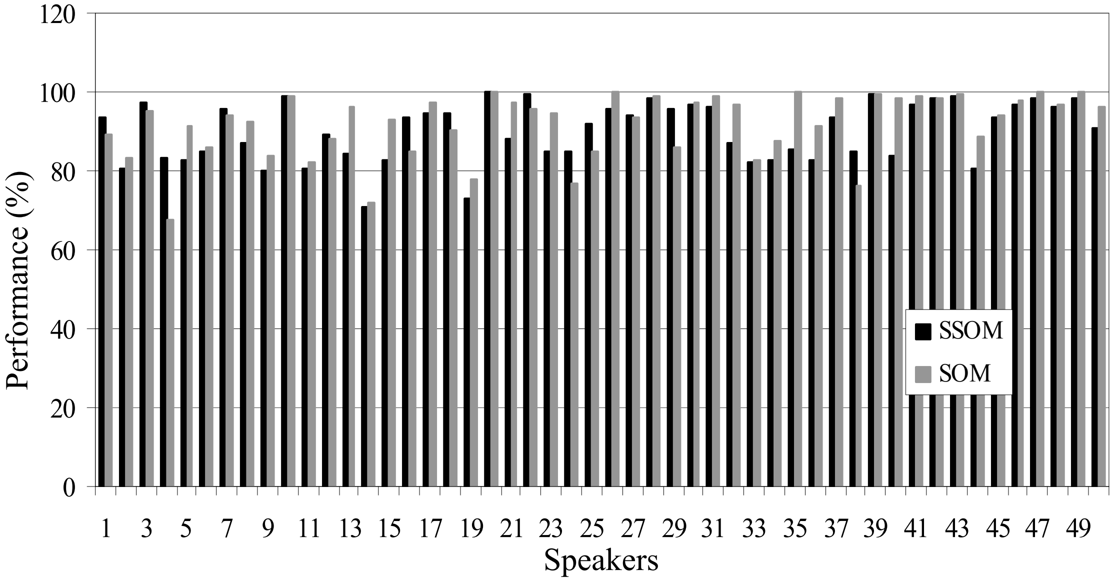

Figure 6.3 – Performance of 50 speakers of CSLU2002 database.

From Table 6.1 it is clear that the performance of the proposed algorithm is comparable to the DFT spectrum-based non-spiking SOM algorithm. No comparison has been made with the SOM+ weighted ED algorithm presented in Chapter 5 since this uses three additional positive samples during the training process in order to overcome user variability. The proposed SSOM here is trained using only **one** positive sample containing one example each of three vowels for each user.

Table 6.1 – Average speaker verification performance.

Method	Feature vector type	Performance (%)
SSOM	Delayed rank order coding	90.1
SOM+ED	DFT spectrum	91.7

6.5 Summary

In this chapter a spiking neural network for speaker verification is proposed that is inspired by the physiology of the hearing in the human auditory system. This

SSOM system uses delayed rank order coding as the input feature vector. During the training phase the network updates the winner neuron only if it is active beyond a certain level.

The proposed algorithm was evaluated using speech samples of 50 speakers from the CSLU2002 speaker verification database over two recording sessions. The algorithm shows an average speaker verification performance of 90.1%. In a direct comparison, the proposed biologically plausible SSOM is seen to be comparable to the non-spiking based SOM algorithm results 91.7% presented in Chapter 5 using the same speech dataset. Due to the short duration of speech data used in training and testing stages (~ 3 sec) of this experiment, the environment can be classified as a very limited data condition scenario. Consequently, the proposed biologically inspired algorithm, even with a slightly lower verification performance, is still preferable to traditional speaker recognition systems, where performance significantly decreases under such environments (Jayanna and Prasanna 2009).

Chapter Seven

Conclusions and Future Work

7. Introduction

The research in this thesis investigated biologically inspired plausible solutions for the speaker verification problem. Most current speaker verification platforms use a speech recognition engine as a pre-processing front-end. However, biologically, this is not the case, as babies are known to develop their speaker verification system (Mehler *et al.* 1978) before they develop their speech recognition capabilities (Ramscar and Gitcho 2007). This leads to the conclusion that speaker verification is possible without the need of a complete speech recognition system, and that, biologically, speaker verification must be based on a lower morphological level of features than sentences and/or words (i.e. phonemes). Based on that evidence, the research objective was to develop a biologically inspired speaker verification algorithm that is based purely on low level feature extraction processes.

Different experiments have been conducted to achieve this target and, in the following sections, the key points highlighted by this work are discussed.

7.1 The Choice of Three Vowels

Three vowels were considered in this research. The choice of these vowels was based on the facts:

- Vowels in general are more intelligible than phonemes (Rabiner and Schafer 2010).
- Vowels contain more information about the speaker identity than other phonemes (Han-Sheng and Mammone 1995a).
- The three vowels adopted in this research are well separated among the ten vowels of English language. This has direct improvement on the clustering efficiency of the modified SOM.

As a result of choosing only three vowels, great saving is accomplished in the training speech data. This is a preferable option for many commercial speaker verification application, where a limited speech data scenario occurs in training and/or testing (Angkititrakul and Hansen 2007). The two experiments using the SOM+ED and SSOM (mentioned in Chapter 6) represent an extreme limited data condition, where only one speech sample of the three vowels is used for registration and another speech sample of three vowels is used for verification.

7.2 The Choice of the Discrete Fourier Transform As Feature Vector

All natural human speech signals are generated as compositions of modulated frequencies produced by resonances of the vocal tract. Therefore, they are fully describable in the frequency domain; making the DFT spectrum an excellent feature vector candidate. This choice of the DFT spectrum is in agreement with the basilar membrane function, which is known to act as a spectral analyser (Greenberg *et al.* 2004; Møller 2006; Rabiner and Schafer 2010).

Although other popular feature formats such as MFCC and LPCC can be derived from the DFT spectrum, they can occlude identity information (Wysoski *et al.* 2010). The results presented in this research (Table 5.1 in Section 5.2.4 and repeated below as Table 7.1) indicate that the DFT spectrum outperforms both the LPC spectrum and MFCC as input feature vector.

Table 7.1 – Verification accuracy.

Type of input for SOM	Verification accuracy (%)
DFT spectrum	92.47
LPC spectrum	91.79
MFCC	92.32

7.3 The Choice of Self Organising Map

An SOM is adopted in this research due to its topological nature which resembles the tonotopic nature of the human auditory system (Young 2008). The SOM also has the ability to extract low morphological levels of the speech signal such as phonemes (Kohonen 1990). This is in agreement with the known to ability of babies to perform speaker verification before they understand speech (Mehler *et al.* 1978).

The modified SOM presented in Section 5.2.2 of this thesis has a dual use. Basically, its main function is to cluster the input data into three output neurons that refer to the three vowels. When trained with one speech sample which belongs to a claimed speaker, the SOM uniquely describes the claimed speaker's vowels characteristics. Therefore, the SOM itself act as speaker verification platform, using only positive samples for the verification process. The SOM in Section 5.3 functions as a coarse speaker verifier that filters frames of data based on the closeness of their input features to the claimed speaker, before feeding them to the MLP fine verifiers. By seeding its outputs, and updating the winner neuron only when the input is within a distance threshold, the SOM is efficiently trained using only the relative feature inputs to the three output patterns. A comparison of the results for the above algorithms is shown in Table 7.2 (repeated from Table 5.2 in Section 5.3.2).

Table 7.2 – Speaker verification performance.

Method	Performance (%)
SOM+ED	89.79
SOM+ weighted ED	92.73
SOM+MLP	94.54

7.4 Spike-Based Features with Spiking Self Organising Map

As explained in Chapter 2, the cochlea converts the audio speech signal from mechanical movement (captured by the tympanic membrane) into an electrical spike discharge. The analysis presented in Section 6.1 demonstrates that the delayed rank order coding feature vector can describe the signal as it travels through the low level of the human auditory system. Although, the delayed rank order coding does not consider phase information, it can still provide an accurate description of the DFT spectrum envelope. A great advantage of using the delayed rank order feature vector is that it is automatically normalised over the dynamic range of the frequency components with in the vector.

It is argued in Chapter 6 that a spiking SOM with the delayed rank order coding as input feature vector, represents a complete biologically plausible spike-based tonotopic solution. The SSOM operates in similar manner to the SOM presented in Section 5.2.2, except that the distance to the target vector is expressed in term of synchronisation of spike train onset times as explained in Figure 5.8b. Table 7.3 (repeat of Table 6.1 Section 6.4) shows that the SSOM has a comparable verification performance to that of the SOM+ED.

Table 7.3 – Average speaker verification performance.

Method	Feature vector type	Performance (%)
SSOM	Delayed rank order coding	90.1
SOM+ED	DFT spectrum	91.7

7.5 Future Work

The results presented in this thesis provide an encouraging baseline for further exploration of biologically plausible speech processing systems. This can be accomplished by following different research trends such as:

7.5.1 Spiking Self Organising Map with Spiking Multi Layer Perceptron

Similar to the SOM in Section 5.3, the SSOM can be combined with spiking MLP neural networks. Spiking MLP networks will allow the use of negative impostors' samples in the training phase. This is expected to improve the verification accuracy in a similar manner to the experiment in Section 5.3. However, implementing a spiking MLP network is not a straight forward task, as a training algorithm is needed which is more sophisticated than the traditional back-propagation. Spiking neural network training algorithms can be found in (Gerstner and Kistler 2002) and (Bohte 2003).

7.5.2 Investigating Other Spike-Based Features

Rank order coding is only one of several spike based feature formats. Other biologically inspired, spike-based features such as Phase, Correlations and synchrony, Stimulus reconstruction and reverse correlation (Gerstner and Kistler 2002), can also be investigated as a spike-based speaker verification platform. However, if spike-based features are used, a spiking neural network classifier is required to produce a feasible system.

7.5.3 Inclusion of Temporal Speech Information

Temporal embedded data within the speech signal also contains useful behavioural characteristics of the speaker such as: accent, rhythm, intonation style, and pronunciation pattern. A good start to investigate the inclusion of temporal data, would be to explore a non-linear movement model of the basilar membrane (Møller

2006). When a temporal pattern is considered, special types of neural networks, that work more efficiently with sequential pattern classification, are recommended for the task: i.e. recurrent neural networks (Briciu 2010) and liquid state machine (Uysal *et al.* 2008).

7.5.4 Further Investigation of the Human Auditory System

Different aspects of the human auditory system are also worth further exploration. This may include investigating the effects of increasing the number of input vector components up to the 3600 component (i.e. full scale of inner hair cells in the basilar membrane) as well as including non-linear input signal processing derived from a more detailed model of the basilar membrane and the non-linear temporal encoding of spikes (Møller 2006). Another potential research area would be to study the dynamic role of outer hair cells in sound normalisation.

References

Alarifi, A., I. Alkurtass and A. M. S. Al-Salman (2011). Arabic Text-Dependent Speaker Verification for Mobile Devices Using Artificial Neural Networks. Machine Learning and Applications and Workshops (ICMLA), 2011 10th International Conference on.

Angkititrakul, P. and J. H. L. Hansen (2007). "Discriminative In-Set/Out-of-Set Speaker Recognition." Audio, Speech, and Language Processing, IEEE Transactions on **15**(2): 498-508.

Badran, E. F. M. F. and H. Selim (2000). Speaker recognition using artificial neural networks based on vowel phonemes. Signal Processing Proceedings, 2000. WCCC-ICSP 2000. 5th International Conference on.

Bing, L., W. M. Yamada and T. W. Berger (2006). Nonlinear Dynamic Neural Network for Text-Independent Speaker Identification using Information Theoretic Learning Technology. Engineering in Medicine and Biology Society, 2006. EMBS '06. 28th Annual International Conference of the IEEE.

Bohte, S. M. (2003). SPIKING NEURAL NETWORKS, Universiteit Leiden. **Ph.D. thesis**.

Briciu, P. M. (2010). Speaker identification using Partially Connected Locally Recurrent Probabilistic Neural Networks. Communications (COMM), 2010 8th International Conference on.

Campbell, W. M., J. P. Campbell, D. A. Reynolds, E. Singer and P. A. Torres-Carrasquillo (2006). "Support vector machines for speaker and language recognition." Computer Speech & Language **20**(2-3): 210-229.

Daqrouq, K. (2011). "Wavelet entropy and neural network for text-independent speaker identification." Engineering Applications of Artificial Intelligence **24**(5): 796-802.

Davis, S. and P. Mermelstein (1980). "Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences." Acoustics, Speech and Signal Processing, IEEE Transactions on **28**(4): 357-366.

Delacretaz, D. P. and J. Hennebert (1998). Text-prompted speaker verification experiments with phoneme specific MLPs. Acoustics, Speech and Signal Processing, 1998. Proceedings of the 1998 IEEE International Conference on.

Dong, E., G. Liu, Y. Zhou and Y. Cai (2002). Voice activity detection based on short-time energy and noise spectrum adaptation. 6th International Conference on Signal Processing.

Fakotakis, N. and J. Sirigos (1996). A high performance text independent speaker recognition system based on vowel spotting and neural nets. Acoustics, Speech, and Signal Processing, 1996. ICASSP-96. Conference Proceedings., 1996 IEEE International Conference on.

Farrell, K. R., R. J. Mammone and K. T. Assaleh (1994). "Speaker recognition using neural networks and conventional classifiers." Speech and Audio Processing, IEEE Transactions on **2**(1): 194-205.

George, S., A. Dibazar, V. Desai and T. W. Berger (2003). Using dynamic synapse based neural networks with wavelet preprocessing for speech applications. Neural Networks, 2003. Proceedings of the International Joint Conference on.

George, S., A. Dibazar, J. S. Liaw and T. W. Berger (2001). Speaker recognition using dynamic synapse based neural networks with wavelet preprocessing. Neural Networks, 2001. Proceedings. IJCNN '01. International Joint Conference on.

Gerstner, W. and W. Kistler (2002). Spiking Neuron Models, Cambridge University Press.

Greenberg, S., W. A. Ainsworth, A. N. Popper, R. R. Fay, A. Palmer and S. Shamma (2004). Physiological Representations of Speech: Speech Processing in the Auditory System, Springer New York. **18:** 163-230.

GuoBin, O., L. Xin, Y. XiaoCao, J. HongBin and Y. L. Murphey (2005). Speaker identification using speech and lip features. Neural Networks, 2005. IJCNN '05. Proceedings. 2005 IEEE International Joint Conference on.

Hadjitodorov, S., B. Boyanov and N. Dalakchieva (1997). "A two-level classifier for text-independent speaker identification." Speech Communication **21**(3): 209-217.

Haigh, J. A. and J. S. Mason (1993). Robust voice activity detection using cepstral features. IEEE Region 10 Conference on Computer, Communication, Control and Power Engineering.

Han-Sheng, L. and R. J. Mammone (1995a). Speaker verification using phoneme-based neural tree networks and phonetic weighting scoring method. Neural Networks for Signal Processing [1995] V. Proceedings of the 1995 IEEE Workshop.

Han-Sheng, L. and R. J. Mammone (1995b). A subword neural tree network approach to text-dependent speaker verification. International Conference on Acoustics, Speech, and Signal Processing ICASSP.

Hmich, A., A. Badri and A. Sahel (2011). Automatic speaker identification by using the neural network. Multimedia Computing and Systems (ICMCS), 2011 International Conference on.

Homayounpour, M. M. and G. Chollet (1995). Neural net approaches to speaker verification: comparison with second order statistic measures. Acoustics, Speech, and Signal Processing, 1995. ICASSP-95., 1995 International Conference on.

Ibrahim, Q. and N. Abdulghani (2012). "Security enhancement of voice over Internet protocol using speaker recognition technique." Communications, IET **6**(6): 604-612.

Ig-Tae, U., R. Jong-Hei and K. Moon-Hyun (2000). Comparison of clustering methods for MLP-based speaker verification. Pattern Recognition, 2000. Proceedings. 15th International Conference on.

Inal, M. and Y. S. Fatihoglu (2002). Self organizing map and associative memory model hybrid classifier for speaker recognition. Neural Network Applications in Electrical Engineering, 2002. NEUREL '02. 2002 6th Seminar on.

Jawarkar, N. P., R. S. Holambe and T. K. Basu (2011). Use of fuzzy min-max neural network for speaker identification. Recent Trends in Information Technology (ICRTIT), 2011 International Conference on.

Jayanna, H. S. and S. R. M. Prasanna (2009). "An experimental comparison of modelling techniques for speaker recognition under limited data condition." Sadhana-Academy Proceedings in Engineering Sciences **34**(5): 717-728.

Jothilakshmi, S., V. Ramalingam and S. Palanivel (2009). "Speaker diarization using autoassociative neural networks." Engineering Applications of Artificial Intelligence **22**(4-5): 667-675.

Kasuriya, S., C. Wutiwiwatchai, V. Achariyakulporn and C. Tanprasert (2001). "Comparative Study of Continuous Hidden Markov Models (CHMM) and Artificial Neural Network (ANN) on Speaker Identification System." International Journal of Uncertainty, Fuzziness & Knowledge-Based Systems **9**(6): 673.

Ke, C. and A. Salman (2011). "Learning Speaker-Specific Characteristics With a Deep Neural Architecture." Neural Networks, IEEE Transactions on **22**(11): 1744-1756.

Kinnunen, T., T. Kilpeläinen and P. Fränti (2000). Comparison of Clustering Algorithms in Speaker Identification. International Conference Signal Processing and Communications (SPC 2000), Marbella, Spain, Proceeding of IASTED

Kinnunen, T. and H. Z. Li (2010). "An overview of text-independent speaker recognition: From features to supervectors." Speech Communication **52**(1): 12-40.

Kishore, S. P. and B. Yegnanarayana (2000). Speaker verification: minimizing the channel effects using autoassociative neural network models. Acoustics, Speech, and Signal Processing, 2000. ICASSP '00. Proceedings. 2000 IEEE International Conference on.

Kishore, S. P., B. Yegnanarayana and S. V. Gangashetty (2001). Online text-independent speaker verification system using autoassociative neural network models. Neural Networks, 2001. Proceedings. IJCNN '01. International Joint Conference on.

Kodukula, S. R. M., S. R. Mahadeva Prasanna and B. Yegnanarayana (2005). Neural network models for extracting complementary speaker-specific information from residual phase. Intelligent Sensing and Information Processing, 2005. Proceedings of 2005 International Conference on.

Kohonen, T. (1990). "The self-organizing map." Proceedings of the IEEE **78**(9): 1464-1480.

Kusumoputro, B., A. Triyanto, M. I. Fanany and W. Jatmiko (2001). Speaker identification in noisy environment using bispectrum analysis and probabilistic neural network. Computational Intelligence and Multimedia Applications, 2001. ICCIMA 2001. Proceedings. Fourth International Conference on.

Lacerda, M. A., R. C. Guido, L. M. de Souza, P. R. F. Zulato, J. Ribeiro and S. H. Chen (2010). "A WAVELET-BASED SPEAKER VERIFICATION ALGORITHM." International Journal of Wavelets Multiresolution and Information Processing **8**(6): 905-912.

Lapidot, I., H. Guterman and A. Cohen (2002). "Unsupervised speaker recognition based on competition between self-organizing maps." Neural Networks, IEEE Transactions on **13**(4): 877-887.

Loiselle, S., J. Rouat, D. Pressnitzer and S. Thorpe (2005). Exploration of rank order coding with spiking neural networks for speech recognition. Neural Networks, 2005. IJCNN '05. Proceedings. 2005 IEEE International Joint Conference on.

Mafra, A. T. and M. G. Simoes (2004). Text independent automatic speaker recognition using selforganizing maps. Industry Applications Conference, 2004. 39th IAS Annual Meeting. Conference Record of the 2004 IEEE.

Mehler, J., J. Bertoncini, M. Barriere and D. Jassikgerschenfeld (1978). "INFANT RECOGNITION OF MOTHERS VOICE." Perception **7**(5): 491-497.

Memon, S., M. Lech and H. Ling (2009). Using information theoretic vector quantization for inverted MFCC based speaker verification. Computer, Control and Communication, 2009. IC4 2009. 2nd International Conference on.

Molla, K. I. and K. Hirose (2004). On the effectiveness of MFCCs and their statistical distribution properties in speaker identification. Virtual Environments, Human-Computer Interfaces and Measurement Systems, 2004. (VECIMS). 2004 IEEE Symposium on.

Møller, A. R. (2006). Hearing: anatomy, physiology, and disorders of the auditory system, Academic Press.

Monte, E., J. Hernando, X. Miro and A. Adolf (1996). Text independent speaker identification on noisy environments by means of self organizing maps. Spoken Language, 1996. ICSLP 96. Proceedings., Fourth International Conference on.

Mubeen, N., A. Shahina, A. N. Khan and G. Vinoth (2012). Combining spectral features of standard and Throat Microphones for speaker identification. Recent Trends In Information Technology (ICRTIT), 2012 International Conference on.

Mueen, F., A. Ahmed, Sanaullah and A. Gaba (2002). Speaker recognition using artificial neural networks. Students Conference, ISCON '02. Proceedings. IEEE.

Oglesby, J. and J. S. Mason (1991). Radial basis function networks for speaker recognition. Acoustics, Speech, and Signal Processing, 1991. ICASSP-91., 1991 International Conference on.

Ouzounov, A. (1997). Text-independent speaker identification using a hybrid neural network and conformity approach. Neural Networks,1997., International Conference on.

Panchev, C. and S. Wermter (2004). "Spike-timing-dependent synaptic plasticity: from single spikes to spike trains." Neurocomputing **58-60**(0): 365-371.

Pandiaraj, S., D. S. Vinothini, H. N. R. Keziah, L. Gloria and K. R. S. Kumar (2011). Speaker identification using pykfec and AANN. Electronics Computer Technology (ICECT), 2011 3rd International Conference on.

Pandya, A. S. and R. B. Macy (1995). Pattern Recognition with Neural Network in C++, CRC Press, Inc.

Peterson, G. E. and H. L. Barney (1952). "Control Methods Used in a Study of the Vowels." The Journal of the Acoustical Society of America **24**(2): 175-184.

Price, R. C., J. P. Willmore, W. J. J. Roberts and K. J. Zyga (2000). Genetically optimised feedforward neural networks for speaker identification. Knowledge-Based Intelligent Engineering Systems and Allied Technologies, 2000. Proceedings. Fourth International Conference on.

Qi, L., Z. Jinsong, A. Tsai and Z. Qiru (2002). "Robust endpoint detection and energy normalization for real-time speech and speaker recognition." IEEE Transactions on Speech and Audio Processing **10**(3): 146-157.

Rabiner, L. R. and B. H. Juang (1993). Fundamentals of speech recognition, PTR Prentice Hall.

Rabiner, L. R. and R. W. Schafer (1978). Digital processing of speech signals. Englewood Cliffs, N.J., Prentice-Hall.

Rabiner, L. R. and R. W. Schafer (2010). Theory and Applications of Digital Speech Processing, Pearson.

Ramscar, M. and N. Gitcho (2007). "Developmental change and the nature of learning in childhood." Trends in Cognitive Sciences **11**(7): 274-279.

Ranjan, R., S. K. Singh, A. Shukla and R. Tiwari (2010). Text-Dependent Multilingual Speaker Identification for Indian Languages Using Artificial Neural Network. Emerging Trends in Engineering and Technology (ICETET), 2010 3rd International Conference on.

Rao, K. S., A. K. Vuppala, S. Chakrabarti and L. Dutta (2010). Robust speaker recognition on mobile devices. Signal Processing and Communications (SPCOM), 2010 International Conference on.

Reynolds, D. A. and R. C. Rose (1995). "Robust text-independent speaker identification using Gaussian mixture speaker models." Speech and Audio Processing, IEEE Transactions on **3**(1): 72-83.

Rodgers, J. L. and W. A. Nicewander (1988). "Thirteen Ways to Look at the Correlation Coefficient." The American Statistician **42**(1): 59-66.

Say Wei, F. and L. Eng Guan (2001). Speaker recognition using adaptively boosted classifier. Electrical and Electronic Technology, 2001. TENCON. Proceedings of IEEE Region 10 International Conference on.

Seddik, H., A. Rahmouni and M. Sayadi (2004a). Text independent speaker recognition using the Mel frequency cepstral coefficients and a neural network classifier. Control, Communications and Signal Processing, 2004. First International Symposium on.

Seddik, H., A. B. S. Rahmouni and M. Sayadi (2004b). Text independent speaker recognition based on the attack state formants and neural network classification. Industrial Technology, 2004. IEEE ICIT '04. 2004 IEEE International Conference on.

Shen, Y. and L. Chen (2011). Performance comparison of new endpoint detection method in noise environments. International Conference on Electric Information and Control Engineering (ICEICE).

Sri Rama Murty, K., S. R. Mahadeva Prasanna and B. Yegnanarayana (2004). Speaker-specific information from residual phase. Signal Processing and Communications, 2004. SPCOM '04. 2004 International Conference on.

Sun, F., B. Li and H. Chi (1991). Some key factors in speaker recognition using neural networks approach. Neural Networks, 1991. 1991 IEEE International Joint Conference on.

Ta-Hsin, L. and J. D. Gibson (1996). Time-correlation analysis of nonstationary signals with application to speech processing. Proceedings of the IEEE-SP International Symposium on Time-Frequency and Time-Scale Analysis.

Tan, J. D. and H. N. Ting (2011). Malay speaker identification using Neural Networks. Information Science and Technology (ICIST), 2011 International Conference on.

Thorpe, S. and J. Gautrais (1998). Rank order coding. New York, Plenum Press Div Plenum Publishing Corp.

Timoszczuk, A. P. and E. F. Cabral (2007). Speaker recognition using pulse coupled neural networks. 2007 Ieee International Joint Conference on Neural Networks, Vols 1-6. New York, IEEE**:** 1965-1969.

Uysal, I., H. Sathyendra and J. G. Harris (2008). "Towards spike-based speech processing: A biologically plausible approach to simple acoustic classification." International Journal of Applied Mathematics and Computer Science **18**(2): 129-137.

Voitovetsky, I., H. Guterman and A. Cohen (1997). Unsupervised speaker classification using self-organizing maps (SOM). Neural Networks for Signal Processing [1997] VII. Proceedings of the 1997 IEEE Workshop.

Wu, J. D. and Y. J. Tsai (2011). "Speaker identification system using empirical mode decomposition and an artificial neural network." Expert Systems with Applications **38**(5): 6112-6117.

Wysoski, S. G., L. Benuskova and N. Kasabov (2007). Text-independent speaker authentication with spiking neural networks. Artificial Neural Networks - ICANN 2007, Pt 2, Proceedings. J. MarquesDeSa, L. A. Alexandre, W. Duch and D. Mandic. **4669:** 758-767.

Wysoski, S. G., L. Benuskova and N. Kasabov (2010). "Evolving spiking neural networks for audiovisual information processing." Neural Networks **23**(7): 819-835.

Yan, Z. and G. Yunian (2010). Human Speaker Recognition Based on the Integration of Genetic Algorithm and RBF Network. Intelligent Human-Machine Systems and Cybernetics (IHMSC), 2010 2nd International Conference on.

Ye, F. and Z. Yabin (2009). PNN-based algorithm for the recognition of speakers. Electronic Measurement & Instruments, 2009. ICEMI '09. 9th International Conference on.

Yegnanarayana, B., S. R. M. Prasanna, J. M. Zachariah and C. S. Gupta (2005). "Combining evidence from source, suprasegmental and spectral features for a fixed-text speaker verification system." Speech and Audio Processing, IEEE Transactions on **13**(4): 575-582.

Yegnanarayana, B., K. Sharat Reddy and S. P. Kishore (2001). Source and system features for speaker recognition using AANN models. Acoustics, Speech, and Signal Processing, 2001. Proceedings. (ICASSP '01). 2001 IEEE International Conference on.

Young, E. D. (2008). "Neural representation of spectral and temporal information in speech." Philosophical Transactions of the Royal Society B: Biological Sciences **363**(1493): 923-945.

Zhang, S., Y. Guo and B. Wang (2009). Auto-Correlation Property of Speech and its Application in Voice Activity Detection. First International Workshop on Education Technology and Computer Science, 2009. ETCS '09.

Appendix A

Comparison between Linear Correlation and Energy Frame Analysis Pre-Processing For Speaker Verification

To evaluate the application level performance of the proposed LCC pre-processing technique presented in Chapter 4, the technique was applied as part of the pre-processing stage of the SOM+ weighted ED speaker verification algorithm presented in Chapter 5. The correlation envelope proposed in this thesis was used to replace the EFA technique which had previously been used to locate the words boundaries. After computing the DFT spectrum for a frame size of 16 msec with an overlap step of 4 msec, the CCE in Equation 4.6 was then calculated using a window size of D=5 (which represents a time interval of 32 msec).

A global correlation speech activity threshold of 91% was set to detect the boundaries of the vowels in the 240 speech waveforms that represent 30 speakers uttering the phrase five/eight/two four times over two sessions. These speech samples were taken from the CSLU2002 speaker verification database. Figure A.1 shows the performance obtained in terms of verification performance (100-minimum average error rate %), where the average error rate is the average of the false reject rate and the false accept rate. For each speaker, the false reject rate was calculated over four real user samples, whilst the false accept rate was calculated over 116 impostor samples (i.e. 29 × 4 other speakers samples).

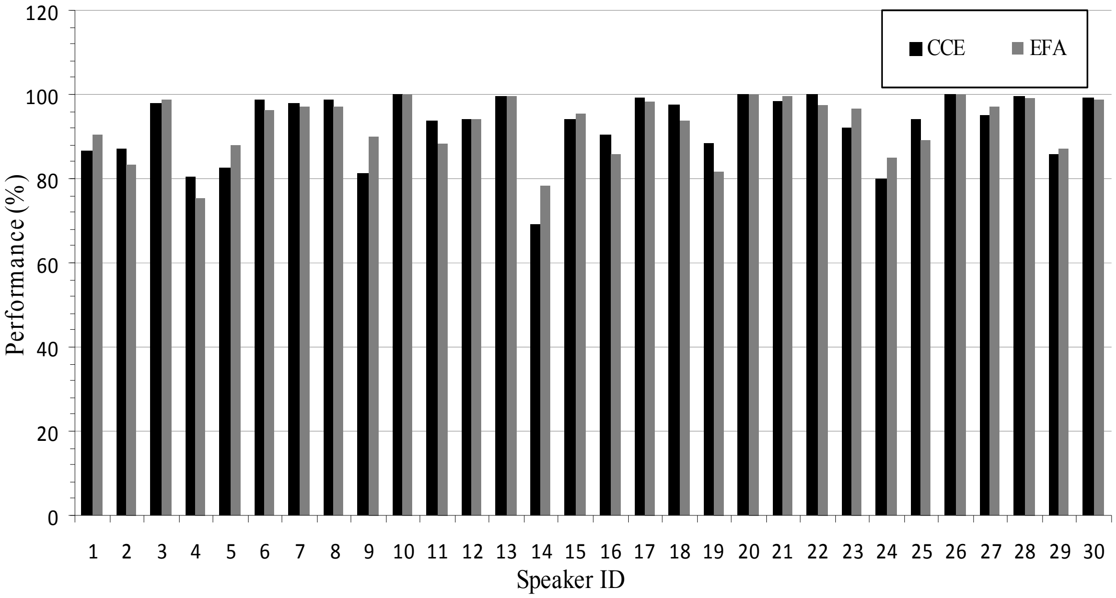

Figure A.1 – Performance of SOM+ weighted ED speaker verification algorithm described in Chapter 5 using Energy Frame Analysis and Correlation Coefficient Envelope.

In general, Figure A.1 shows almost similar verification performance when using EFA or CCE to locate words boundaries. The average performance over the 30 speakers is 92.73% when using EFA and 92.75% when using CCE. However, the advantage of using CCE over EFA is not only the marginal improvement in the average performance, but the simplicity and robustness in word boundary detection when using CCE with a global threshold compared to the need to perform individual word normalisation (with associated individual energy thresholds) required when using EFA. The ability to use a global threshold eases the processing overhead of the CCE algorithm resulting in faster execution.

The EFA and LCC techniques were then evaluated in terms of their performance within a speaker verification application and the LCC technique shows equivalent performance (with reduced processing overhead) compared to the EFA technique previously used in the same application.