DEVELOPMENT OF A MAPPING FEATURE FOR SPEAKER RECOGNITION

A THESIS

submitted by

HEMANT MISRA

for the award of the degree

of

Master of Science

(by Research)

DEPARTMENT OF ELECTRICAL ENGINEERING

INDIAN INSTITUTE OF TECHNOLOGY, MADRAS.

MAY 1999

To

Shri. P. S. Misra, my Papa

and

Smt. Sushila Mishra, my Ma

THESIS CERTIFICATE

This is to certify that the thesis entitled **DEVELOPMENT OF A MAPPING FEATURE FOR SPEAKER RECOGNITION** submitted by **Hemant Misra** to the Indian Institute of Technology, Madras for the award of the degree of Master of Science (by Research) is a bonafide record of research work carried out by him under our supervision. The contents of this thesis, in full or in parts, have not been submitted to any other Institute or University for the award of any degree or diploma.

Prof. K. R. K. Rao	Prof. B. Yegnanarayana
Dept. of Electrical Engg.	Dept. of Computer Science and Engg.

Madras 600 036

Date:

ACKNOWLEDGEMENTS

I do not wish to merely complete a formality when I express my gratitude and admiration for my guides, Prof. B. Yegnanarayana and Prof. K. R. K. Rao. Apart from their help in my research work, I will always be indebted to them for the helping hand they lended me in my hour of crisis. Thank you very much, Sirs. I learned a lot from Prof. Yegnanarayana while attending his lectures, discussing my research problem with him, and on many other occasions. His ideas engrace my thesis in the form of results and performances.

I am thankful to Jwalant Sir who *pushed* me into this M.S. program. Sir, hope I could live upto your expectations. I am thankful to Dr. Vijay who taught me the complexities of speaker recognition in the easiest possible way.

I would like to thank Dr. C. Chandrasekhar, Dr. Hema Murthy and Dr. Sundar with whom I had quite a few good discussions which helped me in my understanding of the concepts. Shri. R. Rajendran helped me by mounting cdroms time and again (courtesy power failures) so that I can run my programs smoothly. I will take this opportunity to thank Dr. Arvind, Dr. Giridhar and Dr. Jalihal for taking personal interest in my career growth. TA work under them helped me very much in terms of knowledge and confidence.

Murthy Bhaiya (MIW) and Anitha (Smt. Mathew) are the two members of the speech and vision laboratory whom I pained at will. They are my guru in Matlab and Latex. I learned a lot from them. Dr. Manish Sarkar is the person from whom I learned ANN fundas (in the laboratory) and fundas of life (in the mess). Thanks to all of you for all the help and encouragement you provided me when I needed it most.

The two other members (many more to follow) of the laboratory, M. Mathew (TT) and Shajith Ikbal (Ikku), were my close companions through the thicks and thins of my life at IIT. Mere words cannot express my feelings for them.

In the hostel zone, Mathew P. J., R. Srinivasan (Tanu), Raja Mutiah and T. G. Venkatesh (Venku) were the persons around whom my life revolved. Their love for me, I can never forget. Thank you for all help and support you rendered me when I required it most.

Now my juniors. Thank you Anjani for going through the first draft of my thesis and many other things. Also, sorry for paining you on many occasions. Thanks Chandana for the first chapter as well as your wishes. Jyotsana, thanks for the discussions we used to have. Talking often helps in relaxing. Also, thanks for Chapter 2. Mahadeva Prasanna (anna) thanks for correcting my synopsis. Your suggestions were indeed helpful. Thank you Pragathi for your understanding and words of encouragement. S. P. Kishore (Kiddu!) pained me a lot. Kid, you are equivalent to Prashant. So you had the rights of a younger brother. Indeed you exercised your rights very well! Thanks for your unconstrained love. Siva Rama Krishna (3 gods!) was always there to boost up the sagging morale. His smiling face can make any person forget his worries. Keep smiling! Bavaraju Murthy (2^{nd} kid) is the person who pained me with his intelligence! His never ending queries even sharpened my own concepts.

Sincere thanks to Narendra Nath for his cslu and tcl/tk help, to Kavitha, Shrikanth, Tamil and Karna for their affection. Thanks to alpha1 and alpha2, the machines which ran my batch jobs, day and night, without any complaints.

In the last I would like to mention the names of the persons without whom I would have been nothing. My ammaji (grandmother), my mummy, Manu (my elder brother), Rakhee (my bhabhi) and Prashant (my younger brother) never let me feel the harsh realities of life. In the time of crisis, they took all the pains and never let me feel even the pinch. With their love, encouragement and support only I could finish my work successful. Also, the name of Vedika deserves a special mention. She made be more patient than ever.

Hemant Misra

ABSTRACT

KEYWORDS: *speaker-specific mapping; mapping feature; linguistic information; linguistic and speaker information; background normalization; network error criterion.*

Automatic speaker recognition is the task of recognizing a person from his/her voice by a machine. In this thesis, a new approach, called as *speaker-specific mapping*, is proposed for text-independent speaker recognition. The possibilty of using the mapping as a feature is explored in the present thesis. The main issue in using mapping as a feature is the choice of parameter vectors for the mapping. An important aspect of the mapping approach is its robustness to channel variations.

In the present work, mapping capabilities of a multilayer feedforward neural network (MLFFNN) are utilized for capturing the speaker-specific mapping feature. The speaker-specfic information is captured by mapping a set of parameter vectors specific to linguistic information (LI) in the speech, to paremeter vectors having linguistic and speaker information (LSI). Suitable parameter vector *pair* for the mapping (the parameter vector having LI and LSI) is obtained from experimental studies.

In order to improve the performance of the system, two techniques are proposed in the present work, namely, *background (BG) normalization* and *network error (Ne) criterion*. In the BG normalization, the concept of relative distance is used for taking a decision. The idea behind the normalization is to measure the distance of a speaker with respect to some common model having information about the other speakers. The Ne criterion is proposed to select frames containing speaker-specific information. The results of the speaker recogntion studies are illustrated with TIMIT and NTIMIT

databases. For speaker verification, an Equal error rate (EER) of 0.5% is obtained for the TIMIT database, and an EER of 5.45% is obtained for the NTIMIT database.

TABLE OF CONTENTS

List of Tables

List of Figures

Chapter 1

AUTOMATIC SPEAKER RECOGNITION: AN INTRODUCTION

1.1 OBJECTIVE OF THE WORK

Every person has a unique voice, therefore voice can be used for recognizing a person [1]. Automatic Speaker Recognition [2, 3] is the task of recognizing a person from his/her voice by a machine. In this thesis, we study a new feature based on *mapping approach* for text-independent speaker recognition. The main idea in the mapping approach is to capture the speaker-specific information. This is achieved by mapping a set of parameter vectors specific to linguistic information in the speech, to parameter vectors having both linguistic and speaker information. The objective of the present work is to explore the possibility of using speaker-specific mapping as a feature for speaker recognition. Some of the issues we address in the present work are:

1. Parameter vector for the mapping approach.
2. Improving the performance of the system using background normalization and selecting frames containing speaker-specific information.
3. Robustness of the mapping approach to channel variations.

1.2 INTRODUCTION

Usually physical tokens, such as identity cards, badges and passwords, are used for person authentication. Biometric methods, such as finger print recognition and recognition by voice, to name a few, can also be used for the same task. Unlike physical tokens used for recognizing a person, biometric methods are superior in the sense that they cannot be lost or forgotten. Recognition by voice has a distinct advantage over other methods used for establishing the identity of a person. The individual need not be present in-person, that is, authentication can be done remotely over a telephone line. The second advantage is that the recognition can be done with least obtrusion to the person's normal work.

Automatic speaker recognition has numerous applications in day-to-day life. Voice can be used as a password for bank transactions over a telephone network, voice dialing, information and reservation services, secure access to confidential systems and many other related services. In short, if a person's identity is required to be checked, and a password is needed to do so, automatic speaker recognition can be used. Another application of automatic speaker recognition could be for forensic purposes [4].

1.3 ISSUES IN SPEAKER RECOGNITION

1.3.1 Sources of variability among speakers

Some of the factors that contribute to the uniqueness of voice of a speaker are discussed in the section. A few of them are qualitative (can be perceived but cannot be estimated), while others are quantitative (can be perceived as well as estimated).

1. Qualitative Factors

    - Speaking style: The way a speaker speaks depends on many factors. Some of them are: (a) the region speaker belongs to (dialect region), (b) mother tongue of the speaker, (c) socio-economic background of the speaker
    - Linguistic cues: Some speakers use some particular words or phases quite often, and in a unique manner.

2. Quantitative Factors

    - Shape and length of the vocal tract: It is reflected in the spectrum of the speech signal.
    - Source characteristics: They can be observed in the pitch frequency as well as roll-off of the spectrum.
    - Speaking rate: This leads to different syllable rates for different speakers.
    - Duration: Sub-word duration, word duration and other durational cues.

1.3.2 Sources of variability in speaker's own voice

Besides different speakers having distinct voices, the voice of a speaker itself changes with time. Some of the sources of variation are because of the speaker himself, but some others are not under his control. While most of them are short-term in nature, a few of them are long-term too. Some of them are listed below.

1. Short-term Variations

- Speech production process: Speech production process is dynamic, and it changes with time. A speaker cannot produce the same sound "exactly" at two different times. Short-term variations because of the speech production process, usually the ones that happen in a day or so, are less significant.

- Emotional state of speaker: Emotional state of a speaker is reflected in his speech, and it changes the voice characteristics of the speaker.

- Ambient noise: In the presence of ambient noise, a speaker's speaking style changes. The speaker starts exerting his vocal chords more, consequently his voice characteristics are affected. This effect is known as "Lombard Effect".

- Throat infection: In the case of throat diseases, such as cough and cold, the voice of a speaker can change.

- Channel effects: The channel through which the speech signal passes, superimposes its own characteristics on the speech signal. Therefore, with the change in channel, the voice characteristics can also change [5, 6].

2. Long-term Variations

   - Speech production system: Apart from its contribution to the short-term variations, the dynamic nature of speech production mechanism plays an important role in the long-term variations too. As compared to the short-term variations, the changes in the voice characteristics of a speaker in the long run are more prominent. One of the reason for the long-term variations is the physiological changes in the speech production system itself.

1.3.3 Components of a speaker recognition system

The three main components of a speaker recognition system are:

1. Parameter extraction from speech signal.
2. Model generation for each speaker.
3. Matching algorithm and decision logic.

In the present work, the first two components are studied using mapping approach.

There are two phases in an automatic speaker recognition system, training and testing. In the training phase, the models for speakers are developed. This learning phase is supervised, and is usually a one-time process. The training phase requires the first two components of the system mentioned above. In the testing phase, given a test utterance, the task is either to identify or to verify the claim of a test speaker. Testing involves the first and the third components of the above mentioned system.

1.3.4 Parameters for speaker recognition

After passing through various channels, the quality of a speech signal degrades. As a consequence, the performance of a speaker recognition system drops [5,6]. But human beings do not seem to have much problem in recognizing a speaker even if the channel characteristics are superimposed on the speech signal. That means, there are certain characteristics of the speech signal which are not degraded by the channel, and hence are robust to the channel variations [7, 8]. Is it possible to extract parameters that are robust to channel degradation? In the absence of right parameters/features, the remaining stages of a speaker recognition system cannot improve the performance of the system.

Parameter extraction [8] is a key step in developing systems for speaker recognition. Some of the desirable properties of the parameters are:

1. High interspeaker discrimination

2. Low intraspeaker variability

3. Invariance to degradations in the speech signal due to channel and noise [9].

Normally the parameters used in the speech systems for recognition (speech recognition as well as speaker recognition) are some variation of the parameters primarily used to represent the speech signal for coding and compression. Most of these parameters are based on spectrum analysis over a short-time of 10-30 ms. The short-time spectral parameters are derived either by computing directly the Discrete Fourier Transform (DFT) spectrum of the speech signal or by assuming a model for speech production mechanism as in the linear prediction (LP) analysis [10]. Each segment of speech is represented in terms of a parameter vector such as linear prediction coefficients (LPCs) [11], LPC derived cepstral coefficients [12], mel-cepstral coefficients and perceptual linear predictive (PLP) [13] derived cepstral coefficients. All the above mentioned parameter vectors are spectral parameters. They depend on the shape and size of the vocal tract of a speaker. Moreover, all these parameters are affected by the ambient noise and channel distortion.

The conjecture in all the traditional text-independent speaker recognition approaches is the following: (a) The parameter vectors obtained from the speech utterances of a speaker are from a probability density function (PDF), and (b) Each speaker has a unique PDF. The underlined principle in all the traditional approaches is to capture this PDF [14]. Schwartz et al. [15] were the first to use the PDF for speaker recognition.

The difficulty with the parameters based on representation of the speech signal for speaker recognition is that normally there is very little discrimination between models for different speakers. Moreover, these speech systems are affected by degradations in the speech during testing. This is because the PDFs of the parameter vectors derived from the degraded test speech, do not match with the PDFs of the parameter vectors used during training. Further, in all these methods, there is no effort to determine the features specific to a given speaker. The methods depend mostly on the statistical distribution of the parameter vectors of each speaker. Chapter 2 deals with some of these aspects in more detail.

1.4 MOTIVATION FOR THE PRESENT WORK

Methods based on speaker-specific mapping as a feature for speaker recognition are relatively new [16–18]. In the mapping approach, the idea is to capture the speaker-specific information. This is accomplished by mapping a set of parameter vectors specific to linguistic information (LI) in the speech, to parameter vectors having both linguistic and speaker information (LSI). Speaker-specific mapping approach was suggested in [16] for text-dependent case and extended to text-independent speaker recognition in [17, 18].

The motivation is to study the effectiveness of speaker-specific mapping as a feature for speaker recognition. The studies are divided into three parts: (a) To derive experimentally the optimal pair of parameter sets for the mapping [19], (b) To explore methods to improve the performance of the system, and (c) To study the affect of channel variations on the performance of the mapping approach.

In order to improve the performance of the system, we study two methods, namely, *background (BG) normalization* and *network error (Ne) criterion* Background normalization is a popular technique used in other approaches such as Gaussian mixture models (GMM) [20] and hidden Markov models (HMM) [21]. The concept of background normalization is developed for the mapping approach. It is fine tuned to improve the performance of the system. The *Ne* criterion is proposed to select the frames containing significant speaker-specific information [22, 23].

Finally, robustness of the speaker-specific mapping approach to channel variations is studied. Two databases, TIMIT[1] and NTIMIT[2], are used for this purpose.

1.5 TYPES OF SPEAKER RECOGNITION SYSTEMS

Automatic speaker recognition is a generic term. It can be divided into the following categories:

1. Depending on Task
    - Automatic speaker identification (ASI)
    - Automatic speaker verification (ASV)
2. Depending on Text Employed for Doing the Task

[1] APPENDIX-A

[2] APPENDIX-B

- Text-dependent methods
- Text-independent methods
- Text-prompted methods

1.5.1 Automatic speaker identification

In ASI [14], given a test speaker's utterance, the task is to find the identity of the speaker. ASI can be further categorized into close-set identification and open-set identification. If there are N speakers enrolled with the system, in the case of close-set identification, the test speaker is always one of these N speakers. It need not be so in the open-set identification task.

Fig.1.1 is the block diagram of a typical closed-set speaker identification system. In close-set identification, the registered speaker whose model matches best with the test utterance is adjudged as the identified speaker. The output of the system is the identity of the test speaker.

In contrast, in open-set identification, the task is performed in two steps. First, the speaker closest to the test speaker is found. Next, the distance of this speaker is compared with a threshold. If the distance is less than the threshold (threshold is either fixed or it might vary from speaker to speaker), then the speaker is labelled as the identified speaker. Otherwise, a decision is made that the test speaker is not among the speakers registered with the system.

Finding an optimum threshold is a difficult problem in itself, and is an important issue in any speaker recognition system. With increase in the population size the performance of a speaker identification system falls [5].

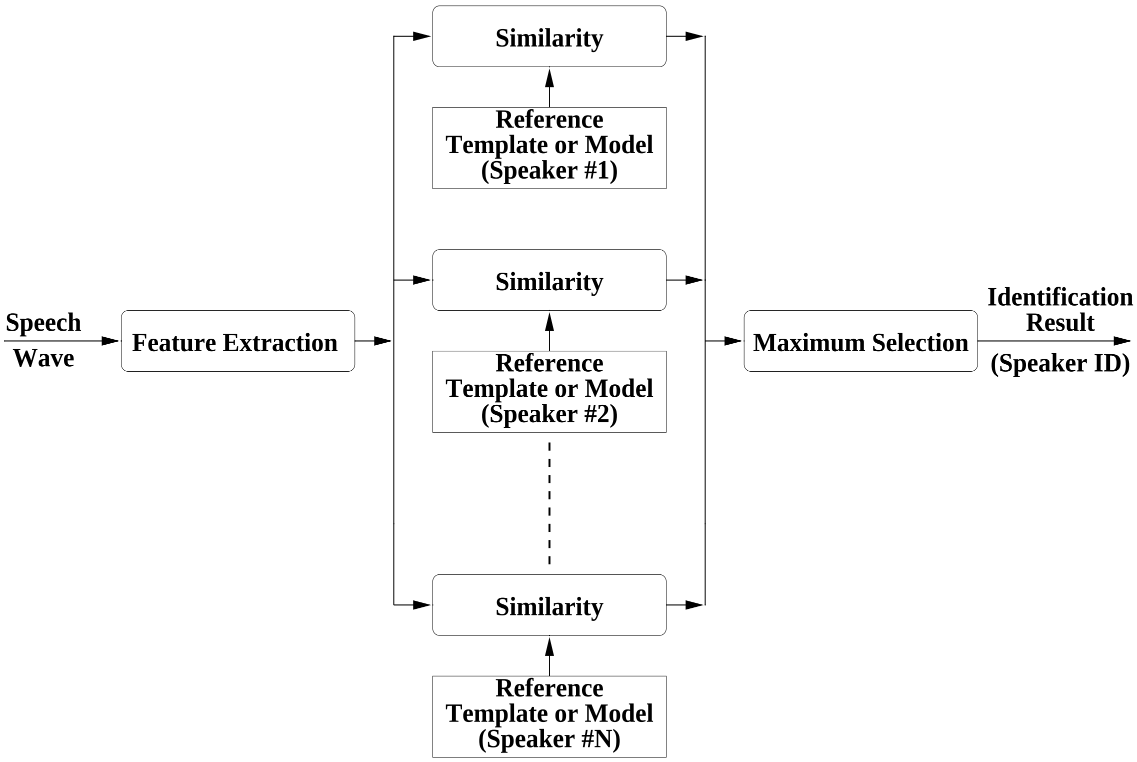


Fig. 1.1: Basic structure of a typical close-set speaker identification system.

1.5.2 Automatic speaker verification

In ASV [24], a test speaker claims himself to be a certain speaker (called as target speaker) registered with the system. The task is to verify the claim of the test speaker. As in the close-set identification task, the concept of threshold is used in ASV also. If the distance of the test speaker utterance to the target speaker model is below the threshold (may be static or dynamic [25]), then the claim of the speaker is accepted, otherwise it is turned down. The confidence with which the claim is accepted or rejected is of importance, and is termed as score. Fig.1.2 shows the block diagram of a typical ASV system. In contrast to the ASI task, in the case of ASV two types of

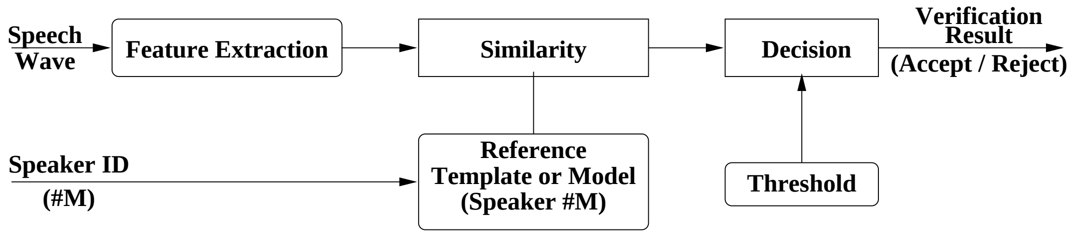


Fig. 1.2: Basic structure of a typical speaker verification system.

errors are involved, false acceptance (FA) and false rejection (FR). If the claim of an imposter is accepted it is FA, while in the case of FR the claim of a genuine speaker is turned down. The FA and FR errors are interlinked, and depend on the threshold. A high threshold (in terms of distance) means that the FR will be low but at the cost of high FA rate. Likewise, a low threshold will lead to a low FA rate, but then the FR will be high. Depending on the level of security required, the threshold can be adjusted. There exists a threshold at which the FA and FR are equal. The error at this threshold is called as equal-error-rate (EER). EER is a widely used measure to judge the performance of ASV systems.

In contrast to ASI, the performance in the case of ASV task is independent of the size of the population. This is because the decision is taken based on the model of the target speaker alone, and not on the models of all the registered speakers.

Fig.1.3 shows the nature of FA and FR curves as a function of threshold. The point of cross-section of the FA and FR curves is shown in the figure, and is marked as *Equal Error Rate*.

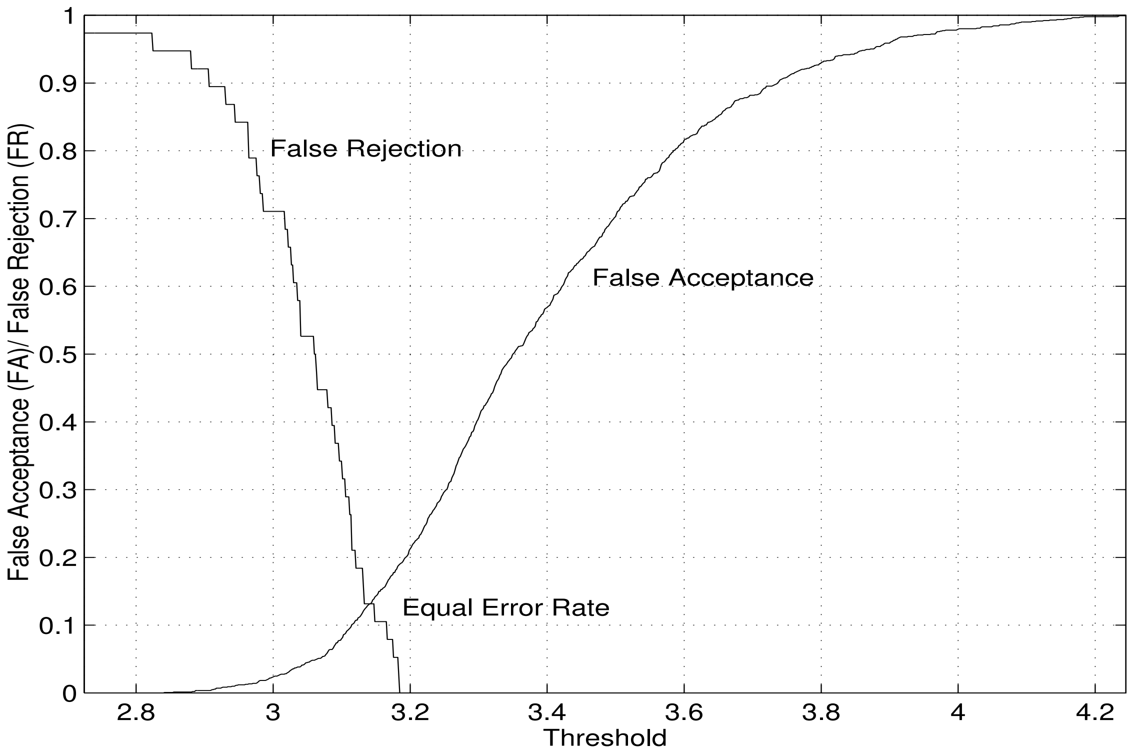


Fig. 1.3: False Acceptance (FA) and False Rejection (FR) curves as a function of threshold.

1.5.3 Text-dependent speaker recognition

In a text-dependent speaker recognition system, the text of the utterances during training and testing is same. Usually matching is done at the acoustic level, and therefore the performance is generally better. One common matching method employed is the dynamic time warping (DTW) [12] technique. The performance of these systems is correlated with the vocabulary that is chosen.

1.5.4 Text-independent speaker recognition

In a text-independent system, the text of the utterances during training and testing need not be the same. In these systems, the statistical behaviour of the speaker's parameter vectors is captured during training. The same statistical behaviour is estimated at the time of testing, and compared with the behaviour captured during the training phase. Because these methods use statistical behaviour, the performance is not as good as in the case of text-dependent systems. Chapter 2 covers the traditional approaches to text-independent speaker recognition and some of their limitations.

1.5.5 Text-prompted speaker recognition

Both the earlier mentioned systems suffer with a common problem. By storing the voice of a speaker on some recording device, and then playing it back at the time of testing, can defeat the systems. A text-prompted system can overcome this problem. In a text-prompted systems, the system prompts the user to speak some random combination of digits, or some randomly selected words from a set of words. The system expects the user to speak the same sequence in a reasonable time. When the speaker utters the digits/words, first the system checks whether the digits/words sequence is the same as prompted by the system. If the sequence matches, the claim is validated the way it is done in a text-dependent system. If the sequence doesn't match, the validation process is not performed, and the speaker's claim is rejected. Fig.1.4 shows the block diagram of a typical text-prompted system. These systems are robust to recording, but have a drawback that speech and speaker recognition systems are integrated into them. Therefore the system is more complex. Moreover, the

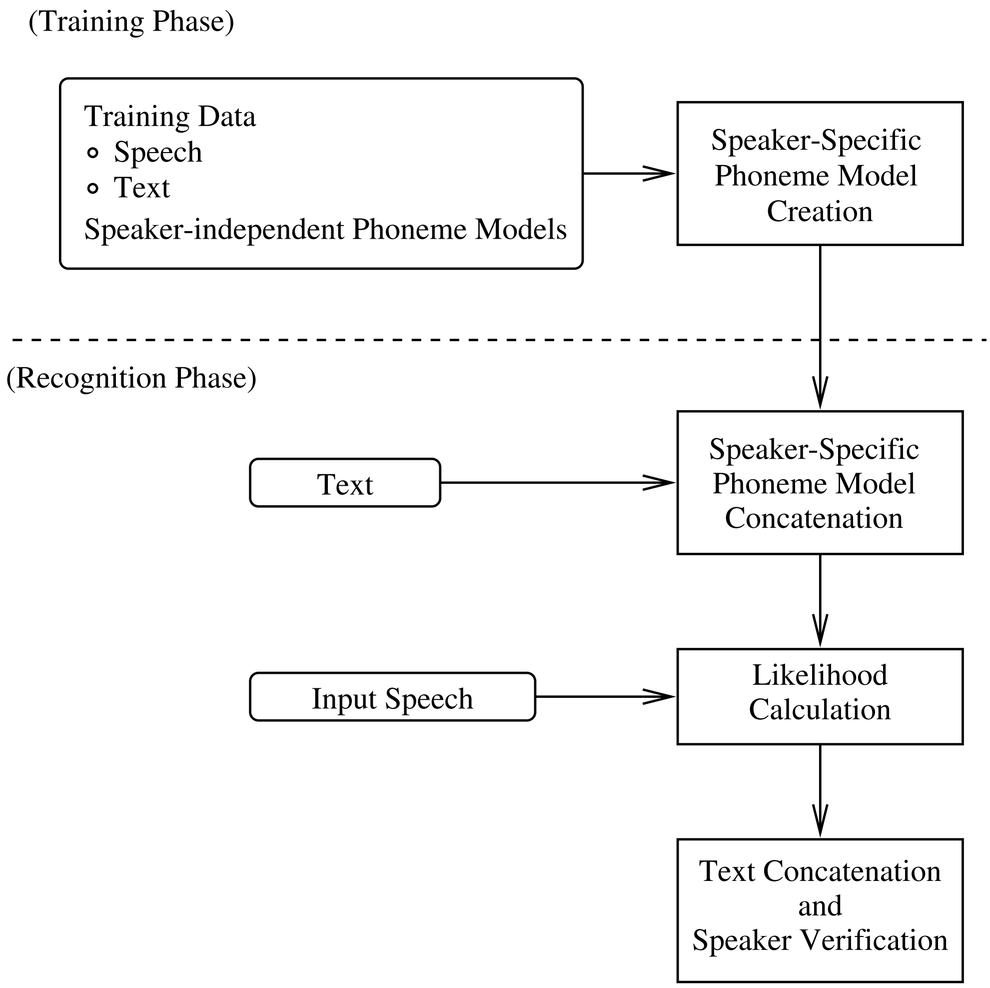


Fig. 1.4: Block diagram of a text-prompted speaker recognition system.

performance of the individual component of the system guides the overall performance of the system.

A summary of the various tasks in speaker recognition, and their dependency on the text is as follows:

1. **Identification Task**

   (a) **Closed-set identification task**

i. **text-dependent methods**

ii. **text-independent methods**

iii. **text-prompted methods**

(b) **Open-set identification task**

i. **text-dependent methods**

ii. **text-independent methods**

iii. **text-prompted methods**

2. **Verification Task**

(a) **text-dependent methods**

(b) **text-independent methods**

(c) **text-prompted methods**

1.6 ORGANIZATION OF THE THESIS

This thesis is organized in eight chapters. A brief overview about the organization of the thesis in terms of the chapters and their contents is as follows:

Chapter 2 introduces the feature space. In this chapter, the concept of speaker-specific PDF is explained, and some traditional approaches to capture these PDFs are reviewed. The chapter concludes with a discussion on the limitations of the parameters, as well as the traditional approaches, and hence the need for new approach.

The idea of mapping approach and the requirement of speaker-specific mapping as a feature for speaker recognition task forms the contents of Chapter 3. The other topics covered in this chapter are: the parameter vectors required for capturing the speaker-specific mapping, analysis of the approach and the model used to realize the mapping in the current work.

In Chapter 4, two parameter vectors are studied to demonstrate the effectiveness of the mapping approach in capturing speaker-specific knowledge. In Section 4.1, the autocorrelation coefficients are used as a parameter vector. The studies presented in this section validate the concept of speaker-specific mapping. In Section 4.2, the results obtained using the LPC cepstral coefficients as a parameter vector are presented. The optimal LPC analysis pair is obtained experimentally.

Chapter 5 and Chapter 6 are devoted to two techniques, namely, Background (BG) normalization and Network error (Ne) criterion, respectively, to improve the performance of the system. Through experimental studies the effectiveness of both the techniques is demonstrated.

The other aspects of the mapping approach are: its performance for other spectral parameters , dependence of its performance on the number of test-utterances, and robustness of the mapping approach to channel variations. These topics are covered in Chapter 7.

A summary of the work is given in Chapter 8, along with the scope for future work.

Chapter 2

FEATURES FOR SPEAKER RECOGNITION

This chapter is divided into three sections. Section 2.1 introduces the idea of feature-space. The concept of parameter vectors being samples from continuous PDFs, and the uniqueness of these PDFs for each speaker, are explained the same section. The next section gives an overview of the traditional approaches used for text-independent speaker recognition. In Section 2.3, we discuss some of the limitations of these methods, as well as parameter vectors, and thus build up a case for the mapping approach, the topic of next chapter.

2.1 FEATURE SPACE

Physiological and behavioural characteristics of a speaker are reflected in the spectral parameters (vocal tract characteristics), as well as in the supra-segmental parameters (source characteristics) of his speech signal. As mentioned in Section 1.3.1, some of these characteristics are difficult to measure explicitly, while some others are captured by various signal processing measurements.

As with any pattern recognition task, the speech signal is first reduced to a sequence of measurements called features, feature vectors or parameters. This reduction is termed as feature (or parameter) extraction. Feature extraction is the process of reducing data while retaining the classification information. The feature extraction module is of importance in a speaker recognition system (Section 1.3.4). Text-independent speaker recognition is often based on the following premise: acoustic-measurements (parameter vectors extracted from a speech signal) derived from the speech utterances of an individual are characterized by a stable and unique PDF. That is, the parameters extracted from the voice of an individual occupy a particular region in the feature-space which is unique to him. The differences between the PDFs allow to discriminate speakers.

In the feature-space, the parameter vectors for different speakers that form the PDFs having more discrimination give better performance than the parameter vectors whose PDFs overlap. The PDFs of the parameter vectors should have high interspeaker variability and low intraspeaker variations (Section 1.3.4). With increase in the population size, "crowding" [5] of the feature-space occurs. This accounts for the degradation in the performance in ASI as the population increases.

Text-independent systems are based on the modeling of the speaker's entire acoustic feature-space. The PDF for a speaker is estimated from the parameter vectors obtained from his speech, and model the entire feature-space of the speaker. Therefore, a large amount of speech data is needed for training to have a reliable estimate of the PDFs. On the other hand, text-dependent systems require the modeling of a sub-space of the speakers. The sub-space modeled depends on the text employed for the task. This is the reason for correlation between the performance of these systems and the vocabulary chosen.

In the next section some traditional methods to estimate the PDFs are discussed. All the methods are applicable for text-independent speaker recognition systems, but can be extended to text-dependent recognition systems.

2.2 METHODS TO CAPTURE PDFs

The approaches to text-independent speaker recognition can be broadly classified into two categories:

1. Parametric Approaches
   - (a) Gaussian Mixture Model (GMM) based
   - (b) Hidden Markov Model (HMM) based
2. Nonparametric Approaches
   - (a) Vector Quantization (VQ) based
   - (b) Artificial Neural Network (ANN) based

The basic assumption in these approaches is: The parameter vectors produced by a speaker are samples from a continuous PDF. Though the PDFs of different speakers overlap, sharing the feature-space, they are ideally distinguishable from each other. Therefore, speaker recognition can be performed using these PDFs. Speaker recognition is accomplished by estimating where voices "spend their time" [14] in the feature-space.

Approaches that assume a model for PDFs characterized by parameters (parameters of the model, not the parameter vectors extracted from the speech signal) are termed as parametric. The parameters of the model are estimated from the training set of feature vectors. In contrast, in nonparametric approaches, no assumption is made about the PDFs.

2.2.1 Parametric Approaches

Parametric approaches are also known as model-based approaches. These approaches assume a model, and the parameters of the model are estimated from the training feature vectors. It is assumed that the model is adequate to represent the PDFs. Two of the widely used parametric approaches are GMM and HMM based.

2.2.1.1 Gaussian Mixture Model (GMM)

In the GMM approach, the underlying PDF specific to each speaker is captured using Gaussian mixtures [26–28]. If the PDFs of different speakers are different, then the GMMs used to model them will also be different. The two parameters of the GMMs (means and variances of the Gaussian bells) are estimated from the training feature vectors. The GMMs of a speaker will be unique to the speaker. The model of each speaker is defined in terms of the means and variances of his GMMs. The differences between GMMs facilitate in discriminating speakers.

Usually the GMM approach is not used directly. Instead, a background (BG) normalization technique [20, 29] is used to improve the performance. In the present work (mapping approach), we have also used the concept of BG normalization. In the

BG normalization, a background GMM is built first. For building the BG model, the feature vectors from the speech utterances of a large number of speakers are obtained. The BG model is built (or trained, i.e., parameters of the model are estimated) using these feature vectors. The speakers' models are built on top of the BG model as follows: The feature vectors from the utterances of the speaker whose model is to be built, are obtained. The BG model is retrained using the feature vectors of the speaker. The parameters obtained after retraining are assigned to the speaker's model. Generation of a general BG model, and then on top of it the speakers' models, constitute the training phase.

At the time of testing, the features extracted from the test utterance are given to the BG as well as the target speaker model (ASV task). The difference between the outputs of the two models (BG and speaker) is computed to obtain the final score.

The fundamental assumption behind the BG model approach is: The BG model contains information about all the speakers and it is smooth. When the BG model is further trained with a speaker's feature vectors, deviations are produced on the overall smooth surface of the BG model. These deviations are produced in that region of the feature-space where the speaker's voice "spends its time". The difference between the BG and the speaker's model score can be attributed to these deviations. Higher the difference is (positive), more is the evidence that the test utterance belongs to the target speaker.

Discussion on the approach:

1. The PDF of speech might be very complex. If the feature vectors do not follow a distribution that can be modeled by GMMs, the performance of the resulting

system will be poor.

2. It is a statistical approach, and hence it needs a large amount of training data to have a good estimate of the model's parameters.

3. To get a good confidence score the testing data should be large.

4. When the training and testing environments are different, performance of the system degrades [5, 6]. It can be attributed to the fact that the channel effects and presence of noise changes the speaker-specific PDFs appreciably. However, the problem is having its root in the feature representation.

2.2.1.2 Hidden Markov Model (HMM)

In this approach, Hidden Markov Models [30] are built for each speaker [25, 31–34]. In the training phase, an HMM for each speaker is obtained (i.e., parameters of the model are estimated) using the speaker's training feature vectors. The parameters of HMM are [30]:

1. State-transition probability distribution: It is represented by $A = \{a_{ij}\}$, where

$$a_{ij} = P[q_{t+1} = j | q_t = i] \qquad 1 \leq i, j \leq N \tag{2.1}$$

   defines the probability of transition from state i to j at time t.

2. Observation symbol probability distribution: It is given by $B = \{b_j(k)\}$, in which

$$b_j(k) = P[\vec{o}_t = \vec{v}_k | q_t = j] \qquad 1 \leq k \leq M \tag{2.2}$$

   defines the symbol distribution in state j, $j = 1, 2, \cdots, N$.

3. The initial state distribution: It is given by $\Pi = \{\pi_i\}$, where

$$\pi_i = P[q_1 = i] \qquad 1 \leq i \leq N \tag{2.3}$$

In the given description, N is the total number of states, and q_t is the state at time t. The states are hidden, however often some physical significance might be attached to them. M is the number of distinct observation symbols per state, and $\vec{o}_t$ is the observation symbol at time t. The observation symbols correspond to the physical output of the system being modeled.

An HMM can be described completely in terms of the following: (a) two model parameters, N and M, (b) specification of the observation symbol, and (c) the specification of three sets of probability measure, A, B, and Π. Compact notation for HMM is

$$\lambda = (A, B, \Pi) \tag{2.4}$$

At the time of testing, $P(\vec{O}|\lambda)$ for each model is calculated, where $\vec{O} = (\vec{o}_1\vec{o}_2\vec{o}_3 \cdots \vec{o}_T)$ is the test feature vectors sequence. The goal is to find out the probability, given the model, that the test feature vectors belong to that particular model. The speaker whose model gives the highest score is declared the identified speaker. GMM model corresponds to the single-state continuous ergodic HMM [35] .

Discussion on the approach:

1. Output of the model is in terms of multiplication of several probabilities, therefore the scores are very small. Efficient scaling methods are required to overcome this scaling problem.

2. As in the case of GMM approach, HMM also is a statistical approach. Hence, to estimate the parameters of the model reliably, a large amount of training data is required.
3. As in the case of GMM, when the training and testing conditions are different, performance of the system drops.

2.2.2 Nonparametric Approaches

In nonparametric approaches no models are assumed. However, the basic idea is still the same: (a) The feature vectors extracted from the utterances of a speaker are samples from a continuous PDF, and (b) every speaker has a unique PDF. Since no model is assumed, therefore estimation of the model's parameters is not required. Instead, from the feature vectors itself the PDF estimation is done. Comparison between the PDFs is done to accomplish the task of speaker recognition. Two popular approaches in this category are VQ and ANN based [22, 36–39].

2.2.2.1 Vector Quantization (VQ)

Essential characteristics of a speaker can be represented by a set of training feature vectors obtained from short-segmental (10-30 ms) analysis. Keeping in mind the computations involved and the storage capacity required, this becomes impossible if the number of feature vectors are very large. VQ is an effective method to get representative vectors for a set of training feature vectors.

In the VQ-based method for speaker recognition, codebooks are obtained for each speaker. Codebooks consist of a small number of representative feature vectors to characterize a speaker. During training, the feature vectors are grouped into certain fixed number of clusters. The mean of each cluster is calculated, and the mean vector thus obtained is the representative of that cluster. The clusters' representative constitute the codebook of the speaker.

Assuming that the PDF from which feature vectors are drawn is unique for each speaker, each speaker will have a unique codebook. A codebook is generated for each speaker that is specific to him.

In the testing phase, the test utterance is vector quantized using the codebook of each reference speaker. **VQ distortion** (distance of the test vector from the codebook element that is closest to it) is calculated for each speaker's codebook. This VQ distortion is accumulated over the entire test utterance. The accumulated distortion is used to arrive at a decision for speaker recognition.

In the VQ approach, instead of giving equal importance to all the distortion scores for each codebook element, some weighting can be done [40]. Weighting might be dependent on how much speaker-specific knowledge can be attributed to a particular element. Moreover, there can be more than one codebook for each speaker [9, 41]. For example, the feature vectors might consist of cepstral coefficients, their derivatives, pitch and its derivative. A voiced segment of speech will contain all the four types of features, while an unvoiced segment will contain only the first two. Separate codebook can be generated for the voiced and unvoiced parts. The scores of both the codebooks can be weighted and summed to arrive at a final score. It has been shown that the pitch information (extra feature) and its derivative, when incorporated along with the spectral envelope features into the system, does improve the overall performance of the system [41].

VQ-based method suffers from the problem of outliers in the distribution of the feature vectors. The outliers vectors can be due to: (a) Inter-session spectral variations, or (b) Difference between the phonetic content of the texts used in the training and testing.

VQ-based speaker recognition systems are easy to build, and are shown to give good results. The concept of sequence is not used in the VQ-based approach. However, in matrix quantization, the concept of sequence can also be brought in. VQ can be considered a degenerate case of single-state HMM with observation probability being replaced by the distance measure [35].

Discussion on the approach

1. VQ based methods suffer from the problem of outliers.
2. The vectors lying at the boundaries of the clusters create problem.
3. A large amount of training data is required to have a good estimate of cluster representatives. Each cluster should have some minimum number of feature vectors so that the representative vector is calculated reliably.
4. Degradation in the performance occurs if the training and testing environments are different.

2.2.2.2 Artificial Neural Network (ANN)

Multilayer feedforward neural networks (MLFFNN) are well known for their mapping [42, 43] and generalization capabilities [44–46]. The potential of MLFFNN lies in its structure [45]. The processing units operate in parallel. The nonlinearity of the processing units can effectively yield nonlinear classification [47] in the feature-space.

Usually MLFFNN consists of one input layer, with linear processing units, one or more hidden layers with either linear or nonlinear processing units, and one output layer which might consist of either linear or nonlinear units. The units in one layer are fully connected to the units in the layer above, with the exception of the output layer, which is the last layer. The connection weights are adjustable, and the process of adjustment of the weights is called as learning.

In the training phase, the feature vectors are fed to the input of the network, and the corresponding output is obtained at the output layer of the network. Actual output of the network is compared with the desired output. The error between the two is propagated back to adjust the weights. The weights are adjusted such that the mean-square-error between the actual output of the network and the desired output is minimized. For this reason, the learning process of MLFFNN is also termed as "Backpropagation" learning. The backpropagation algorithm is a gradient descent method to minimize the cost function. The cost function is the mean-square-error between the desired and the actual outputs of the network.

The computations involved during the training phase can be categorized into two: forward propagation (of the training patterns) and backward propagation (of the error between the actual and the desired outputs). The forward part is meant for calculating the output of the network, while the backward part is used to adjust the weights such that the mean-square error is minimized. The training feature vectors are given one by one. Use of the full set of feature vectors once for adjusting the weights is termed as one “epoch” or “cycle”. Several cycles are used to reduce the error below an acceptable threshold. Each training pattern (feature vector) is used several times.

In general, when using a MLFFNN for speaker recognition task, no model is assumed. Separation between the speakers in the feature-space is “learned” by the MLFFNN from the feature vectors itself during the training phase. “Feedforward neural networks can be regarded as semi-parametric models for conditional probability estimation.” [48]. Article by Bennani and Gallinari [49] gives a comprehensive account of the recent work on speaker recognition using ANN.

In [50, 51], one network for each speaker was used with each network having one output node. The speech frames of all the speakers were interleaved to make every frame contribute equally in the training. The desired output of the networks was kept binary: 1 if the frame belonged to the speaker, otherwise 0. In this approach, the clustering property of MLFFNN were utilized, with the data itself constructing the separation space. In another ANN approach [52], only one network was considered with the number of output units being same as the number of speakers.

A MLFFNN network learns (estimates) the underlying PDF of each speaker in the feature-space without assuming any model. In the feature-space, each speaker will have a different PDF. Separation in the feature-space is obtained by capturing the PDFs specific to every speaker. Looking from a different perspective, neural network (NN) tries to form hyper surfaces in the higher dimensional feature-space to separate the PDFs of the different speakers.

The advantages of using an ANN are many. No assumption about the PDFs of the underlying feature vectors is made. The PDFs are learned from the feature vectors themselves. ANN has good generalization capability. Moreover, the degradation in the performance of an ANN is graceful. Even if some of the weights are corrupted, the performance doesn't fall sharply due to the distributed nature of processing.

Discussion on the approach

1. There is no proof of convergence.
2. Time required for the training is more.
3. There is problem of local minima. Once the network gets trapped in a local minima, it cannot come out of it, leading to poor results.
4. Performance is dependent on the size of the network, as well as training [35,51]. There is no easy method to determine the optimum size of the network.
5. Because of nonlinearity, analysis of the network behaviour is difficult.
6. Over training might lead to loss of generalization capability.

2.3 NEED FOR A DIFFERENT APPROACH

In the above approaches, if the training and testing conditions are different, the PDFs of the feature vectors change, which in turn results in poor performance [5]. Some of the techniques, such as cepstral mean subtraction, RASTA filtering and delta cepstral coefficients, are used to compensate for the first order linear filtering effects. Apart from these simple linear filtering effects, there are other effects due to the handset variability [6]. Such effects cannot be removed easily by the above mentioned techniques. Handset variability occurs when training and testing data are collected with different handsets. The PDF of the feature vectors extracted from the test speech is different from the PDF of the training speech in such cases. This is one of the reason for degradation in the performance with handset variability.

There could be two approaches to the problem of change in the PDFs of the feature vectors with the change in training and testing environment. Either we should find feature vectors that are invariant to the channel variations (whether linear or nonlinear, including handset) [8], or to use the present features such that instead of PDF of the speaker's feature vectors, speaker-specific information is captured. This speaker-specific information should be invariant to the effects of channel for it to be used as a feature for speaker recognition.

In the present work, we adopt the second approach. Speech information is represented by LP cepstral coefficients [12] for each of the 20 ms speech segment. For comparison we have also considered other parameters like mel-cepstral coefficients and perceptual linear prediction (PLP) cepstral coefficients. Speaker-specific information is captured from these parameter vectors. The approach adopted to carry out the present work is ANN based. The aim is to capture the speaker-specific information using the mapping capabilities of MLFFNN [42].

The mapping idea was initially mooted by Gong and Haton [16, 53]. They used the term "Interpolation Vector" for their text-dependent studies that yielded good performance. The idea was extended further by Hynek and Narendranath [18] for their text-independent speaker recognition studies. They introduced the term "Speaker-specific Mapping". The next chapter is devoted to the idea of mapping approach and its qualitative analysis. Realization of the mapping feature is also discussed.

2.4 SUMMARY

In the present chapter, the concept of feature-space was introduced. The conjecture of feature (parameter) vectors being samples from continuous PDFs that are unique for each speaker, was explained. We reviewed some of the traditional approaches to capture these speaker-specific PDFs for speaker recognition. We discussed some of the limitations of the feature vectors. In the process, we build up the case for using the speaker-specific mapping as a feature for the speaker recognition task.

Chapter 3

MAPPING APPROACH FOR SPEAKER RECOGNITION

Speech signal contains information about speaker, emotional state of the speaker, and linguistic information. All the above information is embedded in the speech signal, and it is difficult to separate it out. Linguistic information (**LI**) is the message part of the speech signal, while speaker information (**SI**) is embedded over it.

In Section 3.1, the other work related to the mapping approach are reviewed. Section 3.2 addresses the issue of qualitative representation of LI and linguistic plus speaker information (**LSI**). In Section 3.3, the hypothesis of the mapping approach is explained. Analysis of the mapping approach is presented in Section 3.4. In Section 3.5, we propose a model to realize the speaker-specific mapping feature.

3.1 MAPPING APPROACH: AN INTRODUCTION

Gong and Haton [16, 53] proposed a mapping approach for text-dependent speaker recognition. They called the mapping function as "non-linear vectorial interpolation function". Using MLFFNN with one hidden layer, and the mel-cepstral coefficients of the same order at the input and output of the network, they achieved good results. In their studies, phonetic labelling of the utterances was required, both during training and testing. Hynek [18] suggested speaker-specific mapping approach for their text-independent speaker recognition studies. They used the terms “speaker dependent” and “speaker independent” information. They used the PLP derived cepstral coefficients for their studies. Also, using oriental principal component analysis (OPCA), they obtained an improved EER of 9.3% compared to the best result of 12.5% obtained using 7^{th} order PLP as the input and 14^{th} order PLP as the output. They used a subset of “Switch Board” corpus for their studies. The database consisted of 40 speakers, with the testing on 30s matched conditions. In their studies, they compared the results of the mapping approach with the GMM-based approach for speaker verification.

The main idea in the mapping approach is to capture the speaker-specific information. This is achieved by mapping a set of parameter vectors specific to LI in the speech to the parameters having LSI. Let us assume that for a speaker a parameter vector can be extracted from a segment of his speech such that it contains the LI. Again, let us assume that from the same segment of speech, a parameter vector containing the LSI can be obtained. If a mapping function can be found that transforms the vectors containing LI to the vectors containing LSI, then that mapping function will be specific to the speaker. The goal in speaker recognition is to find a mapping function for every speaker that is unique and specific to him/her.

What parameter vectors can be extracted from a speech segment that satisfy the property of LI and LSI? As mentioned in Section 1.3.4, the choice of parameter vectors is of importance in a speaker recognition system. The parameters suitable for the speaker-specific mapping are not explored well in the literature. Besides, the ambiguous terms LI and LSI in the speech signal are difficult to characterize quantitatively. LP analysis provides some clues to obtain parameters with the above mentioned information embedded in them, namely LI and LSI. The next section gives the qualitative representation of LI and LSI in terms of the LP analysis order.

3.2 QUALITATIVE REPRESENTATION OF LI AND LSI

The order of LP analysis determines the number of peaks of an all-pole system. A complex pole pair accounts for one resonant peak, and the real poles account for the roll-off of the spectrum. A lower order LP analysis leads to a spectrum with the gross features of the spectral envelope. Speaker information is lost in such a representation, while linguistic information is preserved. In contrast, a higher order LP analysis captures the finer details of the spectrum that might be specific to a given speaker. Such a representation retains the linguistic as well as speaker information. The drawback of the LP analysis is, if the order is very high then the model starts picking up spurious peaks. When the model is adequate to represent the LSI, and not picking up spurious peaks, is difficult to decide. If the model is adequate, it will capture the LSI in terms of the finer details of the spectrum. Fig.3.1 shows the results of such qualitative analysis. The speech utterances sampled at 16 KHz were collected from two speakers (1 female and 1 male) over a microphone. Both the speakers spoke the vowel /a/ four times. Steady section of the utterances were taken for the LP analysis. A frame size of 20 ms was considered, and Hamming window was applied to the samples after pre-emphasis (the pre-emphasis factor was 1). Figs.3.1(a) and (b) spectra belong to the female speaker, obtained by performing 6^{th} and 14^{th} order LP

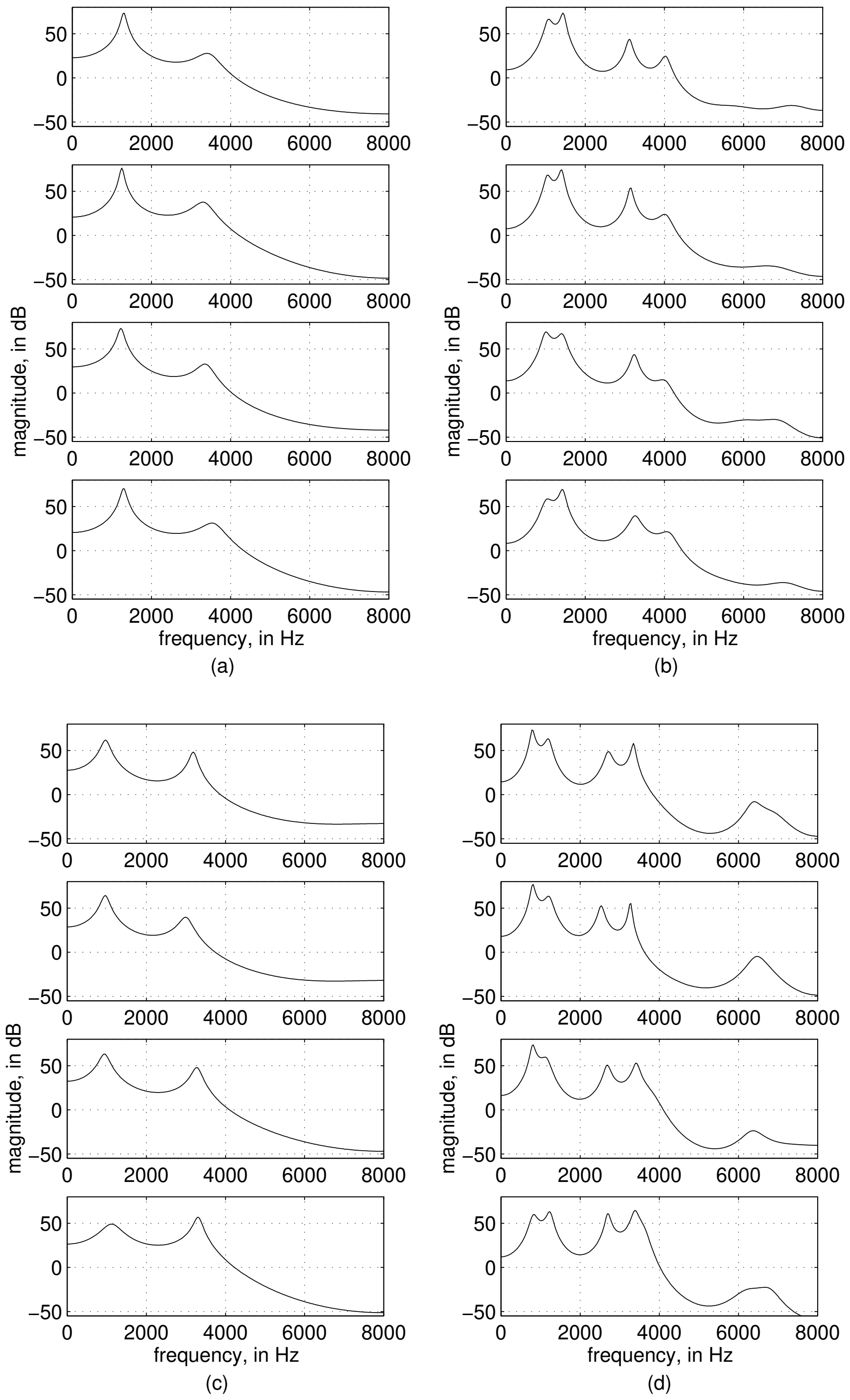


Fig. 3.1: (a) and (b) are 6^{th} and 14^{th} order LP spectra for 4 different utterances of the same vowel sound (/a/) by a female speaker, respectively. (c) and (d) are similar spectra for a male speaker. Segment size is 20 ms and sampling frequency is 16 KHz.

analysis, respectively. Each spectrum pair was obtained for the same segment of an utterance. Similarly, Fig.3.1(c) and (d) spectra belong to the male speaker.

From the plots, we can draw the following conclusions: For the 6^{th} order LP analysis (a) The spectra for the different utterances of the same speaker are similar, and (b) The spectra of the two speakers are also similar. Similarly, for the 14^{th} order LP analysis: (a) The spectra for the different utterances of the same speaker are similar, but (b) the spectra of the two speakers are significantly different. These differences can be attributed to the speakers.

Fig.3.2 shows LP spectra for 4 different speakers. The speech utterances sampled at 16 KHz were collected from four speakers (1 female and 3 males) over a microphone. All the speakers spoke the vowel /a/ once. Steady sections of the utterances were considered for the LP analysis. A frame size of 20 ms was taken, and the samples were Hamming windowed after differencing. Figs.3.2(a) and (b) show the spectra obtained by performing 6^{th} and 14^{th} order LP analysis, respectively, on the four utterances obtained from the four speakers. Each spectrum pair belongs to one speaker such that the lower and higher order LP analysis were performed on the same segment of the utterances.

For the 6^{th} order LP analysis, the spectra for all the four speakers are similar, though small differences are there at the higher frequencies (Fig.3.2(a)). However, for the 14^{th} order LP analysis (Fig.3.2(b)), the spectra for different speakers are significantly different. The 6^{th} order LP analysis qualitatively validates the assumption of the parameter vector containing LI (for the same vowel sound, the spectra are similar for all the speakers). For the 14^{th} order LP analysis, the spectra of the four speak-

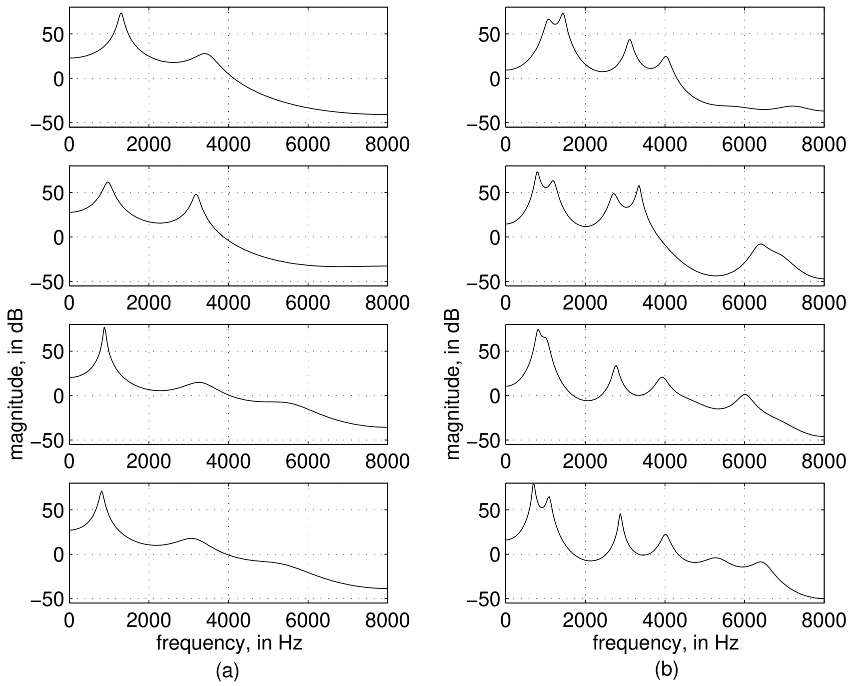


Fig. 3.2: (a) 6^{th} order LP spectra for the same vowel sound (/a/) spoken by 4 different speakers and (b) 14^{th} order LP spectra for the same segment as in part(a). Segment size is 20 ms and the sampling frequency is 16 KHz.

ers are significantly different. These variations can be attributed to the SI (which is specific for each speaker) because the LI is nearly same in all the four cases.

From the above examples two conclusions can be drawn: lower order LP spectrum (6^{th} in the present discussion), retains the LI while looses the SI. At the same time, if the order of LP analysis is high (14^{th} in the present studies), the spectrum envelope captures the finer details. These details are specific to a speaker. Thus, such a representation retains the LSI, where the SI is unique to each speaker. Therefore, the parameter vectors containing the LI and LSI can be extracted from a speech signal by choosing the proper LP analysis order.

3.3 MAPPING OF LI TO LSI

Let us assume that there exists a nonlinear mapping function, as shown in Fig.3.3(a), that transforms the input vector $\vec{X}$ to the output vector $X+Y$. For the mapping func-

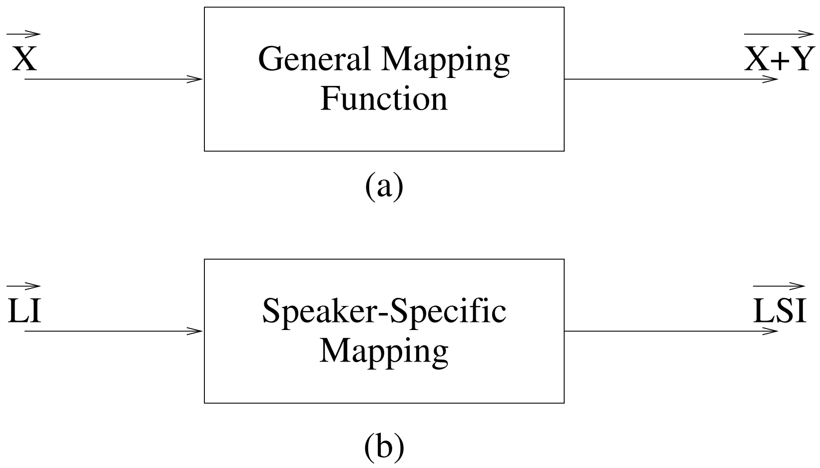


Fig. 3.3: (a) General mapping function that transforms $\vec{X}$ to $X+Y$, and (b) speaker-specific mapping function that transforms $\overrightarrow{LI}$ to LSI.

tion to achieve this, the mapping function should know about $\vec{Y}$. Extending the same logic, for the mapping function of Fig.3.3(b) to accomplish the task of transforming the vector $\overrightarrow{LI}$ to the vector LSI, the function should have the SI. The mapping function that can accomplish this task will be specific to the speaker from whose speech the $\overrightarrow{LI}$ and LSI are obtained, and termed as *speaker-specific mapping function*. Till now we assumed that such a speaker-specific mapping function exists. The next section explains how to derive this mapping function using the parameter vectors (training phase).

3.4 ANALYSIS OF MAPPING APPROACH

Let $\mathcal{F}(\cdot)$ denote a nonlinear mapping from LI to LSI. For N input-output vector pairs, the average error of mapping, $E(\mathcal{F})$, is given by

$$E(\mathcal{F}) = \frac{1}{N} \cdot \sum_{n=1}^{N} |\mathbf{o}_n - \mathcal{F}(\mathbf{i}_n)| \tag{3.1}$$

where, $\mathbf{i}_n$ vector is the LI, $\mathbf{o}_n$ vector is the LSI, and $|\cdots|$ is the norm of difference between the two vectors. The goal is to find $\mathcal{F}$ such that E is minimized. The assumption in this analysis is that $\{\mathbf{i}_n, \mathbf{o}_n\}$, $n = 1, \cdots, N$ are related by the same mapping function $\mathcal{F}(\cdot)$. As $\mathbf{i}_n$ is the LI and $\mathbf{o}_n$ is the LSI, the mapping function that minimizes E should be specific to the speaker from whose speech $\mathbf{i}_n$ and $\mathbf{o}_n$ are extracted. Hence, it can be used as a model for the speaker. The task is to derive this speaker-specific mapping function using $\{\mathbf{I}, \mathbf{O}\}$, where $\mathbf{I} = \{\mathbf{i}_1, \mathbf{i}_2, \cdots \mathbf{i}_N\}$ and $\mathbf{O} = \{\mathbf{o}_1, \mathbf{o}_2, \cdots \mathbf{o}_N\}$.

Let us assume that there exists a function $\mathcal{F}_k$ for the k^{th} speaker that performs a nonlinear mapping between $\{\mathbf{I}, \mathbf{O}\}$. Once the number of free parameters are fixed, $\mathbf{I}$ and $\mathbf{O}$ are fixed, then the free parameters can be obtained by,

$$\frac{\partial E(\mathcal{F}_k)}{\partial a_l^k} = 0 \qquad \forall l, \quad l = 1, \cdots, L \tag{3.2}$$

In the above equation, a_l^k are the free parameters to be determined in order to minimize $\mathcal{F}_k$ for speaker k. The number of such free parameters is L (fixed). The speaker model for the k^{th} speaker is given by

$$M_k = arg \begin{pmatrix} min & \\ \mathcal{F}_k & E \end{pmatrix} \tag{3.3}$$

At the time of identification, the speaker whose model gives the least error is adjudged as the identified speaker.

$$\widehat{k} = arg \begin{pmatrix} min & \\ k & E \end{pmatrix} \tag{3.4}$$

With the parameters suitable for the mapping approach being identified, the next task is to derive the mapping function. The nonlinear speaker-specific mapping function can be derived using MLFFNN since they are considered to be universal approximator [42, 43]. The next section discusses one of the MLFFNN structure used in the present studies.

3.5 REALIZATION OF MAPPING FEATURE

In Eq. 3.4, the attempt is to find a solution for $\mathcal{F}(\cdot)$ that minimizes E, the average error of mapping, in the mean-square-sum sense. It can be achieved using MLFFNN. In MLFFNN, the mean-square-sum is minimized using gradient descent algorithm. An MLFFNN structure used in a few experiments of the present studies is shown in Fig.3.4. In the present work, the mapping capabilities of MLFFNN are utilized.

The input and output layers of the structure shown in Fig.3.4 consist of 19 linear units each. The first hidden layer has 30, while the second has 10 nonlinear units. Nonlinearity is "tanh" function in both the cases. The input-output parameter vectors are 19-dimensional each for the given structure. The parameter vectors obtained from the training data are used to adjust the weights using backpropagation learning law. Models are derived for each speaker using all the frames of that speaker's training data. For testing, the input parameter vector is used as input to the NN. The difference

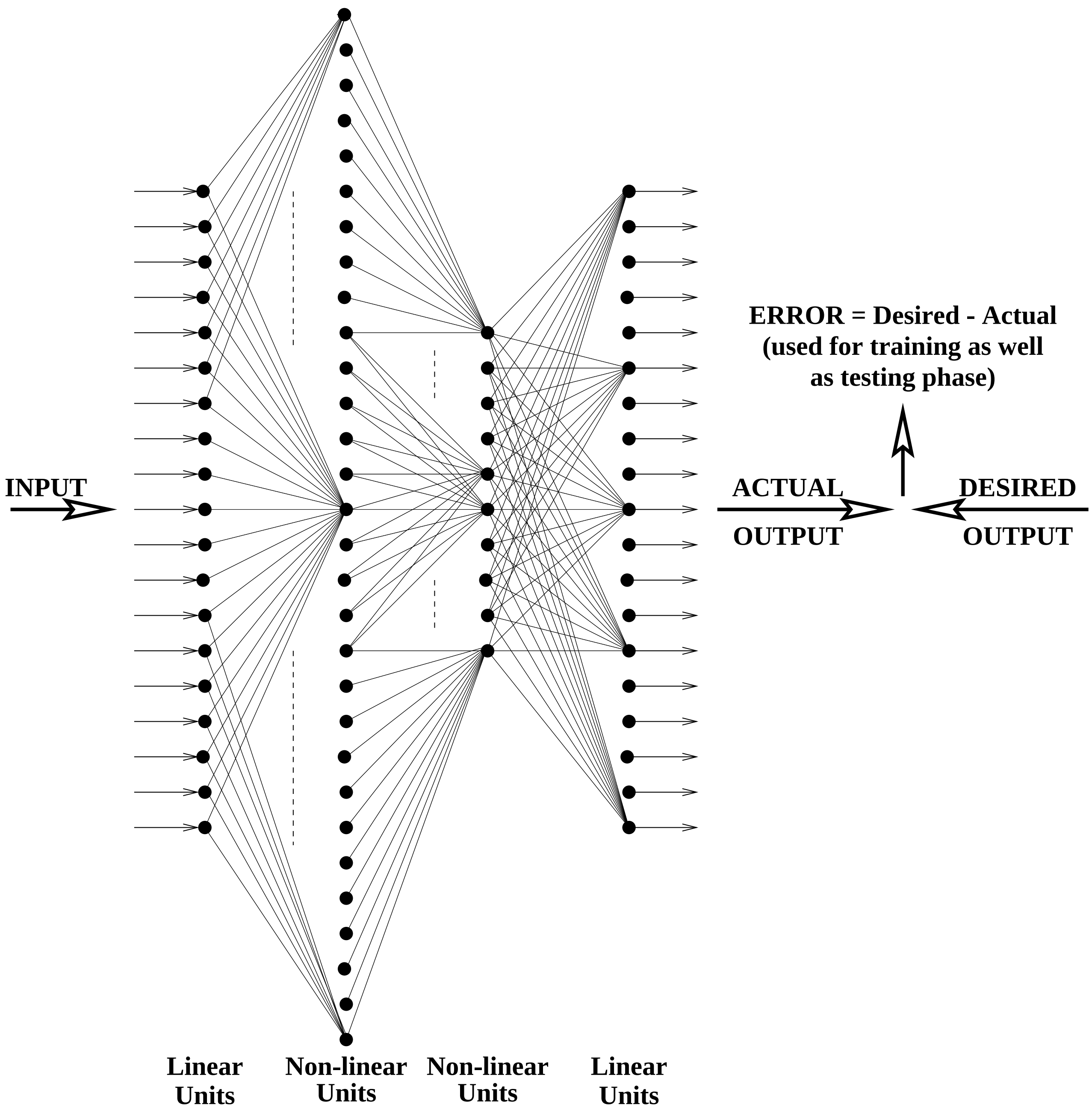


Fig. 3.4: A representative structure of multilayer feedforward neural network model to capture "Speaker-specific Mapping Feature".

between the desired output parameter vector and the actual output vector of the NN is used as a distance of that frame. The total accumulated distance averaged over all the frames gives an indication of the proximity of the test utterance to the speaker model. The test set data is used to study the performance of the speaker models for identification and verification. For the identification task, the parameter vectors of the test utterance of each speaker are given to each of the models. The speaker corresponding to the model that gives the least distance is declared the identified speaker (Eq. 3.4). For the verification task, the performance is expressed in terms of EER. The EER is computed using the $K \times K$ matrix of distances obtained such that the test utterance of each speaker is tested on all the speaker's models. K is the number of registered speakers. The threshold for EER is derived from the distance matrix.

3.6 SUMMARY

In this chapter, we introduced the hypothesis of mapping approach, and discussed the parameter vectors suitable for the mapping approach. The concept of parameter vectors containing LI and LSI was proposed. We showed that such a representation is possible by choosing the proper LP analysis order. The analysis of the approach was presented. Motivated by the mapping capabilities of MLFFNN, we proposed them for capturing the speaker-specific mapping.

Chapter 4

PARAMETERS FOR SPEAKER-SPECIFIC MAPPING

The concept of LI, SI and LSI was explained in the last chapter. In order to explore the parameter vectors that might be suitable for the mapping approach, we studied the following two parameter vectors:

1. Autocorrelation Coefficients [Section 4.1]
2. LPC derived Cepstral Coefficients [Section 4.2]

The autocorrelation coefficients define the short-time spectrum of the speech, and are nonparametric features. In contrast, the LP analysis is a model-based approach, and the LPCs are parameters of an all pole model system. The limitations of both the parameter vectors, and their suitability for the mapping approach, are discussed in the next two sections of this chapter.

4.1 AUTOCORRELATION COEFFICIENTS AS PARAMETERS

The concept of LI and LSI, as in the case of LP analysis, can be extended to the autocorrelation coefficients also. Some preliminary experiments to explore the possibility of using the autocorrelation coefficients as a parameter vector are presented in this section.

In the following experiments, the input and the desired output parameter vectors are autocorrelation coefficients. The dimensions of the input and output parameter vectors are I and $O(> I)$, respectively. The spectrum of the input parameter vector will be smooth. Such a representation will retain LI, while loosing SI. The finer details captured by the spectrum of the output parameter vector might be specific to the speaker, and thus corresponds to LSI.

The following procedure is adopted to extract the autocorrelation parameter vectors from the speech data:

1. **Pre-emphasis:**

$$y[m] = s[m] - s[m-1]$$

where $s[m]$ is the m^{th} speech sample.

2. **Frame Blocking:**

$$y_l[m] = y[M \cdot l + m]$$

where $m = 0, 1, \cdots, M-1$ and $l = 0, 1, \cdots, L-1$

3. **Windowing:**

$$\tilde{y}_l[m] = w[m] \cdot y_l[m] \qquad m = 0, \cdots, M-1$$

where $w[m]$ is a Hamming window of size M, given by

$$w[m] = 0.54 - 0.46 \cdot \cos\left(\frac{2 \cdot \pi \cdot m}{M-1}\right) \qquad m = 0, \cdots, M-1$$

4. **Autocorrelation Analysis:**

$$R_l[k] = \sum_{m=0}^{M-1-k} \tilde{y}_l[m] \cdot \tilde{y}_l[m+k] \qquad k = 0, 1, \cdots, p \tag{4.1}$$

where p is the dimension of the parameter vector required. $p = I$ for the input parameter vector and O for the output.

5. **Input-Output Parameter Vectors:**

$$\mathbf{i}, \text{ input parameter vector } = k \cdot r[k] \cdot \rho^k \qquad k = 1, \cdots, I \tag{4.2}$$

and

$$\mathbf{o}, \text{ output parameter vector } = k \cdot r[k] \cdot \rho^k \qquad k = 1, \cdots, O \tag{4.3}$$

where $r[k] = \frac{R[k]}{R[0]}$, and ρ is a forget factor (≤ 1). The forget factor is used to suppress the higher coefficients smoothly.

The use of $k \cdot r[k]$ is prompted by the following analysis:

$$R[k] \longleftrightarrow S(e^{jw}) \tag{4.4}$$

where $R[k]$'s are the autocorrelation coefficients, and $S(e^{jw})$ their Fourier transform (that is, power spectrum of the signal) given by [54]

$$\begin{aligned}
S(e^{jw}) &= \sum_k R[k] \cdot e^{-jwk} \\
\log S(e^{jw}) &= \log\left(R[0] + \sum_{k,k\neq 0} R[k] \cdot e^{-jwk}\right) \\
\log S(e^{jw}) &= \log\left(1 + \sum_{k,k\neq 0} \frac{R[k]}{R[0]} \cdot e^{-jwk}\right) + \log(R[0]) \qquad (4.5)
\end{aligned}$$

Taking the approximation $\log(1+x) = x - \frac{x^2}{2} + \frac{x^3}{3} \cdots$, and retaining the first term, x being small, gives

$$\log S(e^{jw}) \approx \sum_{k,k\neq 0} r[k] \cdot e^{-jwk} + \log(R[0]) \tag{4.6}$$

Differentiating both the sides w.r.t.w will give

$$\begin{aligned}
\frac{d}{dw}\left(\log S(e^{jw})\right) &= -j \cdot \sum_{k,k\neq 0} k \cdot r[k] \cdot e^{-jwk} \\
&= -\sum_{k,k\neq 0} k \cdot r[k] \cdot \sin(wk) - j \cdot \sum_{k,k\neq 0} k \cdot r[k] \cdot \cos(wk) \qquad (4.7)
\end{aligned}$$

The power spectrum can also be expressed as

$$S(e^{jw}) = |S(e^{jw})| \cdot e^{j \cdot arg[S(e^{jw})]}$$

Taking log on both the sides leads to

$$\log S(e^{jw}) = \log(|S(e^{jw})|) + j \cdot arg[S(e^{jw})]$$

Differentiating both the sides w.r.t.w give

$$\frac{d}{dw}\left(\log S(e^{jw})\right) = \frac{d}{dw}\left(\log(|S(e^{jw})|)\right) + j \cdot \frac{d}{dw}\left(arg[S(e^{jw})]\right) \tag{4.8}$$

Comparing the imaginary part of Eq. 4.7 and 4.8, we get

$$\begin{aligned} -\frac{d}{dw}\left(arg[S(e^{jw})]\right) &= \sum_{k,k\neq 0} k \cdot r[k] \cdot \cos(wk) \\ &= \sum_{k,k\neq 0} k \cdot r[k] \cdot Re[e^{-jwk}] \\ &= Re\left[\sum_{k,k\neq 0} k \cdot r[k] \cdot e^{-jwk}\right] \end{aligned} \tag{4.9}$$

The left hand side of Eq. 4.9 is the group-delay spectrum obtained by taking the Fourier transform of the indexed normalized autocorrelation coefficients, and retaining the real part(Re denotes the real part). The motivation for using $k \cdot r[k]$ as a feature vector is by the above equation of the group-delay spectrum [54].

While using the autocorrelation coefficients as a feature vector, 10 speakers (5 females and 5 males) of the NTIMIT[1] [55] database were used. Out of the 10 utterances of each speaker, 7 were used for training the speaker model and 3 for testing them. A frame-size of 20 ms and frame-shift of 10 ms were used. The speech data was pre-emphasized and windowed by the Hamming window before extracting the input-output parameter vector pairs.

[1]Appendix B

The neural network structure for the studies was IL10N30NOL, where L denotes linear units and N represents nonlinear units. I and O are the dimensions of the input and output vectors, respectively. The number preceding L (or N) denotes the number of nodes in a particular layer.

Table-4.1 shows the performance of the mapping approach using the autocorrelation coefficients as the parameter vectors. In the table, I and O represent the input

Table 4.1: Verification performance (in terms of EER) by mapping approach. Input-output parameter vectors are autocorrelation coefficients; Number of speakers is 10; Network structure is IL10N30NOL

$\frac{O\longrightarrow}{I\downarrow}$	6	8	10	12	14	16	18	20
6	29.0	33.3	30.0	21.1	20.0	20.0	15.0	17.0
8	-	39.0	30.0	29.0	20.0	20.0	13.3	12.5
10	-	-	36.7	23.3	23.4	12.2	12.2	12.2

and output parameter vectors dimensions, respectively. The entries in the table are the EER for the corresponding input-output parameter vector pairs.

From the results, the following analysis can be drawn:

1. In each row, as the output parameter vector dimension is increased, the EER decreases. This validates the hypothesis that if the input parameter vector is specific to the LI, and the output contains LSI, the network captures SI. As the SI at the output is enhanced (by increasing the output order), the mapping

becomes stronger, and the performance improves.

2. The EER of 12.2% for such a small database is not satisfactory, but it validates the concept of speaker-specific mapping feature. Though the mapping exists, it is weak. The cause of mapping being not strong can be attributed to the choice of parameter vectors used in the present study. The input parameter vector is a subset of the output parameter vector.

3. The network structure is IL10N30NOL. As the input-output parameter vector's dimensions change, the network structure itself changes. Because of this implicit change in the network structure, a comparison in terms of the parameters is not possible.

The two shortcomings of the autocorrelation coefficients can be compensated, to a certain extent, by the LPC cepstral coefficients. The next section covers the experimental results and their analysis, when the LPC cepstral coefficients are used as a parameter vector for the mapping approach.

4.2 LPC CEPSTRAL COEFFICIENTS AS PARAMETERS

LP analysis is a model-based approach, and the LPCs are parameters of an all pole model system. The LP analysis yields good approximation for inertial systems. In the case of a speech signal, the LP analysis gives better approximation for a voiced segment of the speech signal as compared to an unvoiced one. The inherent problems of any model-based approach for extracting the parameters are present in the LP analysis too. If the model doesn't fit the data well, the parameters obtained are not reliable. In the present section, the experiments using the LPC cepstral coefficients as a parameter vector are reported.

Let us assume that the order of LP analysis for the input and output vector is N and M, respectively. 20 cepstral coefficients were obtained from the input as well as output LPCs. The 0^{th} cepstral coefficient, being zero, was ignored, and the remaining 19 coefficients were used as a parameter vector. Even though the dimensions of the input and output parameter vector were same (19), their order of the LP analysis were different. The problem encountered while using the autocorrelation coefficients as the parameter vectors (Section 4.1: the input parameter vector is a subset of the output parameter vector) is not present when the LPC cepstral coefficients are used. Moreover, the comparison of parameter vectors is possible because the network structure is independent of the LP analysis order (the variable in the experiments).

The first four stages to extract the LPC derived cepstral coefficients are the same as in the case of autocorrelation coefficients. However, there is a slight modification at the fourth stage. Instead of p being the dimension of the parameter vectors, it is the order of the LP analysis in the present studies. The remaining stages to extract the parameters are as follows:

5. **LPC Analysis (Durbin's Method):**

$$E^{(0)} = R[0] \tag{4.10}$$

$$k_i = \frac{R[i] - \sum_{j=1}^{L-1} \alpha_j^{(i-1)} \cdot R[|i-j|]}{E^{(i-1)}} \qquad 1 \leq i \leq p \tag{4.11}$$

$$\alpha_i^i = k_i \tag{4.12}$$

$$\alpha_j^i = \alpha_j^{(i-1)} - k_i \cdot \alpha_{i-j}^{(i-1)} \tag{4.13}$$

$$E^{(i)} = (1 - k_i^2) \cdot E^{(i-1)} \tag{4.14}$$

And the final solution is given by

$$a_m = \alpha_m^{(p)} \qquad 1 \leq m \leq p \tag{4.15}$$

where a_ms are linear predictive coefficients (LPCs).

6. **LPC Derived Cepstral Coefficients:**

$$c_0 = \ln \sigma^2 \tag{4.16}$$

$$c_m = a_m + \sum_{k=1}^{m-1} \frac{k}{m} \cdot c_k \cdot a_{m-k} \qquad 1 \leq m \leq p \tag{4.17}$$

$$= \sum_{k=1}^{m-1} \frac{k}{m} \cdot c_k \cdot a_{m-k} \qquad m > p \tag{4.18}$$

7. **Linearly Weighted Cepstral Coefficients:**

$$\tilde{c}_m = m \cdot c_m \qquad 0 \leq m \leq 19 \tag{4.19}$$

Weighting the cepstrum is also know as "liftering process". The linearly weighted cepstral coefficients were used as parameter vectors. This was done in order to emphasize the high frequency components which are believed to be more speaker-specific. Besides, it also reduces the "peakiness" of the spectrum envelope. The liftering process tends to reduce the sensitivity without altering the fundamental "formant" structure. The original sharp spectral peaks are highly sensitive to the LPC analysis condition (e.g., window position), and the resulting peakiness creates unnecessary sensitivity in spectral comparison [56].

In the following experiments, 38 speakers (14 females and 24 males) of dialect region "dr1" of the NTIMIT database were used. Out of the 10 utterances of each speaker, 7 were used for training and 3 for testing. A frame size of 20 ms, and frame shift of 10 ms were used. The network structure used for the studies was 19L30N10N19L (Fig.3.4). In the experiments, p (the order of LP analysis) was N and M for the input and output parameter vectors, respectively. Table-4.2 shows the identification performance of the mapping approach using the LPC cepstral coefficients. The entries in the table indicate the number of speakers correctly identified out of the

Table 4.2: Identification performance by mapping approach. Input-output features are LPC derived cepstral coefficients of dimension 20 (0 th coefficient is ignored); Number of speakers is 38; Network structure is 19L30N10N19L

LP ORDER for $\frac{OUTPUT\longrightarrow}{INPUT\downarrow}$	6	8	10	12	14	16	18	20
6	6	20	21	28	24	28	29	28
8	13	7	16	20	20	28	26	26
10	12	12	3	11	18	23	21	21
12	14	19	13	5	12	13	16	13
14	7	19	19	14	5	11	13	15
16	12	12	13	14	9	11	8	8
18	9	18	19	16	9	3	2	8
20	14	19	18	12	16	9	5	5

maximum 38 speakers.

From the upper-triangular matrix of the table, the following conclusions can be drawn:

1. For a given order of LP analysis for the *input* parameter vector, as the order of the LP analysis for the *output* is increased, initially the identification performance increases, reaches a maximum, and then starts decreasing. In terms of LI, SI, and LSI it means that the LI at the input is kept constant while the SI at the output is increased. More SI at the output gives a stronger speaker-specific mapping feature. However, if the order of the LP analysis for the output is very high, the model starts picking up spurious peaks. This leads to degradation in the performance. Fig.4.1 is the plot of the entries of 2^{nd} row of the identification table (Table-4.2) . As the order of LP analysis of the output feature vector

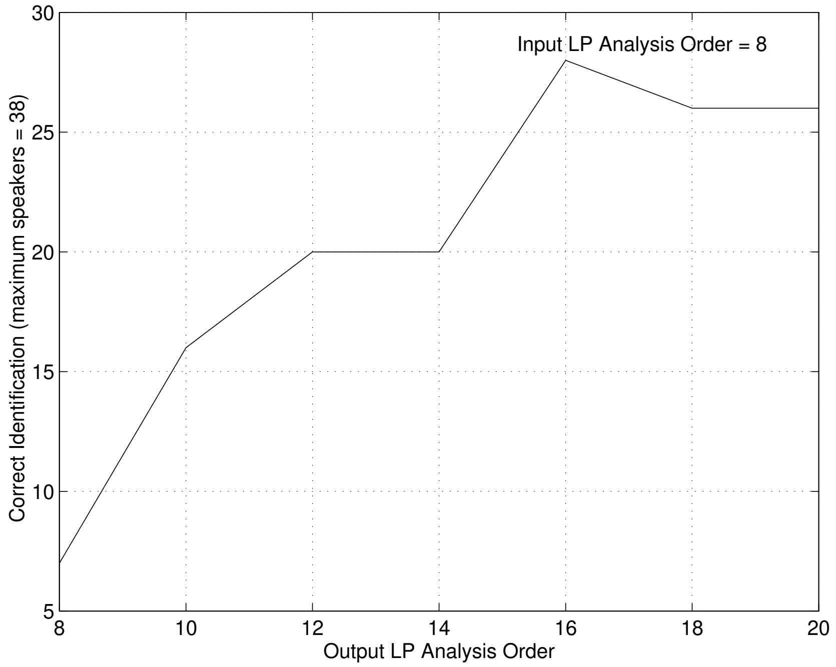


Fig. 4.1: Plot of the entries of a row (2^{nd} row) of the identification table (Table-4.2).

increases, keeping the input LP analysis order fixed at 8, the identification performance improves, reaches a maximum value (at the output order of 16), and then starts decreasing.

2. For a given order of LP analysis for the *output* feature vector, as the order of the analysis for the *input* increases, the identification performance decreases. In each column, from top to bottom, the LSI at the output is kept constant while the SI at the input is increased. More SI at the input weakens the speaker-specific mapping feature. Fig.4.2 is the plot of the entries of column 7 of the identification table. The identification performance decreases monotonically

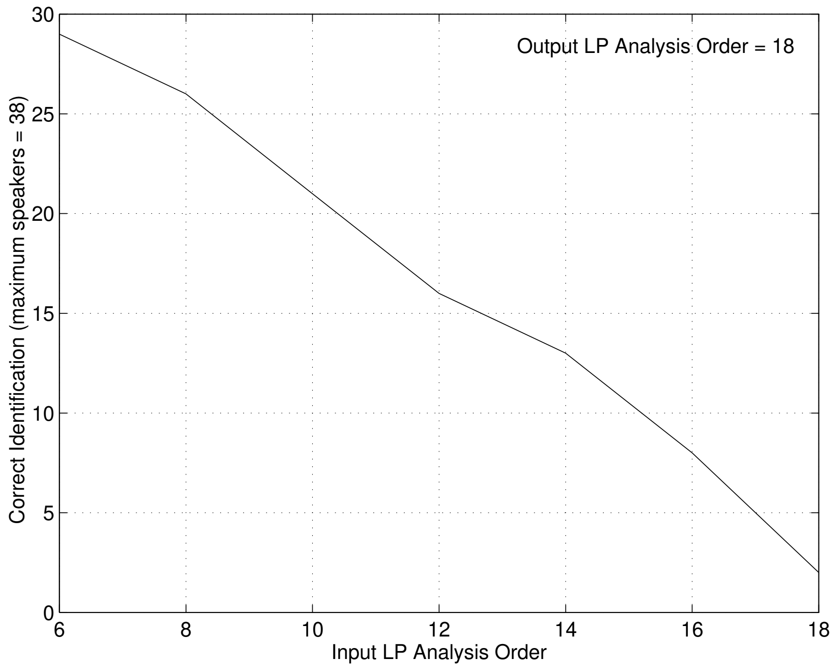


Fig. 4.2: Plot of the entries of a column (7^{th} col.) of the identification table (Table-4.2).

with increase in the input LP analysis order. The output order is fixed at 18.

The best result of 29 correct identification out of the 38 speakers is obtained for the 6^{th} order LP analysis for the input and 18^{th} order for the output parameter vector.

Like Table-4.2, there is a table for the verification results for the same set of experiments. Table-4.3 shows the performance of the mapping approach for the verification task. The entries in the table are the performance for a particular order of the input-

Table 4.3: Verification performance by mapping approach. Input-output features are LPC derived cepstral coefficients of dimension 20 (0 th coefficient is ignored); Number of Speakers is 38; Network structure is 19L30N10N19L

LP ORDER for $\frac{\mathcal{OUTPUT}\longrightarrow}{\mathcal{INPUT}\downarrow}$	6	8	10	12	14	16	18	20
6	27.3	24.1	21.1	15.8	13.2	15.8	18.5	20.4
8	21.1	31.2	24.5	15.8	18.5	21.1	23.7	29.0
10	23.7	22.6	33.6	23.7	23.5	26.4	28.5	29.0
12	20.3	20.7	22.9	36.7	26.3	26.4	30.9	29.0
14	23.7	25.8	18.5	20.3	32.7	28.6	29.5	29.0
16	24.5	20.5	21.1	19.3	28.7	31.6	28.6	27.8
18	27.9	23.7	17.9	16.3	21.9	28.4	36.3	29.0
20	27.0	22.7	23.0	22.4	23.7	27.5	29.0	26.4

output LP analysis pair in terms of EER.

The trends that were present in the identification table (Table-4.2) can be observed in the verification table (Table-4.3) also. From the upper triangular matrix of the verification table, the following conclusions can be drawn:

1. In each row, from left to right, the EER decreases (EER is an error, so the performance improves), reaches a minimum value, and then starts increasing. Fig.4.3 shows the representative plot of one of the row. In the figure, the input

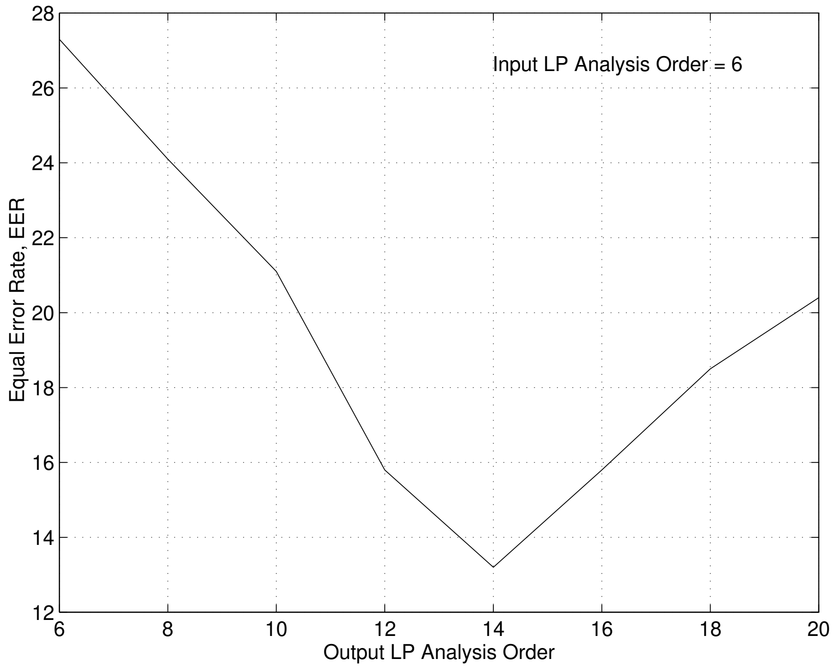


Fig. 4.3: Plot of the entries of a row (1^{st} row) of the verification table (Table-4.3).

order is 6 (fixed) and the output LP analysis order is increased. Initially the EER decreases, reaches a minimum value (at 14), and then starts increasing.

2. In each column, from top to bottom, the EER increases. Fig.4.4 shows the representative plot of one of the column entries. In the figure, the input LP analysis order is varied from 6 to 14, in steps of 2, while the output order is kept

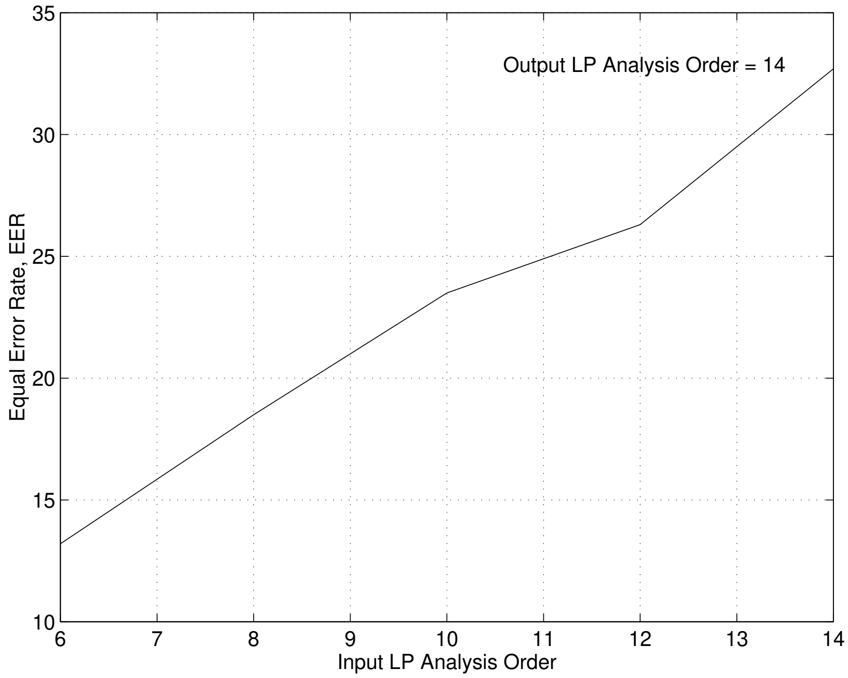


Fig. 4.4: Plot of the entries of a column (5^{th} col.) of the verification table (Table-4.3).

constant (at 14). The EER increases monotonically with increase in the input LP analysis order.

The best performance of 13.2% EER was obtained for the input LP analysis order being 6 and output 14.

It is not that the lower order LP analysis looses all the SI and retains only the LI. From Fig.3.2, though the 6^{th} order LP spectra look similar for different speakers, there are some variations in the spectra at higher frequencies. This might be the reason for the lower triangular matrix showing good performance in some of the cases (Tables-4.2 and 4.3). In the ideal situation of lower LP analysis loosing the SI completely, the performance would have been poor in the lower triangular matrix (the lower triangular matrix corresponds to the case when the input to the network is LSI and the desired output is LI).

4.3 SUMMARY

In this chapter we have shown that the hypotheses of LI at the input and LSI at the output of an ANN helps the ANN in capturing the speaker-specific mapping function. Two parameter vectors, the autocorrelations coefficients and LPC derived cepstral coefficients, were used to validate the hypothesis of speaker-specific mapping. We showed that the mapping function can be used as a feature for the speaker recognition task . Besides, the suitability of the two parameter vectors for the mapping approach was discussed. In the case of LPC derived cepstral coefficients as the parameter vectors, the optimal input-output pair $(6 - 14)$ was obtained using experimental studies.

Chapter 5

DEVELOPMENT OF A MODEL FOR SPEAKER RECOGNITION

In the previous chapter, the optimal LP analysis pair (input 6^{th} order and output 14^{th} order LPC cepstral coefficients) was obtained using experimental studies. It gave the best performance of 13.2% EER. From this point onwards, the thrust is on EER, not on the identification. Accordingly, most of the results are reported in terms of the EER, and the focus is to bring down the EER . The reason for doing so is: verification is independent of the population size, while identification depends on the size of the population. Yet another reason is that 0% EER implies 100% correct identification, but 100% correct identification needn't give 0% EER .

In Section 5.1, the hypothesis of background (BG) model and BG normalization are introduced. In Section 5.2, we explain the algorithm to generate the BG and speaker's model for the mapping approach. The concept of relative score and method to calculate it are discussed in Section 5.3. In the same section, we explain the criterion for taking decision based on the relative score. The BG normalization studies are split into 3 sections. In Section 5.4, the database used is the same set of 38 speakers of the dialect region "dr1" of NTIMIT as used in Section 4.2. The focus in this section is to demonstrate the effectiveness of BG normalization in improving the performance. In Section 5.5, the speaker set is different. In these experiments, 60 speakers (23 females and 37 males) of dialect region "dr2" of the NTIMIT database are used. The aim is to show that the BG normalization is effective for the other set of speakers too. The effectiveness of BG normalization is studied for two different MLFFNN structures. In Section 5.6, we study the role of nonlinear processing units and the MLFFNN structure in capturing the speaker-specific mapping.

5.1 CONCEPT OF BACKGROUND MODEL

In the previous chapter, the distances and scores to arrive at a decision were absolute. A speaker model didn't have information about the other speakers registered with the system. To get the score of a speaker, the Euclidean distance between the output of the speaker model and the desired output parameter vector was calculated. It was assumed that the output of the genuine speaker model will be closer to the desired output as compared to the other speaker's models. Since this distance is absolute, the model of a particular speaker has no knowledge about the distances produced by the models of the other speakers. With absolute scores, fixing a threshold is difficult. Besides, to determine when a speaker's score is small (or similarity is good enough) to declare him a genuine speaker, based on his own score, is difficult.

The concept of relative score is proposed to circumvent the problem of lack of reference. It is a widely used technique, and gives significant improvement in the performance [20,21,29]. The idea is to measure the score with respect to some common model derived from the information about the other speakers. Such a model is called as Background (BG) model.

5.2 METHOD OF GENERATING MODELS

A BG model is generated using the parameter vectors extracted from the speech utterances of a large number of speakers. In the present study, the whole population is used for generating the BG model. The steps involved in generating the BG model and speakers' models, when the concept of BG normalization is used, are:

1. The speech utterances of all the speakers are taken, and the input-output parameter vector pairs are obtained from them. A MLFFNN of appropriate size is trained with these pooled parameter vectors of all the speakers. The model so obtained is termed as the BG model. Some finer details regarding the BG model generation are discussed in the experimental setups.

2. The initial network for a speaker is obtained by initializing his model's weights by those of the BG model's weights. The speaker model is trained using the input-output parameter vectors of the speaker. The orders of the LP analysis for the input-output parameter vectors are kept the same as they were for the BG model.

The BG model obtained by the above mentioned procedure ensures that the model contains information about the whole population. A speaker model is fine tuned to the speaker, over and above the BG model, and thus is specific to the speaker.

The finer details regarding the generation of BG and speakers' models for the best performance are covered in the light of results obtained by the experimental analysis.

5.3 CALCULATION OF RELATIVE SCORE

The relative score for a speaker is obtained using

$$\text{relative score} = \text{BG model score} - \text{the speaker model score} \tag{5.1}$$

Three possibilities about the status of a test speaker are:

1. The test speaker is a true impostor, that is, he is not from the given population.
2. The test speaker is an impostor, however, he belongs to the population.
3. The test speaker is a genuine speaker.

In Case 1, assuming that the true impostor might be similar to at least one of the speakers of the population, the BG model will give a low score (distance). At the same time, the speaker model being fine tuned to the speaker will generate a high score (distance) for the impostor. The relative score will be *negative* in this case. This gives a measure of rejecting the true impostor.

In Case 2, the BG model will generate a lower score as compared to Case 1, while the speaker model will still give a high score. Again, it leads to a negative relative score, and helps in rejecting the impostor.

However, in Case 3, the situation reverses. As compared to the speaker model, that has been fine tuned to the speaker, the BG model gives a higher score. Consequently, the relative score obtained is positive. A high positive relative score means a high confidence that the test speaker is genuine. In contrast, when the BG normalization is not used, a low score implies more similarity (the desired output being much closer to the model output). This role reversal of score leads to an interesting change in FA/FR curve trace. The FA/FR curves *interchange* their positions when the BG normalization is used.

5.4 EFFECTIVENESS OF BG NORMALIZATION

As mentioned earlier, the whole population (in the present experiments, all the 38 speakers of the dialect region "dr1" of NTIMIT) was used for generating the BG model. All the 7 training utterances of the 38 speakers were taken, and the input-output parameter vector pairs were extracted. A frame size of 20 ms and frame shift of 10 ms was used. The order of LP analysis for the input and output parameter vectors were 6 and 14, respectively (optimal pair from Table-4.3). From this pool of parameter vector pairs, 1/6 of the total parameter vectors (in pair) were taken at random. The BG model (the structure of MLFFNN was 19L10N30N19L) was trained using them.

Table-5.1 shows the experimental results obtained with the above mentioned setup. The first two columns of the table are related to the BG model generation (**Iter** = the number of iterations or epochs; **Patt** = the total number of patterns or frames out of which 1/6 frames were selected for training the BG model). The last four columns of the table are specific to the speakers' models generation. The entries of columns 3 and 4 are the EER obtained when the *BG normalization was not used.* The last two columns correspond to the case when *BG normalization was used. Without BG normalization* is the situation when even though the BG model is generated it is not used for the normalization. That is, instead of relative scores, absolute scores are used for evaluating the performance.

The way speakers' models were trained over and above the BG model is the same

Table 5.1: Effect of iterations and randomization on BG "normalization". Number of Speakers is 38; BG is generated using 7 utterances from each speaker and choosing 1/6 of the frames at random; Network structure is 19L10N30N19L.

Iter.	Patt.	Without BG Normalization		Using BG Normalization	
		Sequential	Randomized	Sequential	Randomized
1	1	11.90	12.77	26.52	25.75
10	69303	11.45	12.44	12.91	09.79
20	69303	14.32	12.07	13.30	08.72
30	69303	14.41	13.16	13.16	07.90
40	69303	13.23	13.16	14.25	07.90
50	69303	14.24	13.29	13.50	07.90

for columns 3 and 5 (**Sequential**) and columns 4 and 6 (**Randomized**). The results of columns 3 and 5 were obtained when the speaker's parameter vector frames were used for training the speaker's model in the same sequence in which they were generated. Columns 4 and 6 results were obtained when the speaker's parameter vector frames were randomized first and then used for training the speaker's models.

The following observations can be drawn from the table:

1. If the BG normalization is not used (column 3 and 4), performance is independent of the iterations and number of patterns. The entries of column 3 do not show any trend. In column 4, initially the EER decreases, reaches a minimum

value, and then starts increasing. However, improvement in the performance is insignificant.

2. Column 5 and 6 indicate that the EER is high (performance is poor) if the BG model is not trained well (EER of 26.52 and 25.75, respectively).

3. If the speakers' models are trained sequentially, the advantage of BG normalization is lost (column 5).

4. If the BG model is trained well, and the frames of the speakers are randomized before training the speakers' models, improvement in the performance is significant (Column 6). Initially the EER decreases with increase in the number of iterations, reaches a value of 7.9%, and then it gets stagnant.

5. There is an improvement in the performance by 31.0% in the best case when the BG normalization is used as compared to the case when it is not (column 6, row 6 = 7.90% and column 3, row 2 = 11.45).

The concept of sequential training and randomized training is used due to the following two reasons:

1. Speech is a sequential phenomenon. Because it is the output of an inertial system the neighbouring frames are similar, and

2. MLFFNN learns only when there is an error between the output of the network and the desired output.

If frames are used sequentially, the effective number of frames are reduced. Let us assume that n^{th}, $(n+1)^{th}$, and $(n+2)^{th}$ frame are similar, and belong to the same acoustic event. The parameter vectors extracted from these 3 frames will also be similar. In MLFFNN, when the n^{th} frame is used for training, the error between the output of the network and desired output will be large. Weights will be adjusted to minimize the error for the n^{th} frame. Again, in the sequential training, the $(n+1)^{th}$ frame is presented to the network. Because the $(n+1)^{th}$ frame is similar to the n^{th} frame, and the weights were adjusted to minimize the error for the n^{th} frame, the error for the $(n+1)^{th}$ frame will be small. With small error, the weights adjustment will be less. Next, when the $(n+2)^{th}$ frame is presented for the training, the error will be much smaller and so will be the learning in terms of the weights adjustment. Effectively the training is done for $1-2$ frames instead of 3 frames if the frames are presented in the same order in which they are generated. Therefore, the effective number of frames is far less than the actual number of frames when the sequential training is adopted.

5.5 VALIDATION OF BG NORMALIZATION

In this section, the aim is to study the effect of BG normalization for a different set of speakers and for different structures of MLFFNN. Validation for the different speakers set is required to show that the BG normalization is not fine tuned to the speakers of dialect region “dr1”, but is effective for the other set of speakers too.

The database for the studies was all the 60 speakers (23 females and 37 males) of the dialect region “dr2” of NTIMIT. The whole population was used for generating the BG model. The method of generating the BG model was the same as it was in the previous section.

In Table-5.2, the entries of columns 1 and 2 are related to the BG model generation.

Table 5.2: Effect of structures, iterations and randomization on BG normalization. Number of speakers is 60.

Iter.	Patt.	19L30N10N19L		19L10N30N19L	
		Sequential	Randomized	Sequential	Randomized
1	109819	18.55	18.77	16.67	15.00
11	109819	13.34	09.06	15.00	15.00
21	109819	13.34	07.69	13.78	11.62
31	109819	13.50	07.72	14.72	08.41
41	109819	13.34	08.19	14.12	07.89
51	109819	14.29	08.33	15.00	08.06
61	109819	15.00	08.71	14.64	08.31

The results of columns 3 and 4 correspond to the structure 19L30N10N19L (expansion at 1^{st} hidden layer and compression as 2^{nd}), while 6 and 7 are related to the structure 19L10N30N19L (reverse of the first structure). In the experiments, out of the total 109819 parameter vector pairs (obtained from the 7 utterances of the 60 speakers), 1/6 were chosen at random for training the BG model.

Analysis of the results brings out the following points:

1. Sequential training of the speakers' models is not good (column 3 and 5). The performance hovers around 13%.
2. Both the structures give comparable performance in the best case (19L30N10N19L

gives an EER of 7.69% and 19L10N30N19L gives 7.89%).

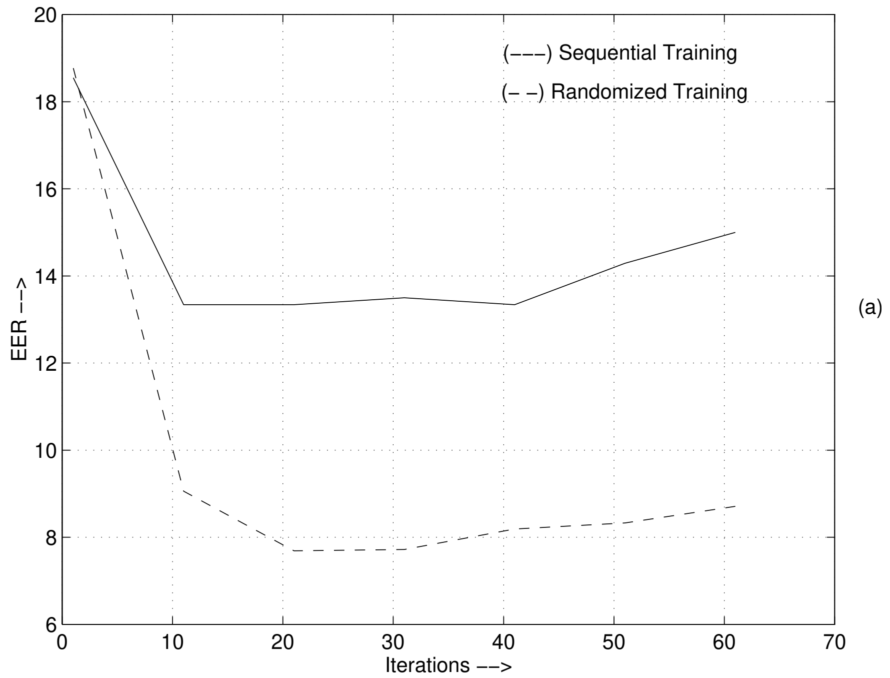
3. Structure 19L30N10N19L needs fewer iterations for the best BG

4. In the case of randomized training of the speakers' models, as the number of iterations are increased, the EER decreases (performance improves), reaches a minimum value and then starts increasing (column 4 and 6).

5. One important point that was not observed in Table-5.1 but comes out in these experiments is, if the BG model is over-trained the performance deteriorates. Both under-training as well as over-training of the BG model gives a poor performance.

The plots of EER Vs the number of iterations for the structure 19L30N10N19L and structure 19L10N30N19L are shown in Figs.5.1(a) and (b) respectively. In both the figures, the solid curves (——) represent the case of sequential training of the speaker's models, while the dotted (- - -) ones correspond to the randomized training.

Figs.6.1(a) and (b) show the plots of FA/FR Vs Threshold for the case when *BG normalization is not used* (solid —) and when *BG normalization is used* (dashed - -), respectively. Significant improvement in the performance is observed when the BG normalization is used (EER1 = 13.16%, EER2 = 7.90%). The FA/FR curves interchange their positions in the two plots. (when the BG normalization is not used, less score implies more similarity and more the confidence that a speaker is genuine. In contrast, when the BG normalization is used, more score means more the confidence that a speaker is genuine.).



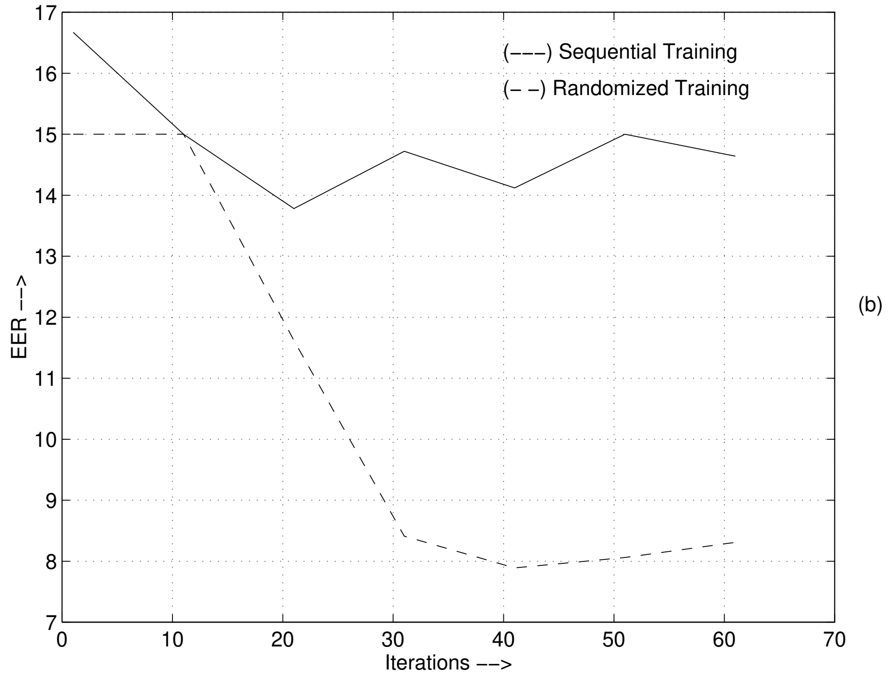


Fig. 5.1: Effect of iterations on BG normalization. Network structure is (a) 19L30N10N19L (b) 19L10N30N19L.

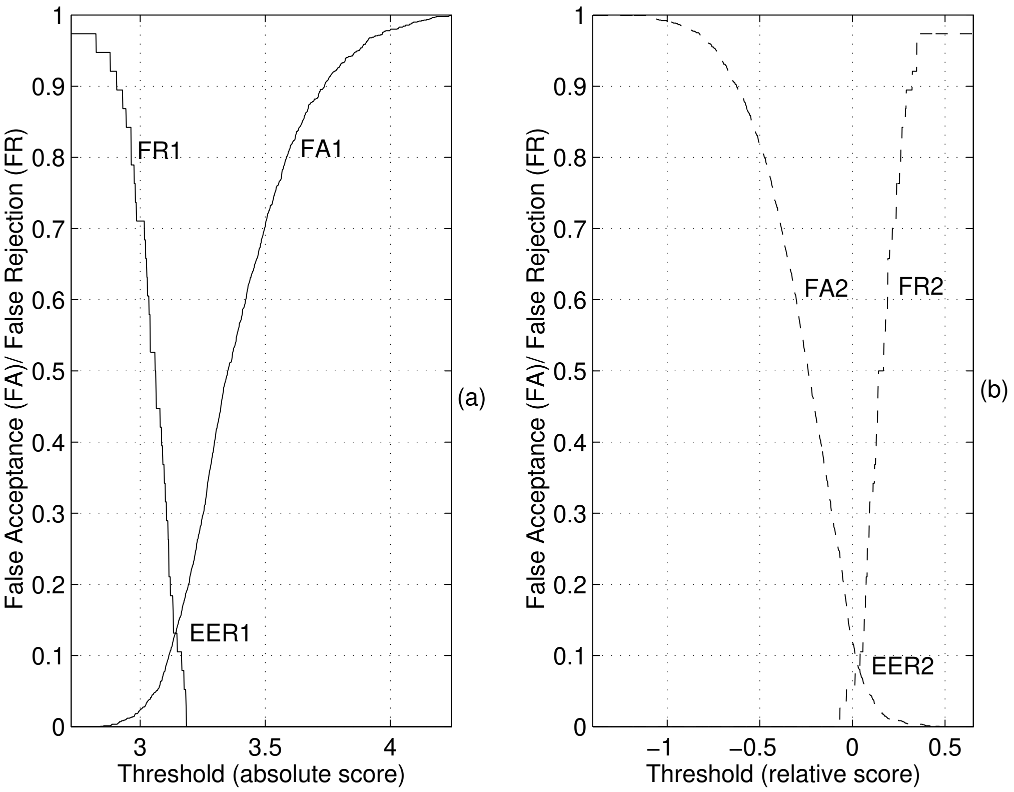


Fig. 5.2: Improvement in EER with Background Normalization. Equal error rate (a) *without BG normalization* (EER1) and (b) *with BG normalization* (EER2). Note that the False Acceptance and False Rejection curves have switched there positions.

5.6 ROLE OF NONLINEAR PROCESSING UNITS

In this section, we study the significance of nonlinear processing units in a network structure for capturing the speaker-specific mapping feature. The same set of 38 speakers of "dr1" region of the NTIMIT database was used. The input and output parameter vectors were 6^{th} and 14^{th} order LPC cepstral coefficients, respectively, of 19-dimensions each. In these experiments BG normalization was used, and speaker's frames were randomized before training the speaker's model. The network structure used was 19L10(N/L)30(N/L)19L (compression at the first hidden layer while expansion at the second). Table 5.3 shows the results of the experiments with the above mentioned setup.

Table 5.3: Nonlinear Hidden Layers Vs Linear Hidden Layers. Input is 6^{th} order LPC derived cepstral coefficients and output is 14^{th} order LPC derived cepstral coefficients. Number of Speakers is 38; BG Normalization is used; Frames are randomized for background as well as speakers' models generation.

Structure	Identification Result	EER
19L10N30N19L	32/38	7.90
19L10L30L19L	28/38	9.54

The following conclusions can be drawn from the results:

1. The performance is better if the network has nonlinear processing units in the hidden layers.
2. However, the performance is comparable for both the structures. Therefore, it is the structure of MLFFNN that gives the model the potential to capture the speaker-specific mapping feature [45]. Nonlinearity of the processing units enhances the effectiveness of the model [38] in capturing the speaker-specific mapping. Improvement in the performance is by 17.19% when the hidden layers have nonlinear units.

5.7 SUMMARY

In this chapter we introduced the concept of relative score and a method to obtain it using the Background (BG) normalization. The algorithmic details to implement the BG normalization for the mapping approach were discussed. Its effectiveness in improving the EER performance was demonstrated for two different sets of speakers and two different structures of MLFFNN. We studied the effectiveness of nonlinear processing units in the MLFFNN structure for capturing the speaker-specific mapping feature.

Chapter 6

FRAME SELECTION FOR PERFORMANCE IMPROVEMENT

6.1 MOTIVATION FOR FRAME SELECTION

In contrast to the problem of speech recognition, where are all the frames of a speech utterance are needed, it is not so in the speaker recognition task. Some frames contain significant speaker-specific information [22]. If such frames can chosen and used for taking a decision, the performance of the system will be better. In other words, if the frames that are confusing can be removed, then the performance will be better.

High energy frames, generally belonging to voiced segments of the speech, are supposed to contain speaker-specific knowledge. In most of the current day systems, energy is used to reject the silence frames. However, in the mapping approach, the energy based selection of the frames doesn't give much improvement in the performance of the system. In the present work, we suggest a *Network Error (Ne) criterion* to select frames having speaker-specific information.

In FA/FR curves, the aim is to bring down the point of intersection of the FA and FR curves. It can be achieved if the FA2 curve can be pulled towards left and the FR2 curve towards right (Fig.6.1(b), - - curves of FA2 and FR2). There is no simple way to achieve this. Still, moving both the curves towards right such that the relative

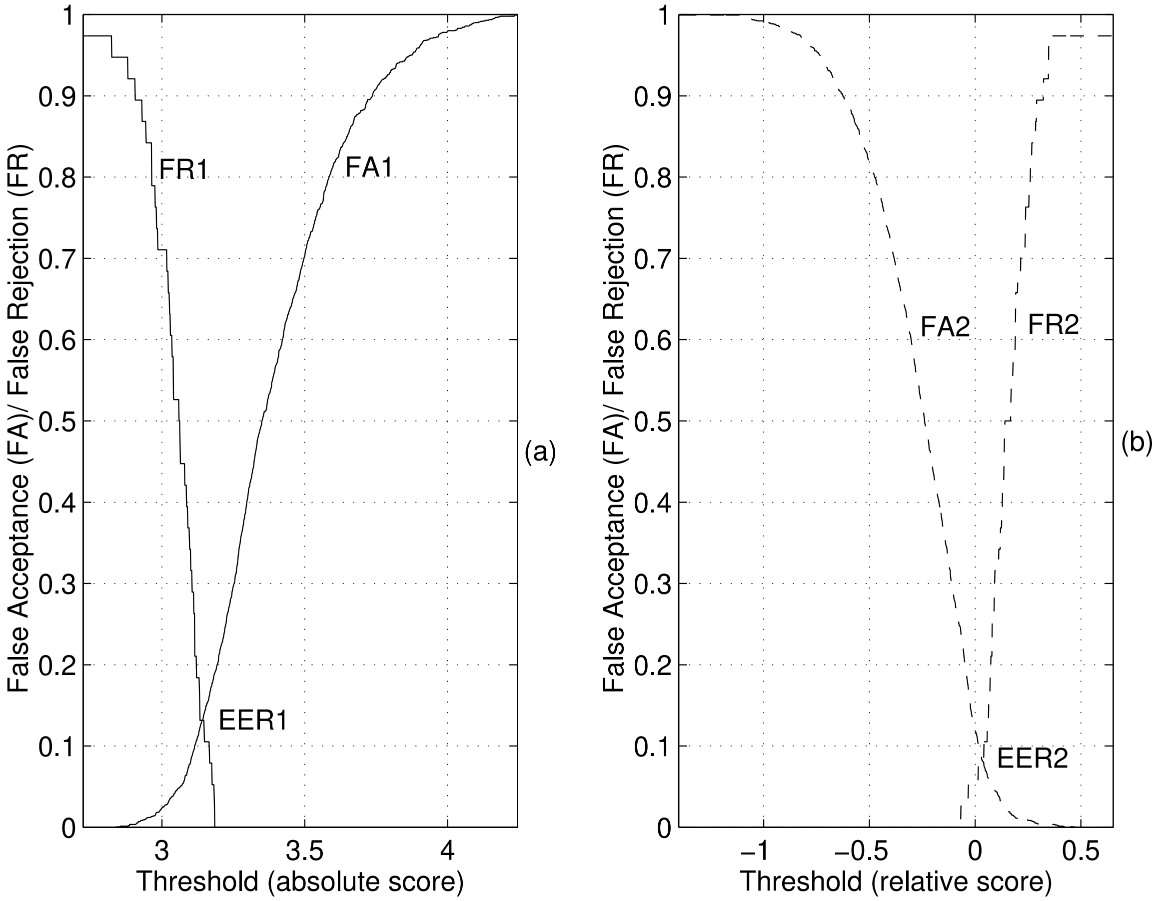


Fig. 6.1: EER (a) without BG normalization, and (b) with BG normalization.

displacement of the FA2 curve is less as compared to that of the FR2 curve can bring down the point of intersection of the FA2 and FR2 curve. *Ne* criterion achieves this by selecting speaker-specific frames.

The concept of *Ne* criterion is proposed for some other reason also. It is argued that one network might not be able to learn all the patterns. Let us assume that a network can learn l out of n patterns in the first few iterations. Weights for these l patterns are adjusted to reduce the error for them. The remaining (n-l) patterns still generate high error. If the weights of the network are adjusted to reduce the error for these (n-l) patterns, the error for the earlier l patterns increases. The mapping learned by the network for the l patterns might be destroyed if the weights are adjusted to reduce the error for the remaining (n-l) patterns. Therefore, it is proposed to have more than one network for each speaker. In the present studies, we have trained two networks for each speaker using the *Ne* criterion. How to divide the frames to train the two networks is an important issue. The algorithm and its implementation are explained in the next section. Section 6.3 covers the results obtained using the *Ne* criterion.

6.2 IMPLEMENTATION OF *NE* CRITERION

There are two phases in the implementation of *Ne* criterion: Training and Testing.

6.2.1 TRAINING PHASE

The *Ne* criterion is used along with the BG normalization. The algorithm presented in this thesis is applicable when both the methods are used simultaneously. The following is the algorithm to divide frames and obtain two networks for each speaker:

1. The speaker's model weights are initialized to the BG model's weights. This initial speaker model is trained using all the frames of the speaker (after randomizing them). However, this training is mild in the sense that fewer number of iterations are used for the training (nearly 50).

2. After this light training, all the training frames are used as test frames on the lightly trained speaker model.

3. The frames that give a high score are termed as "good frames", and are segregated from the "bad frames" (less score frames).

4. The lightly trained speaker model weights are copied to two models, "speaker.0" and "speaker.1".

5. The models "speaker.0" and "speaker.1" are trained using the good and bad frames, respectively. The number of iterations for both the models for this training phase are more (nearly 600).

6. At the end of the training phase, there are two models for each speaker, "speaker.0" and "speaker.1". Besides, there is a common BG model for all the speakers.

6.2.2 TESTING PHASE

At the time of testing, all the test frames are passed through the speaker model "speaker.0". The frames that give a high relative score (Eq. 5.1) are used for taking a decision. The low score frames are passed through the "speaker.1" model. Again the relative score is obtained and a decision is made. The low score frames generate a different score when tested with the "speaker.1" model.

6.3 PERFORMANCE STUDIES

It was found that the performance for the "speaker.1" models was not good, and therefore those results are not considered. This result is on the expected lines. The performance obtained by the low score frames is expected to be poor.

The entries of Table-6.1 show the verification performance when the *Ne criterion is used* and compared with the case when the *Ne criterion is not used.* The aim is

Table 6.1: Effect of (a) Sequential Vs Randomized training (on background model generation as well as speakers' models generation), (b) Number of frames on background model generation, and (c) *Ne* criterion. on the performance. Number of speakers is 38; Network structure is 19L10N30N19L.

Equal Error Rate						
$\frac{\mathcal{SPEAKER}\rightarrow}{\mathcal{BACKGROUND}\downarrow}$			Sequential		Randomize	
Training Mode	No. of Utter.	% of Frames	Without Ne	Using Ne	Without Ne	Using Ne
Sequential	2	100.0	12.53	10.53	11.71	09.89
Sequential	7	100.0	13.55	11.71	12.19	10.72
Randomize	2	16.7	12.68	08.38	12.62	08.30
Randomize	7	16.7	08.95	07.50	08.40	05.81
Randomize	2	100.0	10.52	07.74	07.90	05.45
Randomize	7	100.0	07.02	08.95	07.90	08.23

to study the effect of: (a) Sequential Vs randomized training (of the BG as well as speaker's model), (b) The number of frames used for the BG model generation, and (c) Using the *Ne* criterion on the performance of the system.

The database for these experiments consisted of 38 speakers from the dialect region "dr1" of NTIMIT. The frame size and frame shift were the same as in the previous experiments. The input and output parameter vectors were 19-dimensional cepstral coefficients obtained from 6^{th} and 14^{th} order LP analysis, respectively. Though 20 cepstral coefficients were obtained, 0^{th} coefficient was not used. The network structure used for the study was 19L10N30N19L.

In the table, the first three columns give information about the BG model generation. Rest of the columns specify the method of generation of the speakers' models on top of the BG model.

In the table, "Randomize" indicates that the frames were randomized before training the models. "Sequential" means that the frames were used in the same sequence in which they were extracted from the utterances.

The 2^{nd} column entries show the number of utterances of each speaker used to obtain pooled frames for training the BG model. The 3^{rd} column gives the information about the % of total pooled frames used for training the BG model.

Columns 4 and 5 correspond to the sequential training of the speakers' model(s) when *Ne criterion was not used* and when *Ne criterion was used*, respectively. Columns 6 and 7 are similar to 4 and 5, respectively, except that the speaker's models were trained after randomizing the speaker's frames.

The following conclusions can be drawn from the results:

1. The *Ne* criterion for the frame selection gives better performance in almost all the cases as compared to the cases when it is not used. The only exception to this is the last row entries of the table where the performance deteriorate if the *Ne* criterion is used. The reason for this behaviour is explained later.

2. As in the earlier cases, randomization of the speaker's frames for the speaker model training is always better as compared to the sequential training.

3. In the table, the different conditions used for generating the BG model are also given. It is noticed that the sequential training is not good for the BG model training. In the case of randomized training, following is the analysis: (a) If the number of frames used for the training are less (row 3: 2 utterances, 16.7% of the total frames), the performance is poor. (b) If the number of frames are adequate (row 4: 7 utterances, 16.7% of the total frames, and row 5: 2 utterances and 100% of the total frames), the performance is good. (c) If the number of frames are large (row 6: 7 utterances, 100% of the total frames), the performance is poor. Therefore, both under-training and over-training of the BG lead to the poor performance of the system.

4. It is observed from the results of row 6 that the performance in the particular case is poor when the *Ne* criterion is used as compared to when it is not. The reason for this behaviour is concerned with the Point 3 discussed above. The BG model is over-trained with the frames of the other speakers. If the *Ne* criterion is used to select the frames, the characteristics of the other speakers, via the BG model, reject the speaker-specific frames. Therefore, the BG model being over-trained plays a negative role.

   The plots of FA/FR curves (and their point of cross-section, that is, the EER) *with* (- - -) and *without* (——) the *Ne* criterion being used are shown in Fig.6.2. Both the curves (FA1 and FR1) move in the same direction (right hand side) when the *Ne* criterion is used. The displacement of FR1 curve is more as compared to FA1 curve. Consequently, the point of intersection of the FA2 and FR2 curves is lower as compared to the FA1 and FR1 curves.

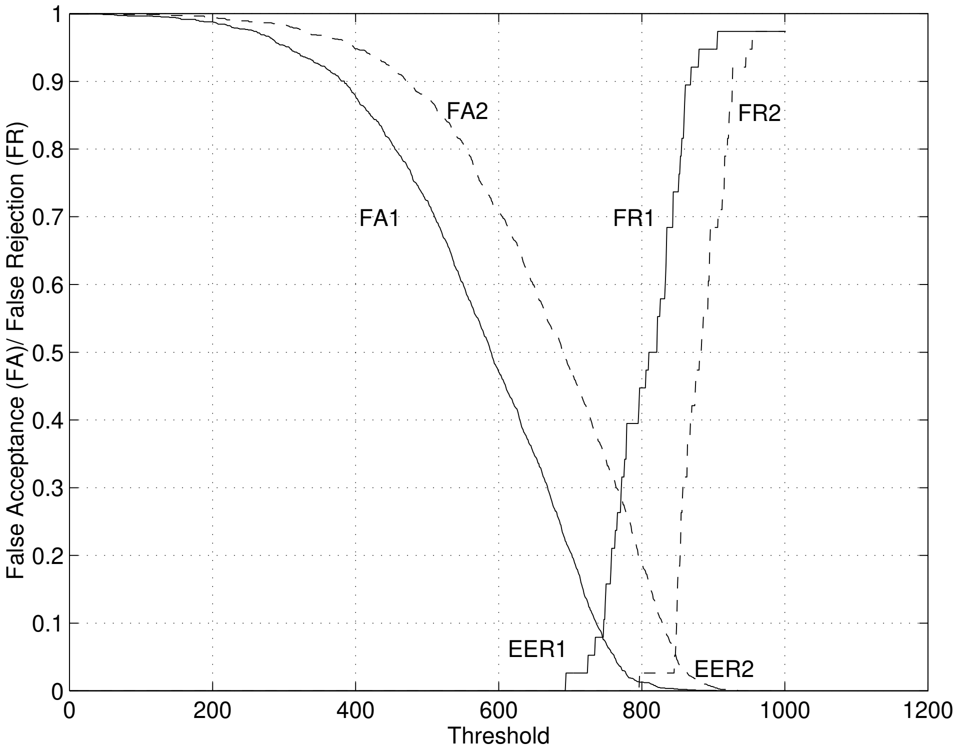


Fig. 6.2: Improvement in EER using the *Ne* criterion for selecting frames. Both the curves (FA1 and FR1) move in the same direction but the displacement of FA1 is more as compared to FR1. As a result EER2 is less than EER1.

6.4 SUMMARY

In this chapter we proposed the Network Error (*Ne*) criterion for selecting the frames containing speaker-specific information. The hypothesis, its implementation, and the performance using the *Ne* criterion were reported. We showed that significant improvement in the performance can be achieved if the *Ne* criterion for the frame selection is used properly.

Chapter 7

SPEAKER RECOGNITION STUDIES

7.1 BACKGROUND

In order to gain knowledge about the other aspects of mapping approach for speaker recognition, a few more studies were performed. In the present chapter, we study the performance of the mapping approach for perception based spectral parameter vectors, such as mel-cepstral coefficients and PLP derived cepstral coefficients. Next, the experiments to study the performance of the system as the number of test utterances are increased, are reported. We address the issue of robustness of the mapping approach to channel variations by comparing the performance with two different databases (TIMIT[1] and NTIMIT databases are used for this study).

7.2 COMPARISON OF SPECTRAL PARAMETER VECTORS FOR MAPPING APPROACH

After studying the performance for the LPC cepstral coefficients pair, the performance for other spectral parameter vector pairs is also studied. For this, the network structure was kept fixed (19L10N30N19L), and the same 38 speakers of region "dr1" of the NTIMIT database were used. BG normalization and *Ne* criterion gave improvement

[1] Appendix A

in the performance for the lpc-lpc mapping. Both the techniques were used in the experiments reported in this section to have a common test-bed for the comparative studies. In all the experiments, the output parameter vector was 14^{th} order LPC derived cepstral coefficients of dimension 20, with 0^{th} coefficient ignored. Three experiments, one for each spectral parameter vector, namely lpc, mel and plp, are reported in this section.

The results using the three different spectral parameter vectors are presented in Table-7.1 The first two columns in the table show the type of input parameter vector

Table 7.1: Comparison of various spectral parameters for mapping approach. Input feature vectors are changed while output feature vector is 14^{th} order lpc derived cepstral coefficients of 20 dimensions, with 0^{th} cepstral coefficient ignored. Number of speakers is 38; Network structure is 19L10N30N19L.

Input		Output		Sequential	Randomized
Feature	Order	Feature	Order	Training	Training
lpc	6	lpc	14	07.64	05.45
mel	19	lpc	14	13.16	11.78
plp	6	lpc	14	12.89	12.26

used and its order. The entry "plp-6" means 6^{th} order PLP derived cepstral coefficients of dimension 20 is the input parameter vector. To derive the mel-cepstral coefficients of order 19 (dimension of the input feature vector should be 19, the number of input units in the ANN structure), 19 filter banks were used. Column 3 and 4 indicate the output feature vector type and its order. It was same for all the three experiments.

From the results obtained, the following analysis can be made:

1. The ordering of the parameter vector pairs in terms of their performance is, lpc-lpc, plp-lpc and mel-lpc.

2. For all the three spectral parameter vectors, randomization of frames for the speakers' models generation yields better performance.

3. Though the lpc-lpc mapping gave the best result, still these experiments do not give the full picture. The whole domain was searched to find the best lpc-lpc pair. The same is not done for the other two parameter vector pairs, namely plp-lpc and mel-lpc.

7.3 PERFORMANCE DEPENDENCY ON NUMBER OF TEST UTTERANCES

In all the earlier experiments reported in this thesis, all the three test utterances were used to get the final score, and subsequently in evaluating the performance. The focus in this section is to study the change in the performance as the number of test utterances are varied [38]. The experiments are divided into the following two sets:

1. The network structure is varied while the input-output parameter vectors are kept same. [Section 7.3.1]

2. The network structure is kept constant while the order of the LP analysis of the output parameter vector is varied. [Section 7.3.2]

7.3.1 For different structures of MLFFNN

In the following set of experiments, the aim is to study the performance of different network structures as the number of test utterances are varied. The network structure used for the experiments was 19L**X**N30N19L, where **X** was varied form 2 to 10. The input and output to the network were 6^{th} and 18^{th} order LPC cepstral coefficients of 19-dimension each (0^{th} coefficient was ignored out of the 20 cepstral coefficients obtained). In Table-4.2, $6-18$ mapping gave the best identification performance (29/38 correct identifications). However, in Table-4.3, $6-14$ mapping gave the best verification result (13.2% EER). It was mentioned in Chapter 5 that the identification performance depends on the population size but the verification is independent of it. Keeping that in mind, the mapping used was $6-14$ in all the experiments. The BG normalization and *Ne* criterion were applied to this pair only. The goal was to improve the performance in terms of the EER. In the present studies, the $6-18$ mapping is considered to cross check whether the $6-14$ mapping is the optimal pair to obtain the best verification performance. The database for the experiments was the same 38 speakers of the "dr1" region of NTIMIT. Tables-7.2(a) and (b) show the % identification results and EER, respectively for the experimental setup explained above.

Analysis of the results leads to the following conclusion: For all the structures studied, as the number test utterances is increased, performance improves. The result is true for the verification (EER decreases) as well as identification (% correct identification increases) tasks. The reason is, as the number of test utterances are increased, more and more frames are involved in taking the decision. Because the final score is

Table 7.2: Performance as the number of test utterances increases. Structure is 19L**X**N30N19L, where **X** is varied from 2 to 10; Number of speakers is 38; Input is 6^{th} order LPC derived cepstral coefficients and output is " 18^{th}" order LPC derived cepstral coefficients. (a) % Identification (b) EER

X, No. of Units→ / No. of Test Utterances↓	2	4	6	8	10
1	50.9	67.5	63.6	62.3	58.8
2	69.7	80.3	79.0	82.9	77.6
3	79.0	84.2	85.5	84.2	89.5

X, No. of Units→ / No. of Test Utterances↓	2	4	6	8	10
1	17.0	12.0	14.0	12.4	13.0
2	13.0	10.5	09.4	10.2	10.52
3	10.5	08.0	09.0	07.97	08.67

the average of individual frame score, therefore statistics are involved. More frames give a better estimation, and subsequently a better performance. Figs.7.1 and 7.2 show the plots of the columns of Tables-7.2(a) and (b), respectively. (the number of test utterances is varied keeping the structure fixed). The vertical lines in the plots indicate the variation around the mean value. In both the plots, the horizontal axis represents the number of test utterances. In Figs.7.1 and 7.2, the vertical axis denotes % correct identification and EER, respectively.

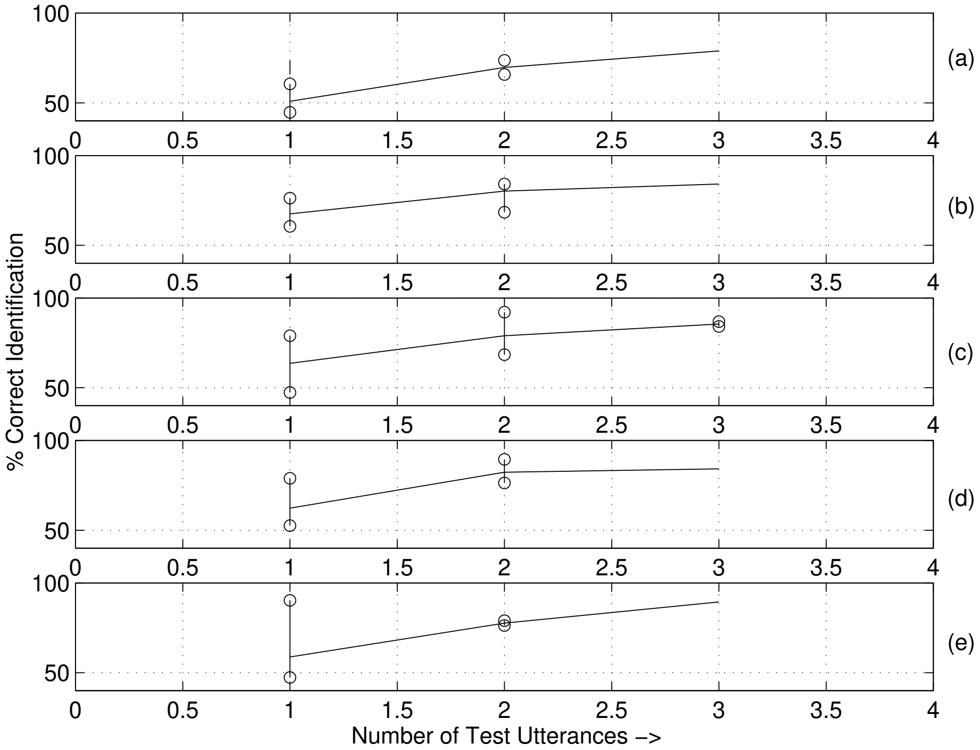


Fig. 7.1: Improvement in % correct identification as the number of test utterances are increased for different structures of MLFFNN (19L**X**N30N19L). **X** for (a), (b), (c), (d) and (e) are 2, 4, 6, 8 and 10, respectively.

7.3.2 For different orders of LP analysis for the output parameter vector

In this section, the aim is to study the performance of the mapping for different output parameter vectors as the number of test utterances are varied. In the following experiments, the network structure was kept constant (19L6N30N19L). The input parameter vector was 6^{th} order LP cepstral coefficients of 19-dimension. It was kept same for all the experiments. The order of the LP analysis for the output parameter vector was varied. The results obtained with the above mentioned setup are presented

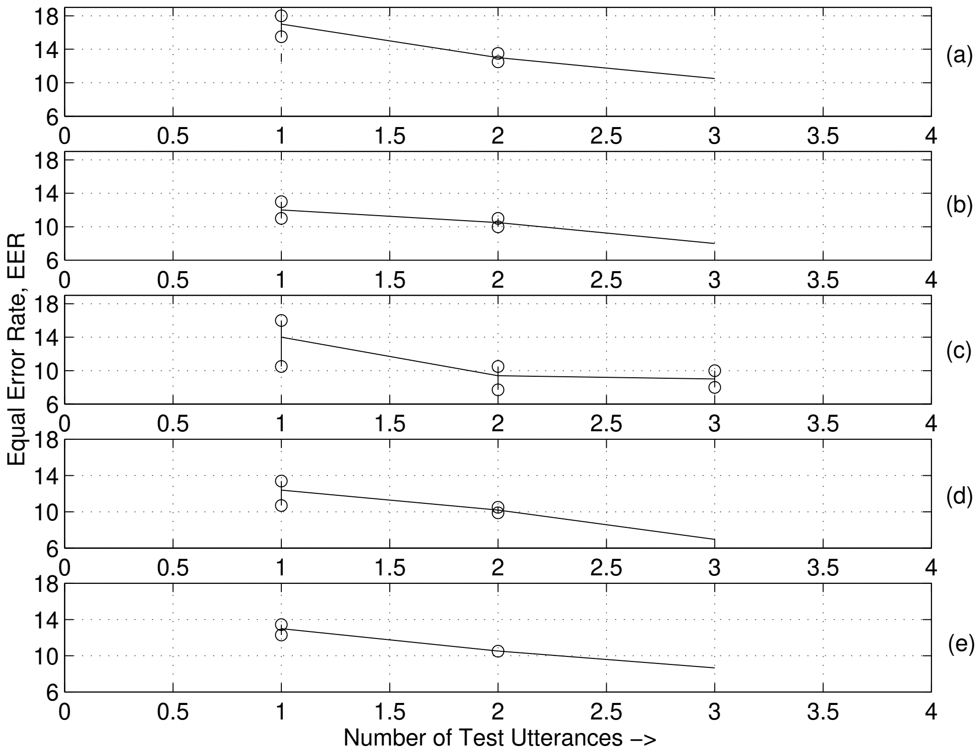


Fig. 7.2: Improvement in EER as the number of test utterances are increased for different MLFFNN structures (19L**X**N30N19L). **X** for (a), (b), (c), (d) and (e) are 2, 4, 6, 8 and 10, respectively.

in Table-7.3

Analysis of the results brings out the following points:

1. For all the LP analysis orders for the output parameter vector, as the number of test utterances are increased, the performance improves. Figs.7.3 and 7.4 show the plots of the columns of Tables-7.3(a) and (b), respectively. The vertical lines in the plots depict the variation in the performance around the mean value. In both the plots, the horizontal axis represents the number of test utterances.

Table 7.3: Performance as the number of test utterances increases. Structure is 19L6N30N19L; Number of speakers is 38; Input is 6^{th} order LPC derived cepstral coefficients. The output LPC order is varied. (a) % Identification, (b) EER,

Output LPC Order→ / No. of Test Utterances↓	13	15	18
1	51.75	59.65	63.60
2	72.37	81.58	78.85
3	84.21	81.58	85.53

Output LPC Order→ / No. of Test Utterances↓	13	15	18
1	15.30	13.90	14.00
2	11.00	08.87	09.40
3	08.00	07.90	09.00

In Figs.7.3 and 7.4, the vertical axis denotes % correct identification and EER, respectively.

2. In each row of Table-7.3(b), initially the EER is high, reaches a minimum value (more SI at the output), and then starts increasing (because of spurious peaks).

From the results of Sections 7.3.1 and 7.3.2, as the number of test utterances are increased (i.e., as the number of test frames are increased), the performance of the system improves.

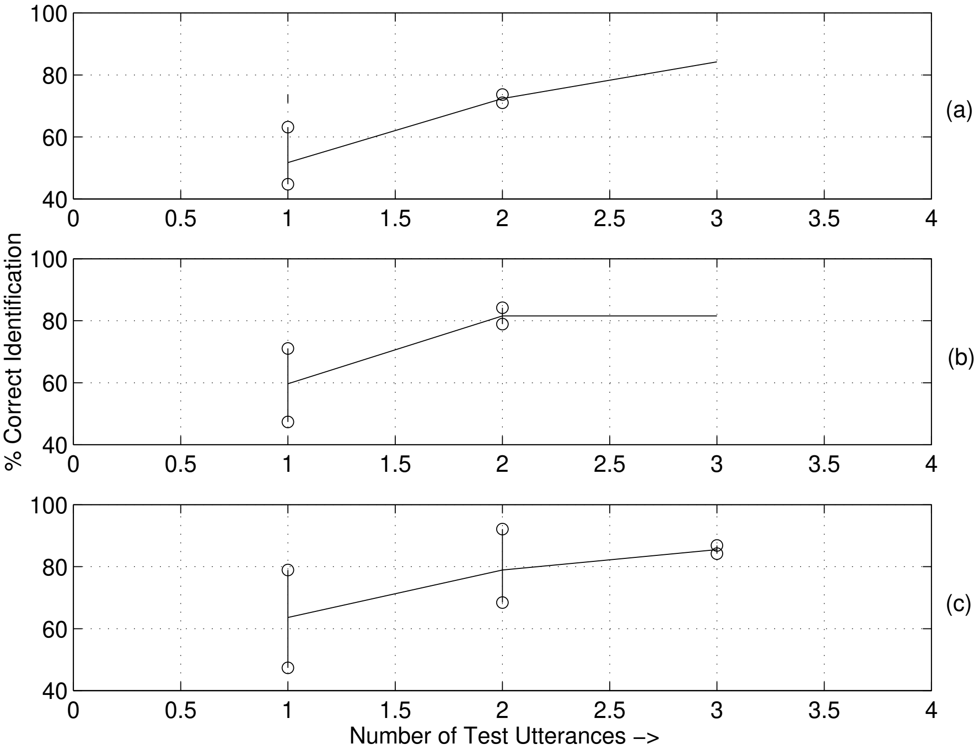


Fig. 7.3: Improvement in % correct identification as the number of test utterances are increased for different output LP analysis orders. Orders of LP analysis for the output feature vector for (a), (b) and (c) are 13, 15 and 18, respectively.

7.4 ISSUE OF ROBUSTNESS

In all the experimental studies presented, the database used was NTIMIT. The NTIMIT database was derived from TIMIT database [2]. In the studies presented in this section, the issue of robustness of the mapping approach to channel variations is addressed. The results using the TIMIT and NTIMIT databases are compared. The same 38 speakers of dialect region "dr1" of the TIMIT and NTIMIT databases are used for

[2] Appendix A

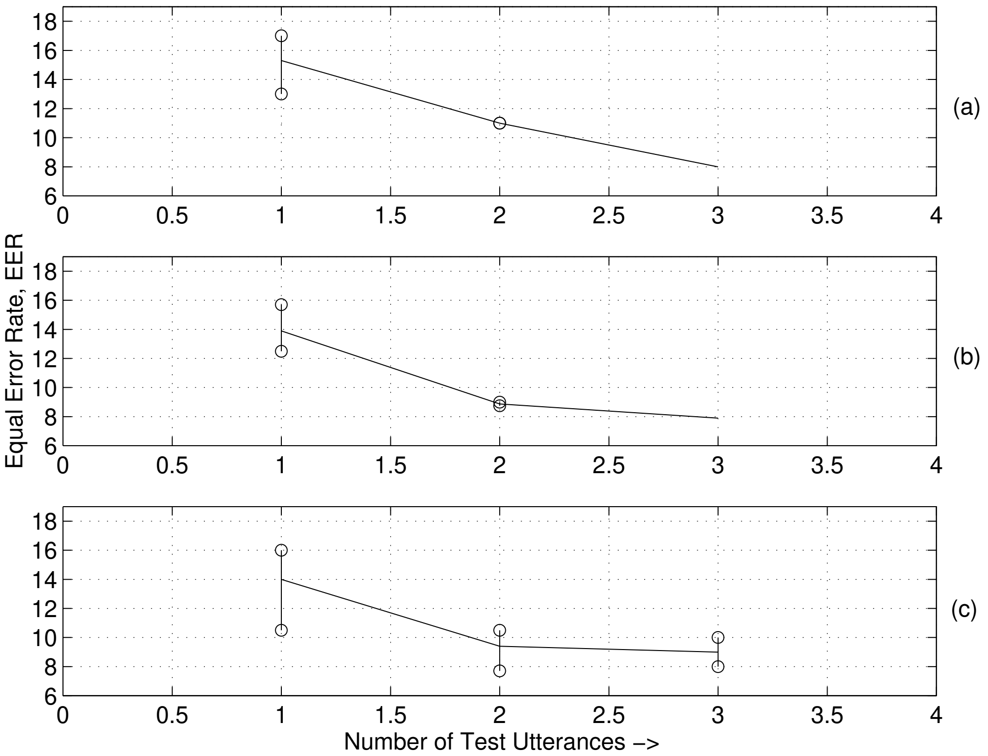


Fig. 7.4: Improvement in EER as the number of test utterances are increased for different output LP analysis orders. Orders of LP analysis for the output feature vector for (a), (b) and (c) are 13, 15 and 18, respectively.

these studies.

Three experiments were performed. In the first experiment, the TIMIT database was used. In the second experiment, the TIMIT database passed through an finite impulse response (FIR) low pass filter (LPF) of order 33 and cut-off 3560 Hz ($\approx 3300Hz$) was used. The purpose of the experiment is two-fold. First, to see the effect of removal of the high frequency components on the performance. Second, to check whether the mapping approach is robust to channel distortions. The frequency response of the filter is shown in Fig.7.5. In the series, the third experiment was

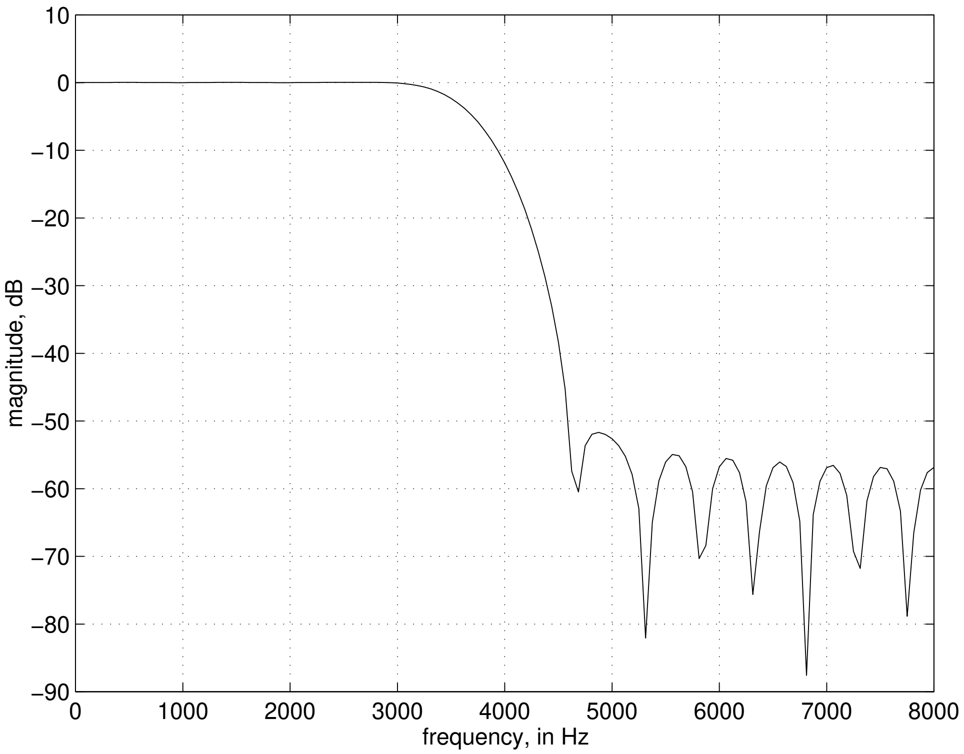


Fig. 7.5: Low Pass Filter used (3 dB cut-off at 3.6 KHz).

using the same set of 38 speakers of the NTIMIT database. The results of the three experiments are shown in Table-7.4. The EER obtained on the TIMIT database was 0.4% and identification was 100%. TIMIT database is clean speech collected with a microphone in a noise free environment and sampled at 16KHz. The same TIMIT utterances when passed through the LPF, gave an EER of 6.06% and 97.37% correct identification. For the same experimental conditions, the performance for the NTIMIT database were 5.45% EER and 73.68 % correct identification. The NTIMIT database is bandlimited to telephone channel.

The following conclusions can be drawn from these results: The effect of low pass

Table 7.4: Studies for different databases to show the robustness of mapping approach. Network structure is 19L10N30N19L; Number of speakers is 38.

Database	Processing	Identification	EER
TIMIT	Bandwidth of 8 KHz	38/38	0.40
TIMIT	Bandwidth of 3.6 KHz	37/38	6.06
NTIMIT	Telephone Channel	28/38	5.45

filtering is common to the TIMIT passed through a LPF of 3560 Hz and NTIMIT (the NTIMIT utterances were obtained by passing the TIMIT utterances through a telephone channel). The NTIMIT utterances have extra degradations because of the channel distortions. Since the performance of the system for the TIMIT passed through the LPF and NTIMIT are comparable, it seems the mapping approach is robust to the channel distortions.

The reason for the robustness might be attributed to the basic hypothesis of the mapping approach. The input and output parameter vector pairs are extracted from the same segment of speech for mapping. Therefore, the channel effects are common to both the parameter vectors. It is assumed that the information that is common at the input and output of the network is not learned by the network (Fig.3.3). Hence, the channel effects are nullified.

It seems from these preliminary results that the mapping approach is robust to the channel effects. In the mapping approach, MLFFNN learns the speaker-specific mapping. It is assumed that this mapping is not affected by the channel variations.

However, a few more experiments on different databases are required to confirm the robustness aspect of the mapping approach.

7.5 SUMMARY

In this chapter, the experiments to study the other aspects of the mapping approach were presented. We compared the performance of the mapping approach for the other spectral parameter vectors, namely mel-cepstral coefficients and plp cepstral coefficients. Improvement in the performance of the mapping approach with increase in the number of test utterances was analyzed. We studied the robustness aspect of the mapping approach to channel variations by experiments on the TIMIT and NTIMIT databases.

Chapter 8

CONCLUSION AND SUMMARY

8.1 SUMMARY OF THE WORK

In this thesis, a mapping approach was proposed for the task of speaker recognition. The approach is based on the following premise: If the input to a MLFFNN is linguistic information (LI) and the desired output is linguistic plus speaker information (LSI), both extracted from the same segment of speech, then the network will capture speaker-specific information (SI). It is assumed that the LI being common at the input and output of the network is not learned by the network. The mapping function captures the speaker-specific information. In the present work we showed that the mapping function can be used as a feature for the task of speaker recognition.

It was shown (using LP analysis) that the parameter vectors having LI and LSI can be extracted from a segment of speech. Two types of parameter vectors were studied. Both of them validated the hypothesis of speaker-specific mapping feature.

The LPC derived cepstral coefficients of order 6 at the input and 14 at the output gave good performance (EER of 13.2%). Moreover, the lpc-lpc pair gave better performance when compared with the other spectral pairs such as, mel-lpc and plp-lpc.

To improve the performance of the system, background normalization technique was proposed. The technique was developed and tuned for it to be used for the mapping approach. We demonstrated the effectiveness of BG normalization in improving the performance of the system. The factors that influenced the effectiveness of the BG normalization were: The number of patterns, number of iterations and the manner of training. The BG normalization, when used optimally, gave an EER of 7.9%.

In this thesis, we proposed a new method called as Network Error (*Ne*) criterion to select frames having significant speaker-specific information. When used together, BG normalization and Ne criterion yielded significant improvement in the performance. An EER of 5.45% was obtained when both the methods (BG normalization and *Ne* criterion) were used together.

While analyzing the mapping approach, we observed that the performance improved as the number of test patterns were increased. The result was validated for different structures of MLFFNN and different combination of input-output parameter vector pairs.

The structure of MLFFNN was shown to give the network the capability to learn the speaker-specific mapping. The significance of nonlinear units in the structure for realizing the mapping was studied.

The mapping approach was shown to be robust to channel affects. However, this observation was drawn from the preliminary studies. A few more experiments are required to confirm the robustness aspect of the mapping approach to channel variations.

8.2 CONTRIBUTIONS OF THE WORK

From the experimental studies presented in this thesis, we can conclude that the mapping approach certainly holds promise for performing the task of text-independent speaker recognition. The following are the contributions of the thesis:

1. The concept of speaker-specific mapping was introduced, and the parameter vector pair suitable for the mapping approach were conceived. The optimal lpc-lpc pair was obtained using experimental studies.

2. The hypothesis of speaker-specific mapping and using it as a feature for speaker recognition task were validated in the present work.

3. The concept of BG normalization was developed for the mapping approach to improve the performance of the system.

4. In this thesis, we proposed the concept of *Ne* criterion to select frames having significant speaker-specific information. Significant improvement in the performance was achieved using the *Ne* criterion.

5. Dependency of the performance of the mapping approach on the number of test frames was studied.

6. The issue of robustness of the mapping approach to channel variations was addressed.

7. Importance of the MLFFNN structure and nonlinearity of the units in the structure in capturing the speaker-specific mapping were studied.

8.3 SCOPE FOR FUTURE WORK

A few of the topics which need attention either to improve the performance or to have a better understanding of the mapping approach are listed below.

1. In our approach, we didn't use the sequence knowledge of speech. Using this knowledge, the system performance can be improved.

2. Instead of using one network, several networks can be obtained for each speaker using different spectral parameters. Combining the evidences [57] provided by the individual networks, can improve the performance of the system.

3. Theoretical study of the mapping approach is not done in the present work. To gain insight, a theoretical analysis is required.

4. Grouping the speakers based on suprasegmental information (such as pitch), and then applying the mapping approach to the obtained sub-groups is another aspect to look into.

5. Use of syllable rate or model spectra for grouping the speakers could also be useful

APPENDIX–A
TIMIT DATABASE

TIMIT is a standard database used for checking the performance of speech as well as speaker recognition systems. Salient features of the TIMIT database are:

1. It consists of 630 speakers, 192 female and 438 male.
2. These speakers are categorized into one of the 8 dialect regions of America. For example, "dr1" (dialect region 1) represents *New England* and "dr2" corresponds to *Northern*.
3. There are 10 utterances per speaker. Each utterance is one sentence. The sentences are grouped into three categories. They are:
   (a) Dialect sentences (2): These were spoken by all the 630 speakers to measure the effect of dialect on various characteristics of speakers.
   (b) MIT sentences (5): These were iteratively designed by hand to provide a rich variety of phonetic segments and phonetic contexts. There are total of 450 MIT sentences, each spoken by 7 speakers.
   (c) TI sentences (3): These were designed by taking a large corpus of written text and augmenting it with a small number of selections from a corpus of spoken sentences. There are 1890 TI sentences, each spoken by 1 speaker.
4. Utterances are orthographically and phonetically transcribed.
5. Recording of the utterance was done over a microphone in noise free environment, and the speech signal was sampled at 16 KHz.

APPENDIX–B
NTIMIT DATABASE

NTIMIT database is obtained from the TIMIT database. The TIMIT utterances were transmitted in an acoustically isolated room through an "artificial mouth" (designed to approximate the acoustic characteristics of a human mouth). After passing through long distance telephone channels, they were recorded back. Recording was done such that orthographical and phonetical transcriptions of the NTIMIT matched with that of the TIMIT.

The assumed degradations in NTIMIT as compared to NTIMIT are:

1. Bandlimiting between 300–3300 Hz because of telephone channel.
2. Channel distortions, both because of microphone (non-linear) and telephone line (linear).
3. Noise

In the present work, the subsets of NTIMIT were used. In all the experiments, 7 utterances were used for training and 3 for testing. A frame size of 20 ms and frame shift of 10 ms was used to extract the parameter vectors from the utterances.

Bibliography

[1] B. S. Atal, "Automatic recognition of speakers from their voices," *Proc. IEEE*, vol. 64, pp. 460–475, Apr. 1976.

[2] G. R. Doddington, "Speaker recognition–identifying people by their voices," *Proc. IEEE*, vol. 73, pp. 1651–1664, Nov. 1985.

[3] D. O'Shaughnessy, "Speaker recognition," *IEEE ASSP Magazine*, pp. 4–17, Oct. 1986.

[4] O. Tosi, H. Oyer, W. Lashbrook, C. Pedrey, J. Nicol, and E. Nash, "Experiment on voice recognition," *J. Acoust. Soc. Amer.*, vol. 51, no. 6, pp. 2030–2043, 1972.

[5] D. A. Reynolds, M. A. Zissman, T. F. Quatieri, G. C. O'Leary, and B. A. Carlson, "The effects of telephone transmission degradations on speaker recognition performance," in *Proceedings of IEEE Int. Conf. Acoust., Speech, and Signal Processing*, vol. 1, (Detroit, Michigan, USA), pp. 329–332, May 1995.

[6] D. A. Reynolds, "The effects of handset variability on speaker recognition performance: Experiments on the Switchboard corpus," in *Proceedings of IEEE Int. Conf. Acoust., Speech, and Signal Processing*, vol. 1, (Atlanta, Georgia, USA), pp. 113–116, May 1996.

[7] B. S. Atal, "Automatic speaker recognition based on pitch contours," *J. Acoust. Soc. Amer.*, vol. 52, no. 6, pp. 1687–1697, 1972.

[8] J. J. Wolf, "Efficient acoustic parameters for speaker recognition," *J. Acoust. Soc. Amer.*, vol. 51, no. 6, pp. 2044–2056, 1972.

[9] F. K. Soong and A. E. Rosenberg, "On the use of instantaneous and transitional spectral information in speaker recognition," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 36, pp. 871–879, June 1988.

[10] J. Makhoul, "Linear prediction: A tutorial review," *Proc. IEEE*, vol. 63, pp. 561–580, Apr. 1975.

[11] S. Hangai and K. Miyauchi, "Speaker identification based on multipulse excitation and lpc vocal-tract model," in *Proceedings of Int. Conf. Spoken Language Processing*, (Kobe, Japan), pp. 1269–1272, Nov. 1990.

[12] S. Furui, "Cepstral analysis technique for automatic speaker verification," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 29, pp. 254–272, Apr. 1981.

[13] H. Hermansky, N. Morgan, A. Bayya, and P. Kohn, "(RASTA–PLP) speech analysis technique," in *Proceedings of IEEE Int. Conf. Acoust., Speech, and Signal Processing*, vol. 1, (San Francisco, California), pp. 121–124, Mar. 1992.

[14] H. Gish and M. Schmidt, "Text–independent speaker identification," *IEEE Signal Processing Magazine*, pp. 18–32, Oct. 1994.

[15] R. Schwartz, S. Roucos, and M. Berouti, "The application of probability density estimation to text-independent speaker identification," in *Proceedings of IEEE Int. Conf. Acoust., Speech, and Signal Processing*, vol. 3, (Paris, France), pp. 1649–1652, May 1982.

[16] Y. Gong and J.-P. Haton, "Non–linear vectorial interpolation for speaker recognition," in *Proceedings of IEEE Int. Conf. Acoust., Speech, and Signal Processing*, vol. 2, (San Francisco, California, USA), pp. II173–II176, Mar. 1992.

[17] N. Malayath, H. Hermansky, and A. Kain, "Towards decomposing the sources of variability in speech," in *Proceedings of EUROSPEECH'97*, (Rhodes, Greece), pp. 497–500, Sept. 1997.

[18] H. Hermansky and N. Malayath, "Speaker verification using speaker–specific mapping," in *Proc. RLA2C*, (Avignon, France), Apr. 1998.

[19] H. Misra, S. Ikbal, and B. Yegnanarayana, "Spectral mapping as a feature for speaker recognition," in *Proc. National Conference on Communications*, (Kharagpur, India), pp. 151–156, Jan. 1999.

[20] D. A. Reynolds, "Comparison of background normalization methods for text-independent speaker verification," in *Proceedings of EUROSPEECH'97*, vol. 2, (Rhodes, Greece), pp. 963–966, Sept. 1997.

[21] A. E. Rosenberg and S. Parthasarathy, "Speaker background models for connected digit password speaker verification," in *Proceedings of IEEE Int. Conf. Acoust., Speech, and Signal Processing*, vol. 1, (Atlanta, Georgia, USA), pp. 81–84, May 1996.

[22] J. P. Eatock and J. S. Mason, "Automatically focusing on good discriminating speech segments in speaker recognition," in *Proceedings of Int. Conf. Spoken Language Processing*, (Kobe, Japan), pp. 133–136, Nov. 1990.

[23] M. Savic and S. K. Gupta, "Variable parameter speaker verification system based on hidden Markov modeling," in *Proceedings of IEEE Int. Conf. Acoust., Speech, and Signal Processing*, vol. 1, (Albuquerque, New Mexico, USA), pp. 281–284, Apr. 1990.

[24] A. E. Rosenberg, "Automatic speaker verification: A review," *Proc. IEEE*, vol. 64, pp. 475–487, Apr. 1976.

[25] T. Matsui, T. Nishitani, and S. Furui, "Robust methods to updating model and a-priori threshold in speaker verification," in *Proceedings of IEEE Int. Conf. Acoust., Speech, and Signal Processing*, vol. 1, (Atlanta, Georgia, USA), pp. 97–100, May 1996.

[26] D. A. Reynolds, "Speaker identification and verification using gaussian mixture speaker models," in *Proc. ESCA workshop of automatic speaker recognition, identification and verification*, (Martigny, Switzerland), pp. 27–30, Apr. 1994.

[27] D. A. Reynolds, "Speaker identification and verification using gaussian mixture models," *Speech Comm.*, vol. 17, pp. 91–108, Aug. 1995.

[28] J. He, L. Liu, and G. Palm, "A discriminative training algorithm for gaussian mixture speaker models," in *Proceedings of EUROSPEECH'97*, vol. 2, (Rhodes, Greece), pp. 959–962, Sept. 1997.

[29] L. P. Heck and M. Wientraub, "Handset–dependent background models for robust text–independent speaker recognition," in *Proceedings of IEEE Int. Conf. Acoust., Speech, and Signal Processing*, vol. 2, (Munich, Germany), pp. 1071–1074, Apr. 1997.

[30] L. Rabiner and B.-H. Juang, *Fundamentals of Speech Recognition.* Englewood Cliffs, New Jersey: Prentice Hall, 1993.

[31] A. E. Rosenberg, C.-H. Lee, and F. K. Soong, "Sub–word unit talker verification using hidden Markov models," in *Proceedings of IEEE Int. Conf. Acoust., Speech, and Signal Processing*, vol. 1, (Albuquerque, New Mexico, USA), pp. 269–272, Apr. 1990.

[32] A. E. Rosenberg, C.-H. Lee, F. K. Soong, and M. A. Mcgee, "Experiments in automatic talker verification using sub–word unit hidden Markov models," in *Proceedings of Int. Conf. Spoken Language Processing*, vol. 1, (Kobe, Japan), pp. 141–144, Nov. 1990.

[33] H. Noda and M. Yanagida, "Extraction of phoneme–dependent individuality using HMM–based segmentation for text-independent speaker recognition," in *Proceedings of Int. Conf. Spoken Language Processing*, (Kobe, Japan), pp. 129–132, nov 1990.

[34] M. E. Forsyth, P. C. Bagshaw, and M. A. Jack, "Incorporating discriminatory observation probabilities (DOP) into semi-continuous HMM for speaker verification," in *Proc. ESCA workshop of automatic speaker recognition, identification and verification*, (Martigny, Switzerland), pp. 19–22, Apr. 1994.

[35] S. Furui, "An overview of speaker recognition technology," in *Automatic Speech and Speaker Recognition* (C.-H. Lee, F. K. Soong, and K. K. Paliwal, eds.), ch. 2, pp. 31–56, Boston: Kluwer Academic, 1996.

[36] S. Vela and H. A. Murthy, "Speaker identification – A new model based on statistical similarity," in *Proceedings of the Int. Conf. Computational Linguistics, Speech and Document Processing*, (Calcutta, India), pp. B7–B11, Feb. 1998.

[37] Y. Gong and J.-P. Haton, "Text–independent speaker recognition by trajectory space comparison," in *Proceedings of IEEE Int. Conf. Acoust., Speech, and Signal Processing*, vol. 1, (Albuquerque, New Mexico, USA), pp. 285–288, Apr. 1990.

[38] H. Hattori, "Text–independent speaker recognition using neural networks," in *Proceedings of IEEE Int. Conf. Acoust., Speech, and Signal Processing*, vol. 2, (San Francisco, California, USA), pp. II153–II156, Mar. 1992.

[39] W.-Y. Li and D. O'Shaughnessy, "Hybrid network based on RBFN and GMM for speaker recognition," in *Proceedings of EUROSPEECH'97*, vol. 2, (Rhodes, Greece), pp. 955–958, Sept. 1997.

[40] W. Ren-Hua, H. Lin-Shen, and H. Fujisaki, "A weighted distance measure based on the fine structure of feature space: Application to speaker recognition," in *Proceedings of IEEE Int. Conf. Acoust., Speech, and Signal Processing*, (Albuquerque, New Mexico, USA), pp. 273–276, Apr. 1990.

[41] T. Matsui and S. Furui, "Text-independent speaker recognition using vocal tract and pitch information," in *Proceedings of Int. Conf. Spoken Language Processing*, vol. 1, (Kobe, Japan), pp. 137–140, Nov. 1990.

[42] K.-I. Funahashi, "On the approximate realization of continuous mappings by neural networks," *Neural Networks*, vol. 2, pp. 183–192, 1989.

[43] K. Hornik, "Approximation capabilities of multilayer feedforward neural networks," *Neural Networks*, vol. 4, pp. 251–257, 1991.

[44] B. Yegnanarayana, *Artificial Neural Networks.* Connaught Circus, New Delhi: Prentice–Hall India, 1999.

[45] J. Sietsma and R. J. F. Dow, "Creating artificial neural networks that generalize," *Neural Networks*, vol. 4, pp. 67–79, 1991.

[46] R. P. Lippmann, "An introduction to computing with neural nets," *IEEE ASSP Magazine*, pp. 4–22, Apr. 1987.

[47] P. Gallinari, S. Thiria, F. Badran, and F. Fogelman-Soulie, "On the relations between discriminat analysis and multilayer perceptrons," *Neural Networks*, vol. 4, pp. 349–360, 1991.

[48] C. M. Bishop, *Neural Networks for Pattern Recognition*, ch. 2, pp. 33–34. Oxford: Clarendon Press, 1995.

[49] Y. Bennani and P. Gallinari, "Neural networks for discrimination and modelization of speakers," *Speech Comm.*, vol. 17, pp. 159–175, 1995.

[50] I.-C. Jou, S.-L. Lee, M.-T. Lin, C.-Y. Tseng, S.-S. Yu, and Y.-J. Tsay, "A neural network based speaker verification system," in *Proceedings of Int. Conf. Spoken Language Processing*, (Kobe, Japan), pp. 1273–1276, Nov. 1990.

[51] J. Oglesby and J. S. Mason, "Optimisation of neural models for speaker identification," in *Proceedings of IEEE Int. Conf. Acoust., Speech, and Signal Processing*, (Albuquerque, New Mexico, USA), pp. 261–264, Apr. 1990.

[52] H. Yin and T. Zhou, "Speaker recognition using static and dynamic cepstral feature by a learning neural network," in *Proceedings of Int. Conf. Spoken Language Processing*, vol. 2, (Kobe, Japan), pp. 1277–1280, Nov. 1990.

[53] Y. Gong and J.-P. Haton, "Non–linear interpolation methods for speaker recognition and verification," in *Proc. ESCA workshop of automatic speaker recognition, identification and verification*, (Martigny, Switzerland), pp. 23–26, Apr. 1994.

[54] B. Yegnanarayana, A. M. Jacob, and A. Bayya, "Spectral estimation through group-delay functions," in *Proc. IEEE DSP workshop*, (Utah, USA), Aug. 1998.

[55] C. Jankowski, A. Kalyanswamy, S. Basson, and J. Spitz, "NTIMIT: A phonetically balanced, continuous speech, telephone bandwidth speech database," in *Proceedings of IEEE Int. Conf. Acoust., Speech, and Signal Processing*, vol. 1, (Albuquerque, New Mexico, USA), pp. 109–112, Apr. 1990.

[56] L. Rabiner and B.-H. Juang, *Fundamentals of Speech Recognition*, ch. 4, pp. 169–170. Englewood Cliffs, New Jersey: Prentice Hall, 1993.

[57] S. Sharma, P. Vermeulen, and H. Hermansky, "Combining information from multiple classifiers for speaker verification," in *Proc. RLA2C*, (Avignon, France), Apr. 1998.

LIST OF PUBLICATIONS

PRESENTATIONS IN CONFERENCES

- H. Misra, M. Shajith Ikbal, and B. Yegnanarayana, "Spectral mapping as a feature for speaker recognition," *Proc. National Conf. Commn.*, IIT Kharagpur, pp. 151-156, Jan. 1999.

- M. Shajith Ikbal, H. Misra, and B. Yegnanarayana, "Analysis of autoassociative mapping neural networks," to appear in *Proc. Int. Joint Conf. on Neural Networks*, Washington, Jul. 1999.