2005

Text-independent speaker recognition

Smitha Gangisetty
West Virginia University

TEXT-INDEPENDENT SPEAKER RECOGNITION

Smitha Gangisetty

Thesis submitted to the
College of Engineering and Mineral Resources
at West Virginia University
in partial fulfillment of the requirements
for the degree of

Master of Science
in
Electrical Engineering

Afzel Noore, Ph.D., Chairman
Ronald L. Klein, Ph.D.
Powsiri Klinkhachorn, Ph.D.
Raymond Morehead, M.D.

Lane Department of Computer Science and Electrical Engineering

Morgantown, West Virginia
2005

Keywords: Speaker Recognition, MFCC, PCA, ICA, GMM

ABSTRACT

Text-independent Speaker Recognition

Smitha Gangisetty

This research presents new text-independent speaker recognition system with multivariate tools such as Principal Component Analysis (PCA) and Independent Component Analysis (ICA) embedded into the recognition system after the feature extraction step. The proposed approach evaluates the performance of such a recognition system when trained and used in clean and noisy environments. Additive white Gaussian noise and convolutive noise are added. Experiments were carried out to investigate the robust ability of PCA and ICA using the designed approach. The application of ICA improved the performance of the speaker recognition model when compared to PCA. Experimental results show that use of ICA enabled extraction of higher order statistics thereby capturing speaker dependent statistical cues in a text-independent recognition system. The results show that ICA has a better de-correlation and dimension reduction property than PCA. To simulate a multi environment system, we trained our model such that every time a new speech signal was read, it was contaminated with different types of noises and stored in the database. Results also show that ICA outperforms PCA under adverse environments. This is verified by computing recognition accuracy rates obtained when the designed system was tested for different train and test SNR conditions with additive white Gaussian noise and test delay conditions with echo effect.

ACKNOWLEDGEMENTS

It is my pleasure to express my sincere gratitude and appreciation to my advisor Dr. Prof. Afzel Noore, for his guidance, encouragement, and moral support throughout the graduate studies at West Virginia University. His constant help, invaluable suggestions, and insightful discussions helped me clarify many questions on various technical aspects. His mentorship is a rewarding experience, which I will treasure my whole life.

I would also like to extend my special appreciation and thanks to my committee members Dr. Ronald Klien, Dr. Powsiri Klinkhachorn, and Dr. Raymond Morehead for their valuable guidance, suggestions, comments and criticism.

I am very grateful to Vijay for his extreme support and encouragement. Special thanks for standing always beside me giving necessary feedback, calming my fear and participating in many useful discussions at all stages of my work.

I would like to convey my special thanks to all my roommates and friends for their timely help, support, humor, and encouragement.

DEDICATION

I would like to express my indebtness and gratitude to my family, who always provide me endless support, patience and encouragement. I dedicate this work to my parents Mr. Vidya Sagar, and Mrs. Usha Rani as I would never have come this far without their guidance through all phases of my life. I would also like to dedicate this work to my sister and my grandparents who have always been a great source of motivation and inspiration.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

ASR	Automatic Speaker Recognition System
AWGN	Additive White Gaussian Noise
DCT	Discrete Cosine Transform
DWT	Dynamic Time Warping
EM	Expectation Maximization
FFT	Fast Fourier Transform
GMM	Gaussian Mixture Model
HMM	Hidden Markov Model
ICA	Independent Component Analysis
KLT	Karhunen-Loeve Transform
LPC	Linear Prediction Coefficients
ML	Maximum Likelihood
MFCC	Mel-Frequency Cepstral Coefficients
NN	Neural Networks
PCA	Principal Component Analysis
PDF	Probability Distribution Function
SNR	Signal to Noise Ratio
VQ	Vector Quantization

CHAPTER 1. INTRODUCTION

Speaker recognition is the process of automatically recognizing a person on the basis of individual information included in speech signals. Campbell defines it more precisely as *the use of a machine to recognize a person from a spoken phrase* [Campbell, 1997]. It is a known fact that speech is a speaker dependent feature that enables us to recognize friends over the phone.

During the years ahead, it is hoped that speaker recognition will make it possible to verify the identity of persons accessing systems; allow automated control of services by voice, such as banking transactions; and also control the flow of private and confidential data [Furui, 1994].

Biometric based authentication measures individuals' unique physical or behavioral characteristics. While fingerprints and retinal scans are more reliable means of identification, speech can be seen as a non-evasive biometric data that can be collected with or without the person's knowledge or even transmitted over long distances via telephone. Biometric authentication has some key advantages over knowledge and token based authentication techniques. Unlike other forms of identification, such as passwords or keys, a person's voice cannot be stolen, forgotten or lost. Speaker recognition with proper statistical, analytical and data processing techniques thus allow for a secure method of authenticating speakers.

1.1 Motivation

1. Build a better text-independent speaker recognition model that would allow capturing speaker discriminating properties and therefore make the system robust against noise.
2. Avoid the shortcomings of present text-independent speaker recognition approaches using lower order statistics like Principal Component Analysis, particularly due to their poor de-correlation property thereby failing to extract additional useful speaker dependent information.
3. Explore the potential for increased robustness of text-independent speaker recognition systems using higher order statistical techniques such as Independent Component Analysis.

1.2 Research Objectives

The objectives of this research are:

1. Develop a new text-independent speaker recognition framework with multivariate dimensional reduction tools such as Principal Component Analysis (PCA) and Independent Component Analysis (ICA) embedded into the recognition system after the feature extraction step.
2. Dynamically train the speaker recognition system with clean and noisy (additive and convolutive) speech signals. Each time a new speech signal is input to the system, additive white Gaussian noise at different values of SNR and echo with varying values of delay are added to the clean speech signals.

3. Investigate the performance of the proposed text-independent speaker recognition system under noisy environments.
4. Compute the accuracy rates of identifying the test speaker in clean and noisy environments using the designed speaker recognition model.
5. Evaluate the robust ability of Principal Component Analysis (PCA) and Independent Component Analysis (ICA) transforms for speaker identification using the proposed approach under clean and noisy conditions.

1.3 Outline of thesis

A brief overview on the topics covered in each of the chapters is presented below.

Chapter 2 discusses the background of various concepts used in speaker recognition.

Chapter 3 summarizes a thorough literature review of text-independent speaker recognition system based on the current state of the speaker recognition technology. It introduces the basic model of text-independent speaker recognition system and its components as a means of explaining the process being carried out in sequential steps. Simultaneously it gives a complete survey of techniques used, work done by other researchers, and the results obtained.

Chapter 4 gives a complete description of the proposed text-independent speaker recognition system. It provides an in depth look at various technical details used in evaluating the proposed model and compares the experimental results with existing work.

Chapter 5 concludes the research with a summary and possible future work in the field of text-independent speaker recognition model.

CHAPTER 2. BACKGROUND

This chapter presents the basic concepts of speaker recognition. It identifies certain classifications, introduces theory of speech signal, and the mechanism of how speech is produced and represented.

2.1 Automatic Speaker Recognition System

Speaker recognition is the process of automatically recognizing a person on the basis of individual information included in speech signals. An Automatic Speaker Recognition System deals with recognizing the speaker at the output.

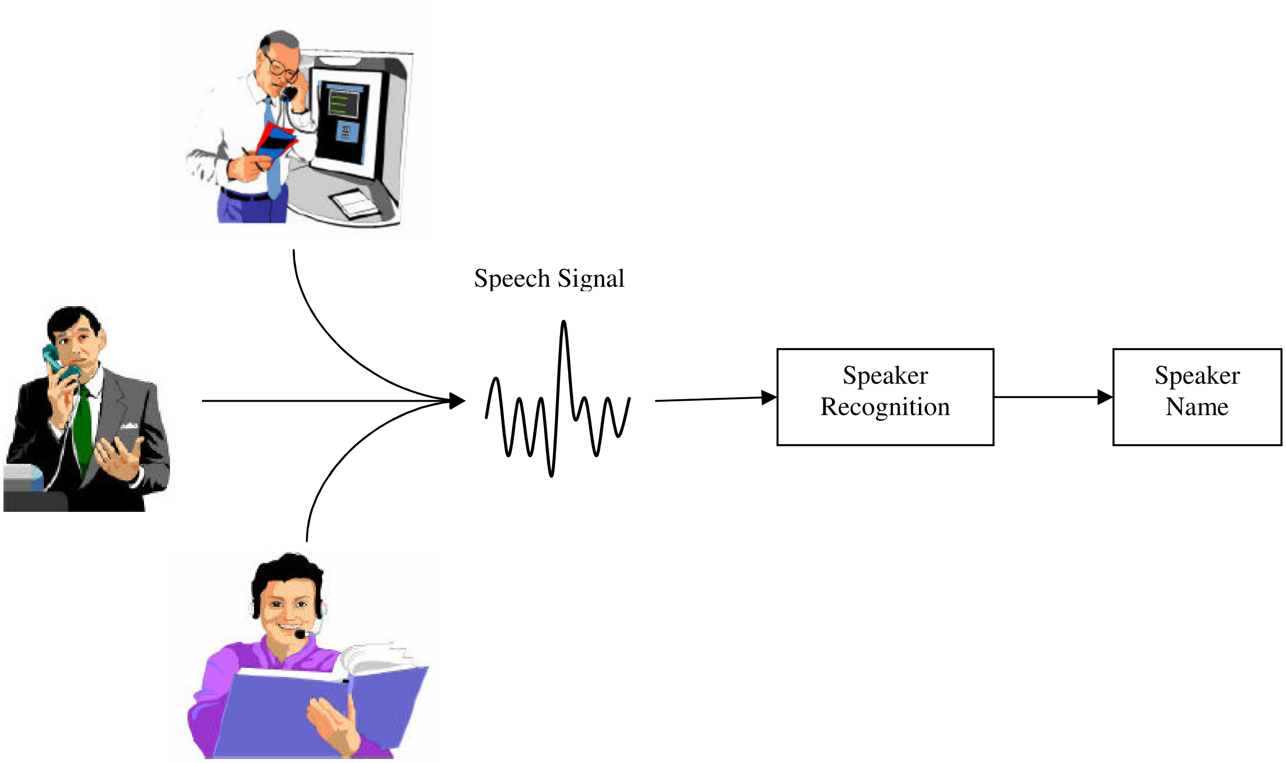

Fig. 2.1 [Douglas, 2000] Generic speaker recognition model

It is different from speech recognition and language recognition since these concepts deal with recognition of speech (i.e. the words that are spoken) and recognition of language (i.e. the language in which the words or sentences are spoken).

2.1.1 Classification

Speaker identification is a process of determining the identity of a person by machine [Gish, 1994]. The terms *speaker identification and speaker recognition* are used interchangeably [Gish, 1994]. Speaker recognition is of two types:

Text-dependent

Text spoken by a person is known to the speaker recognition model. In this process the speaker is asked to utter a prompted or a fixed phrase. Text-dependent recognition is employed in applications with strong control over user input. This type of recognition has an advantage of increasing the performance of the system because of the prior knowledge of the spoken text.

Text-independent

This type of mechanism is used for recognizing any type of conversational speech or user selected phrase. Text-independent recognition system has no prior knowledge of the text spoken by the person. This is generally used in applications with less control over user input.

Speaker recognition can be further subdivided into two categories [Gish, 1994] as shown in Fig. 2.2

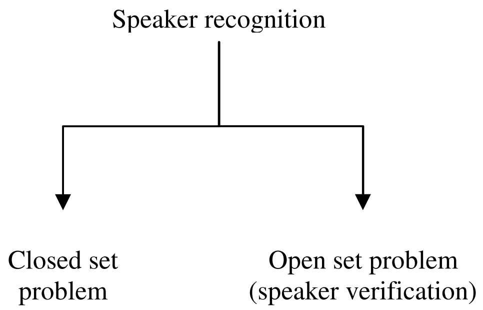

Fig. 2.2 Classification of speaker recognition system

Closed set problem

The closed set problem tries to determine the identity of a person most likely to have spoken the speech from among a set of known voices [Gish, 1994].

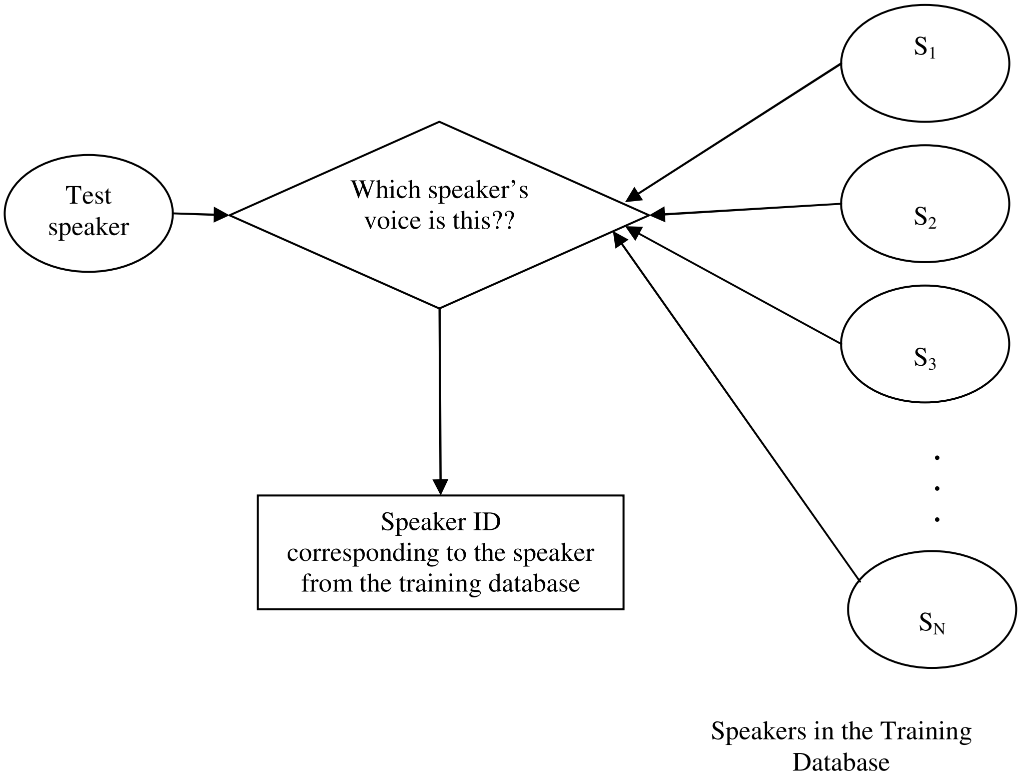
Test speaker
Which speaker's voice is this??
Speaker ID corresponding to the speaker from the training database
S1
S2
S3
SN
Speakers in the Training Database

Fig. 2.3 Closed set problem

This is also referred to as closed identification since it is often assumed that the unknown voices must come from a known set. Closed set problem can be represented by Fig. 2.3.

Open set problem

This problem deals with deciding whether the speaker of the particular test utterance belongs to a group of known speakers [Gish, 1994]. It is called open set problem because the unknown voice could come from a large set of unknown speakers. A special case of open set problem is called speaker verification. It is the task of verifying whether a speaker is who the individual claims to be from a given speech [Reynolds, 1995]. In this case, the speaker makes an identity claim. Open set problem can be represented in Fig. 2.4.

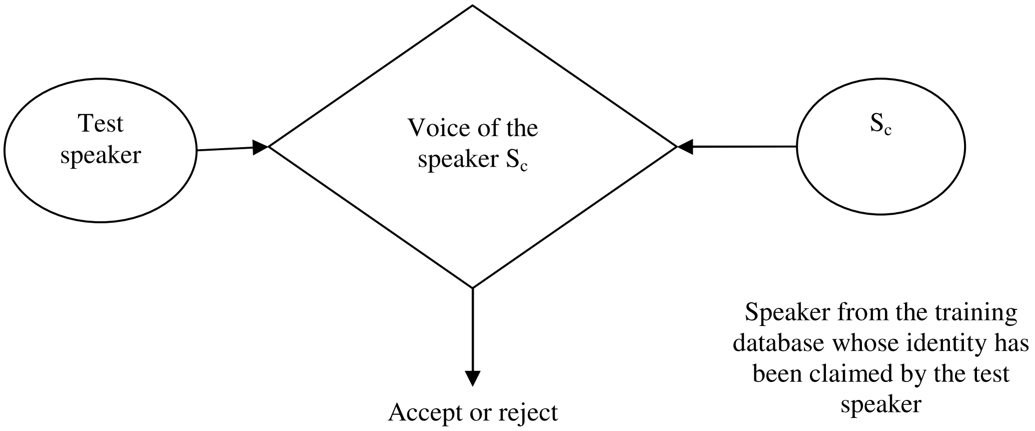
Test speaker
Voice of the speaker S_c
S_c
Accept or reject
Speaker from the training database whose identity has been claimed by the test speaker

Fig. 2.4 Open set problem

2.2 Theory of Speaker Recognition

Speech is a complex signal. This chapter will attempt to focus on the theory of a speech signal, and the mechanism of how speech is produced and represented.

2.2.1 Speech signal

Sound is defined as the longitudinal waves that propagate through a medium like solid, liquid or gas. A *voice signal* can be defined in terms of time domain and frequency domain. In time domain it gives the volume, pitch and tone and in frequency domain it gives spectral information contained in the voice signal that is unique to a particular speaker. S*peech* is the act of producing sound through vocal chords. The signal carrying the message information is also referred to as *acoustic waveform*. Information contained in the speech signal is of the discrete form and can be represented by a concatenation of elements from finite set of symbols called *phonemes* [Rabiner, 1978]. Speech signal can be transmitted, stored and processed in many ways and these are the three basic steps carried out in any communication system.

2.2.2 Speech Production Technique

In humans, pushing out air from the lungs through vocal chords and mouth produces speech. Lungs act as a source of producing sound and vocal tract acts as a filter. Articulators are soft palate, tongue, lips and jaw (Fig. 2.5). Speech is produced as a result of movements of different components of the vocal tract in different configurations producing voiced and unvoiced speech. As a result, pressure wave is generated in front of the lips. A speech signal is nothing but the sampled version of this pressure wave.

Vocal tract consists of connection from *esophagus* to mouth *(pharynx)* or oral activity. The overall shape of the vocal tract varies over time with the movement of the articulators thus causing corresponding variations in resonance properties. Therefore if we could track the changes in resonances then we will probably be able to track the articulator movements, and hence analyze the speech signal. Acoustically, this information can be obtained from the frequency spectrum of the speech signal at a particular instant [Campbell, 1997].

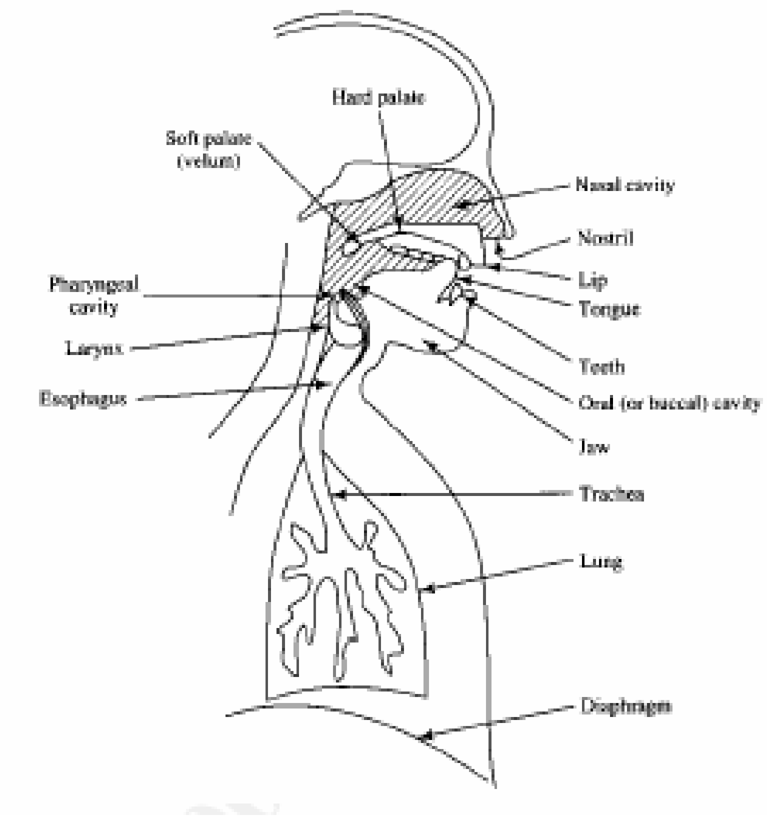

Fig. 2.5 [Flanagan, 1972] Human vocal system

The bandwidth of a speech signal is wide around 10 kHz [Kent, 1992]. Generally below 3 kHz, we can find the information regarding the linguistic content of speech signals with the higher frequency components mainly carrying the information particular to the speaker.

The frequencies at which the vocal tract resonates are called formants [Campbell, 1997] and they are important for the analysis of the speech signal. For voiced speech, about 4-5 disjoint formants are found below 5 kHz and for unvoiced sounds, formants tend to be more suppressed resulting in flatter spectrum containing less total energy. A speech signal is considered to be a random signal since humans probably cannot repeat the same cycle of articulator movements. The presence or absence of vocal chord vibration always tends to vary the distribution of the samples, therefore globally a speech signal is considered to be non-stationary. But due to the limitations imposed by human anatomy we have to assume that signal is locally stationary. For this purpose we have to fragment the signal into small isolated frames of approximate time durations of 10-20 ms. This assumption is extremely useful to avoid certain problems of deriving tractable speech production models. The property of irreproducibility by human beings can be used here. There can be many possible realizations of same utterance. Human speech perception system is capable of accepting all these realizations as conveying the same meaning. Now it becomes evident from the waveforms that, though speech signals may vary numerically or vary in duration, they may still carry the same linguistic content. Even the signals representing same utterance from the same individual vary considerably.

2.2.3 Voiced, Unvoiced and Plosive speech

Speech is the acoustic wave that is produced or radiated by sub-glottal system, by the air expelled from the lungs and is perturbed by some constraint at some moment, somewhere in the vocal tract.

Vocal tract can be modeled as a linear time varying filter. Fig. 2.6 represents the appropriate model for speech production derived from the speech production mechanism. This is an all-pole model capable of representing all sounds. Generally nasal and fricative sounds consider poles and zeros but once the order of the filter is very high it acts as an all pole model. This summarizes the fact that vocal tract response represents an all pole model.

Depending upon the mode of excitation, speech sounds can be classified into three categories:

1. Voiced
2. Unvoiced
3. Plosive

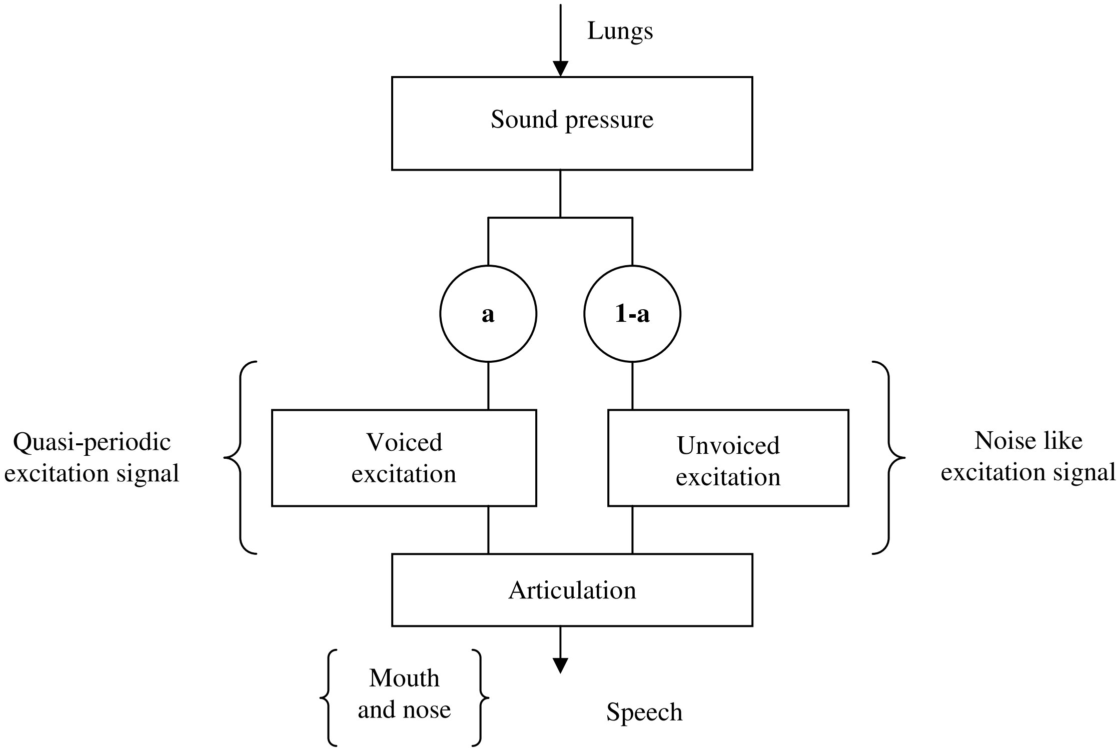

Fig. 2.6 Model describing speech production

Voiced Speech

Vocal tract is excited by producing *quasi-periodic pulses* of air [Fant, 1973] (see Fig. 2.6). Therefore voiced speech exhibits quasi-periodic behavior. *Vowels* are usually classified as voiced sounds [Fant, 1973]. These types of sounds have high average energy levels and very distinct formant frequencies (Fig. 2.7). Such sounds are produced by forcing the air from the lungs over the vocal chords as a result of which vocal chords vibrate in a periodical pattern and generate series of air pulses called glottal pulses [Campbell, 1997], [Fant, 1973]. These glottal pulses or air pulses travel through rest of the vocal tract to mouth, where some frequencies resonate. *Pitch* of the sound is defined as the rate at which vocal chords vibrate. Generally in women and children, due to a faster rate of vibration of the vocal chords while producing voiced speech, pitch is believed to be higher than in men [Fant, 1973], [Kent, 1992]. Therefore pitch is also an important parameter to be included for analysis or synthesis of voiced sounds.

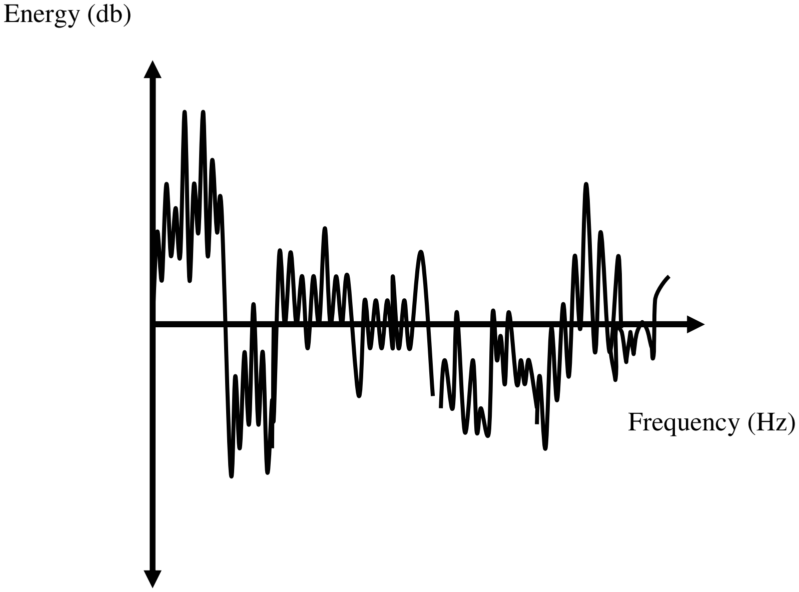

Fig. 2.7 Voiced speech

Perceived Pitch

The perceived pitch differs with the gender and age of the speaker. Its range for humans lie between 50 and 500 Hz. Children have the highest pitch voices followed by females and then males with the lowest pitch. Pitch varies with time and tells about the prosody of utterance. With age, females tend to lower their pitch and male voices tend to rise in pitch. The acoustical counterpart of pitch is fundamental frequency.

Stress

Information about the meaning and also about the language can be revealed depending upon the way stress is applied to certain parts of an utterance. An acoustical counterpart of stress is the energy of speech signal. Energy of the signal can also be used to detect or track the salient periods preceding the burst release of glottal stops and is higher during voiced speech.

Unvoiced speech

Sounds produced due to unvoiced speech have a random behavior and are generated by forming a constriction at some point in the vocal tract towards the mouth and forcing the air through the constriction at a very high velocity to produce turbulence [Flanagan, 1972]. Thus noise is generated to excite the vocal tract.

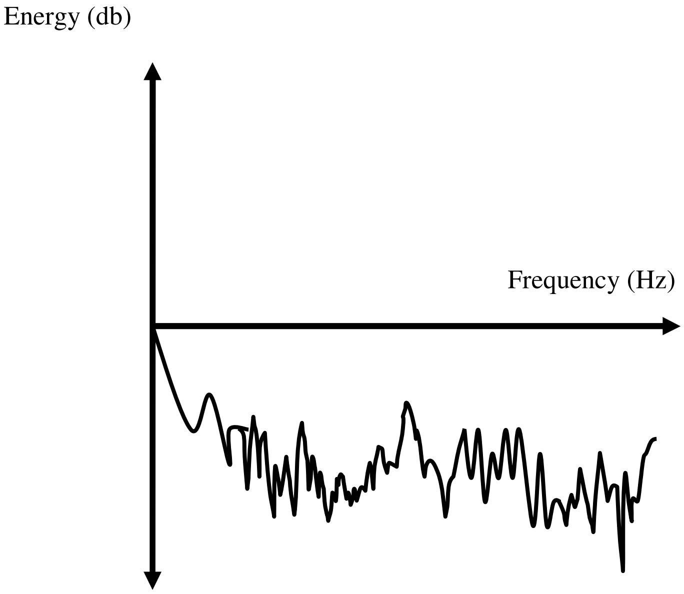

Fig. 2.8 Unvoiced speech

Unvoiced speech is also referred to as fricative speech. *Consonants* are classified as unvoiced sounds [Fant, 1973]. Unvoiced sounds have lower energy levels and high frequencies than voiced sounds (Fig. 2.8). Unvoiced sounds are produced when air is forced through the vocal tract with vocal chords open until the sound is produced in a turbulent flow. There is no vibration of vocal chords taking place here and therefore pitch does not come into picture.

Plosive Sounds

These sounds are generated due to complete closure towards the front of the vocal tract causing pressure to build up behind the closure and abruptly releasing it [Campbell, 1997]. The resonant frequencies of the vocal tract are called *formant frequencies* and depend upon the shape and dimensions of the vocal tract.

CHAPTER 3. LITERATURE REVIEW

This chapter introduces the basic concepts of text-independent speaker recognition system with a detailed sequence of steps that characterize the system. It also presents a complete literature review of text-independent speaker recognition system and sheds light on work done by other researchers in this field.

3.1 Text-Independent Speaker Recognition System

Text-independent speaker recognition is the task of identifying a speaker by machine [Campbell, 1997]. In this research, only text-independent speaker recognition is considered. This involves two phases: Training and Testing.

Training

This is a process of making the system know the speakers and deals with collecting data from the utterances of people to be identified.

Testing

It is the task of identifying an unknown utterance. This is accomplished by making some kind of comparison between the unidentified utterance and the training data.

This technique should work irrespective of the text either in training or testing process. The system does not have any prior knowledge of the text spoken by the person. Practically, designing a text-independent recognition system is more difficult than designing a text-dependent system but has an advantage of being more flexible.

Applications [Gish, 1994]

Text-independent speaker recognition system has many potential applications. They are:

Security Control

Speaker recognition systems can be used for law enforcement. They can help identify suspects. Some security applications employ sophisticated techniques to check whether a speaker is present where that particular speaker is supposed to be.

Telephone Banking

Access to bank accounts may be voice controlled. Such systems may want to verify whether the authorized person is trying to access the accounts, private and personal details. Intelligent machines may be programmed to adapt and respond to the user.

Information retrieval systems

Participants in conferences or meetings may be identified by special machine technology. Automatic transcriptions containing a record of who said what can be also obtained from large quantities of audio information if such machine technology is used in conjunction with continuous speech recognition systems.

Speech and Gender recognition systems

Speaker recognition systems can be usefully employed by speech recognition systems. Many speaker independent speech recognizers are already using gender recognition system for improving the performance.

Fig. 3.1 illustrates the schematic diagram of a typical text-independent speaker recognition system. Each block represents a unique component of the system. A text-independent speaker recognition system comprises of two parts: front-end (feature extraction) and back-end (actual recognition). These systems use processed form of speech signals instead of using raw speech signal as it is obtained. This is to reduce the time consumed in identifying the speaker and to make the process easy by reducing the data stream and exploiting its advantage of being redundant.

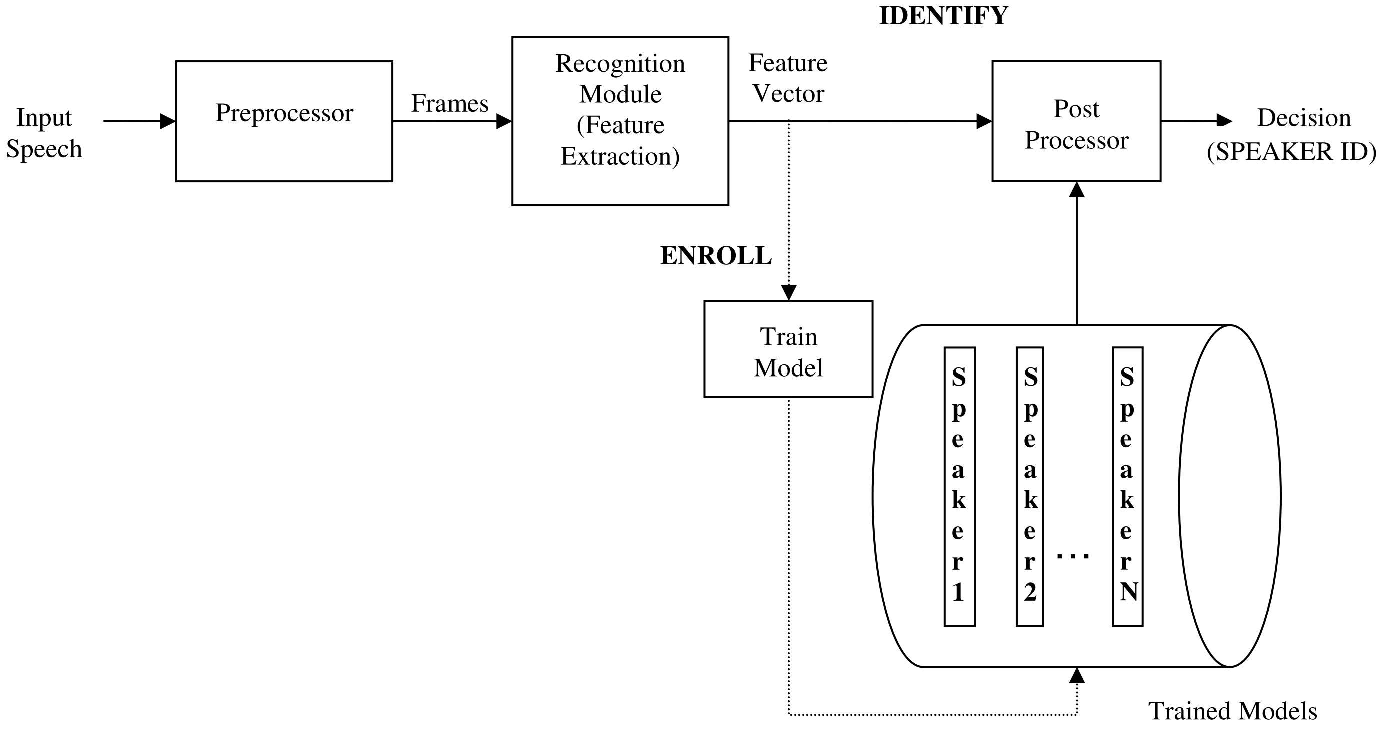

Fig. 3.1 Block diagram of a text-independent speaker recognition system

Computation of Cepstral coefficients using preprocessing and feature extraction phases play a major role in text-independent speaker recognition systems. Various studies [Zhu, 1994] and [Furui, 1981] have shown that computing Cepstral coefficients is the best among all the parameters for any type of speaker recognition. It was proved by Erell and Weintraub [Erell, 1993] that the performance of the speech recognizers can be improved using Cepstral representation of the signals for both clean speech and noisy environment. Most widely used techniques are the frequency representations of the signals and they are Fourier Transforms, Linear Prediction Analysis and Mel-Frequency Filter Banks [Umesh, 2002]. One of the main advantages of using Cepstra is that they can be considerably modeled by multivariate Gaussian distribution functions [Gish, 1994]. This involves short term speech parameterization which is defined as an efficient method of representing spectral and temporal information contained in non-stationary speech signals. Speech parameterization includes: Mel-frequency Cepstral coefficients (MFCCs) [Reynolds, 1995] and Linear Prediction Coefficients (LPC) [Campbell, 1997]. Mel-frequency Cepstral coefficients (MFCCs) are one of the most commonly used features in variety of applications [Gish, 1994], [Shannon, 2004]. Transforming the spectral coefficients into Cepstral domain using Discrete Cosine Transform (DCT) thereby removing the correlations between the adjacent coefficients generates these coefficients. Linear Prediction Coefficients (LPC) Cepstrum is another such feature that is often found in the literature [Campbell, 1997]. Furui [Furui, 1981] has shown that Cepstral coefficients work well even with Linear Prediction Models. The generation of a LPC Cepstrum involves autocorrelation sequence of a speech frame. Though LPC Cepstrum is less expensive, it is not as effective as MFCCs [Somervuo, 2003]. A traditional MFCC feature extractor was used in our research work and the description of basic components of this system is given below.

3.1.1 Preprocessor

Initially speech signal is processed with the help of a preprocessor (Fig. 3.2). The main purpose of preprocessing is to reduce the amount of data to be processed by the rest of the system. Preprocessing involves: A/D Conversion and Pre-emphasis filtering.

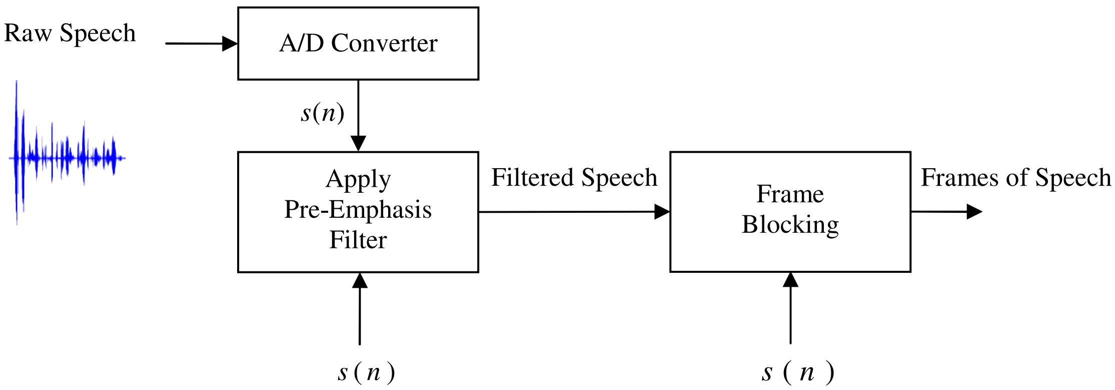

Fig. 3.2 Preprocessor

3.1.1.1 A/D Conversion

The digital speech signal $s(n)$ is captured by an analog-to-digital converter (ADC) at sampling frequency fs. It is applied to a pre-emphasis filter for further processing.

3.1.1.2 Pre-emphasis filtering

Pre-emphasis filtering is a process in which the frequency response of the filter has emphasis at a particular frequency range. The input speech signal is filtered with a first order high pass filter whose transfer function is given by

$$H(z) = 1 + \alpha z^{-1} \tag{3.1}$$

where α typically lies within the range of -1.0 and -0.4 and reflects the degree of pre-emphasis [Picone, 1993]. Pre-emphasis filtering is traditionally used to compensate for the -6dB/octave spectral slope of the speech signal.

Frame Blocking

This is a process of dividing or splitting the pre-emphasized signal into equal frames of finite length N. Each frame begins at the offset of the previous frame by L samples as shown in Fig. 3.3. The second frame begins at $L+1$ and the third frame begins at $2L+1$ and so on. If $L \leq N$, the adjoining frames overlap. The transitions from frame to frame can be smoothed out by introducing the overlap. In a system where the sampling frequency is 8 kHz, typical values of L and N are 80 and 160 respectively, related to a frame length of 20 ms with an overlap of 10 ms [Gish, 1994].

If x_i is the i^{th} segment of the sampled speech $\hat{s}$, and I is the required number of frames, then frame blocking can be described as

$$x_i(n) = \hat{s}(Li + n) \tag{3.2}$$

for $n = 0,1,....,N-1$ and $i = 0,1,....,I-1$

Thus by dividing the apparently stochastic acoustic data into frames, it is now possible to calculate some of the useful features on each frame.

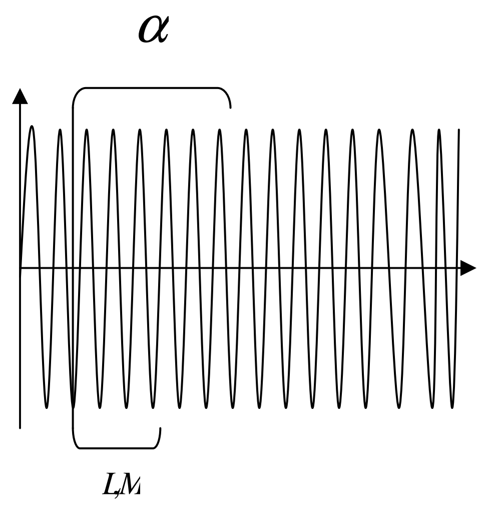

Fig. 3.3 Parameters in frame blocker

3.1.2 Recognition Module (Feature Extraction)

This is the core or heart of any text-independent speaker recognition system. The main purpose of this module lies in obtaining reliable and efficient smoothing of the frequency response of a human vocal tract. Calculating the Cepstral coefficients for a speech signal involves the following steps: windowing, followed by Fourier transformation, Mel-spectrum generation and discrete cosine transformation (DCT) for each time-frame [Picone, 1993]. Fig. 3.4 represents the block diagram for generating the Cepstrum.

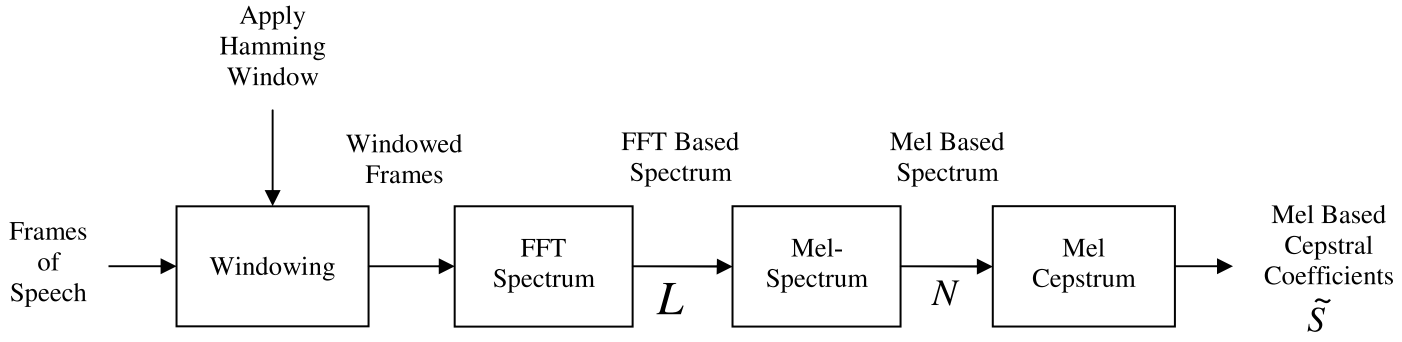

Fig. 3.4 Feature extraction process

3.1.2.1 Windowing

It is a process in which each pre-emphasized frame is multiplied by a time window of given shape to emphasize pre-defined characteristics of the signal. Use of windowing ensures that all parts of the signal are recovered and the possible gaps between the frames are eliminated. Hamming window is one of the most commonly used windowing techniques [Picone, 1993]. This is done to enhance the harmonics, smooth the edges and to reduce the edge effect while taking the *FFT* on the signal. The output windowed segment can be defined as [Picone, 1993]:

$$x(n) = x_i(n)w(n),\ n = 0,1,....,N-1 \tag{3.3}$$

and $w(n)$ is the Hamming window defined as:

$$w(n) = 0.54 - 0.46\cos\left(\frac{2\pi n}{N-1}\right) \tag{3.4}$$

3.1.2.2 FFT Spectrum

FFT is the Fourier transformation. Short-term power spectrum is computed by applying Fourier Transform (FFT) to each windowed signal, directly taking the magnitudes of Fourier coefficients raised to the power of two and is represented as $s(k)$. The schematic diagram given below describes the sequence of generating power spectrum for each windowed frame obtained from the previous section.

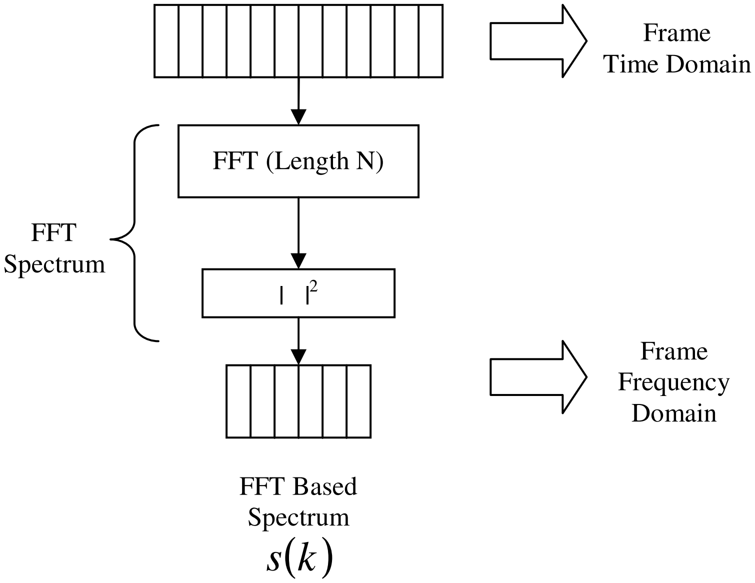

Fig. 3.5 Schematic diagram for generating FFT spectrum

3.1.2.3 Mel-Spectrum

Mel-Spectrum is computed by passing the Fourier transformed signal through a set of band-pass filters known as Mel-Filterbank. The filters are designed such that their shapes correspond to the Hamming window. The magnitude of each filter is computed by multiplying each filter in the bank with the spectrum. This process involves simple matrix operations and makes the formants more clearly identifiable.

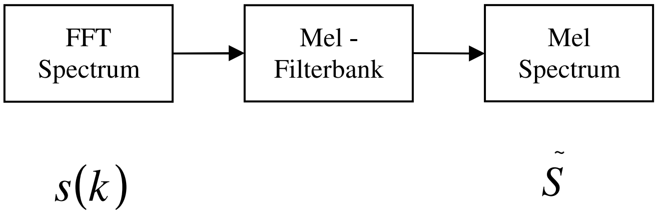

Fig. 3.6 Schematic diagram for generating Mel-spectrum

Mel-Scale Formulation

Mel-scale was first proposed in 1937 by Stevens, Volkman and Newman [Umesh, 2002]. Human ear tends to perceive the frequencies below 1000 Hz in a linear way and frequencies above 1000 Hz in a non-linear manner. A mel is a unit of measurement of percieved frequency (pitch) of the tone [Umesh, 2002]. Mel-scale formulation is given as

$$fmel = 2595 \log_{10}\left(1 + \frac{f}{700}\right) \tag{3.5}$$

where $fmel$ is the frequency in Mel-scale corresponding to the actual frequency f [Klabbers, 2001]. The mapping or transformation taking place in Mel-scale formulation is illustrated by Figures 3.7 and 3.8. Fig. 3.7 represents the mapping on an ordinary scale where as Fig. 3.8 represents the mapping on a logarithmic scale.

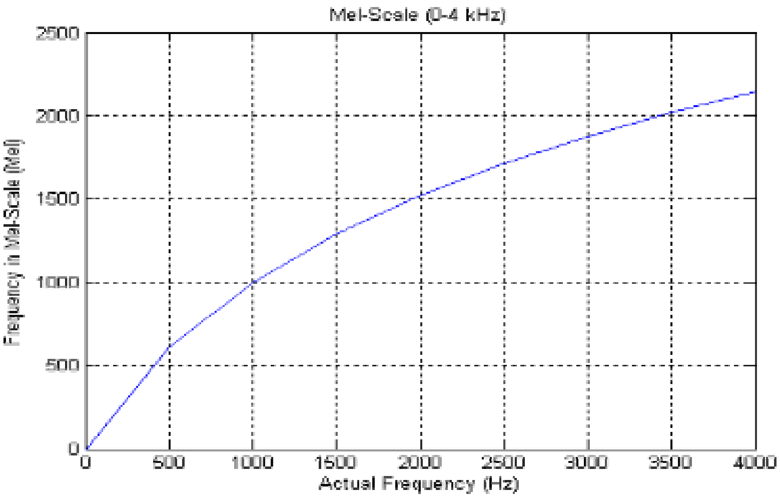

Fig. 3.7 Mel-scale formulation on linear scale (0-4 kHz)

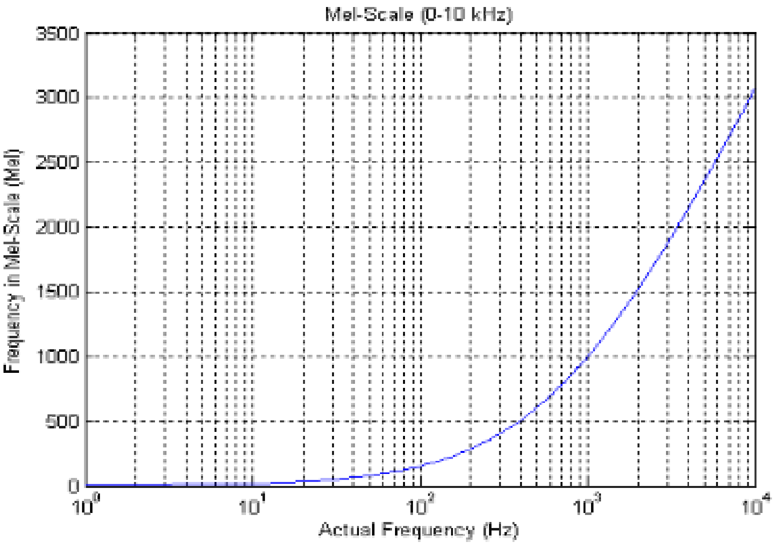

Fig. 3.8 Mel-scale formulation on log scale (0-10 kHz)

Mel-Frequency Filterbank

The Mel-filter bank is designed to simulate band pass filtering occurring in auditory system such that it is approximately linear up to 1 kHz and in actual frequency domain is logarithmic at higher frequencies [Picone, 1993]. Such a model allows a constant bandwidth and constant spacing on the Mel-frequency scale and exploits the fact that the speech signal is stationary for short periods of time. It is modeled by constructing the required number of triangular band-pass filters with 50% overlap. Triangular band-pass filters are generated with Mel-frequencies to be the centers of the triangles (Fig.3.9: Mel-Filterbank for 20 filters).

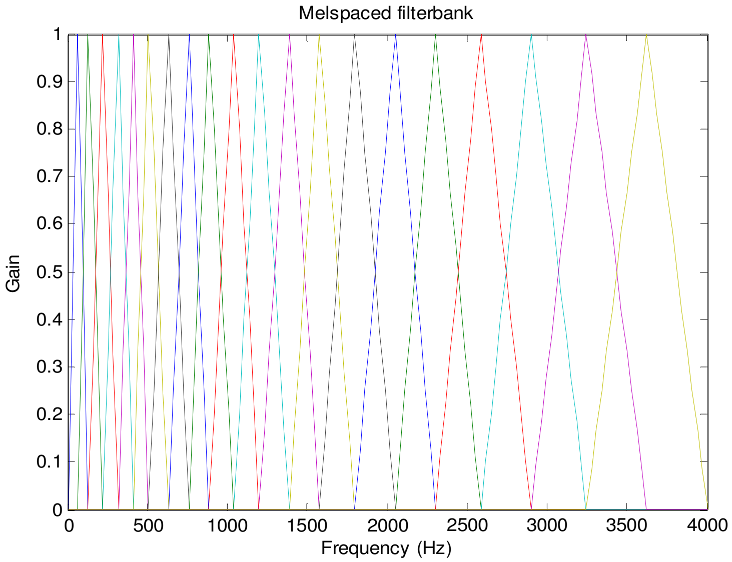

Fig. 3.9 Mel-filter bank

3.1.2.4 Cepstral Coefficients

Cepstrum $(c(n))$ in its simplest form is the discrete cosine transformation of the Mel-spectrum of a signal in logarithmic amplitudes and can be mathematically defined as [Rabiner, 1993]

$$c(n) = ifft(\log|fft(s(n))|) \tag{3.6}$$

where $s(n)$is the signal obtained from the convolution of an excitation signal $p(n)$, approximately a periodic impulse train and synonymous with frequency and $h(n)$ representing the transfer function of a filter practically the impulse response of all the things that get in the way of speech emanating from the lungs e.g. teeth, nasal cavity, lips etc [Picone, 1993] .

$$s(n) = h(n) \otimes p(n) \tag{3.7}$$

The figure briefly describes the process involved in computing Cepstral coefficients:

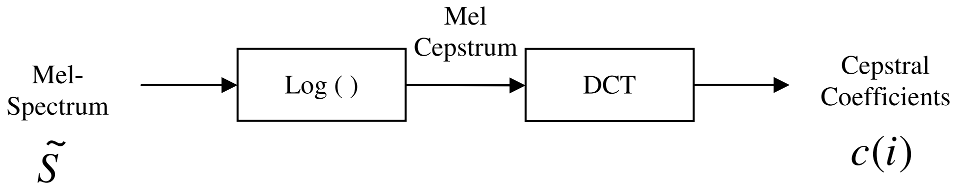

Fig. 3.10 Speech cepstrum parameterization

The Mel-spectrum given by $\tilde{S}$ is usually represented on a log scale because the shape of the log power spectrum is preserved independent of the input signal strength. The discrete cosine transformation applied to the transformed logarithmic-scaled energies produces a set of Cepstrum coefficients (c) given as [Molau, 2001], [Picone, 1993]

$$c(i) = \sqrt{\frac{2}{Nof}} \sum_{m=1}^{Nof} \log\left(\tilde{S}(m)\right) * \cos\left(\frac{\pi}{Nof} i(m - 0.5)\right),$$

$$i = 0,1,....C - 1 \tag{3.8}$$

where Nof is the number of filters and $c(i)$ are the Cepstral coefficients and C is the number of Mel-Cepstral coefficients. Cepstral analysis thus converts logarithmic-scaled energies to generate a signal in the Cepstral domain with a que-frequency peak corresponding to the pitch and lot of formants. Mel-Cepstral coefficients [Davis, 1980] are highly useful parameters since they perceptually capture the most important characteristics of speech. Since most of the signal information is represented by the first few Mel-Cepstral coefficients, the system is made robust by extracting only those coefficients [Gish, 1994].

3.1.3 Training the text-independent speaker recognition system

Training the model includes ENROLL Phase which is one of important phases used in the text-independent speaker recognition employed after the feature extraction step.

Each speaker model is trained with the extracted feature vectors and is stored in the trained database with corresponding speaker ID which is unique. There are two types of models that can be used for training the input data [Gish, 1994]. They are parametric and nonparametric models.

Parametric Models

These models have a particular structure characterized by a set of parameters. By defining the structure, the form of the model has been specified and limited to a specific requirement. This ensures that it makes an efficient use of the data in estimating the model parameters. Another major advantage in using parametric model is that the changes in the parameters can be easily determined by the changes in the data [Gish, 1994]. Parametric models include Gaussian mixture models (GMM), Hidden Markov Models (HMM) and Neural Networks (NN). Literature shows that many researchers have implemented parametric models in the text-independent speaker recognition system [Poritz, 1982], [Tishby, 1991], [Gish, 1994], [Reynolds, 1995] and [Seddik, 2004]. The use of a five state HMM for text-independent speaker recognition is proposed by Poritz [Poritz, 1982] and expanded to 8 states in [Tishby, 1991] by Tishby. Seddik, Rahmouni and Sayadi in [Seddik, 2004] have proposed an implementation of neural networks in text-independent speaker recognition system. Text-independent speaker recognition with Gaussian mixture model was proposed by Reynolds [Reynolds, 1995]. GMM is most commonly used parametric model for training purposes [Gish, 1994]. We therefore implemented GMM in our model to increase robustness and performance of the designed approach.

Gaussian Mixture Models (GMM)

GMM is a classic parametric method best used to model speaker identities due to the fact that Gaussian components have the capability of representing some general speaker dependent spectral shapes. Gaussian classifier has been successfully employed in several text-independent speaker identification applications since the approach used by this classifier is similar to that used by the long term average of spectral features for representing a speaker's average vocal tract shape [Gish, 1986].

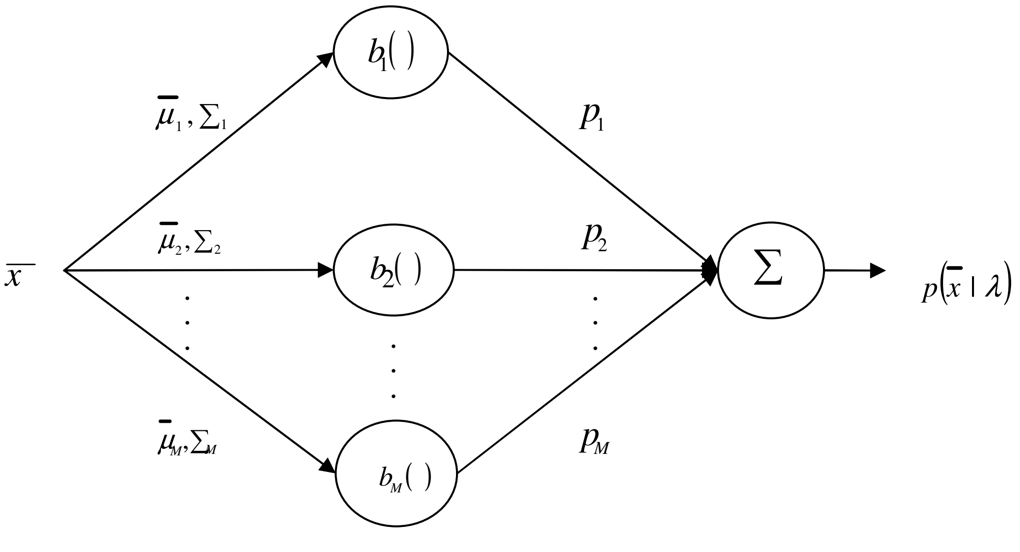

Fig. 3.11 [Reynolds, 1995] M component Gaussian mixture density

In a GMM model, the probability distribution of the observed data takes the form given by the following equation [Reynolds, 1995]

$$p(\bar{x} \mid \lambda) = \sum_{i=1}^{M} p_i b_i(\bar{x}) \tag{3.9}$$

where M is the number of component densities, $\bar{x}$ is a D dimensional observed data (random vector), $b_i(\bar{x})$ are the component densities and p_i are the mixture weights for $i = 1,...,M$.

$$b_i(\bar{x}) = \frac{1}{(2\pi)^{D/2} \left|\sum_i\right|^{1/2}} \exp\left\{-\frac{1}{2}(\bar{x} - \bar{\mu}_i)^T \sum_i^{-1} (\bar{x} - \bar{\mu}_i)\right\} \tag{3.10}$$

Each component density $b_i(\bar{x})$ denotes a D- dimensional normal distribution with mean vector $\bar{\mu}_i$ and covariance matrix $\sum_i$. The mixture weights satisfy the condition $\sum_{i=1}^{M} p_i = 1$ and therefore represent positive scalar values. These parameters can be collectively represented as $\lambda = \{p_i, \bar{\mu}_i, \sum_i\}$ for $i = 1,...,M$. Each speaker in a speaker identification system can be represented by a GMM and is referred to by the speaker's respective model λ. Fig. 3.11 represents a Gaussian mixture density modeled as weighted sum of M component densities.

The parameters of a GMM model can be estimated using maximum likelihood (ML) [McLachuo, 1998] estimation. The main objective of the ML estimation is to derive the optimum model parameters that can maximize the likelihood of GMM. Unfortunately direct maximization using ML estimation is not possible and therefore a special case of ML estimation known as Expectation-Maximization (EM) [Dempster, 1977] algorithm is used to extract the model parameters.

The GMM likelihood for a sequence of T training vectors $X = \{\bar{x}_1,.....,\bar{x}_T\}$ can be given as [Reynolds, 1995]

$$p(X \mid \lambda) = \prod_{t=1}^{T} p(\bar{x}_t \mid \lambda) \tag{3.11}$$

The EM algorithm begins with an initial model λ and tends to estimate a new model $\bar{\lambda}$ such that $p(X \mid \bar{\lambda}) \geq p(X \mid \lambda)$ [Reynolds, 1995]. This is an iterative process where the new model is considered to be an initial model in the next iteration and the entire process is repeated until a certain convergence threshold is obtained.

Hidden Markov Model (HMM)

HMM is a simplified stochastic process model based upon the Markov chain [Rabiner, 1989]. The working principle of a Hidden Markov Model (HMM) is similar to that of a finite state automation system. Its main aim is to generate a model containing whole set of possible realizations of each word.

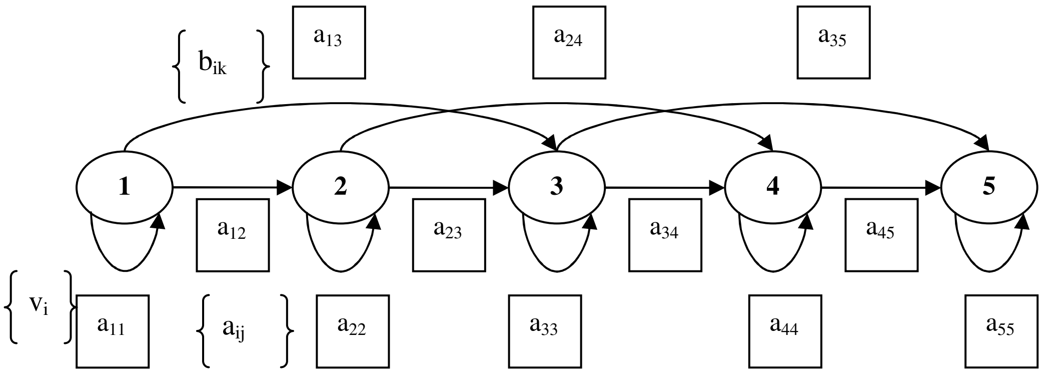

Fig. 3.12 State diagram representing HMM

Given the inputs, the probabilities of each of the HMMs in the system are calculated. This results in a possible pattern sequence. The input is then identified as one represented by the HMM having the highest of the probabilities. The parameters a_{ij}, b_{ik}, v_i are determined by training the system such that the probabilities are maximized. Ideally the training procedure employed in a HMM speaker recognizer should be optimized to minimize the training error rate. Also the system must be trained on a large speech database to achieve superior performance [Picone, 1993].

Neural Networks (NN)

Neural networks attempt to simulate some or all of the characteristics of biological neurons that form the structural constituents of the brain. Similar to the HMM's, Neural networks have to be trained to simulate the highly parallel and distributed way of information processing in the brain. Such systems can adapt themselves to the changes in the surrounding environments by modifying their synaptic weights. Neural networks also have the capability of handling imprecise, noisy, fuzzy and probabilistic information. [Seddik, 2004] has shown application of a neural network model to a text-independent speaker recognition system using MFCC.

Nonparametric Models

Nonparametric models differ from the parametric models like the way in which the space is dichotomized. Only the minimal assumptions regarding the probability density functions are made. Vector Quantization and Dynamic Time Warping (DTW) are examples for nonparametric models. Vector Quantization is used for text-independent speaker recognition where as Dynamic Time Warping (DTW) is used for text-dependent speaker recognition. Vector Quantization was first applied to speaker recognition by Soong *et al.* [Soong, 1985]. A description and a comparison of VQ model with HMM for text-independent speaker recognition system is given by Matsui and Furui [Matsui, 1994].

Vector Quantization (VQ)

Vector Quantization (VQ) based classifier is used for text-independent speaker recognition. VQ codebook has a small number of highly representative vectors that efficiently represent the speaker specific characteristics. This is a method used for reducing or compressing the number of training vectors required in a recognition system. Now a days these are being replaced by Gaussian mixture model based classifiers.

Dynamic Time Warping (DTW)

This is one of the classification techniques used earlier for speaker identification. In a pattern matching process the time duration of two utterances i.e. the input speech vector and the stored pattern vector may not be same though they may represent same utterance. In simple words, length of the preprocessed input does not necessarily match the pattern vector. DWT is a dynamic programming used to align similar parts of two utterances at a time [Gish, 1994]. DTW algorithm also combines both the warping and distance measurement into one simple procedure. This type of recognition module technique ignores the inherent variability in speech. Though time distortions are overcome, they do not allow proper scaling. Therefore most of the modern ASR systems replace this technique by a stochastic approach such as HMM.

3.1.4 Post Processor

Post processing involves IDENTIFY Phase. This phase uses the identification process where the test feature vectors are identified belonging to one of the speakers in the train database. The goal of classification is to build a set of models that can correctly predict the class of the different objects. Input to these methods is a set of objects (i.e., training data), the classes which these objects belong to (i.e., dependent variables), and a set of variables describing different characteristics of the objects (i.e., independent variables). Designing a pattern classifier depends on a number of different factors such as the distribution of the training data, and the assumptions made concerning the distribution.

In ASR systems, during the classification phase, the sequence of feature vectors is compared with acoustic models generated for each of the speakers in the training database to produce a similarity measure that relates the test utterance with each speaker. The speaker identification system then recognizes the identity of the speaker using this measure. Calculating the matching score in this process is computationally the most expensive step in speaker identification. The pattern classifier is designed such that it yields an (in some sense) optimal response for a given pattern under the expected operating conditions or the test conditions. The design of a classifier can have a major impact on the systems effectiveness and efficiency.

Various types of classifiers have been used for speaker identification. These can be grouped into either template or stochastic based classifiers [Gish, 1994]. Template matching methods were employed earlier before the development of stochastic or probabilistic models. They have proved to be sensitive to different variations in channel and background noise which could result in altering the feature properties [Gish, 1994]. Our research uses the probabilistic technique, the Bayes' decision, for speaker identification.

Stochastic Models

This type of modeling deals with computing probability distributions rather than distances to average features as in template models. Domingos and Pazzani in

[Domingos, 1997] reported through an experiment that the naive Bayes' classifier proved to be a good classification tool when compared to several other classical learning algorithms on large ensemble of data sets.

Bayes' Decision Rule

If distributions for all the speakers are assumed to be known and if p_i is assumed to be the continuous densities, then the likelihood that a feature vector x is generated by the i^{th} speaker is $p_i(x)$ [Gish, 1994]. Using the Bayes' rule, the probability that the speaker is the i^{th} speaker is [Gish, 1994]

$$p(\text{speaker} = i \mid \bar{x}) = \frac{p_i(\bar{x})P_i}{p(x)} \tag{3.12}$$

where P_i is defined as the prior probability that the utterance came from the i^{th} speaker and the probability of feature x occurring from any speaker is given as $p(x)$.

$$p(x) = \sum_{i=1}^{I} p_i(x)P_i \text{ , } I \text{ is the number of speakers} \tag{3.13}$$

The prior probabilities for each of the speakers are typically assumed to be equal. Therefore if the prior probabilities for all the speakers are assumed to be equal then the identified speaker will be the one with the highest probability distribution or likelihood. Probabilistic modeling was first applied by Schwartz et al. [Seddik, 2004] to the speaker identification task. This method is very useful in robust identification systems [Gish, 1994].

Template Models

Classifiers based on template models are the simplest of all. Template models use distance measures to compare the test utterances to the training templates in speaker identification applications. Most commonly used template models are distance metrics.

Distance metrics

The techniques used for template matching also vary based on the distance metrics used. There are several types of distance metrics [Gish, 1994] and Euclidean is one of the simplest and commonly used among them.

Euclidean

Euclidean distance D_E is defined as the measure of dissimilarity and is given by the equation [Brummer, 1997]

$$D_E = \sqrt{\sum_{i=1}^{\dim}(x_i - y_i)^2} \tag{3.14}$$

where x_i and y_i are the given vectors. Euclidean distance is also defined as Mean Square Error (MSE), a measure of the quality of the codebook generated from training.

Mahalonobis

Another distance metric available is Mahalonobis distance which is defined as [Gish, 1994]

$$r^2 = (\bar{x} - \bar{\mu}_i)^T \sum_i^{-1} (\bar{x} - \bar{\mu}_i) \tag{3.15}$$

where $\bar{x}$ is the average of test feature vectors, $\bar{\mu}$ is the mean and $\sum$ is the covariance.

3.2 Robust Speaker Identification

Practically in any speaker recognition application the input speech signals may not always be clean and may be corrupted in many ways. Noise may contain uncharacteristic speech sounds, crosstalk or speech from multiple speakers. The identification performance degrades considerably due to the presence of noise. This was observed by Lockwood and Boudy [Lockwood, 2001] and can create a major obstacle in the design of a commercial recognition system that is required to be used in normal day-to-day environments. This causes a call for robust recognition systems that would be able to improve recognition rates even in the presence of noisy environments or during the changes in the speaker's voice due to the external noise. In order to reduce the mismatch between test data in noisy environments and speech models trained under clean conditions, one solution is to add the noise experienced under test conditions to the training data. Furui [Furui, 1992] has been able to show that the use of such training data contamination gives good improvements in a number of recognition systems [Furui, 1992]. Therefore we propose an approach wherein the available database is trained with clean and noisy speech signals generated under different noise environments. The use of data contamination can also be helpful for learning algorithms to perform better recognition. The robust approach of our research is based on computing speaker analysis on relatively short time frames of speech. This can be used with any class of recognizers used and we used Gaussian mixture model with Bayes' classification rule for speaker identification.

3.3 Related Work

Prior researchers have applied several analytical approaches to the problem of text-independent speaker recognition [Reynolds, 1995], [Gish, 1994], [Seddik, 2004], [Tishby, 1991] and [Matsui, 1994]. Considerable work has been done by Douglas A. Reynolds [Reynolds, 1995] in the field of robust text-independent speaker recognition using Gaussian mixture models. The model implemented the use of traditional MFCC feature extraction as front end and Gaussian mixture models with Bayes' decision rule as a back end for speaker identification. Spectral analysis was carried out on 20 ms short time segments of speech and followed the sequence of steps involving preprocessing, FFT spectrum and Mel-spectrum to compute the Cepstral coefficients. The results were reported on the KING database with 16 speakers taken from a total set of 51 male speakers in the database [Reynolds, 1995]. Each speaker had 10 conversations of approximately 45 seconds of speech, each recorded during 10 separate sessions. Speaker identification performance with GMM was investigated with varying component densities of 8, 16 and 32. Bayesian classifier was used to determine the unknown speaker. A maximum speaker identification of 94.5 ± 1.8 % was obtained with 5 seconds of clean test speech utterances. It was observed that good identification results could be obtained by increasing the number of component densities used by GMM model and by increasing the population size of the data base.

Principal Component Analysis (PCA) and Independent Component Analysis (ICA) have been widely used in image processing, especially in face recognition, identification and tracing. However, their application in the field of speech is relatively increasing these days. PCA tries to linearly transform the uncorrelated components of a high dimensional vector into a low dimensional space. Thus PCA uses only the second order cumulants for solving the recognition problem. ICA attempts to solve the problem by generalizing PCA to generate statistically independent components rather than simply transforming the uncorrelated components. Thus ICA tends to use higher order cumulants. Projecting the original feature set into smaller subspaces using PCA and ICA transforms not only reduces the dimensions of the original feature vectors but also the correlation among the elements of the signals. This consequently reduces the computational overhead involved in the subsequent processing stages thereby retaining maximal variances.

Literature shows that related work has been done on using PCA [Wanfeng, 2003], [Ding, 2001] and ICA [Ding, 2001] in speaker recognition systems. These multivariate dimensionality reduction techniques (PCA/ICA) can be sometimes applied to Mel-spectral energies [Ding, 2001] or the Mel-Cepstral feature vectors [Wanfeng, 2003] after the feature extraction phase. The correlation present among the elements of speech feature vectors obtained through MFCCs makes the dimension reduction possible and more efficient [Wanfeng, 2003]. This is because the cepstrum vector characteristics agree with the assumptions made in these algorithms (PCA/ICA) [Somervuo, 2003].

In [Wanfeng, 2003] Zhang Wanfeng *et al.* implemented a new speaker identification framework with PCA embedded into the text-independent speaker recognition system after the feature extraction phase. Their model made use of traditional MFCC feature extraction as front end and Gaussian mixture models with Bayes' decision rule as a back end for speaker identification. Spectral analysis was carried out on 16 ms short time segments of speech with an overlap of 10 ms and followed the sequence of steps involving preprocessing, FFT spectrum and Mel-spectrum to compute the Cepstral coefficients. The results were reported on the YOHO database [Campbell, 1995] with 50 speakers taken from a total set of 138 speakers in the database. Speaker identification performance with GMM was investigated with varying component densities of 8, 16 and 32 and 64. Bayesian classifier was used to determine the unknown speaker. A maximum speaker identification of 99.2 % was obtained with clean test speech utterances. Another database called PHONE [Wanfeng, 2003] was generated by them to check the performance of the recognition system under noisy conditions. An accuracy of 86.3 % was reported using a 32 component density model. It was observed that good identification results could be obtained by embedding such multivariate algorithm like PCA after the feature extraction step.

We propose a new approach where Independent Component Analysis (ICA), a more robust dimensionality reduction method when compared to Principal Component Analysis (PCA) is embedded into the text-independent speaker recognition system. We compare the performance results of embedding Principal Component Analysis (PCA) and Independent Component Analysis (ICA) in a text-independent speaker recognition system.

CHAPTER 4. SYSTEM DESCRIPTION AND RESULTS

This chapter outlines the text-independent speaker recognition system designed in this research, including the training and testing conditions implemented by the system for identifying speakers. This also includes description of the key operating parameters used by different components of our speaker recognition model. We propose a new framework of text-independent speaker recognition system with dimensionality reduction tools such as Principal Component Analysis (PCA) and Independent Component Analysis (ICA) embedded into the traditional speaker recognition system of Fig. 3.1. We evaluated the robustness of our new text-independent speaker recognition system by contaminating input speech signal with various kinds of noise occurring in real world scenario. In our model each time an input speech signal is read, different kinds of noisy signals are generated. We have conducted several test runs to evaluate the performance and measure the robustness of the speaker recognizer using PCA and ICA under different experimental conditions (Table 4.1). Initially, speech is transformed into frame-based acoustic features by means of signal processing methods. Further processing incorporated the use of an appropriate model for extracting Mel-frequency Cepstral features using frame based Cepstral analysis. Dimensionality reduction algorithms such as PCA and ICA are applied to the MFCC coefficients to obtain the linear transformations of the data. Dimensionally reduced data is fed to a Gaussian Mixture Model to train the model. Probability Distribution Functions (PDFs) are computed using Bayes' decision rule and the unknown speaker is identified as one with the largest PDF.

4.1 Speaker Recognition Model using Principal Component Analysis (PCA) and Independent Component Analysis (ICA)

Fig. 4.1 represents the block diagram of the proposed text-independent speaker recognition framework with PCA and ICA embedded in the system after feature extraction.

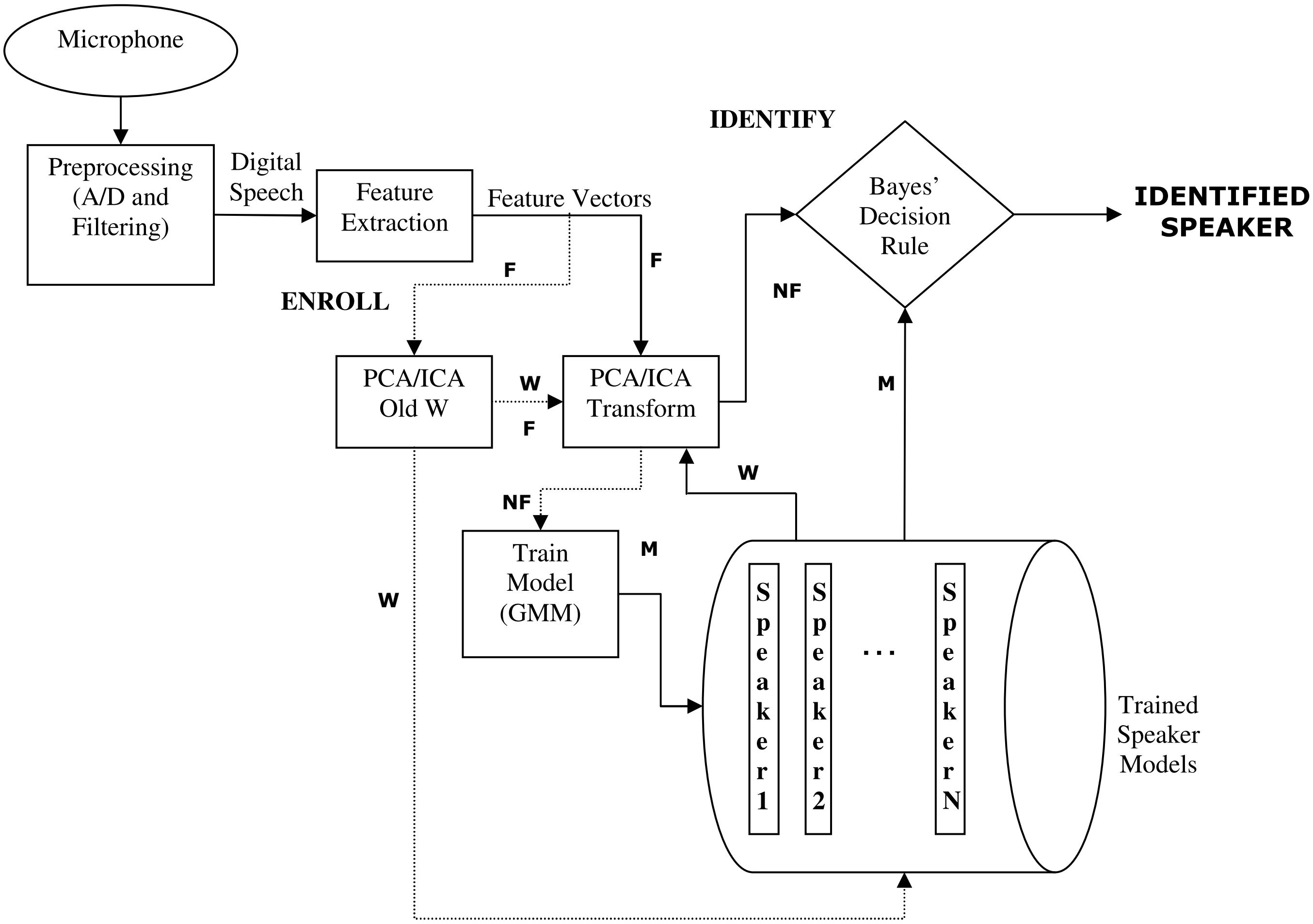

Fig. 4.1 Proposed text-independent speaker recognition system

In this block diagram F represents feature vectors, NF represents new feature vectors after application of PCA and ICA transforms, W is the transformation matrix and

M represents the trained model. The dotted lines in Fig. 4.1 represent the ENROLL phase. The solid line after ENROLL phase is the IDENTIFY phase.

Preprocessing and feature extraction constitute front end processing of our text-independent speaker recognition model. This part of the model is responsible for "signal processing" that involves converting raw speech into digitized form, filtering it and dividing it into frames through preprocessing and converting it into feature vectors using feature extraction. Most commonly extracted features are the Cepstral coefficients [Gish, 1994]. The proposed model computes Cepstral coefficients because it is believed to be the best choice for representing short term spectrum [Gish, 1994], [Zhu, 1994].

4.1.1 Preprocessing

A/D conversion

In real time scenario speech signals may come from sources like telephone or microphone. Analog to digital converter is used to produce digitized speech signal $s(n)$ from a sound pressure wave. Practically we implemented our model using speech signals from YOHO database which is considered to be noise free, collected by ITT Defense Communication Division [Campbell, 1995].

Corpus

YOHO corpus has a total of 138 speakers (106 males and 32 females) [Campbell, 1995]. There were four ENROLL sessions and ten VERIFY sessions. For each speaker there were 24 phrases in each ENROLL session with a total of 96 phrases and 4 phrases in each VERIFY session with a total of 40 phrases. The corpus was composed of combination lock phrases with each phrase being a combination of three doublets e.g. "twenty-six", "eighty-one", "fifty-seven". All the sessions were recorded using a high quality telephone handset in a noise free office environment and were sampled at 8 kHz. We used first 10 speakers from the YOHO corpus with all the ENROLL sessions being used for training the speaker models and all the VERIFY sessions for identifying the speakers. YOHO corpus on CD ROM is available from the Linguistic Data Consortium [Reynolds, 1995] for research and development purposes.

Setting up Train and Test Conditions

Speaker recognition systems of today yield high accuracy rate in clean environments when noise strength is considerably low or can be neglected. But when the speech signal is distorted by acoustic environmental influences such as noise or background speech, the results deteriorate significantly [Lockwood, 2001]. There are certain regions in speech signals that contain relatively high information content whose emphasis leads to increase in perceived intelligibility. Addition of background noise or effects such as echo or reverberations, when a person speaks, results in various changes of vocal tract characteristics. This affects many factors such as amplitude of the speech signals, pitch, formant frequencies, intelligibility, high frequency to low frequency energy ratios, and the duration of the speech signal. As a result, these variations in speaker's voice modify the articulations and degrade the auditory feedback by excess levels of noise. This phenomenon is known as *Lombard Effect* [Junqua, 1993]. A speaker recognition system can be called a noise robust system if its performance is independent of environmental disturbances. To make our model robust to different environmental conditions, we generated noisy signals by adding various types of noises to the input speech signals. The research by [Wanfeng, 2003] also gives an implementation of similar speaker recognition model with just PCA embedded into their system under noisy conditions. We therefore trained the database with clean and noisy signals. Three sets of experiments were conducted.

Training with Clean Signals

The first set of experiment uses only clean test signals. A sufficiently clean signal has a value of signal to noise ratio (SNR) at which the features of the speech signal are not masked by the presence of noise.

Training with Noisy Signals

Noises are of different types [Liria, 2003], [Kleinschmidt, 2002] and [Zhao, 2000].

Additive noise

Additive noise comes from sources surrounding the speaker of interest, going about their every day activities. Such types of noises are unpredictable, uncontrollable and changing constantly. Many sophisticated techniques have been designed to model such noises.

Convolutive

A second type of noise is multiplicative in nature and is called *convolutive* noise. This results from analog transmission channels through which the acoustic signals travel [Liria, 2003]. Convolutive noises may also occur due to the modification of the signal characteristics by the acoustics of physical structures surrounding the speaker thereby reflecting it with distortion and delay such as echoes.

The second and third set of experiments deal with noisy test signals with additive white Gaussian noise and echo added to the clean signals.

Additive White Gaussian Noise

Additive White Gaussian Noise (AWGN) [Jacobsen, 2003] is a stationary random process with a frequency spectrum that is continuous and uniform over a specified frequency band. AWGN is described as a process which has a Gaussian probability density function and a white power spectral density for all the frequency values and can be added linearly to whatever signal is being analyzed. Signal to noise ratio is defined as the ratio of the amplitude of desired signal to the amplitude of noise signal at a particular point of time. Additive White Gaussian Noise (AWGN) is added to the clean signal at SNR of 35 dB during the training process to evaluate the robustness of the ASR system in noisy environment. Figures 4.2 and 4.3 represent clean speech signal and signal with AWGN respectively.

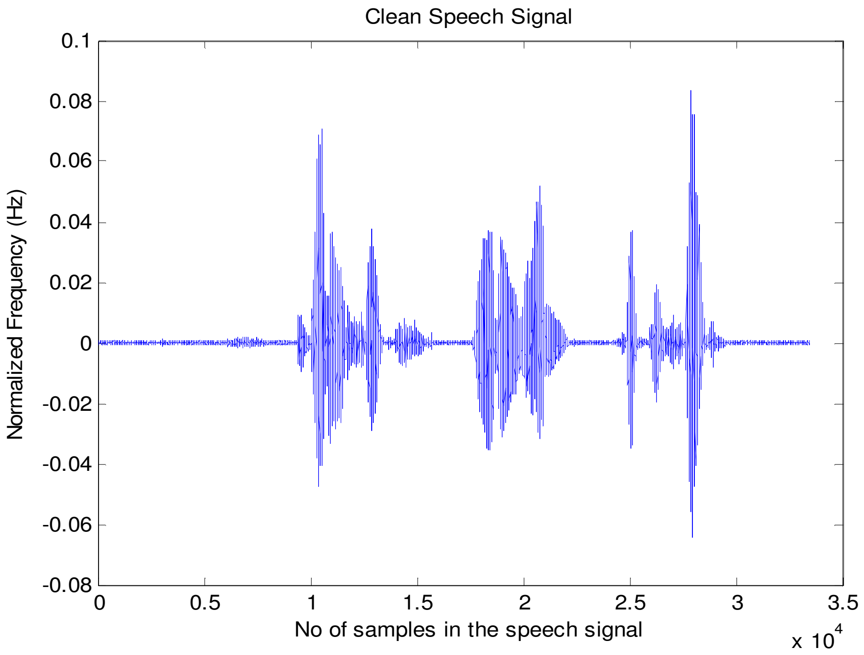

Fig. 4.2 Clean speech signal (of the first speaker from YOHO corpus)

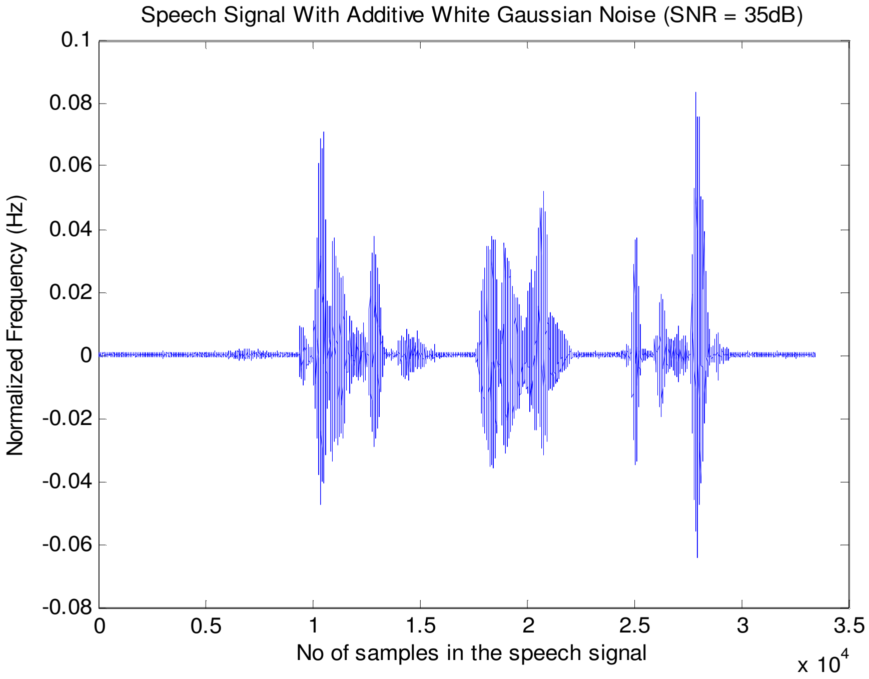

Fig. 4.3 Speech signal (of the first speaker from YOHO corpus) with AWGN

During the testing process, noisy test signals were generated by adding AWGN at four different SNRs: 30dB, 20dB, 10dB, and 0dB trying to practically simulate different kinds of background noise present in the environment.

Echo

Echo effect is created when a speech signal is bounced off by some surrounding objects. As a result, the signal arrives few milliseconds later. It is a type of multiplicative or convolutive noise that can degrade the quality of the speech signals.

Echo effect is a simple digital audio processing effect that can be simulated using a simple echo filter that has the following difference equation [Caputi, 1998]:

$$y(n) = x(n) + ax(n-D) \qquad (4.1)$$

The transfer function $H(z)$ and the impulse response $h(n)$ of this filter are given as [Caputi, 1998]

$$H(z) = 1 + az^{-D} \tag{4.2}$$

$$h(n) = \delta(n) + a\delta(n - D) \tag{4.3}$$

D is delay in seconds and a the coefficient of the filter is taken to be 0.5 since it is the measure of the reflection losses such that $|a| \leq 1$ [Caputi, 1998]. Fig. 4.4 represents an echo affected speech signal with a delay of 0.2 ms.

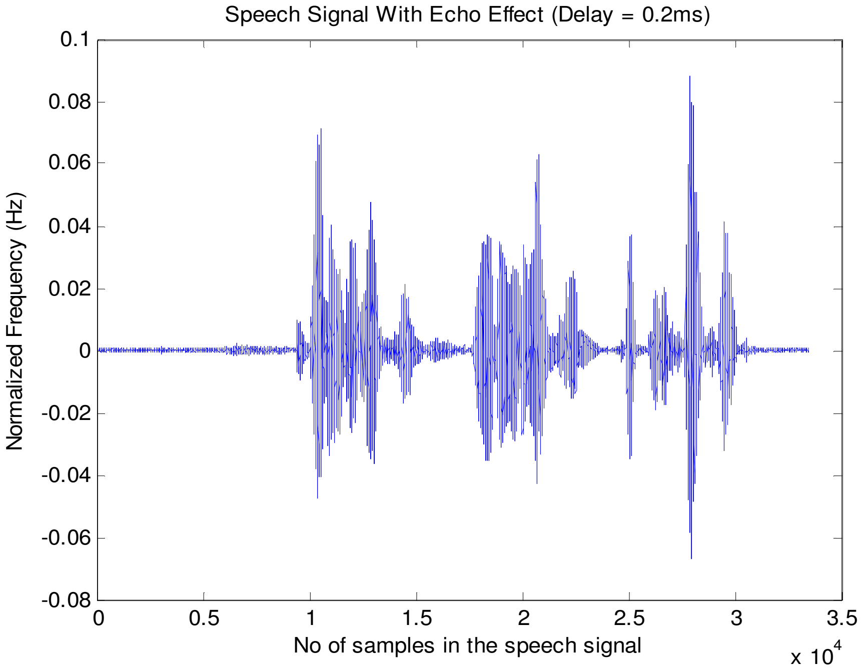

Fig. 4.4 Speech Signal (of the first speaker from YOHO corpus) with Echo

Echo can cause undesirable detection effects. The signal quality suffers or diminishes as the delay increases with increasing echo effect. Speakers with their speech uttered from an outgoing prompt affected with echo, for example, may be incorrectly recognized as imposters. Echo effect can be greatly reduced by integrating echo cancellation and noise reduction techniques into the devices. This would prevent spoken utterances from being echoed and would increase the efficiency of Automatic Speaker Recognition systems. In the third set, echo affected test signals were generated with varying delays. Performance of the speaker recognition system was tested at four different values of delay: 0.25 ms, 0.3 ms, 0.35 ms and 0.4 ms. Several test runs involved in the experiments are listed in Table 4.1. Table 4.2 represents different noises used in this research. Table 4.3 gives a representation of the input speech signals.

Table 4.1 Experimental conditions used for evaluating the performance of the proposed model

Experiments	Training Database	Train Conditions	Test Database	Test Conditions	
				SNR (dB)	Delay (ms)
Experiment - 1	Clean signals	No noise	Clean signals	-	-
Experiment - 2	AWGN added signals	35 dB	AWGN added signals	30	-
				20	
				10	
				0	
Experiment - 3	Echo added signals	0.2 ms	Echo added signals	-	0.25
					0.30
					0.35
					0.40

Table 4.2 Representation of noise used

Type of Noise	Symbol
Additive white Gaussian noise	N_1
Echo Effect	N_2

Table 4.3 Representation of input signals

Mathematical Representation	Signal Representation	Description
x_{11}	**SIGNAL 1**	Clean speech signal of **Speaker 1**
$x_{12} = x_{11} + N_1$, $N_1 : AWG$	**SIGNAL 2**	Signal obtained by adding AWGN to the clean speech signal of **Speaker 1**
$x_{13} = x_{11} + N_2$, $N_2 : ECHO$	**SIGNAL 3**	Signal obtained by adding echo effect to the clean speech signal of **Speaker 1**

Pre-emphasis Filtering

Pre-emphasis filtering is a process of emphasizing most important frequency components of a speech signal. This can be implemented by a simple high order finite response filter (FIR) with a difference equation given below [Picone, 1993].

$$H(z) = 1 + \alpha z^{-1} \tag{4.1}$$

Each input signal is pre-emphasized using this equation. α is the pre-emphasis coefficient and its optimal value is taken close to -1.0 about -0.95 since this allows an efficient implementation in fixed point hardware systems [Picone, 1993]. This results in boosting up of the signal spectrum towards higher frequencies and reducing its susceptibility to finite precision effects at a later stage [Picone, 1993]

Frame Blocking

The short-time representation of signals was computed on frames. The input speech signal was divided into frames by the frame blocker to carry out frame based Cepstral analysis. $N(= f_s * t_s)$ the length of each frame is also the number of samples contained in each frame (where f_s is the sampling frequency and t_s is the sampling rate) and M is the overlap or offset between the adjacent frames are the two important parameters in this phase. With the sampling frequency of 8 kHz we extracted frames of length 18.60 ms which overlap by 10 ms, which corresponds to: N (48) samples and M (80) samples. We have chosen these values because the most important spectral information unique to a person is contained in short time spectrum of the speech signal [Gish, 1994].

4.1.2 Feature Extraction and Parameter Estimation

Speech is intrinsically a highly non-stationary signal. Therefore, speech analysis must always be carried out on short segments across which the speech signal is assumed to be stationary. Typically, feature extraction is performed on 18.6 ms windows with 10 ms shift between two consecutive windows as given above. The experimental / analytical values selected in this work can be justified by the fact that practically only the first 20-30 milliseconds and the last 10-20 milliseconds of sound contains vital information [Currie, 2003]. This is due to the non stationary nature of the speech signal due to which it is assumed to be stationary for only a small frame of time period [Gish, 1994].

Speech parameterization can be obtained by computing Cepstral coefficients from either Mel-frequency filterbank (MFCC) or Linear Prediction models. In this thesis, we investigate the use of MFCC feature set for speaker identification since these features have proven to be more robust for speech recognition [Reynolds, 1995].

Specifications

This section gives a brief overview of extracting the required features from all frames of speech obtained from preprocessing step together with the specifications of the parameters used to model our text-independent speaker recognition system. Feature extraction involves the following steps.

Windowing

A windowing function $w(n)$ is used to taper the start and end of each frame or segment. This is done to reduce the spectral leakage caused by the discontinuities present at the ends of each framed speech. The best solution is to consider a hamming window defined as [Picone, 1993]

$$w(n) = 0.54 - 0.46\cos\left(\frac{2\pi n}{N-1}\right) \tag{4.2}$$

The window is applied to each speech segment through

$$x(n) \Rightarrow x_i(n)w(n), \ n = 0,1,....,N-1 \tag{4.3}$$

Application of hamming window also aims at improving the accuracy of the spectral estimate of the input signal [Picone, 1993].

FFT Spectrum

Each windowed frame is converted into power spectrum $s(k)$ by applying Fast Fourier Transform. We implemented 256-point FFT for computing the spectrum of signal [Davis, 1980]. The number of points used in FFT is taken as the power of 2 and must be greater than the frame size. The number of points in FFT also depends on the FFT length. The power spectrum of half the number of coefficients is preserved.

Mel-Spectrum

The resulting power spectrum is windowed by a set of 20 triangular filters equally spaced filterbank generated prior to pre-emphasis to obtain Mel-Spectrum$\left(\tilde{S}\right)$. This is done to further simplify the spectrum without any significant loss of data. Experimental results obtained on human hearing determine the bandwidths and center frequencies of a Mel-Filterbank.

Mel-scale Formulation

Mel-scale formulation given below is implemented to convert the normal frequencies into Mel-frequencies [Klabbers, 2001].

$$fmel = 2595\log_{10}\left(1 + \frac{f}{700}\right) \tag{4.4}$$

where $fmel$ is the frequency in Mel-scale associated with actual frequency f .

Mel-scale frequency representation of speech signal is the most popular way of extracting the feature vectors from the speech signal because it attempts to mimic the human ear with respect to how it perceives the frequencies of incoming sound and how they are resolved [Umesh, 2002].

Mel-Filter bank

A filtering analysis of speech determines the amount of energy in specific frequency regions, therefore resulting in some kind of spectral analysis [Kent, 1992]. Filter bank based on Mel-scale frequency representation of speech signal gives good estimates of its spectral envelop. This tends to separate the frequency bandwidth of the signal into number of frequency bands, where the energy of the signal can be measured. Thus a Mel-Filterbank with 20 triangular band-pass filters [Davis, 1980] equally spaced is constructed with 50% overlap. It also smoothes out the noise and pitch harmonics present in the speech signal.

Cepstral Coefficients

Since the vocal tract is smooth, the energy levels in adjacent bands tend to be correlated. The discrete cosine transformation applied to the transformed Mel-frequency coefficients produces a set of Cepstrum coefficients $c(i)$. Prior to computing Cepstral coefficients the Mel-spectrum $\tilde{S}$ is usually represented on a log scale. The shape of the log power spectrum is preserved independent of the input signal strength due to the property of log function. Thus Cepstral based analysis converts logarithmic-scaled energies, largely un-correlated in the energy levels, tend to be correlated in the adjacent bands [Picone, 1993].

$$c(i) = \sqrt{\frac{2}{Nof}} \sum_{m=1}^{Nof} \log\left(\tilde{S}(m)\right) * \cos\left(\frac{\pi}{Nof} i(m - 0.5)\right),$$

$$i = 0,1,....C - 1 \tag{4.5}$$

where Nof is the number of filters, $(\tilde{S})$ is the Mel-spectrum and $c(i)$ are the Cepstral coefficients and C is the number of Mel-Cepstral coefficients.

This results in a signal in the Cepstral domain with a que-frequency peak corresponding to the pitch of the signal and a number of formants representing low que-frequency peaks. Since most of the signal information is represented by the first few MFCC coefficients, the system can be made robust by extracting only those coefficients ignoring or truncating higher order DCT components. Traditional MFCC systems use only 8 to 13 Cepstral coefficients [Wang, 2000]. To increase the performance of our system we extracted 34 MFCC coefficients.

4.1.3 Training

The feature extraction and parameter estimation is thus carried out for all the signals in the ENROLL and VERIFY sessions. At this point of design we introduce a new approach of embedding dimensionality reduction algorithms like Principal Component Analysis (PCA) and Independent Component analysis (ICA). Therefore training (ENROLL) and identification (IDENTIFY) phases in the proposed model differ from that of the traditional model shown in Fig. 3.1. A similar implementation of a text-independent speaker recognition model was introduced in [Wanfeng, 2003] with only PCA embedded into the model. In the literature PCA and ICA have also been applied to the Mel-spectral energies [Ding, 2001]. We applied PCA and ICA to the extracted Cepstral coefficients because the dimension reduction is more efficient due to the correlation present among the elements of speech feature vectors obtained using MFCCs [Wanfeng, 2003]. In this research we investigate the robustness of embedding PCA and ICA into speaker text-independent speaker recognition system under clean and noisy conditions. This section presents a brief overview of Principal Component Analysis and Independent Component analysis and further continues with the implementation of ENROLL phase.

4.1.3.1 Principal Component Analysis (PCA), Independent Component Analysis (ICA) and Projection Pursuit

Automatic Speaker Recognition system is a highly complex model associated with a huge number of free parameters. Analysis of such a model is a challenging problem. Under such circumstances dimensional reduction of the data becomes a major requirement for obtaining good identification results. Principal Component Analysis (PCA) [Hotellings, 1933], [Shlens, 2003] and Independent Component Analysis (ICA) [Hyvarinen, 2001] are the two most powerful tools available for high dimensional multivariate analysis. Application of these tools to speech synthesis results in computational and conceptual simplicity.

PCA and ICA are both linear and unsupervised dimensional reduction techniques. These algorithms therefore can be implemented by simple matrix multiplications [Furui, 1992]. PCA extracts orthogonal principal components of variations by de-correlating the second order moments corresponding to low frequency property. ICA is not necessarily orthogonal but tends to make unknown linear mixtures of multi-dimensional random variables as statistically independent as possible. It also allows reduction of higher-order statistical dependencies which makes ICA perform better than PCA [Somervuo, 2003].

Another important difference between PCA and ICA is that: PCA extracts components with largest magnitudes where as ICA extracts independent components even with smaller magnitudes. This section gives a brief description and analysis of Principal Component Analysis and Independent Component Analysis algorithms.

Principal Component Analysis (PCA)

Principal Component Analysis is an approximation of Karhunen-Loeve Transform (KLT) algorithm used to extract few first eigenvectors which mostly retain the variations present in all the original variables. It is a mathematical method used to orthogonally project the features of high dimensional space into low dimensional subspace.

Principal Component Analysis exhibits three important features: (1) It is optimal in terms of mean squared error, i.e. it is a linear scheme used for compressing a set of high dimensional vectors into low dimensional vectors and then reconstructing them. (2) The parameters of the model can be directly obtained from the data by diagonalizing the covariance matrix. (3) Using PCA, operations used to compute the model parameters require only matrix multiplications reducing complexity and time consumed. In spite of all these advantages, PCA however has some shortcomings. It is a naive method used to compute the principal component direction and ends up having trouble with large number of data points and high dimensional data [Somervuo, 2003].

Principal components of the data set can be obtained by computing the covariance matrix of the data set and then finding the eigenvectors corresponding to the largest eigenvalues. Suppose there are N feature vectors given as $\{x_1, x_2, ..., x_N\}$. The mean of the feature vectors is represented by $\bar{x}$ and is calculated as [Smith, 2002]

$$\bar{x} = \frac{1}{N}\sum_{i=1}^{N} x_i \qquad (4.6)$$

The covariance matrix C is a square and symmetric matrix of order $N * N$ and can be computed as [Smith, 2002], [Shlens, 2003]

$$C = \frac{1}{N}\sum_{i-1}^{N} \tilde{x}_i \tilde{x}_i^{\,T} \text{ ,} \qquad (4.7)$$

where

$$\tilde{x}_i = x_i - \bar{x} \qquad (4.8)$$

Covariance matrix C is also observed to exhibit correlation and data dispersion. Eigenvalue decomposition of the covariance matrix results in eigenvalues and eigenvectors [Rabiner, 1993]. Eigenvectors can be computed from the following equation [Smith, 2002], [Shlens, 2003]

$$CV_k = \lambda_k V_k \text{ , } k = 0,1,..., N-1 \qquad (4.9)$$

where V_k is the k^{th} eigenvector and λ_k is the corresponding eigenvalue. Eigenvectors corresponding to M $(M < N-1)$ largest eigenvalues are selected to reduce the dimensions of the data set. The transformation or projection matrix is defined as the transpose of thus obtained eigenvector matrix and is given as [Smith, 2002], [Shlens, 2003]

$$W_{PCA} = V^T \text{ ,} \qquad (4.10)$$

where

$$V^T = V_0, V_1, ..., V_{M-1} \qquad (4.11)$$

The final step is to derive the new data set, the projection of the feature vectors on to the space formed by PCA. This is simply established by multiplying the projection matrix with the original dataset (mean adjusted data). This can be represented as [Smith, 2002]

$$NewDataSet = W_{PCA} * MeanAdjustedOriginalData \tag{4.12}$$

Independent Component Analysis (ICA)

Independent component analysis, a recently developed technique, aims at finding out linear representation of non-Gaussian data so that the components are statistically independent. ICA helps in capturing some of the essential features of data in many applications including Automatic Speaker Recognition Systems [Hyvarinen, 2001].

Estimation of the Model by ICA

Each person's voice has distinguishing properties and features which makes them unique. Air stream pumped by the lungs modifies itself to generate desired sequence of sounds every time a person tries to speak. This implies that there exist some differences in the characteristics of speech depending on the changes in the shape of the vocal tract, vibration of the vocal chords and the nasal cavity. Vocal tract can then be considered as a set of filters that change or alter a set of excitation signals.

ICA aims at extracting a set of statistically independent vectors from the matrix of training data, the Mel-frequency Cepstral feature vectors derived from the original signal. It tends to find directions of minimum mutual information. It aims at capturing certain correlations among the frequencies present in the spectral based representation of a speech signal. This is achieved by ICA in the form of linear combinations of basic filter functions specific to each person. Specific sounds are then generated by combining these functions in a statistically independent nature.

Consider a signal x_t. A set of MFCC coefficients derived from the original signal with frames placed in the columns and observations in the rows. This signal is used since it is a proper Mel-Cepstral based representation of the original signal and the data can be observed as a set of multivariate time series resulting from a hidden linear mixing process A of independent functions s [Potamitis, 2000], [Hyvarinen, 2001]. Linear combination of such sources or functions can be summarized as [Cardoso, 1996]

$$x = As \tag{4.13}$$

The problem of ICA is to determine both the excitation signal s and the scalars A and the only known component is the matrix of the MFCC coefficients of the input speech signal. s can be computed as follows [Hyvarinen, 1997]

$$s = A^{-1}x \tag{4.14}$$

Computing A is a problem and a possible solution is to consider x as a vector of observations where each observation is expressed as a linear combination of independent components. In order to estimate one of the independent components, a linear combination of x_i is chosen such that [Hyvarinen, 1997], [Michael, 2002]

$$y = w^T x = \sum_i w_i x_i \tag{4.15}$$

With respect to the condition stated in equation (4.13) and equation (4.14), the linear combination represented in equation (4.15) is a true combination of independent components if w were one of the columns of the inverse of A.

Nongaussianity

According to the central limit theorem by Hyvarinen and Oja [Hyvarinen, 2001], the sum of the independent variables has a distribution that is closer to Gaussian than the distribution of the original variables [Michael, 2002]. This concludes that the distributions of x are more Gaussian than source or excitation signal since the signal x is the weighted sum of the components of the excitation signal (equation (4.13)). Thus nongaussianity of the excitation signal enables the application of ICA to this problem and the obvious solution for finding $w = A^{-1}$ is to maximize the nongaussianity of $y_i (\approx s_i)$.

To exploit the property of nongaussianity in ICA estimation, we must have some means of quantitatively measuring this characteristic. Negentropy is one of the ways of measuring nongaussianity and its approximation can be given as [Hyvarinen, 1999]

$$J(y) = \left(E\{G(y)\} - E\{G(v)\}\right)^2 \qquad (4.16)$$

$E\{G(v)\}$ is a constant Gaussian variable with zero mean and unit variance.

$E\{G(y)\}$ is a non-quadratic function. Some commonly used functions are *Cosh, Gaussian and Kurtotis-based approximation*. We have chosen the optimal representation, the Gaussian function since it results in minimum estimation error than other approximation functions [Hyvarinen, 1999].

Preprocessing

In an automatic speaker recognition system environment, the columns of the input signal x are the smoothed Mel-spectra of the frames of speech data. Speech excitation s is the cause of the speech and activates speech features represented by A resulting in original speech frames X and using ICA for speaker recognition. Statistically independent coefficients are generated by filtering the speech with filters W known as de-mixing matrix.

Before estimating w (component of the matrix W), the input signal is preprocessed for good and accurate detection results. Preprocessing involves centering of the input speech signal. This is obtained by subtracting their mean value from the signal [Michael, 2002].

$$\hat{x} = x - E(x) \tag{4.17}$$

Whitening

The next step is to whiten the centered data. Whitening is done using eigenvalue decomposition of the covariance matrix $E\{xx^T\}$ very similar to the PCA technique [Hyvarinen, 2001]. As a result, eigenvectors and the diagonal matrix are computed from the covariance matrix. Whitening is done by multiplying the centered signal with a transformation or permutation matrix P given by $inv\left(ID^{-1/2}\right) * EE^T$, where EE is the matrix containing eigenvectors and ID is the diagonal matrix containing eigenvalues corresponding to eigenvectors in EE. Whitening is performed so that the signals are linearly transformed and hence the components of the signal become uncorrelated and possess unit variance [Hyvarinen, 2001]. Thus we obtain the following equation

$$\tilde{x} = P\hat{x} \tag{4.18}$$

Similarly, the mixing matrix A is multiplied with the transformation matrix P given as $\tilde{A} = PA$ which is orthogonal and the covariance of the whitened data equals to identity matrix. Keeping in mind that W is inverse of A and from the orthogonal property of $\tilde{A}$ i.e. $\tilde{A}^T = \tilde{A}^{-1}$ we can deduce $\tilde{W}$ such that $W = \tilde{W}P$ [Michael, 2002].

The final equation obtained after preprocessing and whitening is given as [Michael, 2002]

$$s \approx y = Wx = \tilde{W}Px \tag{4.19}$$

Fast ICA Algorithm

Fast ICA algorithm [Hyvarinen, 1999] is used to estimate w_i which constitutes the rows of $\widetilde{W}$. Since the components are considered to be statistically independent, the variance between them is high. This adds an optimization clue for solving the above problem. Therefore, we need to estimate w_i that maximizes the non-Gaussianity $J(w_i^T \widetilde{x})$ under the constraint $\|w_i\| = 1$ meaning norm equals to one. Assuming the gradient in equation (4.16) to be a Gaussian approximation, it is solved for non-Gaussianity by applying the optimization constraint. Two maximas, w_i and $-w_i$ with same non-Gaussianity are obtained for each component.

Theory of optimization states that the extrema of $E\{G(y)\}$ can be determined at the point where the gradient of the *Lagrange function* is zero (Kuhn-Tucker condition [Luenberger, 1969]). The constraint $\|w_i\| = 1$ can be written as $w^T w - 1 = 0$, and when applied to the *Lagrange function,* we obtain the following equation [Hyvarinen, 1999]

$$L(w, \lambda) = E\{G(w^T x)\} - \lambda(w^T w - 1) \tag{4.20}$$

The gradient of equation (4.20) can be obtained by differentiating it with respect to w [Hyvarinen, 1999]

$$w\ L_w^{'}(w, \lambda) = E\{xg(w^T x)\} - 2\lambda w \tag{4.21}$$

In Fast ICA algorithm, Newton's method (first introduced in [Hyvarinen, 1997]) is iteratively used to solve the equation $L_w^{'}(w, \lambda) = 0$. Each component must have one solution, therefore the optimization has to be run for one component at a time. While performing different iterations, a de-correlation technique is performed to prevent same solution from being found more than once. Newton's method is initialized by making a guess for w_i and the order in which the components are determined depends on this initial guess. Stopping criteria is set, so that the algorithm continues until this criterion is satisfied. Convergence condition can be checked by comparing w_i obtained in iteration with that obtained in the previous iteration [Michael, 2002].

The final step is to project the signals into the space created by ICA.

$$NewDataSet = W_{ICA} * MeanAdjustedOriginalData \tag{4.22}$$

where W_{ICA} is the transformation matrix obtained from Fast ICA algorithm.

ENROLL Phase

This deals with the training of the model. We implemented two ENROLL phases:

ENROLL phase with PCA

Two PCA components are added as shown in block diagram of Fig. 4.1.

a) "PCA Old W"

This function is used to acquire the transformation matrix W_{PCA} of M obtained after applying PCA to the extracted MFCC feature vectors of the speech signals from ENROLL session (YOHO database). By applying PCA we extracted $18(M)$ eigenvectors corresponding to 18 (M) largest eigenvalues and reduced the dimensions of MFCC from 34 (N) to $18(M)$. The total number of eigenvectors that can be obtained are N and M is the number of first few eigenvectors that are used to build the eigenspace. We chose M as 18 since the last eigenvectors $(N-M)$ have relatively smaller values. The output of this function i.e. the transformation matrix $W_{PCA}(34*18)$ and the feature vectors $\boldsymbol{F}(nof*34)$ are given as an input to the function "PCA Transform".

b) "PCA Transform"

This function is responsible for projecting the feature vectors ***F*** in to the eigenspace created by PCA using the equation $NF = W_{PCA} * MeanAdjustedOriginalData$. NF has a size of $18 * nof$ where nof is the number of frames in the respective signal.

The transformation matrix W_{PCA} of each of the speaker from ENROLL session (YOHO database) are stored with a corresponding unique ID (Fig. 4.1) in the trained database and the projected new feature vectors NF are input to the Gaussian mixture model component for training each speaker.

ENROLL phase with ICA

Two ICA components are added as shown in block diagram of Fig. 4.1.

a) "ICA Old W"

This function works similar to PCA but acquires the transformation matrix $W_{ICA}(18 * 34)$ obtained after applying all the steps in ICA (Preprocessing, Whitening, Fast ICA) to the extracted MFCC feature vectors of the speech signals from ENROLL session (YOHO database). This process as a whole was implemented using the FastICA package version 2.3 (published on 27.7.2004) for MATLAB developed by Jarmo Hurri [Hurri, 1998-2004].

b) "ICA Transform"

This function is responsible for projecting the feature vectors ***F*** into the space created by ICA using the equation $NF = W_{ICA} * MeanAdjustedOriginalData$. NF is a matrix of size $18 * nof$ where nof is the number of frames in the respective signal.

The transformation matrix W_{ICA} of each of the speaker from ENROLL session (YOHO database) are stored with a corresponding unique ID (Fig. 4.1) in the trained database and projected new feature vectors NF are input to the Gaussian mixture model component for training each speaker and representing the speaker identities.

4.1.3.2 Gaussian Mixture Models (GMM)

Literature shows that probabilistic models like GMM for have yielded better performance results for training both text-dependent and text-independent speaker recognition applications [Reynolds, 1995]. Due to the probabilistic property of a GMM, it can also be applied to speaker recognition applications in the presence of different noises increasing the channel robustness [Reynolds, 1995] and therefore more suited to this research.

Using a GMM model, for speaker identification, a group of $S:\ 1,2,.....,S$ speakers can be represented by their unique model parameters $\lambda_1, \lambda_2, ..., \lambda_S$. Identity of each speaker λ can be represented as a combination of three parameters: p_i (mixture weights for $i = 1,...,M$ where M is the number of component densities), $\overline{\mu}_i$: (mean vector with D- dimensional normal distribution) and $\sum_i$ (covariance matrix). Collectively λ is represented as $\lambda = \{p_i, \overline{\mu}_i, \sum_i\}$ for $i = 1,...,M$. We investigated the performance of the system by choosing the value of M to be 32. [Reynolds, 1995] and [Wanfeng, 2003] have implemented GMM for training text-independent speaker recognition systems with different values of M and have found that good identification results are obtained with greater values of M.

Depending upon the choice of covariance matrix, GMM can take different forms. The covariance matrix can be classified into three different types; (1) Nodal Covariance: one covariance matrix per Gaussian component, (2) Grand Covariance: one covariance matrix for all Gaussian components in a speaker model or (3) Global Covariance: a single covariance matrix shared by all the speaker models. In addition, the covariance matrix can also be full or diagonal. In this thesis, nodal and diagonal covariance matrices are primarily used for speaker modeling. The parameters of a GMM model were estimated using Expectation-Maximization (EM) algorithm.

4.1.4 Identification using Bayes' decision rule

The goal of a speaker recognition system is to identify the unknown speaker from a group of known speakers.

IDENTIFY Phase

Two IDENTIFY phases were implemented one with PCA and second one with ICA corresponding to two ENROLL phases PCA and ICA respectively. During the identification phase the feature extraction method similar to that used in ENROLL process was carried out with the all test signals (clean and noisy). 34 MFCC feature vectors were extracted from each of the test utterance.

IDENTIFY Phase with PCA

The extracted feature vectors from each test speaker were applied to the function "PCA Transform" (Fig. 4.1) and were projected into the eigenspace created by the associated speaker with unique speaker ID. This was done by calling the already stored projection matrix W_{PCA} associated with that particular ID from the trained speaker models. The new feature vectors of the test utterances and the trained models were fed to a suitable decision rule and the corresponding test speaker was determined.

IDENTIFY Phase with ICA

A similar process was implemented in this phase using ICA. The extracted feature vectors from each test speaker were applied to the function "ICA Transform" (Fig. 4.1) and were projected into the space of ICA created by the associated speaker with unique speaker ID. This uses the stored W_{ICA} from the trained speaker model. The new feature vectors of the test utterances and the trained models were fed to a suitable decision rule and the corresponding test speaker was determined.

Bayes' Decision rule

Bayesian classifier is stochastic based classifier that computes probability distribution functions rather than computing distances to average features as in template models. Bayes classifier is the best choice for identification applications which employ large group of data sets [Domingos, 1997].

$p(i \mid \bar{x}_t, \lambda)$ is called *a posteriori* probability for an acoustic class i and is defined by the following equation

$$p(i \mid \bar{x}_t, \lambda) = \frac{p_i b_i(\bar{x}_t)}{\sum_{k=1}^{M} p_k b_k(\bar{x}_t)} \tag{4.23}$$

For a given observation sequence the main goal is to find the speaker model that has the maximum *a posteriori* probability represented as [Reynolds, 1995]

$$\hat{S} = \arg \max_{1 \le k \le S} \Pr(\lambda_k \mid X) \tag{4.24}$$

$$\hat{S} = \arg \max_{1 \le k \le S} \frac{(X \mid \lambda_k) \Pr(\lambda_k)}{p(X)} \tag{4.25}$$

Equation 4.25 is obtained due to Bayes' rule. The above classification rule can be further simplified by (i) assuming equally likely speakers (equivalent to $\Pr(\lambda_k) = 1/S$) and (ii) observing that $p(X)$ is same for all the speaker models. Therefore equation 4.24 reduces to

$$\hat{S} = \arg\max_{1 \le k \le S} p(X \mid \lambda_k) \tag{4.26}$$

The speaker identification system finally computes $\hat{S} = \arg\max_{1 \le k \le S} \sum_{t=1}^{T} \log p(\bar{x}_t \mid \lambda_k)$ using the logarithms and the independence between the observations.

4.2 Experimental Results

Three sets of experiments were carried out using experimental conditions listed in Table 4.1. The results are reported on a 10 speaker subset taken from YOHO database. Each speaker had 96 utterances in ENROLL session and 40 utterances in VERIFY session. We computed and tabulated the average percentages of recognizing the input VERIFY signals in these runs.

Experiment – 1

This experiment involves Clean Train and Clean Test signals. Results show that recognition rates obtained using ICA outperformed that of PCA.

Table 4.4 Performance using PCA and ICA with Clean Test Signals

(Average Percentages)

Accurate Recognition Rate (%)		
Feature	**PCA**	**ICA**
M: 32	90.50%	94.12%

M: Number of GMM components

A similar framework was implemented by Zhang Wanfeng *et al.* [Wanfeng, 2003] using only PCA. The results were reported on the YOHO database with different number of speakers. It also involved implementation of speaker recognition model using PCA under adverse conditions. Noise came from several sources like people in the background, noise from the adjoining rooms, etc. Only one set of experiment was performed using 16 GMM components and the results were reported on a noisy database "PHONE" developed by them. They achieved a recognition rate of 77%.

We trained our speaker recognition model under different noisy conditions occurring in real world scenario. Additive white Gaussian noise (AWGN) and echo affected signals were generated. The model was trained dynamically. Each time a speech signal was read, AWGN at different values of SNR and echo with varying delays were added to the clean speech signals from YOHO database. Experiment 2 and 3 report the identification results with PCA and ICA for different train and test conditions.

Experiment – 2

This experiment involves only AWGN affected train signals at a particular SNR of 35 dB and AWGN affected test signals with varying SNR's. From the table, we could find that as signal to noise ratio increases, the recognition rate also increases. The performance of speaker recognition is improved using ICA when compared to PCA even in noisy conditions. By this we show that ICA is more robust than PCA for text-independent speaker recognition under adverse conditions.

Table 4.5 Performance of PCA and ICA with Variation in SNR of the test signals (Average Percentages)

Train: 35dB, M: 32	Test values of SNR in dB			
Transformations	0	10	20	30
PCA	40.33%	62.41%	76.70%	87.00%
ICA	51.00%	70.00%	85.50%	89.60%

M: Number of GMM components

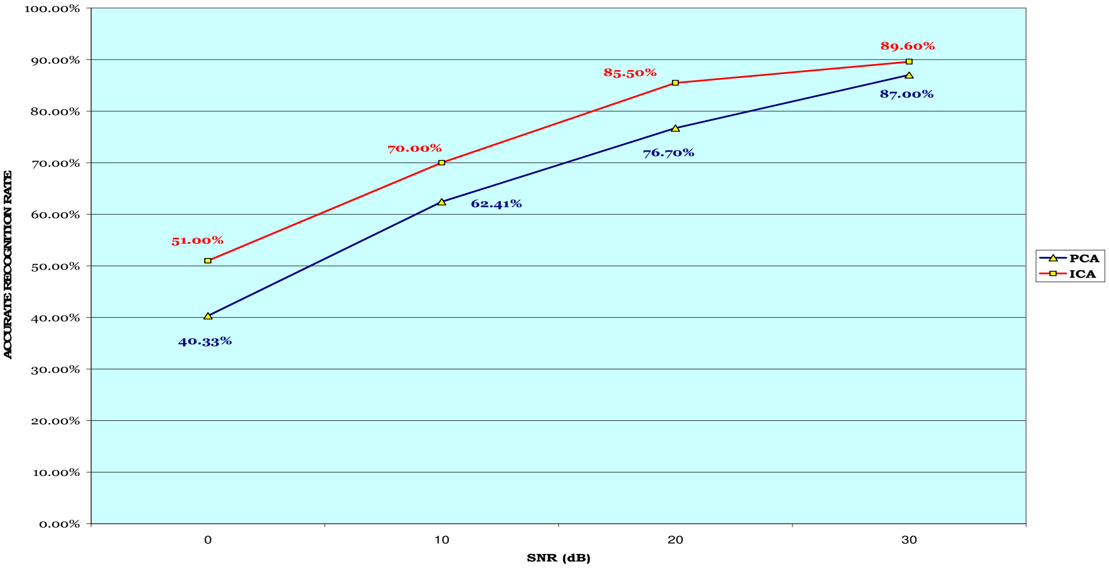

Fig. 4.5 Performance of PCA and ICA with additive white Gaussian noise added to the test signals

Experiment – 3

This set of experiment includes echo affected train and test signals. Train signals are generated with a specific delay of 0.2 ms and test signals with varying delay. The trend we observe from Table 4.7 is that as the delay of the echo affected signal increases, there is a lot of variation in the speech signal, thereby reducing the recognition rate. Again, ICA was more robust than PCA.

Table 4.6 Performance of PCA and ICA with Variation in Echo length or Delay of the test signals (Average Percentages)

Train: 0.2ms, M: 32	Test values of delay in ms			
Transformations	**0.25**	**0.30**	**0.35**	**0.40**
PCA	88.00%	80.33%	75.45%	72.43%
ICA	90.00%	85.00%	78.33%	75.00%

M: Number of GMM components

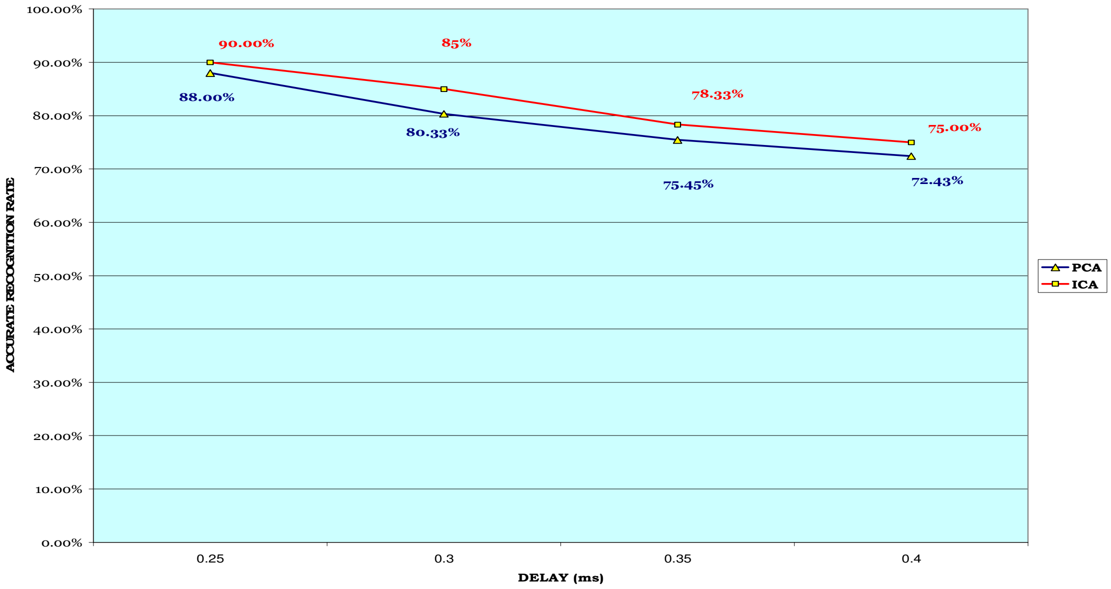

Fig. 4.6 Performance of PCA and ICA with Echo added to the test signals

The proposed model therefore evaluates the performance of a new text-independent speaker recognition model with PCA and ICA embedded into it after the feature extraction step and compares the robustness of PCA and ICA transforms under multi-environment training scenario.

CHAPTER 5. CONCLUSIONS AND FUTURE WORK

In this research we propose an approach focusing on multi-environment training in concatenation with application of dimensionality reduction algorithms for improving the recognition rate of a text-independent speaker recognition system. To evaluate the robustness of our model, we developed a scenario where in different types of noise (additive and convolutive) occurring in real world were added to the clean speech signals. The text-independent speaker recognition system was designed based on Mel-Cepstral analysis. The proposed model uses a new framework where PCA and ICA were embedded after the Mel-Cepstral feature extraction process. Mel-Scaled FFT analysis described in this research work takes into account the behavior and psychoacoustic characteristics of human auditory system and is thus a robust technique. Experiments were performed on a subset of 10 speakers (including all the ENROLL and VERIFY sessions for each of the speaker) from YOHO corpus with Gaussian mixture model and Bayes' classifier to evaluate the performance of the designed system. MATLAB code was written to implement the approach. We show that by embedding Independent Component Analysis, recognition rates of a text-independent speaker recognition system can be improved considerably. The recognition accuracy rate obtained using PCA was 90% where as ICA was 94% for clean signals. Though PCA gains over conventional methods, this approach fails to achieve the lowest-possible dimensions because of the bases being generic and not able to un-correlate the data under consideration optimally [Potamitis, 2000]. For noisy signals the recognition accuracy rates ranged from 40.33 % to 87 % (PCA) and 51% to 89.6% (ICA) for increasing values of SNR and 72.43 % to 88% (PCA) and 75 % to 90 % (ICA) were for decreasing values of delay. These values are presented in Tables 4.4, 4.6 and 4.7. Thus the accuracy percentage rates of identifying the test speakers under adverse conditions using ICA were more than that obtained using PCA. It is observed that ICA outperformed PCA. This is because PCA is capable of removing only the 2nd order dependencies between the feature vectors where as ICA also removes higher order dependencies [Somervuo, 2003]. Independent components extracted by ICA method contain most of the important data in the speech thus ICA tacitly enables the exploitation of the discriminating features of the speech data and hence very popular in many applications of speaker recognition systems.

Results of identification using feature transformations can be improved by exploiting more detailed acoustic models. Future work is being concentrated on ways and methods of concatenating different algorithms such as Principal Component Analysis (PCA), Linear Discriminant Analysis (LDA) and Independent Component Analysis (ICA) aimed at increasing the accuracy rates of speaker recognition systems particularly text-independent speaker recognition systems. More class specific cues from the input signals can be detected by integrating all these feature transformations. Many researchers are also currently working on increasing the robustness of automatic speaker recognition systems in the presence of increased noise. Future trends may also include the enhancement of the recognition systems by taking into account all other parameters such as reverberations other than noise or echo affecting the system. Work may also focus on evaluating the performance of text-independent speaker recognition systems using different classifiers such as hidden markov models (HMM) and neural networks (NN) to improve the identification results.

REFERENCES

[Bashir, 2003] Bashir, F. I., "Content Based Indexing and Retrieval of Audio Data Using PCA for Dimensionality Reduction," Department of Electrical and Computer Engineering, UIC, Apr. 2003.

[Brummer, 1997] Brummer, J. N. L., and Strydom, L. R., "An Euclidean distance measure between covariance matrices of speech Cepstra for text-independent speaker recognition," *Proceedings of the 1997 South African Symposium on Communications and Signal Processing (COMSIG),* pp. 167-172, Sept. 1997.

[Campbell, 1995] Campbell, J. P., "Testing with the YOHO CD-ROM voice verification corpus," *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP),* vol. 1, pp. 341-344, May. 1995.

[Campbell, 1997] Campbell, J. P., "Speaker recognition: a tutorial," *Proceedings of IEEE*, vol. 85, no. 9, pp. 1437-1462, Sept. 1997.

[Caputi, 1998] Caputi, M. J., "Developing Real-Time Digital Audio Effects for Electric Guitar in an Introductory Digital Signal Processing Class," *IEEE transactions on education*, vol. 41, no. 4, pp. 341-341, Nov. 1998.

[Cardoso, 1996] Cardoso, J. -F., and Laheld, B. H., "Equivariant adaptive source separation," *IEEE Transactions on Signal Processing,* vol. 44, no. 12, pp. 3017-3030, Dec. 1996.

[Currie, 2003] Currie, D., "Shedding Some Light on Voice Authentication," Sans Institute, 2003.

[Davis, 1980] Davis, S., and Mermelstein, P., "Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences," *IEEE*

Transactions on Acoustics, Speech, and Signal Processing (ICASSP), vol. 28, no. 4, pp. 357-366, Aug. 1980.

[Dempster, 1977] Dempster, A., Laird, N., and Rubin, D., "Maximum likelihood from incomplete data via the EM algorithm," *Journal of the Royal Statistical Society,* vol. 39, pp. 1-38, 1977.

[Ding, 2001] Ding, P., and Zhang, L., "Speaker Recognition Using Principal Component Analysis," *8th International Conference on Neural Information Processing,* 2001.

[Ding, 2001] Ding, P., Kang, X., and Zhang, L., "Personal Recognition Using ICA," *8th International Conference on Neural Information Processing,* 2001.

[Domingos, 1997] Domingos, P., and Pazzani, M., "On the optimality of the simple Bayesian classifier under zero-one loss," *Machine Learning*, vol. 29, pp. 103-130, 1997.

[Douglas, 2000] Douglas, A. R., and Larry, P. H., "Automatic Speaker Recognition Recent Progress, Current Applications, and Future Trends," *Computers and Speech Symposium*, Feb. 2000.

[Erell, 1993] Erell, A., and Weintraub, M., "Filter-bank-energy estimation using mixture and Markov models recognition of noisy speech," *IEEE Transactions on Speech and Audio Processing,* vol. 1, no. 1, Jan. 1993.

[Fant, 1973] Fant, G., "*Speech Sounds and Features,"* Cambridge, MA: MIT Press, 1973.

[Flanagan, 1972] Flanagan, J. L., "*Speech Analysis, Synthesis and Perception,"* New York: Springer, 1972.

[Furui, 1981] Furui, S., "Cepstral analysis technique for automatic speaker verification," *IEEE Transactions on Acoustics, Speech, and Signal Processing (ICASSP),* vol. 29, pp. 254-272, no. 2, Apr. 1981.

[Furui, 1992] Furui, S., “Towards robust speech recognition under adverse conditions,” *Proceedings of the ESCA Workshop in Speech Processing under Adverse Conditions,* pp. 31-41, Nov. 1992.

[Furui, 1994] Furui, S., “An overview of speaker recognition technology,” *Proceedings of ESCA Workshop on Automatic Speaker Recognition Identification and Verification*, pp. 1-9, Apr. 1994.

[Gish, 1986] Gish, H., Krasner, M., Russell, W., and Wolf, J., “Methods and experiments for text-independent speaker recognition over telephone channels,” *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP),* vol. 11, pp. 865-868, Apr. 1986.

[Gish, 1994] Gish, H., and Schmidt, M., “Text-independent speaker identification,” *IEEE Signal Processing Magazine*, vol. 11, no. 4, pp. 18-32, Oct. 1994.

[Hotellings, 1933] Hotellings, H., “Analysis of a complex of statistical variables into principle components,” *Journal of Educational Psychology,* vol. 24, pp. 417-441, 498-520, 1933.

[Huang, 2002] Huang, H. -J., and Hsu, C. -N., “Bayesian classification for data from the same unknown class,” *IEEE Transactions on Systems, Man and Cybernetics*, vol. 32, no. 2, pp. 137-145, Apr. 2002.

[Hurri, 1998-2004] Hurri, J., Gävert, H., Särelä, J., and Hyvärinen, A., “FastICA package for MATLAB,” Laboratory of Information and Computer Science in the Helsinki University of Technology, 1998-2004.

[Hyvarinen, 1997] Hyvarinen, A., “A family of fixed-point algorithms for independent component analysis,” *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP),* vol. 5, pp. 3917-3920, Apr. 1997.

[Hyvarinen, 1999] Hyvarinen, A., “Fast and Robust Fixed-Point Algorithms for Independent Component Analysis,” *Proceedings of the IEEE Transactions on Neural Networks,* vol. 10, no. 3, pp. 626-634, May. 1999.

[Hyvarinen, 2001] Hyvarinen, A., Karhunen, J., and Oja, E., *“Independent Component Analysis,”* John Wiley & Sons, 2001.

[Jacobsen, 2003] Jacobsen, J. D., “Probabilistic Speech Detection,” *Informatics and Mathematical Modeling,* DTU, 2003.

[Junqua, 1993] Junqua, J. C., “The Lombard reflex and its role on human listeners & automatic speech recognizers,” *Journal of the Acoustical Society of America*, vol. 93, no. 1, pp. 510-24, Jan. 1993.

[Kent, 1992] Kent, R. D., and Read, C., *“The Acoustic Analysis of Speech,”* San Diego: Singular Publishing Group, 1992.

[Klabbers, 2001] Klabbers and Veldhuis, “Reducing Audible Spectral Discontinuities,” *IEEE Transactions on Speech and Audio Processing*, vol. 9, no. 1, Jan. 2001.

[Kleinschmidt, 2002] Kleinschmidt, M., “Spectro-temporal Gabor features as a front end for automatic speech recognition,” *International Conference on Spoken Language Processing,* Sep. 2002.

[Liria, 2003] Liria, A., Masi, M., Muriel, J. A., Palou, M., Aichner, R., and Samuelsson, J., “Acoustic quality enhancement in mobile radio communications applications for public emergency services," *Proceedings of International Conference on Telecom I+D,* vol. 0, Nov. 2003.

[Lockwood, 1992] Lockwood, P., Boudy, J., and Blanchet, M., "Non-linear spectral subtraction (NSS) and hidden Markov models for robust speech recognition in car noise environments," *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP),* vol. 1, pp. 265-268, Mar. 1992.

[Luenberger, 1969] Luenberger, D. G., *"Optimization by Vector Space Methods,"* John Wiley & Sons, 1969.

[Matsui, 1994] Matsui, T., and Furui, S., "Comparison of text-independent speaker recognition methods using VQ-distortion and discrete/ continuous HMMs," *IEEE Transactions on Speech Audio Process,* no. 2, pp. 456-459, 1994.

[McLachuo, 2002] McLachuo, G., *"Mixture Models,"* New York: Marcel Dokker, 1998.

[Michael, 2002] Michael, V., *"Independent Component Analysis of Evoked Potentials in EEG,"* DTU, Dec. 2002.

[Molau, 2001] Molau, S., Pitz, M., Schluter, R., and Ney, H., "Computing Mel-frequency cepstral coefficients on the power spectrum," *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP),* vol. 1, pp. 73-76, May. 2001.

[Picone, 1993] Picone, J. W., "Signal modeling techniques in speech recognition," *Proceedings of IEEE*, vol. 81, no. 9, pp. 1215-1247, Sep. 1993.

[Poritz, 1982] Poritz, A., "Linear predictive hidden Markov models and the speech signal," *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP),* vol. 2, pp. 1291-1294, 1982.

[Potamitis, 2000] Potamitis, I., Fakotakis, N., and Kokkinakis, G., “Spectral and Cepstral projection bases constructed by Independent Component Analysis,” *Proceedings of the 6th International Conference on Spoken Language Processing (ICSLP),* vol. 3, pp. 63–66, Oct. 2000.

[Rabiner, 1978] Rabiner, L. R., and Schafer, R. W., *“Digital Processing of Speech Signals,”* Englewood Cliffs: Prentice Hall, 1978.

[Rabiner, 1989] Rabiner, L. R., “A tutorial on hidden Markov models and selected applications in speech recognition,” *Proceedings of IEEE,* vol. 77, no. 2, pp. 257-286, Feb. 1989.

[Rabiner, 1993] Rabiner, L., and Juang, B. H., “*Fundamentals of Speech Recognition*,” Englewood Cliffs: Prentice Hall, 1993.

[Reynolds, 1995] Reynolds, D. A., and Rose, R. C., “Robust text-independent speaker identification using Gaussian mixture speaker models,” *IEEE Transactions on Speech and Audio Processing,* vol. 3, no. 1, pp. 72-83, Jan. 1995.

[Seddik, 2004] Seddik, H., Rahmouni, A., and Sayadi, M., “Text independent speaker recognition using the Mel frequency cepstral coefficients and a neural network classifier,” *First International Symposium on Control, Communications and Signal Processing*, pp. 631-634, 2004.

[Shannon, 2004] Shannon, B., and Paliwal, K. K., “MFCC computation from magnitude spectrum of higher lag autocorrelation coefficients for robust speech recognition,” *Proceedings of International Conference on Spoken Language Processing,* Oct. 2004.

[Shlens, 2003] Shlens, J., “A Tutorial on Principal Component Analysis Derivation, Discussion and Singular Value Decomposition,” Version 1, 2003.

[Smith, 2002] Smith, L. I., "A tutorial on Principal Components Analysis," 2002.

[Somervuo, 2003] Somervuo, P., "Experiments with linear and nonlinear feature transformations in HMM based phone recognition," *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP),* vol. 1, pp. 52-55, Apr. 2003.

[Somervuo, 2003] Somervuo, P., Chen, B., and Zhu, Q., "Feature Transformations and Combinations for Improving ASR Performance," *Proceedings of the 8th European Conference on Speech Communication and Technology (Eurospeech)*, pp. 477-480, Sep. 2003.

[Soong, 1985] Soong, F. K., Rosenberg, A. E., Rabiner, L. R., and Juang, B. H., "A vector quantization approach to speaker recognition," *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP),* vol. 1, pp. 387-390, March. 1985.

[Tishby, 1991] Tishby, N. Z., "On the application of mixture AR hidden Markov models to text independent speaker recognition," *IEEE Transactions on Signal Processing,* vol. 39, pp. 563-570, 1991.

[Umesh, 2002] Umesh, S., Cohen, L., and Nelson, D., "Frequency warping and the Mel-scale," *IEEE Signal Processing Letters*, vol. 9, no. 3, pp. 104-107, Mar. 2002.

[Wanfeng, 2003] Wanfeng, Z., Yingchun, Y., Zhaohui, W., and Lifeng, S., "Experimental evaluation of a new speaker identification framework using PCA," *IEEE International Conference on Systems, Man and cybernetics,* vol. 4147-4152, pp. 5-8, Oct. 2003.

[Wang, 2000] Wang, X., Dong, Y., Hakkinen, J., and Viikki, O., “Noise robust Chinese speech recognition using feature vector normalization and higher-order Cepstral coefficients,” *5th International Conference on Signal Processing Proceedings*, vol. 2 , pp. 738-741, Aug. 2000.

[Zhao, 2000] Zhao, Y., “Frequency-domain maximum likelihood estimation for automatic speech recognition in additive and convolutive noises,” *IEEE Transactions on Speech and Audio Processing,* vol. 8, no. 3, pp. 255-266, May. 2000.

[Zhu, 1981] Zhu, X., Millar, B., Macleod, I., Wagner, M., Chen, F., and Ran, S., “A Comparative Study of Mixture-Gaussian VQ, Ergodic HMMs and Left-to-Right HMMs for Speaker Recognition,” Australian National University, 1981.