EXPLORING FEATURES AND SCORING METHODS FOR SPEAKER VERIFICATION

A THESIS

submitted by

GURUPRASAD. S

for the award of the degree

of

MASTER OF SCIENCE

(by Research)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

INDIAN INSTITUTE OF TECHNOLOGY MADRAS.

OCTOBER 2004

THESIS CERTIFICATE

This is to certify that the thesis entitled **Exploring Features and Scoring Methods for Speaker Verification** submitted by **Guruprasad. S** 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 my 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.

Madras - 600036	Prof. B. Yegnanarayana
Date:	Dept. of Computer Science and Engg.

ACKNOWLEDGEMENTS

I am immensely grateful to Prof. B. Yegnanarayana for his guidance and encouragement during my research work. His relentless pursuit of excellence, his unconventional approach to research and his inexhaustible levels of energy have been a source of inspiration to me. I am thankful to him for the patience he has shown in me.

I thank Dr. C. Chandra Sekhar and Dr. Hema A. Murthy for their advice, care, comments, criticism and also for their active involvement and keen observations during seminars and presentations. I would like to imbibe their sense of discipline and cheerful outlook.

I thank Prof. S. Raman for providing excellent facilities in the department for research. I am highly indebted to him for his kind consideration and help, towards the continuation of my research work.

I thank the members of the General Test Committee, Prof. D. J. Ram and Prof. V. V. Rao for their interest in my research work and their valuable comments and suggestions on my work.

I thank Dr. S. Rajendran for his encouragement, advice and his reassuring presence in the laboratory. I am grateful to his support during the days of NIST evaluation.

I am thankful to all the faculty and non-teaching staff within the department as well as outside the department for all the help that I have received during my stay at IIT Madras.

I express my sincere gratitude to IIT Madras Alumni Association and the organizers of IJCNN 2003, for providing me with financial assistance to help me attend IJCNN 2003.

I thank my senior laboratory members Surya, Nayeem, Prasanna, Srinivasa Rao, Palanivel, Leena, Shahina, Anitha, Krishnamohan and Kumaraswamy for all the co-operation, understanding and help I have received from them. I thank Sharat, Gupta, Kiran, Jinu, Vinod, Venkatesh, Anvita, Ram and Shiva for the good times. I have thoroughly enjoyed the company of Dhananjaya, Anil, Satish, Suresh, Chaitanya, Sriram, Anand and Balavamsi. My special thanks to Anil from whom I have learnt a few things in the past one year. I also thank my hostel friends Krishna Bhopali, Bharat, Kiran and Paul for the good times. My special thanks to Smt. Kumari, Smt. Nirmala and Smt. Anitha for their affection and sumptuous food, which were most welcome breaks from the routine.

I would like to thank all my teachers, relatives and friends whose blessings and wishes have helped me reach this far. My special thanks to Dr. V. K. Ananthashayana, my guide during B. E., who sparked my interest in the area of signal processing.

No praise is ever too high for the one-and-only Dhananjaya whose friendship, I consider, is an honour. I am grateful to his parents and sister who have treated me with care and affection, and have always considered me as a part of their family.

I am grateful to my brother Dr. S. Muralidhar and my sister-in-law Smt. Ranjitha for their care and affection. I fondly recollect the time we spent together in Portland. My sincere thanks to Dr. S. Muralidhar for motivating me towards studying further.

Words can not express the profoundness of my gratitude to my parents
Smt. S. Sharadamba and Shri. S. Seshadri, whose love and blessings are a constant source of strength to me.

ABSTRACT

Keywords: *speaker verification, pitch synchronous analysis, difference cepstral coefficients, autoassociative neural network models, score normalization, complementary features, combination of evidences.*

The objective of automatic speaker verification is to validate a speaker's claim of identity, based on the speaker's voice. Speaker verification consists of three steps, namely, feature extraction, modeling and score normalization. The objective of this research work is to address certain issues in feature extraction and score normalization. Most methods of feature extraction consider uniform blocks of speech of 10-30 ms duration for analysis, overlooking the position of window of analysis. In this work, the significance of pitch synchronous analysis of speech is studied for accurate estimation of short-time spectral characteristics. Spectral features such as linear prediction cepstral coefficients (LPCC) and mel-frequency cepstral coefficients represent characteristics of both the sound unit and the speaker. We propose difference cepstral coefficients for deemphasizing the sound unit information in the short-time spectrum. The effectiveness of difference cepstral coefficients for speaker verification and its ability to provide complementary information to spectral features is demonstrated. Autoassociative neural network (AANN) models are used to estimate the probability density function of feature vectors in the feature space. An important advantage of AANN models is that they do not make a priori assumptions about the shape of the probability density function. Due to difference in training and test utterances, the scores obtained from the models need to be calibrated, before comparison with a decision threshold. In this work, methods of normalization are proposed for weighting the scores of different test segments, which result in an improvement over the existing methods. Traditionally, speaker verification systems use a single feature for representing speaker-specific information. In this work, combination of evidences from three complementary features, namely, LPCCs, difference cepstral coefficients and excitation source features, is shown to result in a significant improvement in the performance of verification.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ABBREVIATIONS

LP	- Linear Prediction
PLP	- Perceptual Linear Prediction
LPC	- Linear Prediction Coefficients
LPCC	- Linear Prediction Cepstral Coefficients
MFCC	- Mel-Frequency Cepstral Coefficients
DFT	- Discrete Fourier Transform
LF	- Liljencrants-Fant
VQ	- Vector Quantization
GD	- Group Delay
GC	- Glottal Closure
ANN	- Artificial Neural Network
AANN	- Autoassociative Neural Network
MLFFNN	- Multilayer Feedforward Neural Network
GMM	- Gaussian-Mixture Model
HMM	- Hidden Markov Models
WAD	- Within-speaker to Across-speaker Dissimilarity
NIST	- National Institute of Standards and Technology
EER	- Equal Error Rate
DET	- Detection Error Tradeoff

CHAPTER 1

INTRODUCTION TO AUTOMATIC SPEAKER RECOGNITION

Speech is one of the most basic forms of communication among human beings. Speech is a composite signal that contains information about the message to be conveyed, the characteristics of the speaker and the language of communication. The unique characteristics of the voice of a speaker are due to anatomical and physiological factors. Anatomical factors relate to the physical aspects of speech production mechanism, namely, the vocal tract system and the vocal folds. Physiological factors reflect the speaking habits of a person, such as speaking rate, accent and mannerisms. These features are embedded in the speech signal, and hence, are useful in recognizing the speaker.

Automatic speaker recognition is the task of recognizing a person by a machine, using the information obtained from his/her speech signal. Automatic speaker recognition systems are useful in applications where access to a facility needs to be controlled. Although techniques such as automatic fingerprint analysis, face recognition, retinal scanning and magnetic cards with passwords are employed for such applications, they are limited by cost and ease of usage. Also, systems based on alphanumeric passwords can be compromised. On the other hand, speech is a natural and convenient form of input that carries the signature of the speaker. Moreover, speech is inexpensive to collect and analyze, and is hard to mimic. Therefore, automatic speaker recognition is suitable for such applications. Automatic speaker recognition systems can be used as a preprocessing stage in automatic speech recognition systems, to improve the performance of the speech recognizer. They can be used for machine identification of participants in meetings, conferences or conversations. They can also be used in conjunction with automatic speech recognizers for analyzing multi speaker data, to obtain a record of speech uttered by different speakers. In law enforcement, speaker recognition systems can be used to help identify suspects. Thus, speaker recognition systems have a number of important applications.

1.1 SPEAKER RECOGNITION BY HUMANS

An insight into the ability of human beings to identify speakers from their speech may offer clues for automatic speaker recognition. Human beings can recognize speakers from their voices with ease, given a certain degree of familiarity. This is due to their ability to extract specific cues for a given speaker, and also due to their ability to integrate higher sources of knowledge such as context, manner of speaking and language. In [1], 2-3 seconds of speech was observed to be sufficient for subjects to identify familiar voices, while the performance of recognition decreased for unfamiliar voices. Also, when the utterances were played backward, the performance of recognition reduced drastically, thus highlighting the importance of timing and articulatory cues. Human beings can easily perceive mimicry of familiar voices [2]. The ability of human beings to recognize familiar voices in adverse conditions is remarkable [2]. However, the performance of machines can exceed that of human beings, when the test utterance is short and the speakers are unfamiliar. This is because the time required by human beings to learn a new voice is normally long and machines may be trained much faster.

1.2 CATEGORIES OF AUTOMATIC SPEAKER RECOGNITION

Automatic speaker recognition can be divided into two categories: speaker identification and speaker verification. The speaker identification task is to determine if the speaker of an unknown (test) utterance is present in a given set of speakers, and if so, to establish the identity of that speaker. The task is called closed-set identification, if it is known that the speaker is always a member of that set. If the speaker need not be a member of that set, then the task is called open-set identification. The speaker verification task is to determine if the speaker is indeed the person he / she claims to be, i.e., to validate the claim of the speaker. In speaker identification, the number of decision alternatives is equal to the size of the population, whereas in verification there are only two alternatives, acceptance or rejection of the claim.

Speaker recognition can be performed in a text-dependent or text-independent manner. A text-dependent system requires a speaker to utter a set of predefined phrases or sentences while collecting the training and test utterances. A text-independent system does not depend on the text of the training or test utterances. The objective of this thesis is to address issues in text-independent speaker verification.

1.3 ISSUES IN AUTOMATIC SPEAKER VERIFICATION

As mentioned in Section 1.1, human beings extract certain cues from the speech of a speaker, that help them to identify the speaker. But the exact nature of these cues is not fully understood. Moreover, the tools available for speech processing are not adequate to represent the higher sources of knowledge, such as the speaking mannerisms of the individual. Hence, automatic speaker recognition is approached as a statistical pattern recognition problem. In this section, we discuss the general approach to automatic speaker verification and issues involved in the task.

Automatic speaker verification entails the following steps:

1. Representation of speaker-specific characteristics and their efficient measurement from the speech signal, known as *feature extraction*
2. Development of a model (prototype) for each speaker using reference features extracted from the speech of that speaker, known as *modeling*
3. Comparison between the reference features and the features extracted from a test utterance, called *matching*
4. Decision mechanism for verification based on the score obtained during matching, known as *scoring*

The objective of feature extraction is the quantitative representation of speaker-specific properties and the efficient measurement of these properties from the acoustic speech signal. It is desirable that these features have the following properties [3]:

- High interspeaker-to-intraspeaker variability
- Robustness to the characteristics of transmission channel, microphone and ambient noise
- Ease of extraction from the speech signal
- Robustness to aging of the speaker
- Not subject to mimicry

Typically, short-time analysis of speech is performed to extract features which represent the characteristics of the two components of speech production mechanism, namely, the excitation source and the vocal tract system. Although high-level features such as speaking rate, accent and verbal mannerisms of the speaker convey significant speaker-specific information, the existing techniques of feature extraction are not adequate to represent such information. Most of the current speaker recognition algorithms are based on short-time features extracted from speech signal.

Once the features are extracted from the speech signal, the next step is to develop a model to represent the set of features. Models can be classified as parametric or nonparametric models. Parametric models assume a structure characterized by certain parameters, which are estimated from the given features. In general, a model may represent any information derived from the set of features. For example, the model may represent the following:

- Statistical average of the features computed over long utterances (of several seconds or minutes) of speech
- Estimate of the probability density function of the features in the feature space
- Estimate of the temporal information present in the sequence of features

Some issues in the choice of models are as follows:

- The choice of features for modeling speaker-specific characteristics
- The amount of speech data required to reliably estimate the parameters of the model
- The ability of the model to generalize the characteristics of the speaker from the given set of features

The model of a given speaker is presented with the features extracted from a test utterance, whose speaker is unknown. Comparison between the reference features and the test features depends on the nature of the model. The following cases are possible:

- If statistical averages of the features are used, a distance metric is required for comparison.

- If the model represents an estimate of probability density function of the features, then likelihood is one measure of similarity between the reference and test features.
- If the model represents an estimate of the temporal information, then temporal matching score or likelihood can be used as measures of similarity.

The comparison generates a score that indicates the similarity between the reference features and the test features. Based on this score, a decision needs to be made on the validity of the claim. Due to differences in the reference and the test utterances, the score needs to be calibrated before setting a threshold for decision. Hence, normalization and scoring methods are needed for this purpose.

1.4 ISSUES ADDRESSED IN THIS THESIS

The objective of this research work is to address certain issues related to a text-independent speaker verification system. The focus of research is: (a) To explore features for effective representation of speaker-specific characteristics and (b) to explore techniques of score normalization for verification. The significance of the position of window for analyzing speech is discussed. Pitch synchronous analysis of speech is studied for accurate estimation of short-time spectral characteristics. Difference cepstral coefficients are proposed as a feature for speaker verification, by deemphasizing the linguistic information present in the speech signal. The ability of this feature to provide complementary information for speaker verification is also demonstrated. Speaker-specific models based on autoassociative neural networks are used to estimate the probability density function of feature vectors. The problem of score normalization for speaker verification is discussed. Techniques for normalization of scores are proposed, and a comparison with the existing methods is presented. Most speaker verification systems use a single feature for representing speaker-specific information. In this work, evidences due to several complementary features are combined for increasing discrimination between genuine and impostor speakers.

1.5 ORGANIZATION OF THE THESIS

The thesis is organized as follows:

In Chapter 2, a brief review of the existing approaches to speaker verification is presented.

Chapter 3 describes a baseline system for speaker verification using spectral features and autoassociative neural network models, and describes a few refinements for performance enhancement of the system.

Chapter 4 discusses pitch synchronous analysis of speech for extraction of short-time spectral features, and illustrates its advantages over the traditional block-based analysis.

In Chapter 5, the development of difference cepstral coefficients for speaker-specific characterization is described. A speaker verification system based on the above feature is also discussed.

In Chapter 6, methods for normalization of scores are proposed and evidences due to different features are combined for speaker verification.

Chapter 7 presents a summary of the work and outlines the scope for further research.

CHAPTER 2

REVIEW OF APPROACHES FOR SPEAKER VERIFICATION

This chapter presents a brief review of approaches for speaker verification. In particular, features for speaker verification, methods for modeling speaker-specific characteristics and techniques for score normalization are reviewed. Features for speaker verification are mostly obtained by short-time analysis of speech, which normally represent the characteristics of excitation source and vocal tract system. These are reviewed in Section 2.1. Speakers can be modeled with features derived from speech signal, using parametric or nonparametric models. Section 2.2 reviews approaches for modeling speaker-specific characteristics. Due to mismatch between training and test data, the scores resulting from the models cannot be compared to a common threshold for decision. Hence, the scores are calibrated using methods of normalization. Section 2.3 reviews the issue of normalization and some existing methods of normalization.

2.1 FEATURES FOR SPEAKER VERIFICATION

Speech is produced by exciting a time-varying vocal tract system with a time-varying input. Speaker-specific information is present in both these components of speech production mechanism. Short-time analysis of speech is an effective tool for extraction of such information.

2.1.1 Features Based on Vocal Tract System

The vocal tract system can be considered as a cascade of cavities of varying cross sections. The size and shape assumed by the vocal tract while producing various sound units is a characteristic of the sound unit and the speaker. Formants are resonances of the vocal tract system. They vary in frequency, bandwidth and relative amplitude, depending on the sound unit being produced and the speaker uttering the sound. However, accurate extraction of formants from speech signal is a difficult task [4] [5], and distances based on formant frequencies are not sufficiently discriminative between speakers for text-independent systems.

Linear prediction (LP) analysis of speech [6] provides an approximation to short-time spectrum of the transfer function of the vocal tract filter, as well as the source of excitation to the filter. In [7], different parametric representations of speech derived from LP analysis of speech were investigated for their effectiveness for automatic speaker recognition. These were, the predictor coefficients, the impulse response of the vocal tract system, the autocorrelation of the impulse response and the cepstrum derived from the logarithmic transfer function of the vocal tract system. In [8], long term averaging of reflection coefficients (obtained during LP analysis) was shown to increase the ratio of interspeaker-to-intraspeaker variability. In [9], adaptive component weighting cepstral coefficients were proposed, to emphasize the formant structure of the speech spectrum obtained by LP analysis and attenuate the broad bandwidth spectral components. In [10], a method called orthogonal linear prediction was proposed and a small subset of the resulting orthogonal coefficients was shown to exhibit significant interspeaker variation. In [11], principal spectral components were derived from LP coefficients for speaker verification task. In [12], cepstral coefficients extracted by means of LP analysis, called linear prediction cepstral coefficients (LPCC) were shown to yield nearly the same performance of speaker recognition as that due to cepstral coefficients obtained by short-time analysis using DFT. In [12] and [13], orthogonal polynomial representations were proposed to characterize transitional spectral information. Mel-frequency cepstral coefficients (MFCC) have been used for speaker recognition [14]. They are obtained by warping the frequency scale in such a way as to resolve the spectrum finely at lower frequencies and relatively coarsely at higher frequencies [15].

2.1.2 Features Based on Excitation Source

During the production of speech, the vibration of vocal folds provides quasi-periodic impulse-like excitation to the vocal tract system. Linear prediction (LP) residual, obtaining by inverse filtering the speech, is an approximation to the source of excitation of the vocal tract system. In [16], a feature called real cepstrum was computed from the LP residual by ignoring the phase information, retaining the amplitude spectrum and by introducing a logarithmic nonlinearity. Long-term average of the real cepstrum was shown to have a low intraspeaker and high interspeaker variability. In [17], a nonlinear prediction model based on neural networks was used to compute an error signal. Certain measures were defined over LP residual, such as mean square error, mean absolute error and variance of the residue, that were shown to reduce the error rate in speaker recognition. Liljencrants-Fant (LF) model has been used as a parametric model to characterize glottal flow derivative [18]. In [19], estimate of glottal flow derivative was obtained using LF model to capture its coarse structure, while the fine structure was represented by energy and perturbation measures. Both coarse and fine-structure glottal features were shown to result in the reduction of error in a speaker identification system, when used in conjunction with Mel-frequency cepstral coefficients. However, in the above methods, the features of excitation source were modeled using a probabilistic framework. In [20], excitation source information present in the LP residual was extracted using autoassociative neural network models.

Here, the goal was to capture the higher order relationship existing among the samples of the LP residual. The effect of the order of LP analysis on speaker verification was studied. An experimental study on the significance of excitation sources corresponding to different sound units was also conducted, and some sounds were observed to be more significant for speaker verification than others.

Pitch is the fundamental frequency of vibration of vocal folds. Pitch is a unique characteristic of each speaker due to the differences in physical structure of vocal folds among different speakers. It can also be different due to speaking style and accent imposed by different speakers. A summary of various algorithms for pitch extraction was presented in [21]. Unlike spectral features that are affected by channel variations, noise and distance between the speaker and microphone, pitch is insensitive to the above factors. In [22], linear transformation of vectors representing the pitch contours was shown to improve the ratio of interspeaker to intraspeaker variance, for a text-dependent speaker recognition system. In [8], long-term averages of pitch and standard deviation of pitch were shown to be speaker dependent. In [23], a lognormal distribution of pitch was proposed instead of a Gaussian distribution. A probabilistic model for estimated pitch was suggested, using a mixture of three lognormal distributions with tied means and variances.

Variation of pitch as a function of time is called intonation. While a speaker's average pitch may be mimicked, it is difficult for an impostor to mimic the local variations of pitch. Intonation has been more useful in text-dependent speaker recognition. In [22] and [24], similarity between the intonation patterns of reference and test utterances was measured using dynamic time warping algorithm. Two other features related to pitch are jitter and shimmer. Jitter is defined as the perturbation of pitch, while shimmer represents the variation in peak amplitudes of the signal in successive pitch periods [25].

2.2 MODELING SPEAKER CHARACTERISTICS

Parametric and nonparametric models have been studied for speaker verification. In [26], a nearest-neighbour distance measure was proposed, based on the similarity of distributions of features extracted from reference and unknown utterances. The measure did not assume any form of the distributions involved. A relationship was established between the distance measure and Kullback-Leibler divergence [27].

In [28], vector quantization (VQ) codebook was used as a means for characterizing the short-time spectral features of a speaker. A VQ codebook was developed for each speaker. The decision on the identity of the unknown speaker was based on a minimum distance classification rule. The effect of different parameters on the performance of verification was studied. These parameters were the codebook size, phonetic content of the text and difference in recording sessions.

In [14], Gaussian mixture models (GMM) were proposed for text-independent speaker identification. The basis for such a model is that the individual Gaussian components of a GMM represent speaker-dependent spectral shapes that are useful for modeling speaker identity, and also that Gaussian mixtures can model arbitrary densities. The experiments reported in [14] deal with algorithmic issues such as model initialization, variance limiting and model order selection. Techniques such as cepstral mean subtraction, difference coefficients and frequency warping were applied to compensate for spectral variability due to telephone channel and handsets.

The methods mentioned above model only the distribution of feature vectors and do not make use of the temporal correlations that exist in the sequence of feature vectors. In [29], a hidden Markov model (HMM) was proposed to incorporate temporal correlations in the VQ model. In this approach, short-term stationary regions were modeled by states, while the slower variations of the signal were modeled by the transitions between such states. The signal in each state was modeled by a type of

HMM called linear predictive HMM.

Artificial neural network models with different topologies can perform different pattern recognition tasks [27] [30]. In [31], the ability of a neural network model to discriminate between patterns of different classes was exploited for speaker recognition. A global classifier for a set of speakers was developed, whose utility was limited to a small number of speakers. Each model was trained to discriminate between speech data of the given speaker and a small set of impostors. In [32–34], mapping ability of neural network models was exploited to capture speaker-specific knowledge. In [35], the ability of AANN models to estimate arbitrary densities was demonstrated. It was illustrated experimentally that a network can be designed such that the training error surface relates to the distribution of the given data, depending on the constraints imposed by the structure of the network. The effectiveness of AANN models for speaker verification was also demonstrated. In [20] [36], AANN models were used to acquire the temporal relationship between the samples of linear prediction residual, to model speaker-specific characteristics.

Methods based on speaker-specific mapping of features have been used for speaker verification. The goal of this approach was to capture speaker-specific information by mapping a set of feature vectors specific to linguistic information (message part) in the speech, on to a set of feature vectors representing both the linguistic and speaker-specific information. In [37], a nonlinear vectorial interpolation function was proposed for text-dependent speaker recognition using the mapping property of a multilayer feedforward neural network (MLFFNN), to obtain the interpolation vector for each speaker. In [32], speaker-specific mapping approach was investigated for text-independent speaker recognition , using cepstral coefficients derived from perceptual linear prediction (PLP) as features. In [33], parameters for representing linguistic information and linguistic plus speaker-specific information were extracted from speech.

Speaker-specific information was captured by nonlinear mapping using a multilayer feedforward neural network.

2.3 DECISION LOGIC FOR VERIFICATION AND IDENTIFICATION

Once a model is developed for a speaker, decision on the validity of the claim is made based on the output score obtained from the model for a test utterance. Due to mismatch between training and test data, this score is specific to the model and the test utterance. The objective of score normalization is to transform the scores into a range where a common threshold for decision may be set, which is valid for any pair of training and test data.

2.3.1 The Problem of Score Normalization

Given a speech utterance x and a claimed identity λ, the objective of speaker verification is to decide if x was uttered by the genuine speaker λ, or by an impostor. This decision can be based on the comparison of a similarity measure (or a distance measure) between the speaker's model and the utterance x to a threshold. In the probabilistic framework, let O denote the set of observations corresponding to the test utterance x and let M denote the statistical model corresponding to speaker λ. According to Bayes theorem,

$$P(M/O)p(O) = p(O/M)P(M), \tag{2.1}$$

where $P(M/O)$ is the a posteriori probability of the hypothesized speaker model M given the set of observations O, $p(O)$ is the probability density function of the observations, $p(O/M)$ is the likelihood of M with respect to O and $P(M)$ is the prior probability of occurrence of the model M. For speaker verification, we need to evaluate $P(M/O)$. However, the output of a statistical model is an estimate of $p(O/M)$.

Assuming the occurrence of each model to be equally likely, the identity claim can be accepted if

$$p(O/M) > \beta, \tag{2.2}$$

and rejected otherwise, where β is the decision threshold. This decision rule cannot be used in practice due to the following reasons:

1. Due to differences in the training data of different speakers, the resulting models are not equally representative of the speaker-specific characteristics. The assumption is that with sufficient speech, the distribution of features in the feature space is a good representation of the sounds of the speaker. The amount of speech data available to model a speaker may not always conform to this assumption. The ability of a model to represent the distribution of features of a speaker is also affected by the intraspeaker variability of sounds within the speaker. Thus, some speakers are difficult to model, while some are easily modeled [38].

2. Due to mismatch between training and test data, the identity claim can be rejected due to a low likelihood score, even if the claim is legitimate. The main source of this mismatch is the channel through which speech is received, which induces variability in the features, causing them to move around in the feature space. Another source of this mismatch is that, some sound units occurring in the test data of a speaker might not have occurred adequately in the training data of that speaker. This results in poor modeling of that sound unit and consequently, a low likelihood score.

The objective of normalization is to transform the scores to a range where a common threshold can be determined for all tests.

2.3.2 Approaches to Score Normalization

Speaker verification systems based on Gaussian mixture models achieve a certain degree of normalization by using a speaker-independent world model M_w to model speech in general. A normalized log-likelihood score is obtained as

$$S(M,O) = log(p(O/M)) - log(p(O/M_w)) \tag{2.3}$$

Here, the mismatches that occur between the test utterance and the model M will have a corresponding effect on the world model M_w, thus removing the bias in $p(O/M)$ [39]. A similar approach used a set of cohort speakers who were close to the target speaker, thus viewing the cohorts as replacement for the world model [40]. The selection of cohorts can be done during training or testing. During training, a similarity measure was used to compare the speaker model with cohort models [41] [42].

In zero normalization (Z-norm) method [43] [44], a model was tested against example impostor utterances and the log-likelihood scores were used to estimate the mean μ_I and standard deviation σ_I of the impostor distribution. The quantities μ_I and σ_I are specific to the model of each speaker and can be estimated offline. The normalized score was computed as

$$S = \frac{S(M,O) - \mu_I}{\sigma_I}. \tag{2.4}$$

Zero normalization is equivalent to scaling the distribution of speaker-specific scores.

In test normalization, the objective is to estimate the statistics of an impostor for a given test utterance, which can be used to discriminate the genuine speaker from impostors. In T-norm [43], a given test utterance was presented to a set of background models, and the mean and variance of the resulting scores were computed. The normalized score was computed in a manner similar to that of zero normalization. The use of variance parameter is to estimate the distribution of the background scores more accurately. Also, acoustic mismatch between training and test utterances, that is still possible in zero normalization method is avoided here.

In the review of approaches, the scope is limited to review of algorithms / techniques for speaker verification and it does not include the review of speaker verification systems developed in other laboratories over the world, and industries. In this regard, [45] and [46] are useful references for the interested reader. These sources briefly discuss the speaker recognition systems being developed at various research laboratories, and also provide a performance comparison for the NIST Speaker Recognition Evaluation task.

2.4 MOTIVATION FOR THE PRESENT WORK

In this chapter, a brief review of the standard approaches to speaker verification was presented. In general, spectral and source features are extracted from speech signal to represent speaker-specific characteristics. Most of the methods analyze speech over uniform blocks of 10-30 ms duration for extracting spectral features. These methods use an arbitrary positioning of the window of analysis for feature extraction. In this thesis, pitch synchronous analysis is studied to obtain an accurate estimation of the short-time spectral features. The existing spectral features do not aim to specifically represent the characteristics of the speaker, since they also contain information about the sound unit. We propose a method to deemphasize the speech-specific information present in the short-time spectrum. Most of the existing approaches model the probability density function of feature vectors using a parametric model such as GMM. This approach assumes that the number of clusters in the feature space of the speaker is known a priori, and that the probability density function of these clusters is Gaussian in shape. In the present work, AANN models have been used for estimation of probability density function of the features. AANN models do not make assumptions about the nature the of probability density function of features. Score normalization methods help in reducing the effect of acoustic and channel related mismatch between training and test utterances. Existing methods are based on scaling the distribution of the scores of genuine and impostor speakers and they give equal weightage to all the frames of the test utterance. We propose normalization methods to weight the scores of different frames of the test utterance. Most speaker verification systems are based on a single feature. In this work, we discuss the importance of complementary sets of features for speaker verification. Combination of evidences from complementary sets of features is shown to improve the performance of speaker verification.

CHAPTER 3

A BASELINE SPEAKER VERIFICATION SYSTEM

In this chapter, we describe a baseline text-independent speaker verification system using spectral features and autoassociative neural network (AANN) models, which provides a framework for further experiments and performance evaluation. Section 3.1 describes the components of the baseline system. The database used for experiments and the metrics for performance evaluation are discussed in Sections 3.2 and 3.3 respectively. Certain refinements to the baseline system are proposed in Section 3.4.

3.1 COMPONENTS OF THE BASELINE SYSTEM

3.1.1 Feature Extraction

Speech signal is preemphasized and frames of 20 ms duration are Hamming windowed with a window shift of 5 ms. Short-time analysis of speech is performed using 14^{th} order linear prediction analysis. A 19 dimensional weighted linear prediction cepstral coefficient (LPCC) vector is computed from the linear predictor coefficients (LPC) of each frame of data [15]. Cepstral mean subtraction is performed to minimize the effect of slowly varying characteristics of transmission channel [12].

3.1.2 AANN Models for Speaker Verification

Autoassociative neural networks (AANN) are feedforward neural networks that perform an identity mapping of the input space [30]. A three-layer AANN model with linear units can capture the principal orthogonal components of a feature set, while a five-layer AANN with nonlinear units in the hidden layers can capture the probability surface of the feature vectors [30]. The backpropagation learning algorithm for multilayer feedforward neural networks is described in Appendix A of this thesis. Fig. 3.1 shows a five-layer AANN model that performs nonlinear principal component analysis.

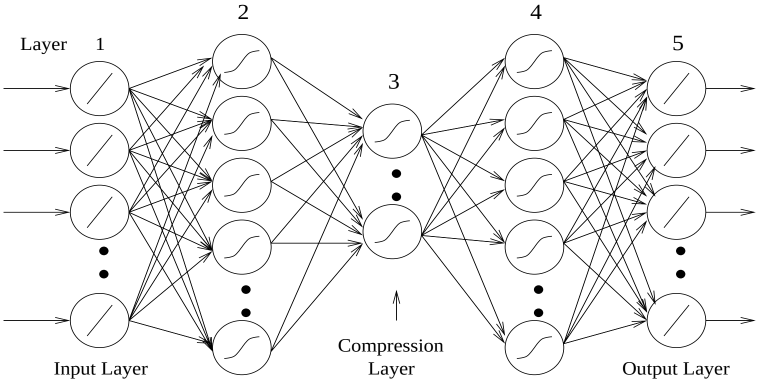


Fig. 3.1: A five-layer AANN model.

The ability of AANN models to capture nonlinear subspaces was demonstrated in [35]. The importance of error surface of the training data in the feature space was studied. It was observed that the average error was lower for the most frequently occurring input vectors than for the less frequently occurring ones. It was demonstrated experimentally that a network can be designed such that the training error surface relates to the distribution of the given data, depending on the constraints imposed on the structure of the network. AANN models are advantageous compared to Gaussian mixture models (GMM), when the surface representing the distribution of features is highly non-linear. This is because, GMMs assume the shape of the components of the distribution to be Gaussian, which need not be the case. Moreover, a GMM requires specification of the number of mixtures a priori.

For the baseline system, a 5-layer AANN model is developed for each speaker. The structure of the model is $19L$ $38N$ $4N$ $38N$ $19L$, where the numbers indicate the number of nodes in each layer. The symbols L and N denote, respectively, linear and nonlinear nature of the activation function of the nodes in each layer. The models are trained using backpropagation learning rule [27]. Each model is trained for 50 epochs, where one epoch denotes that all the feature vectors are presented to the model exactly once.

3.1.3 Normalization of Scores

In the baseline speaker verification system, two existing methods, Z-norm and T-norm, are applied for normalization of scores [43] [44]. For Z-norm, the impostor data collected from background speakers is presented to a claimant model, and the mean and variance of the scores are computed. For test normalization, 20 background models are used. A given test utterance is presented to the 20 background models along with the claimant model. The mean and variance of the scores of the background models are computed. The normalizations are performed as described in Section 2.3.

3.2 DATABASE FOR SPEAKER VERIFICATION

The database used in this study was selected from NIST 2003 speaker recognition evaluation [46]. The speech data was collected over cellular channel and sampled at 8 kHz. The database contains 149 male and 191 female speakers. The duration of training data for each speaker is about 2 minutes. The duration of a test utterance is between 15 and 45 seconds. 500 test utterances of male speakers are considered for verification. Each test utterance has 11 claimants, and the speaker of the test utterance may or may not be present among these 11 claimants. There are no cross gender tests. When a test utterance is presented to the model of a claimant speaker, a score is obtained which indicates the probability that the claimant speaker is the speaker of the test utterance. Thus, claimants are categorized as genuine and impostor claimants.

3.3 PERFORMANCE EVALUATION

The score resulting from a model is compared against a threshold, for accepting or rejecting the claim of the model. Two types of errors are possible in a speaker verification system: (a) False acceptance or false alarm error where an impostor is identified as the genuine speaker, and (b) false rejection or missed detection error, where a genuine speaker is classified as an impostor. The cost of false acceptance is higher than that of false rejection. For a low value of threshold, false rejection error is low but false acceptance error is high. As the threshold is increased, false rejection error increases but false acceptance error decreases. For a particular threshold, the two types of error are equal. The error at that threshold is called equal error rate (EER). Smaller the value of EER, better is the performance of the system.

The probability of false acceptance can be plotted against that of false rejection to observe the error characteristics. Detection error trade-off (DET) curves plot the normal deviates corresponding to the error probabilities [47]. These curves are linear and help in comparing the performance characteristics of different systems. On the DET curve, the point where the line $y = x$ intersects the curve indicates the EER. The DET curves for the normalized and unnormalized scores obtained for the baseline system are shown in Fig. 3.2.

The EER measure can be used to evaluate the performance of a speaker verification system. In this work, EER is computed for three types of scores:

1. Raw (unnormalized) scores obtained from the models
2. Normalized scores obtained by calibrating the raw scores
3. Scaled scores which are obtained as follows: For each test utterance, all the 11

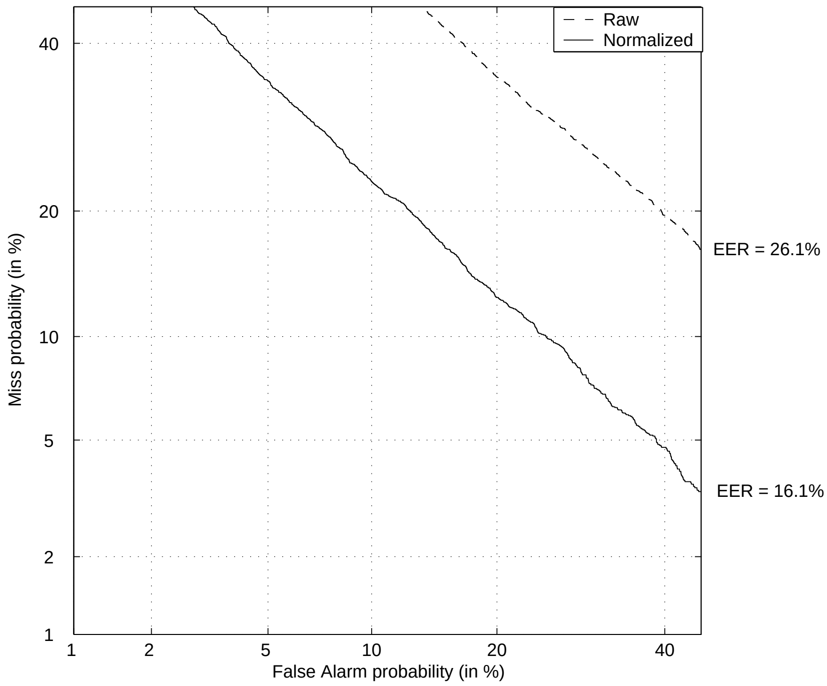


Fig. 3.2: DET curves for the unnormalized (raw) and normalized scores, for the baseline system.

raw scores are scaled by the maximum value among the 11 scores.

In addition, the percentage of the first ranks obtained by the genuine speakers is also computed over all the test utterances. These ranks are computed for the raw scores. The significance of scaled scores is that they transform the scores of all test utterances between zero and one. In all those test cases where the genuine speaker has obtained the first rank, the scaled score of the genuine speaker is one. Thus, scaling is equivalent to test normalization, which brings about a reduction in EER. This is observed from Table 3.1, which lists the performance of the baseline system. The scaled scores act as a reference against which the performance of normalized scores can be compared. Normalized scores are discussed greater detail in Chapter 6.

Table 3.1: Performance of baseline speaker verification system.

% of first ranks	EER(%) for raw scores	EER(%) for normalized scores	EER(%) for scaled scores
72.2	26.1	16.1	12.9

3.4 REFINEMENTS TO THE BASELINE SYSTEM

In this section, we propose certain refinements to the baseline system in the manner of selecting features for training the AANN models. These refinements are based on the interpretation that the feature space for a given speaker consists of a set of clusters of varying locations and densities.

3.4.1 Temporal Smoothing of Feature Vectors

Feature vectors extracted from the speech signal can be viewed as points in a multidimensional feature space. For each speaker, the feature vectors extracted from a given category of sound unit can be expected to form a cluster in the feature space. Temporal smoothing of features can be performed to make the clusters more cohesive in the feature space. Such smoothing reduces the effect of outliers generated during the extraction of features. As a result, the training error of AANN models is reduced, during the estimation of probability density function of feature vectors. This is illustrated in Fig.3.3, where the training error is plotted for AANN models trained on LPCC features and smoothed LPCC features, for a given speaker. It is evident that the models trained on smoothed LPCC features attain a lower value of training error. However, for a given vowel sound, there is overlap between formant frequencies of different speakers, leading to an overlap of clusters in the feature space. If this overlap between two speakers is significant, then the discrimination between them is reduced due to the smoothing of feature vectors.

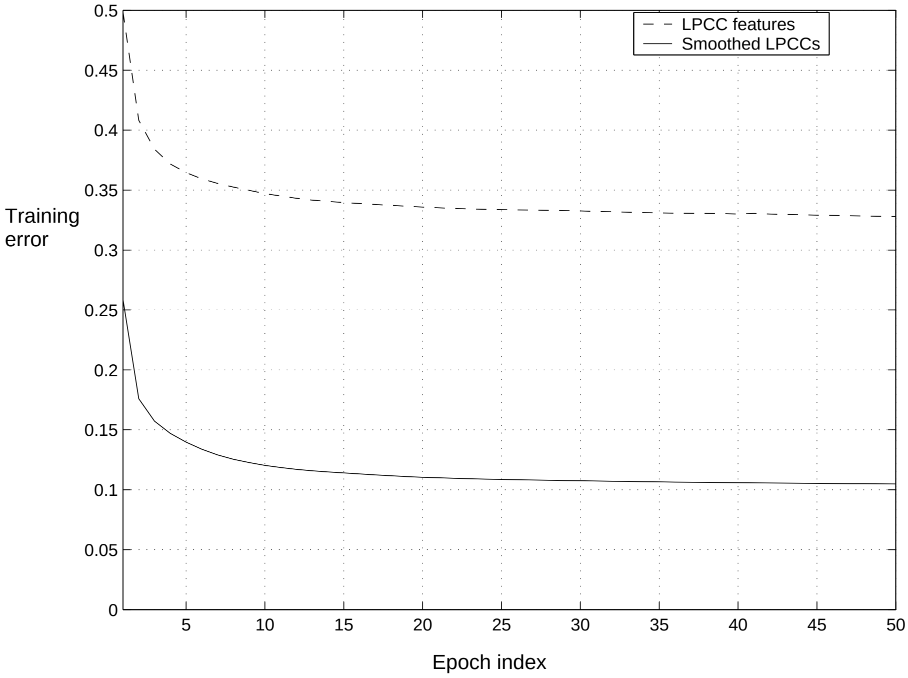


Fig. 3.3: Training error curves, when LPCC features and smoothed LPCC features were used for training AANN models for a given speaker.

Table 3.2 compares the performance of speaker verification for LPCC and smoothed LPCC features. Due to smoothing, the confidence scores of genuine and impostor models for a given test utterance increase, but the discrimination is nearly the same as the case without smoothing. Thus, the advantage of smoothing is offset by the loss of discrimination.

3.4.2 Selection of Feature Vectors for Training

When the parameters of AANN models (initial weights and learning rate) are selected suitably, the training error surface is representative of the probability density function of feature vectors. In [35], it was observed that the training of AANN models is

Table 3.2: Performance of speaker verification system after the refinements.

	Speaker verification system based on		
	LPCC features	Temporal smoothing of LPCCs	Selection of LPCCs during training
% of first ranks	72.2	72.9	70.1
EER for scaled scores (%)	12.9	13.0	13.3

influenced by the patterns that occur more frequently. Also, the training error was lower for the patterns occurring more frequently. Hence, these patterns may be more important for the estimation of probability density function as compared to the less frequently occurring ones. The former may be viewed as the denser regions of the feature space, while the latter may be termed as outliers. For speaker verification, features extracted from steady voiced regions of speech signal can be considered to lie in the denser regions of a cluster in the multidimensional feature space, while feature vectors extracted over weak voiced, unvoiced or noisy speech segments can be treated as outliers. The influence of such outlier patterns should be minimized, since they do not contain significant speaker-specific information.

The outliers in the multidimensional feature space can be eliminated to a certain extent while training the AANN model. While training, the mean and standard deviation of error is computed at regular intervals (10 epochs) for all the training patterns. Patterns having a higher deviation from the mean error are progressively eliminated from the training set. Thus, after every subsequent 10 epochs, certain number of outliers are pruned out. The model is now trained on those patterns that are significant for estimation of probability density function. The criterion for stopping the training is that either a certain number of epochs (50) be completed, or a certain minimum change in error between successive epochs is achieved, whichever happens earlier. While the elimination of outliers during training may lead to a better representation of the distribution of feature vectors, it may also reduce the possibility of matching between training and test data. This is because, the training data available for a given speaker is often limited and may not adequately represent all the categories of sound units. Due to acoustic and channel variabilities, an exact matching between the clusters of training and test data may not be achieved. The effect of mismatch seems to offset the advantage gained by the elimination of outliers, as indicated in Table 3.2.

Thus, the advantage of the methods discussed in Section 3.4.1 and the present section is that they reduce the effect of outliers in the training of AANN models. However, these methods may also reduce the feeble discrimination between the speakers even further. Hence, no major improvements are observed. Since the exact form of the probability density function of the feature vectors is not known, it is difficult to analyze the effects of these methods.

3.5 SUMMARY

This chapter described a baseline speaker verification system using LPCC features and AANN models. AANN models are used to estimate the probability density function of feature vectors. The database and performance measures to evaluate the system were discussed. Refinements were suggested, based on smoothing the feature vectors and selection of feature vectors for training. This highlights the issues of mismatch and loss of discrimination. These issues are addressed during the normalization of scores which is described in Chapter 6.

CHAPTER 4

PITCH SYNCHRONOUS ANALYSIS OF SPEECH

The goal of this chapter is to study the importance of the position of analysis window for extraction of features from speech signal. In Section 4.1, we discuss the importance of the position of analysis window with respect to the production characteristics of speech signal for accurate estimation of the vocal tract characteristics of a speaker. The instant of glottal closure, a significant event in the production of voiced speech, is described in Section 4.2, and a method to derive the same from voiced speech is reviewed. The ability of pitch synchronous analysis to accurately bring out the temporal variations of the spectral characteristics is illustrated in Section 4.3. A quantitative measure is also described, to denote the ability of a feature for effectively representing speaker-specific information. Features extracted from two methods, namely, block-based analysis and pitch synchronous analysis, are compared using quantitative measure. A speaker verification system based on pitch synchronous extraction of features is described in Section 4.4.

4.1 SIGNIFICANCE OF PITCH SYNCHRONOUS ANALYSIS OF SPEECH

Short-segment analysis of speech is performed to extract spectral information present in the signal. For this purpose, speech signal is windowed in time domain. The size of the window is dictated by the desired resolution in frequency domain and also, by the region over which speech signal can be considered quasi-stationary. The shape of the window is chosen so as to reduce the edge effects, that manifest in the frequency domain due to abrupt termination of the signal. For segmental analysis of speech, the size of the window is typically chosen in the range containing 2-4 pitch periods (30 ms) during which the characteristics of speech can be considered nearly stationary. There is another important aspect of analysis, namely, the position of the window relative to the speech signal, that is not given due consideration.

The position of analysis window is critical for extracting the dynamic source and system characteristics from speech signal. Block processing methods consider 10-30 ms of speech to estimate the characteristics of the vocal tract system in that interval. However, this smears the information within the analysis window. Consequently, the estimate of the spectrum corresponds to an average behaviour and is not accurate [48]. For instance, if the analysis window contains a region of dynamic sound, accurate temporal variation of the spectral characteristics can not be obtained by block processing. Secondly, if the analysis window contains more than one pitch period, the resulting spectrum estimate is influenced by the fundamental frequency. This is more pronounced in the case of high-pitched voices, where the short-time spectral envelope and the linear-prediction spectrum are affected by the pitch harmonics [49]. Thus, apart from the size and shape of the analysis frame, the position of the window with respect to the signal is important for accurate estimation of short-time spectrum.

In order to position the analysis window suitably, it is necessary to locate well defined events in the production of speech signal. The instant of significant excitation of the vocal tract system is one such event. For voiced sounds, the instants of significant excitation correspond to the instants of glottal closure. Once such successive events are derived from the speech signal, the analysis window can be placed relative to the events. This ensures that the segments chosen for analysis are always at the same relative position in each pitch period. Hence, the estimated spectral characteristics are more consistent across successive pitch periods. Also, temporal variation of spectral characteristics can be obtained more accurately.

4.2 DETERMINATION OF INSTANTS OF SIGNIFICANT EXCITATION

The instants of glottal closure are manifested in the voiced regions of speech. During the production of voiced sounds, air expelled from the lungs is chopped by the vibration of vocal folds, causing a quasi-periodic excitation to be delivered to the vocal tract system. The degree of opening and closing of vocal folds regulates the amount of excitation delivered to the vocal tract system. While the opening of vocal folds is gradual, the closing is relatively abrupt. It is at the instant of complete closure of vocal folds that the maximum excitation is delivered to the vocal tract system. This is called the instant of glottal closure (GC), or the instant of significant excitation [50]. This is a well manifested event in the voiced regions of speech signal, and one that can be derived from the speech signal accurately. Here, an algorithm for the determination of the instants of significant excitation is briefly reviewed.

A group-delay based method for determining the instants of significant excitation from speech signals was proposed in [51] [52]. Here, the speech signal is preemphasized and 10^{th} order LP analysis is performed on frames of 10 ms duration, with a shift of 5 ms. Speech signal is inverse filtered to obtain the LP residual signal. For each frame of LP residual of 10 ms duration, the group delay $\phi^{'}(\omega)$ is computed using the relation

$$\phi^{'}(\omega) = -\frac{X_R(\omega)Y_R(\omega) + X_I(\omega)Y_I(\omega)}{X_R^2(\omega) + X_I^2(\omega)}, \tag{4.1}$$

where $X(\omega) = X_R(\omega) + jX_I(\omega)$ and $Y(\omega) = Y_R(\omega) + jY_I(\omega)$, $X(\omega)$ is the Fourier transform of the LP residual signal $x(n)$, $Y(\omega)$ is the Fourier transform of $nx(n)$, $n = 0, 1, ..., N - 1$. The length of the signal $x(n)$ is N samples. This computation is repeated for successive frames which are obtained by sliding the window with a shift of one sample at a time. Thus, the group delay is obtained as a function of time. The average group delay for each frame known as phase slope function is computed. The phase slope function is smoothed with an 8-point (N =8) Hamming window given by

$$w(n) = 0.54 - 0.46cos\left(\frac{2\pi n}{N-1}\right), \qquad 0 \leq n \leq N-1. \tag{4.2}$$

The positive zero crossings of the phase slope function are hypothesized as the instants of glottal closure. Certain spurious instants can also be hypothesized as instants of glottal closure, in both nonspeech and speech regions. Cues based on frame energies, strength of the instants and time difference between successive instants are used to eliminate spurious instants.

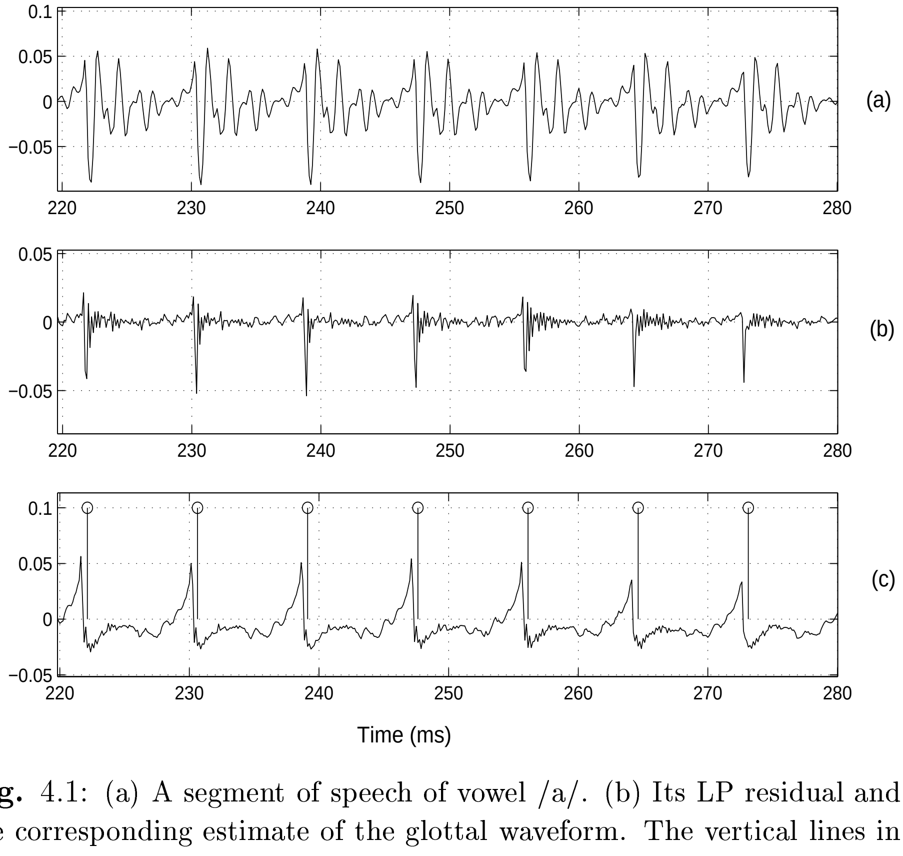
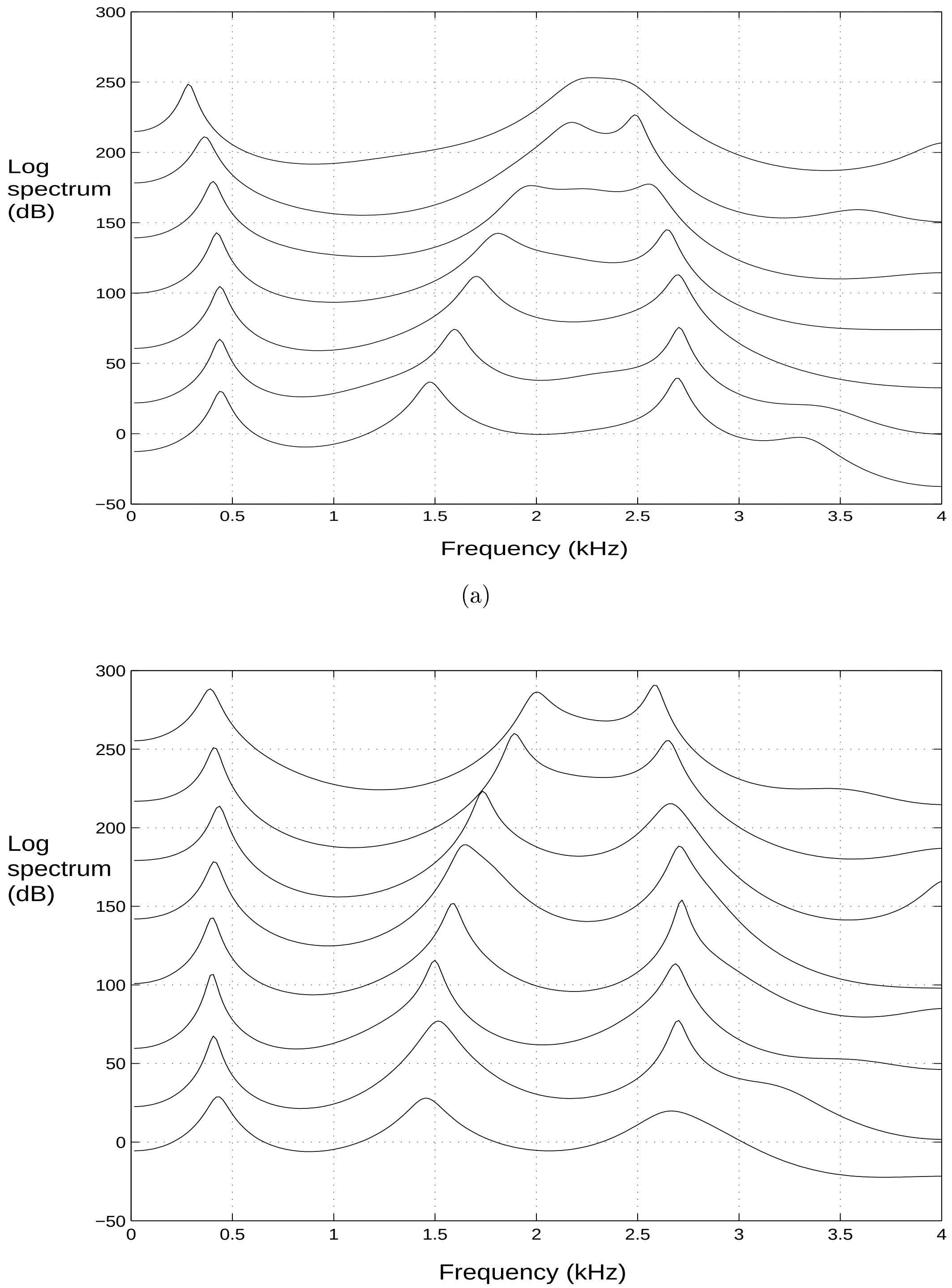


Fig. 4.1: (a) A segment of speech of vowel /a/. (b) Its LP residual and (c) the corresponding estimate of the glottal waveform. The vertical lines in (c) indicate the instants of glottal closure.

Fig. 4.1 shows a segment of speech signal of vowel /a/ for a male speaker, the corresponding LP residual and an estimate of glottal waveform. The estimate of the glottal waveform is obtained by integrating the LP residual. The instants of glottal closure are also marked on the estimate of glottal waveform. The abruptness of the glottal closure event can be observed from the glottal waveform.

4.3 EFFECTIVENESS OF PITCH SYNCHRONOUS ANALYSIS

Once the instants of glottal closure are derived from the speech signal, the next step is to select a region for analysis that encloses one pitch period. For this purpose, the analysis window is placed from a few samples to the left of a GC instant to a few samples to the left of the next GC instant, thus enclosing one complete pitch period. During the linear prediction analysis of speech, the autocorrelation coefficients evaluated using pitch synchronous window represent the properties of only the chosen pitch period, and do not suffer from the effects of smoothing as in block processing. For steady voiced regions, the spectrum does not vary appreciably from one pitch period to another. Hence, the effect of smoothing of autocorrelation coefficients due to block processing is not pronounced. However, in voiced regions with spectral transitions, block processing does not allow an accurate estimation of the spectrum. This is illustrated in Fig. 4.2. Here, a segment of speech corresponding to the word ‘they’ (in the sentence ‘have they come ?’) uttered by a male speaker is collected at 8 kHz. For the segment of 80 ms duration, 12^{th} order LP spectra are computed for block based analysis and pitch synchronous analysis. For block based analysis, a window of 20 ms was chosen with a shift of 10 ms. The effect of smoothing due to block processing can be clearly observed in Fig. 4.2(a). Pitch synchronous analysis brings out the spectral variations between successive pitch periods better than block processing. Also, peaks of the second and third formants are sharper, as shown in Fig. 4.2(b). In contrast, block processing smears the spectrum in the second and third formants, as observed from their bandwidths.



(a)

(b)

Fig. 4.2: LP spectra (12^{th} order) for successive frames in the word 'they', for a male speaker, obtained by (a) block processing and (b) pitch synchronous analysis.

Analysis of speech for closed glottis and open glottis regions was investigated in [48]. It was observed that the tracking of damped formants could be effectively done by analyzing successive frames of closed glottis. This is mainly due to the decoupling between the source of excitation and the vocal tract during the interval of closed glottis . The effectiveness of pitch synchronous analysis for high-pitched voices was also dealt with in [48]. Here, due to a short analysis frame, covariance estimates were averaged over a few successive pitch periods for reliable extraction of the vocal tract parameters.

The significance of pitch synchronous analysis for applications such as prosody manipulation and speech enhancement has been demonstrated in the literature [53] [54]. In the above applications, the effectiveness of the method of analysis is reflected in terms of the perceptual quality of the resulting speech. On the other hand, text-independent speaker verification task is based on the matching between reference features and test features. In this section, a measure of within-speaker to across-speaker dissimilarity of sounds is described, that can be used to measure the effectiveness of a feature for speaker characterization [55]. Then, features extracted using block processing and pitch synchronous analysis can be compared, based on this measure.

Let us consider a set of L speakers given by $S = \{s_1, s_2, ..., s_L\}$. Let $V = \{v_1, v_2, ..., v_M\}$ denote the set of M different sounds. For each speaker, let there be N utterances of each sound. Let $v_{i,k}$ denote the k^{th} utterance of the i^{th} sound. The within-speaker dissimilarity of a given sound v_i, for all the speakers, is given by

$$w(v_i) = \frac{1}{L}\frac{1}{N}\frac{1}{N-1}\sum_{l=1}^{L}\sum_{k=1}^{N}\sum_{n \neq k}^{N} d((v_{i,k}, s_l), (v_{i,n}, s_l)), \tag{4.3}$$

where, $d((v_{i,k}, s_l), (v_{i,n}, s_l))$ is the dissimilarity between the (sound, speaker) pairs $(v_{i,k}, s_l)$ and $(v_{i,n}, s_l)$. The across-speaker dissimilarity of a given sound v_i is given by

$$a(v_i) = \frac{1}{L}\frac{1}{N}\frac{1}{L-1}\frac{1}{N}\sum_{l=1}^{L}\sum_{k=1}^{N}\sum_{j \neq l}^{L}\sum_{n=1}^{N} d((v_{i,k}, s_l), (v_{i,n}, s_j)). \tag{4.4}$$

The within-speaker to across-speaker dissimilarity (WAD) ratio is given by

$$\alpha(v_i) = \frac{w(v_i)}{a(v_i)}. \tag{4.5}$$

The overall WAD ratio, across all sounds, is given by

$$\gamma = \frac{1}{M}\sum_{i=1}^{M} \alpha(v_i). \tag{4.6}$$

In the above equations, the sounds can be represented by any feature, and the WAD ratio is computed for that feature. A small value of γ (less than 1), for a given feature indicates the ability of the feature to provide better discrimination between speakers. Hence, the feature can be deemed more suitable to represent speaker-specific information. A larger value of γ (greater than 1) indicates that the interspeaker variability of the feature is less, and hence the feature is more suitable for representing speech information. In our experiments, a data set containing $L = 5$ speakers was considered. For every speaker, isolated utterances of $M = 5$ voiced sounds (vowels /a/, /i/, /u/, /e/ and /o/) were collected. For every speaker, $N = 5$ utterances (examples) of each sound were collected. Two approaches of analysis of speech, namely, block-based analysis and pitch-synchronous analysis were performed. For block-based analysis, frames of 20 ms were considered with a shift of 10 ms. For pitch synchronous analysis, the instants of glottal closure were determined using the algorithm described in Section 4.2. Then, a region anchored around two successive instants was chosen as an analysis frame. An LP analysis of 12^{th} order was performed using each approach and 19-dimensional LPCCs were computed. Every utterance was characterized by a unimodal, multivariate Gaussian probability density function, using the feature vectors extracted from voiced regions of that utterance. The Kullback-Leibler distance [27] was used as a measure of dissimilarity between the distributions.

Table 4.1 lists the WAD values for five sounds, computed for both the approaches of analysis. Although both the approaches compute LPCCs due to 12^{th} order LP

Table 4.1: Comparison of features extracted by block-based and pitch synchronous methods of analysis, in terms of within-speaker to across-speaker dissimilarity values.

	WAD ratio					
	/a/	/i/	/u/	/e/	/o/	Overall
LPCCs (block based)	0.153	0.1919	0.2165	0.0928	0.136	0.158
LPCCs (pitch synchronous)	0.1269	0.1350	0.1620	0.0234	0.0845	0.1064

analysis, pitch synchronous analysis results in lesser values of the WAD ratio. This can be observed for the different sounds, and hence, for the overall WAD ratio. Thus, pitch synchronous spectral features seem to be better suited for speaker verification compared to those obtained by block processing.

4.4 SPEAKER VERIFICATION STUDIES

For speaker verification studies, the database described in Section 3.2 is considered. For feature extraction, the instants of glottal closure are derived and pitch synchronous spectral features (LPCCs) are computed. The features are modeled using AANN models. Each utterance is tested against 11 claimants. The performance of verification is evaluated in terms of the percentage of first ranks obtained by genuine speakers, and EER for the scaled scores. The performance of pitch synchronous analysis for speaker verification is listed in Table 4.2. It is evident that there is only a slight improvement in the performance. The consistently lower values of WAD ratio suggested that pitch synchronous LPCCs may be more suited than block-based LPCCs. However, it is likely that the averaging of feature vectors in block-based LPCCs, which is an artifact of block processing, may actually be working to its advantage. The smoothing of spectrum due to block processing (especially the high frequency formants) may lead to a better match between training and test feature vectors. This was observed in Fig. 4.2(a). Thus, the advantage of pitch synchronous LPCCs may be offset. However, pitch synchronous analysis is important from the perspective of accurate estimation of short-time spectral characteristics for representing speaker-specific information.

Table 4.2: Comparison of block processing and pitch synchronous analysis in terms of the performance of speaker verification system.

	LPCCs computed by	
	Block-based analysis	Pitch synchronous analysis
% of first ranks	72.2	71.1
EER (%) (scaled scores)	12.9	12.2

4.5 SUMMARY

This chapter described the significance of the position of analysis window for accurate estimation of short-time spectral features. A method to detect the instants of glottal closure from voiced speech was reviewed. These instants serve as anchor points around which short-time spectral features can be extracted. The ability of pitch synchronous analysis to track the temporal variations of spectral characteristics, especially for dynamic sounds, was illustrated. A measure called within-speaker to across-speaker dissimilarity (WAD) was described, which reflects the suitability of a feature for speaker verification. The WAD values obtained on a sample dataset indicate that LPCCs extracted by pitch synchronous analysis are better suited for speaker verification, compared to those extracted by block processing. Pitch synchronous LPCCs performed better than those due to block processing for speaker verification experiments.

CHAPTER 5

EXPLORING FEATURES FOR REPRESENTATION OF SPEAKER-SPECIFIC INFORMATION

The primary step in speaker verification is the extraction of features from the speech signal. These features should characterize speaker-specific information, and they should also be robust to channel variations. Typically, spectral features such as MFCCs and LPCCs extracted from segmental analysis of speech are used for speaker verification. However, these features do not aim to specifically represent speaker-specific characteristics. In this chapter, difference cepstral coefficients are proposed as a feature for speaker verification, with the objective of highlighting speaker-specific characteristics. Section 5.1 describes the logical development of the proposed feature. A speaker verification system based on the above feature is described in Section 5.2. The ability of the proposed feature to add complementary evidence to the existing feature (LPCC) is also demonstrated.

5.1 DIFFERENCE CEPSTRAL COEFFICIENTS FOR SPEAKER CHARACTERIZATION

In this section, a brief review of linear prediction (LP) analysis of speech is presented. This is followed by a discussion on gross and fine spectra of speech, which are computed from lower and higher orders of LP analysis, respectively. This provides motivation for the extraction of difference cepstral coefficients.

5.1.1 Linear Prediction Analysis of Speech

Linear prediction analysis of speech signal [6] [15] predicts a given speech sample at time instant n as a linear weighted sum of the previous p samples, and the predicted sample is given by

$$\hat{s}(n) = \sum_{k=1}^{p} a_k s(n-k) \tag{5.1}$$

where $s(n)$ is the speech sample at time n, and $\{a_k\}$, $k = 1, 2, ...p$, is the set of predictor coefficients [6].

The prediction error $e(n)$ is defined as

$$e(n) = s(n) - \hat{s}(n). \tag{5.2}$$

The mean square of the prediction error over an analysis frame of N samples is given by

$$E = \sum_{n=0}^{N-1} e^2(n). \tag{5.3}$$

Minimizing E with respect to the set of predictor coefficients $\{a_k\}$ results in a set of p normal equations. The set of predictor coefficients $\{a_k\}$ is obtained by solving the p normal equations.

Linear prediction analysis of speech provides a reasonable approximation to both the components of speech production mechanism, namely, the source of excitation and the vocal tract system. The vocal tract system is modeled as an all-pole filter whose spectral response is described by the set of predictor coefficients $\{a_k\}$. The prediction error signal $e(n)$, also known as linear prediction residual, is a model for the source of excitation to the vocal tract system. The prediction order p has significant bearing on the ability of the all-pole filter to closely approximate the short-time spectrum of speech. Typically, the vocal tract system can be characterized by a maximum of five prominent resonances in the 0-4 kHz range. For very small orders of prediction such as

2 or 4, the LP spectrum may represent only one or two resonances. For larger values of p from 16 to 30, the LP model tries to match spurious spectral peaks of the speech signal and also the individual pitch harmonics. Therefore, an LP order of 10 to 14 is appropriate for speech signal sampled at 8 kHz to estimate the short-time spectrum, although the exact order is not very critical.

5.1.2 Interpretation of Gross and Fine Spectra of Speech Signal

The short-time spectrum of speech for a voiced sound has two components: Harmonic peaks due to periodicity of voiced speech, and gross envelope of the spectrum that reflects the vocal tract response and glottal-pulse shape [56]. The periodicity of voiced speech is due to the vibration of vocal folds, which is a property of the source of excitation. The spectral envelope is shaped by formants, that reflect the resonances of the vocal tract. Formant locations and bandwidths show variation between different speakers, even for a given category of sound unit [57]. This is due to the varying vocal tract shapes and lengths for different speakers. This variation is more pronounced in the finer fluctuations of the spectral envelope, as compared to the gross spectral envelope. To illustrate this point, speech utterances for vowel /a/ were collected from two speakers (one female and one male) over a microphone. Four such utterances were collected from each speaker at a sampling rate of 8 kHz. The instants of glottal closure were detected, and both 6^{th} order and 14^{th} order LP spectra were computed over pitch synchronous windows. Figs. 5.1(a) and (b) show the LP spectra for a female speaker, obtained by 6^{th} and 14^{th} order of LP analysis, respectively. Both the spectra are computed for the same region of speech. Similarly, Figs. 5.1(c) and (d) show the corresponding LP spectra for the male speaker. The following are observed:

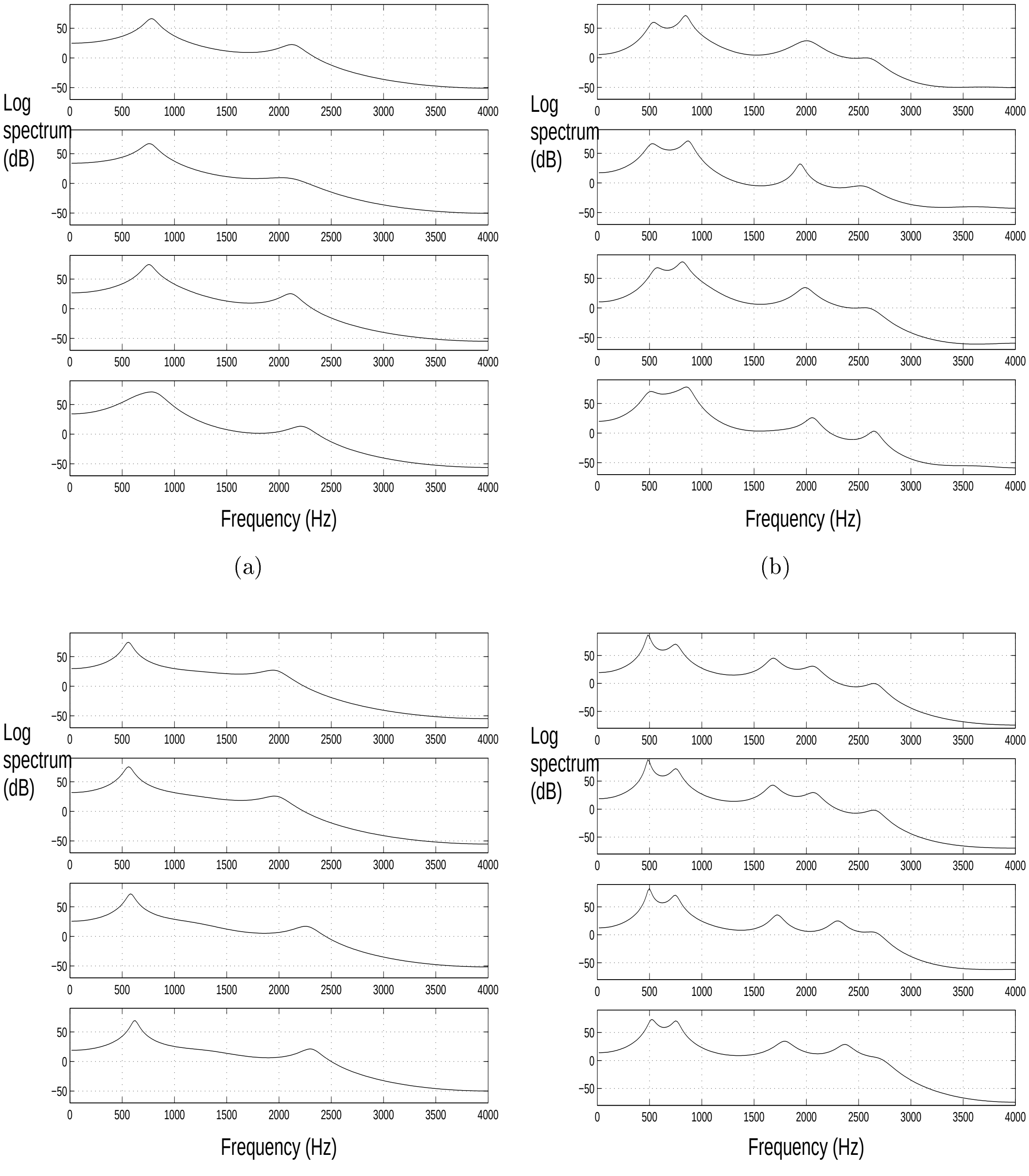
1. For different utterances of a given speaker, the corresponding 6^{th} order LP spectra are similar. The 6^{th} order LP spectra of the two speakers are also similar.

This is observed from Figs. 5.1(a) and (c).

2. While the 14^{th} order LP spectra are similar for different utterances of the same speaker, there are significant differences between the 14^{th} order LP spectra of the two speakers. This is evident from Figs. 5.1(b) and (d).

The above observations imply that the similarity between 6^{th} order LP spectra of the two speakers is due to the same underlying sound unit, while the differences between the 14^{th} order LP spectra of the two speakers is due to the speaker-specific characteristics which are different.

Fig. 5.2 shows the 6^{th} order and 14^{th} order LP spectra for five different speakers, for vowels /a/ and /i/. For the 6^{th} order LP analysis, the spectra for all the speakers are mostly similar for a given sound unit. This can be seen from Figs. 5.2(a) and (c). But, for the 14^{th} order LP analysis, the spectra of the speakers are significantly different even for the same sound unit. This is observed from Figs. 5.2(b) and (d). Thus, the gross spectrum estimated by 6^{th} order LP analysis can be viewed as representing information specific to the speech sound, while the fine spectrum estimated by the 14^{th} order LP analysis represents both sound unit information as well as speaker-specific information.



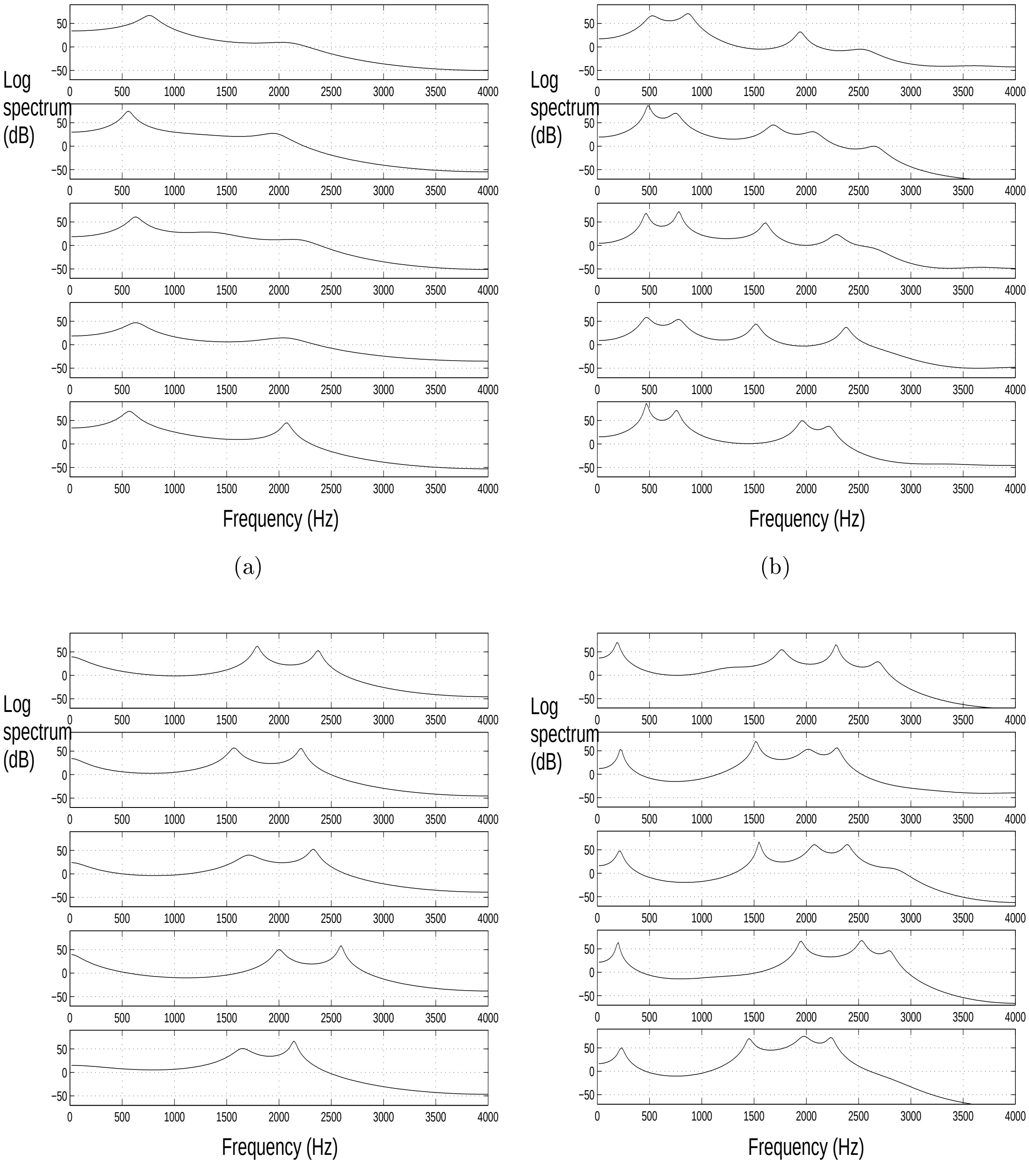
(a)

(b)

(c)

(d)

Fig. 5.1: (a) and (b) are, respectively, the 6^{th} order and 14^{th} order LP spectra for four different utterances of the same vowel /a/, as uttered by a female speaker. (c) and (d) are similar plots for a male speaker. The sampling frequency is 8 kHz.



(a)

(b)

(c)

(d)

Fig. 5.2: (a) and (b) are, respectively, the 6^{th} order and 14^{th} order LP spectra of five different speakers, for the vowel /a/. (c) and (d) are similar plots for vowel /i/. In each plot, the first speaker is female and the remaining speakers are male. The sampling frequency is 8 kHz.

5.1.3 Extraction of Difference Cepstral Coefficients

In order to deemphasize the influence of the sound unit, the difference of the fine spectrum and the gross spectrum is considered. This difference still preserves the finer spectral variations that represent speaker-specific characteristics. For the purpose of representation, this subtraction is done in the cepstral domain. Firstly, the set of cepstral coefficients is derived from the LP coefficients [15]. Cepstral coefficients provide a compact representation of the resonances and the spectral roll-off characteristics of the vocal tract system. The set of cepstral coefficients $\{c_k\}$, $k = 0, 1, ..., m$, is obtained from the set of predictor coefficients $\{a_k\}$, $k = 1, 2, ..., p$, using the following recursive relation:

$$\begin{aligned} c_0 &= logE_{min} \\ c_k &= -a_k + \sum_{j=1}^{k-1} \frac{j}{k} c_j a_{k-j} && 1 \leq k \leq p \\ c_k &= \sum_{j=k-p}^{k-1} \frac{j}{k} c_j a_{k-j} && p < k \leq m \end{aligned} \tag{5.4}$$

where m is the number of cepstral coefficients, and E_{min} is minimum mean squared prediction error.

The set of difference cepstral coefficients $\{d_k\}$, $k = 1, 2, ..., m$ can be expressed as

$$d_k = k(c_k^h - c_k^l) \qquad 1 < k \leq m \tag{5.5}$$

where $\{c_k^h\}$ is the set of cepstral coefficients due to a higher order of LP analysis, $\{c_k^l\}$ is the set of cepstral coefficients due to a lower order of LP analysis. The comparable range of amplitudes of the cepstral coefficients of the two spectra leads to noise in the difference cepstral coefficients. Hence, the difference cepstral coefficients d_k are averaged over a window of M contiguous frames of a region of voiced speech, as follows:

$$\hat{d}_{k,j} \;=\; \frac{1}{M}\sum_{i=j-\frac{M}{2}}^{j+\frac{M}{2}} d_{k,i} \qquad\qquad 1 < k \leq m, \qquad (5.6)$$

where $\{\hat{d}_{k,j}\}$ is the set of averaged difference cepstral coefficients for segment j of the region of voiced speech, and $\{d_{k,i}\}$ is the set of difference cepstral coefficients for frame i.

The differencing of the cepstra also reduces the influence of the transmission channel characteristics on the speech signal. This obviates the need for cepstral mean subtraction, that is normally employed to remove the mean of the time trajectory of each cepstral coefficient [12] [7].

5.2 SPEAKER VERIFICATION USING DIFFERENCE CEPSTRAL COEFFICIENTS

A speaker verification system is developed using difference cepstral coefficients, on similar lines to that of the baseline system described in Section 3.1. Difference cepstral coefficients are extracted as described in Section 5.1.3. A 5-layer AANN model of structure $19L$ $38N$ $4N$ $38N$ $19L$ is used, which is trained using difference cepstral coefficients. This choice of the structure of AANN model for LPCC features was based on a study reported in [58]. In that study, the number of units in layers 2 and 4 were chosen empirically to be twice the dimension of the input vector. The number of units in the compression layer was arrived at, after systematic experimentation. The study was repeated for difference cepstral coefficients and it was observed that the same structure of AANN model was suitable. Each model is trained for 50 epochs. Each utterance is tested against 11 claimants.

The rank of the genuine speaker among the 11 claimants is computed for each test utterance. The number of test utterances where the genuine speaker secures the first

Table 5.1: Performance of speaker verification for LPCCs and difference cepstral coefficients.

	Speaker verification system based on		
	LPCC features	Difference cepstral coefficients	Combination using OR logic
% of first ranks	72.2	67.3	77.7

rank is also computed. A combination of the ranks is performed using OR logic. Table 5.1 compares the performance of LPCC features and difference cepstral coefficients, in terms of the percentage of first ranks. The figure in the third column represents the percentage of first ranks obtained by the genuine speaker, using either LPCC features or difference cepstral features or both. It is observed that the combination results in an improved performance of verification. This indicates that difference cepstral coefficients do contain speaker-specific features that are complementary in nature to LPCC features. Here, the combination of ranks has been performed using the OR logic. This is only to establish that difference cepstral coefficients indeed contain speaker-specific information that is complementary to LPCC features. However, when considering a system that uses both LPCC features and difference cepstral coefficients, the performance analysis in terms of EER requires that the scores due to the two features be combined suitably. This is a combination at the level of measurements and is not as straightforward as a logical OR operation performed on the ranks. Hence, the combination of evidences due to multiple features, and the performance analysis of such a system are discussed in Section 6.3.

5.3 SUMMARY

In this chapter, the development of a feature for representing speaker-specific information was described. The gross spectrum was shown to be representative of the sound unit, while the fine spectrum was shown to contain both speech and speaker-specific characteristics. These spectra were estimated using different orders of LP analysis. Difference cepstral coefficients were extracted from the cepstral representations of gross and fine spectra. A speaker verification system based on difference cepstral coefficients was shown to provide some complementary evidence for verification.

In Chapters 4 and 5, extraction of features for speaker verification was discussed. Probability density function of the feature vectors was estimated using autoassociative neural network models. Once a model is built, it is presented with the feature vectors derived from an unknown utterance. The decision for accepting or rejecting the claim is based on the score output by the model. In the next chapter, we discuss the issue of score normalization for speaker verification.

CHAPTER 6

SCORE NORMALIZATION FOR SPEAKER VERIFICATION

The decision mechanism for speaker verification depends on the score output by the model of a speaker, when presented with an unknown (test) utterance. This score is compared to a threshold in order to accept or reject the claim of the speaker. But generally, the scores obtained from different models and test utterances are not in the same range. The task of computing a calibrated score is known as score normalization. In Section 6.1, the need for score normalization in speaker verification is discussed. Some methods for score normalization are proposed in Section 6.2. The performance of the proposed approaches is compared against that of the existing approaches. Section 6.3 discusses combination of evidences from complementary features for improving the performance of speaker verification. Section 6.4 compares the performance of the speaker verification system described in this thesis with that of a few other systems.

6.1 NEED FOR SCORE NORMALIZATION

The raw scores obtained from the models can not be used for decision making as discussed in Chapter 2. To summarize:

1. The nature of training data differs from one speaker to another. Specifically, the difference is due to the amount of training data, composition of the data in terms of acoustic categories, and the channel effects.

2. Mismatch between the training and test data can lead to low scores, even from the model of genuine speaker. This is due to channel effects, or inadequate representation of certain acoustic categories in the training data.

The ability to discriminate between genuine and impostor speakers differs among models. This ability also differs among test utterances for a given model. To illustrate the effect of these factors, the distributions of the confidence scores of genuine and impostor speakers are observed for the baseline system. Fig. 6.1 shows the estimated distributions of the confidence scores for genuine and impostor speakers. If a significant overlap exists between the two, it makes the task of setting a decision threshold difficult. Thus, there is need to improve the discrimination between the scores of genuine and impostor speakers for reliable decision-making. Table 6.1 shows the EER obtained for the raw and scaled scores. Each test has 11 claimants, and the scaled scores are obtained by dividing all the 11 scores by the maximum. The scaling of scores serves

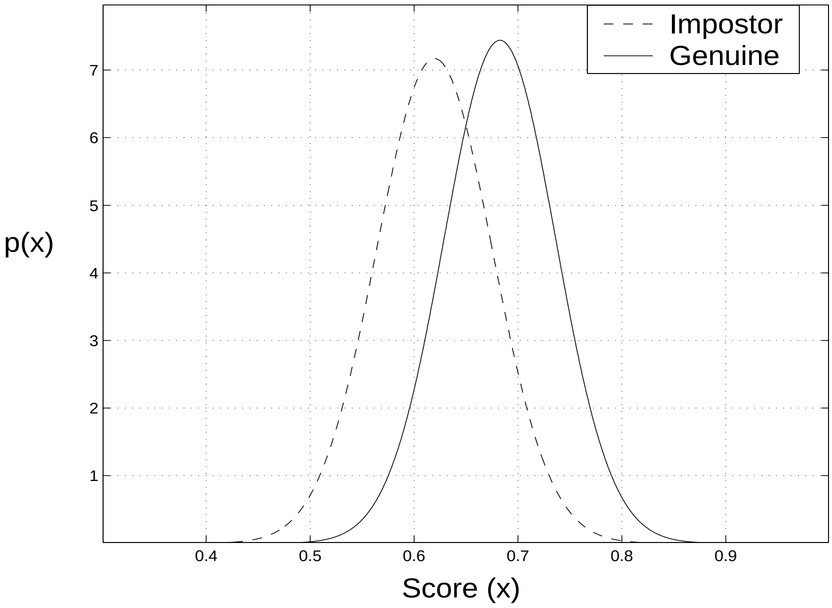


Fig. 6.1: Estimated distributions of the confidence scores of genuine and impostor speakers.

as a normalization because genuine speakers who are winners in their respective tests, have the same score of 1 after normalization. This normalization is reflected in the value of EER for scaled scores. However, the decision of verification should be based on the score of a given model alone. Hence, scaling the scores as mentioned above is not appropriate. This necessitates the need for a common threshold for a given speaker verification system.

Table 6.1: Performance of speaker verification for raw and scaled scores.

% of first ranks	EER(%) for raw scores	EER(%) for scaled scores
72.2	26.1	12.9

6.2 METHODS FOR SCORE NORMALIZATION

Methods of score normalization can be classified as model normalization and test utterance normalization. In model normalization, a speaker's model is tested against example impostor utterances and the resulting scores are used to estimate speaker-specific statistics. In test utterance normalization, the test utterance is compared against the model of a claimant speaker, and also, background/cohort models. The scores of the background models are used to normalize the speaker's score for that utterance. In this section, three different methods of score normalization are proposed. Section 6.2.1 describes a method of model normalization. Sections 6.2.2 and 6.2.3 describe two methods of test utterance normalization.

6.2.1 Modeling Speaker-specific Distribution of Impostor Scores

The training data available to develop a model differs from one speaker to another. Hence, the likelihood/confidence scores resulting from different models cannot be compared to a single threshold for acceptance or rejection. The task of model normalization is to compute the calibrated scores, so that a common threshold for decision can be used across all the speakers.

Let a sequence of feature vectors $X = \{\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_N\}$, derived from the speech of one or more impostors, be presented to the model of a speaker, denoted by M. Speech from 50 impostors was used, with 20 seconds of speech for each impostor. The model M outputs a corresponding sequence of scores $C = \{c_1, c_2, ..., c_N\}$, whose mean and standard deviation are denoted by m_i and σ_i respectively, where the subscript i denotes impostor. The idea of presenting the model M with the feature vectors derived from impostors is to estimate the behaviour of the model for impostors. This is typically done offline. During verification, feature vectors derived from a test utterance are presented to the model M. Let this sequence of test feature vectors $Y = \{\mathbf{y}_1, \mathbf{y}_2, ..., \mathbf{y}_L\}$, when presented to the model M, result in a sequence of scores $S = \{s_1, s_2, ..., s_L\}$ having a mean m_t and standard deviation σ_t, where the subscript t denotes the test utterance. The existing method of normalization (Z-norm) [43] computes the normalized score as $c_{norm} = \frac{s-m_i}{\sigma_i}$, where $s = \frac{1}{L}\sum_{k=1}^{L} s_k$. This method uses only the average value s to compute the normalized score and does not exploit the distribution of the scores C and S. Instead, a method is proposed where the probability density functions of the scores are estimated from C and S. Observation of histograms of scores obtained from C and S for several cases showed that the histograms can be approximated by Gaussian probability density functions. For estimation of $p_i(c)$, features are collected offline and typically, the number of feature vectors (and hence, the number of scores) is in excess of 1,00,000. For estimation of $p_t(c)$ from the test data, the number of scores is typically above 10,000 and almost always, above 5000. This is large enough to obtain the histogram of scores, by dividing the interval 0 to 1 into 10 equally spaced bins.

Thus, the probability density functions of the scores obtained from C and S can be modeled as Gaussian densities. Let $p_i(c)$ and $p_t(c)$ represent the estimates of the probability density functions of the scores C and S, respectively. Then, $p_i(c) = N(m_i, {\sigma_i}^2)$ and $p_t(c) = N(m_t, {\sigma_t}^2)$, where $N(m, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(\frac{-(x-m)^2}{2\sigma^2}\right)$, represents a normal (Gaussian) density with mean m and standard deviation σ. Due to the availability of substantial amount of data from impostors, $p_i(c)$ is a good estimate of the actual probability density function. If the test utterance belongs to an impostor, then $p_t(c)$ is expected to match $p_i(c)$ closely. However, if the test utterance belongs to the genuine speaker of model M, the match between $p_t(c)$ and $p_i(c)$ should reduce, with $m_t > m_i$. Thus, the decision for verification can be based on the degree of match between $p_t(c)$ and $p_i(c)$. Fig. 6.2 illustrates a case where a model is presented with three different test utterances. When comparing $p_t(c)$ and $p_i(c)$, the following cases were observed:

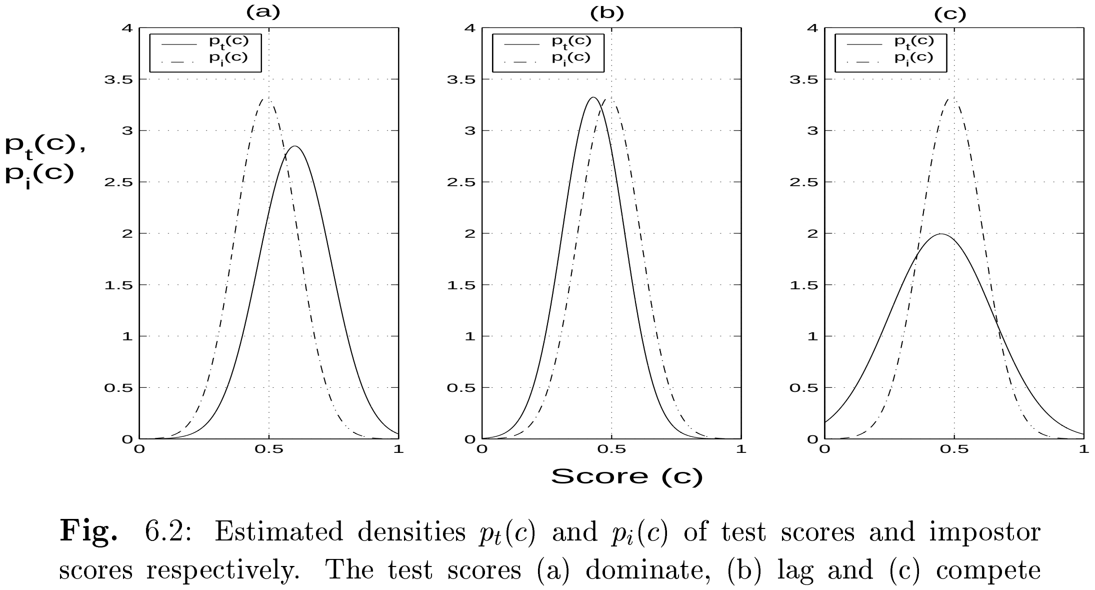


Fig. 6.2: Estimated densities $p_t(c)$ and $p_i(c)$ of test scores and impostor scores respectively. The test scores (a) dominate, (b) lag and (c) compete against the impostor scores.

1. $p_t(c)$ has a significant region that does not overlap with $p_i(c)$, and $m_t > m_i$, as shown in Fig. 6.2(a). It is likely that the test utterance belongs to the genuine speaker, and the model M reasonably represents the distribution of feature vectors of the training and test data. Though less likely, it is also possible that the test utterance belongs to an impostor. This indicates that, due to intraspeaker variability or the effects of the channel, the test feature vectors now 'fall' more often into the clusters represented by the model M.

2. $p_t(c)$ and $p_i(c)$ overlap mostly, with $m_t < m_i$, as shown in Fig. 6.2(b). The more likely inference here is that the test feature vectors belong to an impostor, since feature vectors from the genuine speaker should have resulted in a better match with the model M. A less likely inference is that the test utterance belongs to the genuine speaker.

3. The distributions $p_t(c)$ and $p_i(c)$ lie very close, but $p_t(c)$ 'crosses over' $p_i(c)$ as shown in Fig. 6.2(c). This indicates a good match between $p_t(c)$ and $p_i(c)$ leading to the inference that the test utterance belongs to an impostor. However, in the region of high scores (say, $0.6 < c < 1$), $p_t(c)$ exceeds $p_i(c)$. The scores of $p_t(c)$ in this region may correspond to those frames of the test utterance that closely match the model M. Thus, it is still possible that the test belongs to the genuine speaker.

The above cases are not exhaustive, but they are representative of the general behaviour. Based on these observations, a matching score needs to be computed for verification. A quantitative measure of the match between the two distributions can be computed from the plots of $p_t(c)-p_i(c)$. Figs. 6.3(a), (b) and (c) show $p_t(c)$ for three different tests, against the same model. The corresponding plots of $p_t(c) - p_i(c)$ are shown in Figs. 6.3(d), (e) and (f), respectively. If the area under the curve $p_t(c)-p_i(c)$ is positive in the region of high scores, as in Fig. 6.3(d), then the test utterance is likely to belong to the genuine speaker. If this area is negative as shown in Fig. 6.3(e), then the test speaker is likely to be an impostor.

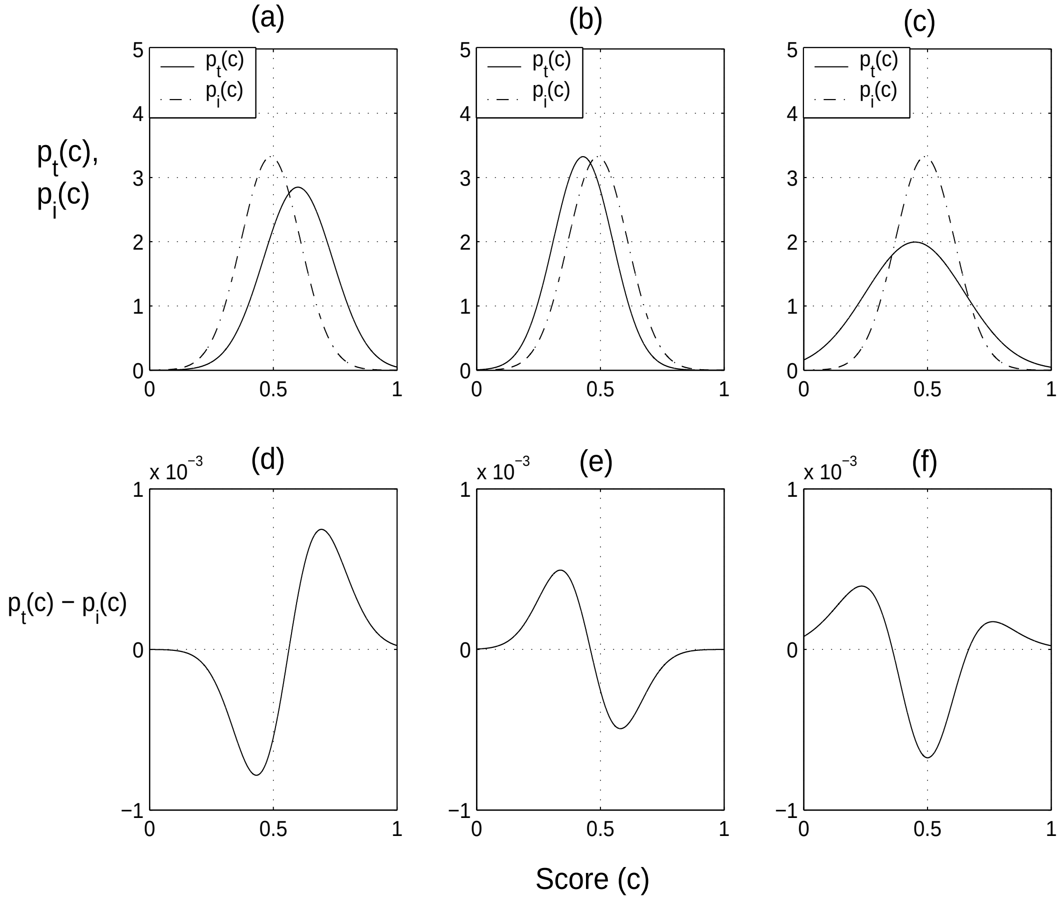


Fig. 6.3: Plots (a), (b) and (c) show $p_t(c)$ for three different test utterances, for a given model. $p_i(c)$ is estimated a priori for the given model. Plots (d), (e) and (f) show $p_t(c) - p_i(c)$ for (a), (b) and (c) respectively.

Figs. 6.3(c) and (f) show a case where $p_t(c)$ 'crosses over' $p_i(c)$ in the region of higher scores. Here too, a positive area exists in the region of higher scores, indicating that the test speaker may be genuine. Thus, positive area under the curve $p_t(c)-p_i(c)$, in the region of high scores, should be considered for scoring. The normalized score can be obtained as $c_{norm} = \sum_{c=m_i}^{1} c(p_t(c) - p_i(c))$. The lower limit of c is chosen as m_i to exclude non-contributing scores. Multiplication by c is intended to provide more weightage to the scores of greater magnitude. The algorithm for model normalization is summarized in Table 6.2.

Table 6.2: Sequence of steps involved in model normalization.

```
1. Present $X = \{\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_N\}$ to $M$, to obtain $C = \{c_1, c_2, ..., c_N\}$.
2. Compute $m_i$ and $\sigma_i$ from $C$.
3. Present $Y = \{\mathbf{y}_1, \mathbf{y}_2, ..., \mathbf{y}_L\}$ to $M$, to obtain $S = \{s_1, s_2, ..., s_L\}$.
4. Compute $m_t$ and $\sigma_t$ from $S$.
5. Obtain the estimates $p_i(c) = N(m_i, {\sigma_i}^2)$ and $p_t(c) = N(m_t, {\sigma_t}^2)$.
6. Compute $c_{norm} = \sum_{c=m_i}^{1} c(p_t(c) - p_i(c))$.
```

Table 6.3 compares the performance of the proposed method against Z-norm. It is observed that the proposed method does not result in appreciable improvement in EER, compared to that obtained from Z-norm. The logic behind model normalization is that the example impostor utterances can represent the response of a given model for any impostor data. However, the conditions under which the test speech is collected may differ from those of the example impostor utterances. Thus, the acoustic mismatch between the test utterance and the example impostor utterances limits the effectiveness of model normalization. This issue is addressed in test normalization.

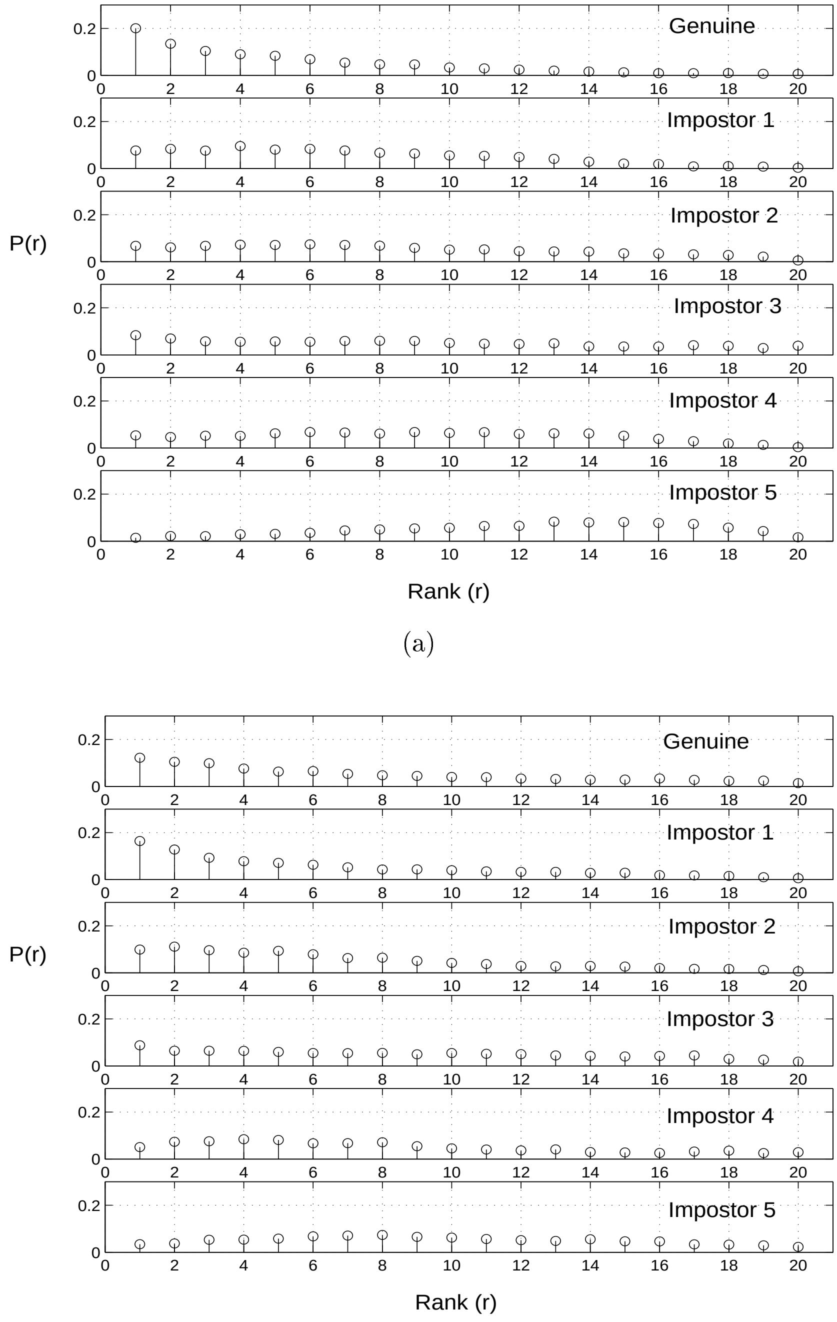
Table 6.3: Performance of different model normalization methods.

	Raw scores	Z-norm	Proposed model normalization
EER (%)	26.1	24.0	23.5

6.2.2 Rank-based Normalization of Scores

A disadvantage of the existing test normalization methods (notably T-norm) is that they consider only the average value of the scores output by a model for a given test utterance [43] [44]. This provides equal weightage to all the frames of the test utterance. However, it is not necessary that all frames of the test utterance are equally important for speaker verification. Some methods consider the sum of only the top M ranked scores, where M is less than the number of segments in the test utterance [59]. Such methods help in eliminating the less significant frames, but a disadvantage is that the sum of top M ranked scores is not normalized across different test utterances. In the proposed method, a set of N background models is used for score normalization. Background models help in estimating the behaviour of impostors. These background models are randomly chosen and are common to all the test utterances. A given test utterance is presented to a claimant model along with the N background models. For every frame of the test utterance, the score due to the claimant model is ranked among the scores due to the N background models. Thus, the rank of the claimant can vary between 1 and $N+1$. The normalized score is computed as the percentage of the total number of frames where the genuine speaker wins over all the background models. The choice of N, the number of background models, should result in a reasonable estimate of the behaviour of impostors. A large value of N such as 50 dilutes the evidence due to the genuine speaker. On the contrary, with a small value of N such as 5, the possibility of an impostor obtaining as many first ranks as the genuine speaker is high. Thus, not enough background models are there to challenge the genuine or impostor speakers. In this experiment, 20 background models have been chosen. Fig. 6.4 shows the fraction $P(r)$ of the total number of frames to have obtained rank r. To illustrate, two test utterances are considered. Fig. 6.4(a) shows a case where an utterance is tested against the genuine speaker, 5 impostors and 20 background models. The genuine speaker scores over the impostors, as observed from the value of $P(1)$. Fig. 6.4(b) shows another case where Impostor 1 has a slightly higher value of $P(1)$ as compared to that of the genuine speaker, leading to false acceptance. The choice of $P(1)$ as the normalized score implies that only those frames that rank first are considered for normalization. However, the number of frames that rank second or third may still be important for discrimination. This issue is addressed in Section 6.2.3. The algorithm of rank-based normalization of scores is summarized in Table 6.4.

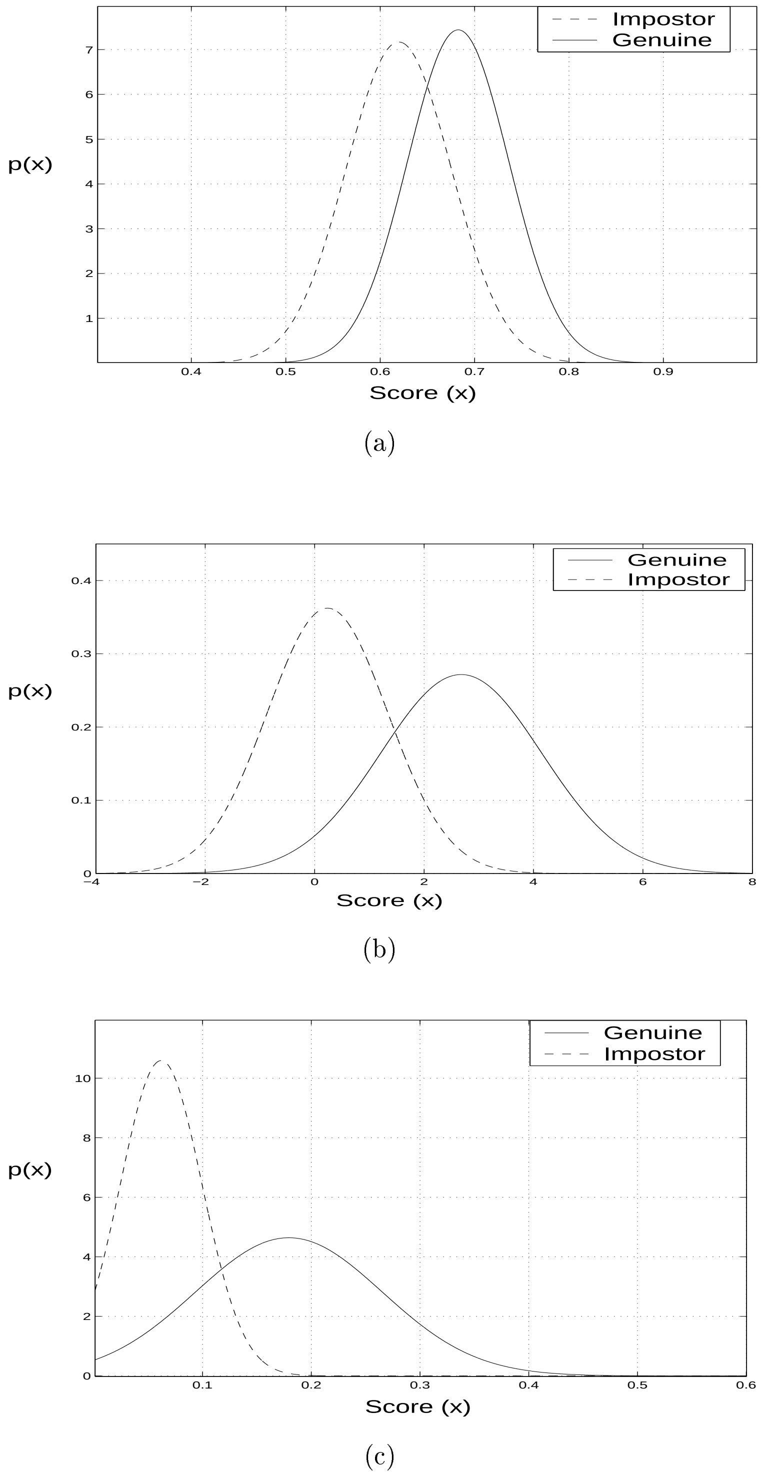
Experiments were conducted on NIST 2003 database, and normalized scores were evaluated for genuine and impostor speakers. Fig. 6.5 plots the estimates of probability density functions, of scores obtained from the models of genuine and impostor speakers. Figs. 6.5 (a), (b) and (c) show, respectively, the densities of raw scores, normalized scores due to the existing methods (Z-norm + T-norm), and normalized scores due to the proposed rank-based approach. From Figs. 6.5 (a) and (c), it is evident that the proposed method significantly improves the discrimination between genuine and impostor speakers, as compared to the raw scores. Figs. 6.5 (b) and (c) indicate that the scores of impostors have lesser variance for the proposed method, compared to the existing method. This is significant for setting the decision threshold. This indicates some uniformity in the behaviour of the normalized scores of impostors. For genuine speakers, the variance of the normalized scores is greater than that of the raw scores, because the degree of discrimination between genuine and impostor speakers may vary from one genuine speaker to another. The performance of this method will be discussed in Section 6.2.4.



(a)

(b)

Fig. 6.4: $P(r)$ is that fraction of the total number of frames, which has obtained rank r. (a) A test utterance where the genuine speaker scores over the impostors. (b) A test utterance where impostors (1 and 2) compete with the genuine speaker.



(a)

(b)

(c)

Fig. 6.5: Estimates of the probability density functions of genuine and impostor scores. (a) Raw scores, (b) scores normalized by Z-norm + T-norm and (c) scores normalized by rank-based method.

Table 6.4: Sequence of steps involved in rank-based normalization.

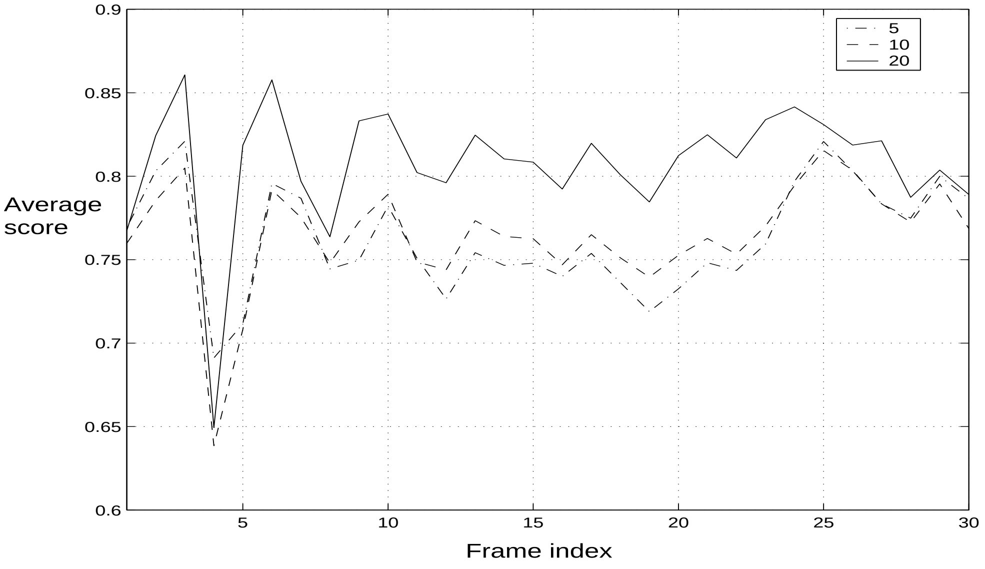
1. Select an appropriate number (N) of background models.
2. Present the test utterance to the claimant model and N background models.
3. For each frame, compute the rank (r) of the claimant among the N background models.
4. Compute $P(r)$, $r = 1, 2, ..., N + 1$, i.e., that fraction of the total number of frames which has obtained rank r.
5. $P(1)$ is the normalized score.

6.2.3 Method Based on Frame-level Weighting of Scores

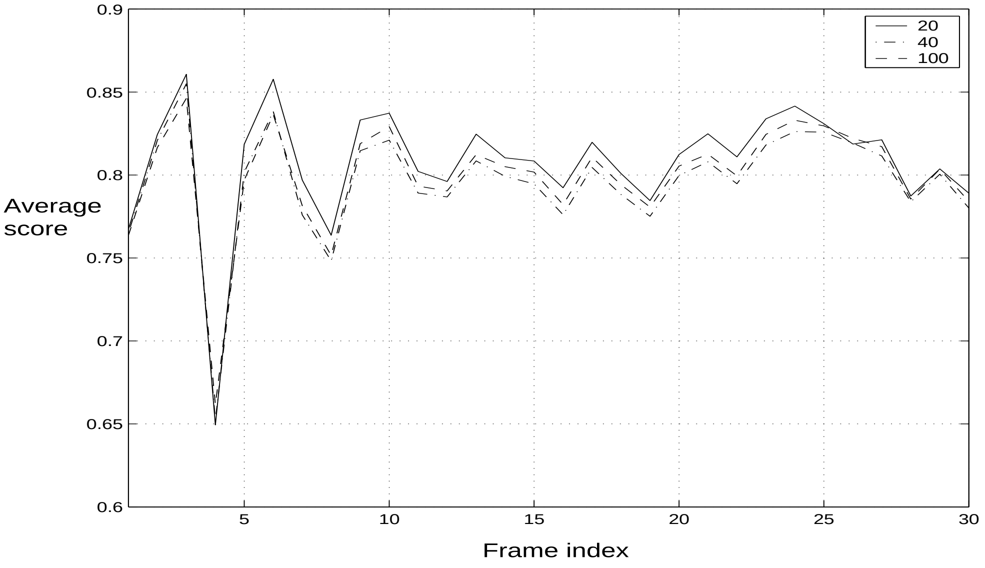
The degree to which a test utterance matches the corresponding (genuine) speaker model varies for different test utterances. To an extent, this degree of match depends on the nature of the test utterance. In test normalization, the objective is to estimate the average behaviour of impostors for the test utterance. The test normalization scheme T-norm described in [43] computes the mean and standard deviation of the average scores of several background models for a given test utterance. By averaging the scores due to all the frames, this method provides equal weightage to all the frames of the test utterance. However, some frames of the test utterance may contain greater speaker-specific information compared to other frames. In [60], it is shown that statistical modeling of speaker-specific characteristics using only two broad phonetic categories (vowel + diphthongs and glides + nasals) resulted in better verification performance than the case when all the phonetic categories were used. The phonetic categorization of frames was achieved by using an automatic speech recognizer. Apart from phonetically less-significant frames, the test utterance may also contain spurious frames in nonspeech regions of the signal, inspite of using a good speech-nonspeech detection in the preprocessing stage. This is possible in the case of energy based-methods of speech-nonspeech detection. The removal of such spurious frames may be achieved by using a suitable signal-processing algorithm. The aim of the current experiment is to weight the frames of the test utterance at the scoring level.

In the proposed method, a test utterance is presented to a claimant model and a set of N background models. For every frame of the test utterance, average of the scores of background models is computed. The reason for computing the average of frame-level scores of background models is to provide different weightages to different frames of the test utterance. This will be described later in this section.

The first issue is the number of background models to be selected. In the experiment, utterances were tested against varying number of background models. Each utterance was tested against 5, 10, 20, 40, 60, and 100 background models, and frame-level average of scores was computed. Figs. 6.6(a) and 6.6(b) show the plots of frame-level scores averaged over different number of background models. It is observed that the variation of the average of frame-level scores is not significant beyond 20 background models. Hence, a set of 20 background models is used in further experiments on test normalization.



(a)



(b)

Fig. 6.6: Frame-level average of scores for varying number of background models, for a given test utterance. (a) For 5, 10 and 20 background models. (b) For 20, 40 and 100 background models. Legend indicates the number of background models.

Once the average of frame-level scores of the background models is computed, the next step is to compute the normalized score. Let $\{c_k\}, k = 1, 2, ..., L$ represent the sequence of frame-level scores obtained when a test utterance is presented to a claimant model. Let $\{b_k\}, k = 1, 2, ..., L$ represent the sequence of average of frame-level scores of the background models for the same utterance. Here, L denotes the total number of frames in the test utterance. The difference score can be defined as

$$d_k = c_k - b_k \qquad\qquad k = 1, 2, ..., L \tag{6.1}$$

To select only those frames where the claimant score exceeds the average of background scores, we define

$$f_k = \begin{cases} 1, & d_k > 0, \qquad\qquad k = 1, 2, ..., L \\ 0, & otherwise. \end{cases}$$

The normalized score can be computed as

$$s = \frac{1}{L}\sum_{k=1}^{L} f_k d_k. \tag{6.2}$$

However, all such frames are given equal weightage in the above scoring scheme. Hence, a weighting function is derived using the frame-level average of background scores, as follows:

$$w_k = \frac{b_k - b_{min}}{b_{max} - b_{min}} \qquad\qquad k = 1, 2, ..., L. \tag{6.3}$$

The difference scores are then weighted with this function for only those frames where the claimant score exceeds the average of background scores. The final score is computed as

$$s_{norm} = \frac{1}{L}\sum_{k=1}^{L} f_k w_k d_k. \tag{6.4}$$

The algorithm is summarized in Table 6.5.

The reason for computing the weight function is the following: If a test frame is poor / spurious, it is likely to result in a lower value of confidence score from most of the background models. If the test frame belongs to a well-manifested region of speech, it is likely to result in a higher value of confidence score from most of the background models. Thus, the frame-level average of scores of background models is a representative of the nature of the test utterance. Fig. 6.7 shows the variation of frame-level confidence scores for a given test segment for genuine and impostor speakers. The percentage of frames, where the frame-level score exceeds the frame-level average of the backgrounds, is a factor of normalization. The performance of this method of normalization is discussed in Section 6.2.4.

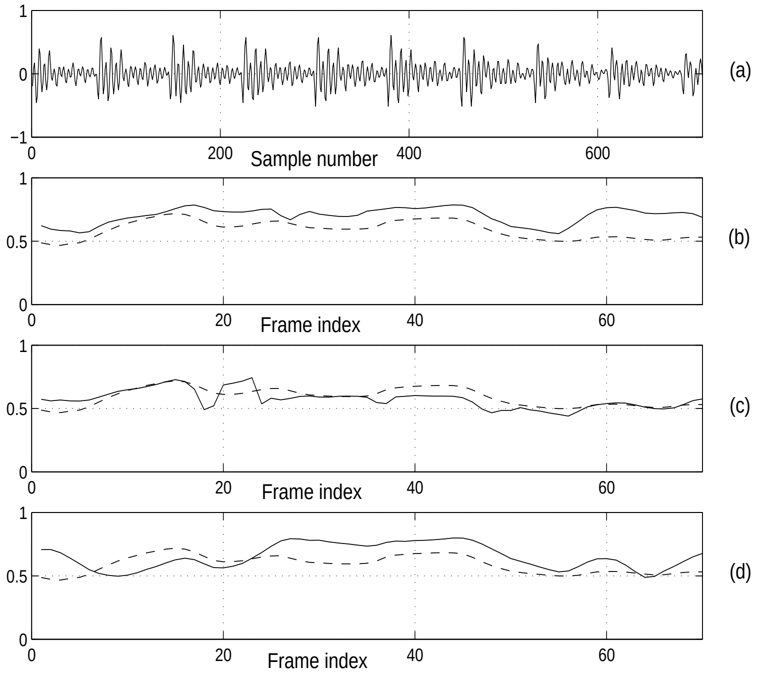


Fig. 6.7: (a) A segment of test speech signal. The corresponding framewise confidence scores obtained from the model of (b) genuine speaker and (c), (d) impostor speakers, shown by solid lines. In (b), (c) and (d), the broken lines represent the framewise average of scores of background models.

Table 6.5: Sequence of steps involved in frame-level weighting of scores.

1. Select a suitable number of background models, by experimentation.
2. Present the test utterance to claimant model, to obtain the scores $\{c_k\}, k = 1, 2, ..., L$.
3. Present the test utterance to N background models and compute the frame-level average of background scores $\{b_k\}, k = 1, 2, ..., L$.
4. Compute the difference score $d_k = c_k - b_k$, $k = 1, 2, ..., L$.
5. Compute the binary weight $\{f_k\}, k = 1, 2, ..., L$.
6. Compute the weight function $\{w_k\}, k = 1, 2, ..., L$.
7. Compute the normalized score as $s_{norm} = \frac{1}{L}\sum_{k=1}^{L} f_k w_k d_k$.

6.2.4 Results and Discussion

In this section, we discuss the performance of the proposed methods of test normalization. Table 6.6 lists the results of the proposed methods of test normalization, along with the existing (Z-norm + T-norm) scheme. The performance of the rank-based approach is better than that of T-norm, and comparable to T-norm + Z-norm scheme.

The existing approach estimates the mean and variance of the scores of background models, using all the frames of the test utterance. In comparison, the rank-based method described in Section 6.2.2 considers only those frames for computing the score, that consistently win over the background models. Also, the normalized score is limited to a range of 0 to 1. On the other hand, in Z-norm and T-norm, the scaling of scores by variance causes the normalized scores to acquire a greater range.

A limitation of the rank-based approach is that it does not consider those frames for scoring that are ranked second or third. To overcome this limitation, the normalized score was computed as a weighted average of the percentage of first, second and third ranks. However, this did not result in the reduction of EER.

Table 6.6: Performance of different test normalization methods.

	T-norm	Z-norm + T-norm	Rank based normalization	Frame-level weighting of scores
EER (%)	19.1	16.1	16.5	15.2

This may be due to the extent of discrimination between the genuine and impostor speakers for different frames of the test utterance. For example, all the frames that secure first ranks may not be equally significant in terms of speaker-specific information. Similarly, frames securing second and third ranks may also be useful for decision, although the rank-based method does not use this information.

The method based on frame-level weighting of scores overcomes this limitation to a certain extent. The selection of only those frames where the claimant scores exceed the average of background scores, is a policy that is similar to the rank-based approach. Yet, it is not as harsh as ignoring the second and third ranked frames altogether. The weight function derived from the background scores serves as a measure of significance of each frame for scoring. The improvement obtained by this method over the existing methods is indicated in Table 6.6.

6.3 COMBINING EVIDENCES FROM MULTIPLE FEATURES

The goal of speaker verification is to validate the identity of a speaker, based on the voice characteristics of the speaker. Traditionally, speaker verification systems use a single feature to represent speaker-specific information and a single modeling technique. In pattern classification problems, studies have shown that it is possible to improve the reliability of classification by using different types of features and models simultaneously [61–63]. In the context of speaker verification, different features can be extracted from speech to represent speaker-specific information. These features may represent the vocal tract system or the source of excitation. The features may be extracted over different levels of analysis. For instance, combination of evidences due to subsegmental, segmental and suprasegmental features has been studied for text-dependent speaker verification [24] [64]. For speaker verification, it is advantageous if the features are complementary in nature, i.e, they represent different aspects of voice characteristics of a speaker.

The method of modeling may depend on the description of features. Due to different representations, it may not be possible to model different features within a single framework. Hence, different models can be used for different features, and the resulting evidences can be combined. The effectiveness of combining the evidences due to different features for speaker verification depends on the following factors:

1. Effectiveness of the individual features for speaker verification

2. Complementary nature of the features

3. Method of combining the scores due to individual features

In the present study, we discuss the combination of evidences due to three different features extracted from the speech signal. These are:

1. Linear prediction cepstral coefficients (LPCC)

2. Difference cepstral coefficients
3. Excitation source features present in the linear prediction (LP) residual [36] [20]

The LPCCs obtained by the 14th order LP analysis represent the resonant frequencies of the vocal tract system and their bandwidths. The LPCCs contain information about the sound unit as well as the speaker. Difference cepstral coefficients are obtained by deemphasizing the gross spectral envelope from the fine spectrum, to suppress the sound unit information while preserving the finer variations of the short-time spectrum. The excitation source features are derived from the 12^{th} LP residual. These features represent the characteristics of the glottal vibrations, and are uncorrelated with the characteristics of the vocal tract system. Thus, the three features can be viewed to provide somewhat complementary information about the characteristics of the speaker. The development of AANN models for speaker verification based on LPCCs and difference cepstral coefficients was described in Sections 3.1.2 and 5.2, respectively. In Section 6.3.1, we briefly review the development of AANN models to represent the excitation source features present in the LP residual of speech signal. Combination of evidences for speaker verification due to the three features is described in Section 6.3.2.

6.3.1 Excitation Source Features for Speaker Verification

Linear prediction analysis of speech results in the LP coefficients which represent the vocal tract characteristics. The error signal obtained by inverse filtering the speech signal is termed as LP residual. LP residual contains excitation source information, which can be captured using a five-layer AANN model [20]. Consecutive blocks of samples of the LP residual are presented to an AANN model, and the blocks are separated by a shift of one sample. When raw data such as the samples of LP residual are presented to the AANN, the interpretation of the behaviour of AANN in terms of capturing the distribution of feature vectors is not appropriate. The reason is, though the adjacent frames may be widely separated in the input space, temporal relationship still exists among the adjacent frames since the samples of the LP residual are not entirely decorrelated. Thus, the objective of training the AANN model using the samples of LP residual is to acquire the higher order relations among the samples, that may contain useful speaker-specific characteristics. The effectiveness of the features of excitation source for speaker verification has been demonstrated in [20] [36]. In [20], significance of the regions of LP residual around the instants of glottal closure was also illustrated for speaker verification.

6.3.2 Approaches for Combining Evidences

An important issue in combining evidences from different classifiers is the nature of output associated with each classifier. The output of a classifier could be a class label, or a set of ranks corresponding to different labels, or a set of measurements to indicate the confidence of the classifier in a given class label. The strategy for combining the evidences depends on the representation of the outputs. If only the class labels or the label rankings are available, a majority vote is used [65] [66]. If continuous outputs like a posteriori probabilities are available, an average or linear combination or a Bayes classifier could be used [67] [63]. When the classifier outputs are available as fuzzy values or belief values, belief functions and Dempster-Shafer techniques are used [68] [69]. In [70], a theoretical framework was suggested for classifier combination. It was shown that the commonly used schemes of combination such as the product rule, sum rule, min rule, max rule and the majority voting are special cases which can be derived from the given framework under different assumptions and approximations. It was found that the sum rule outperformed other classification schemes, and was resilient to estimation errors, under certain assumptions. In our experiments on speaker verification, the sum rule is used for combining evidences.

The ability of difference cepstral coefficients to provide complementary evidence for speaker verification was illustrated in Section 5.2, in terms of the first ranks secured by the genuine speakers. Table 6.7 lists the performance of combination of evidences due to LPCCs and difference cepstral coefficients. A reduction in EER is achieved due to the combination.

Table 6.7: Combining evidences from LPCCs and difference cepstral coefficients.

	LPCCs	Difference cepstral coefficients	Combination by sum-rule
EER (%)	16.1	20.2	15.0

Table 6.8: Combining evidences from LPCCs, difference cepstral coefficients and excitation source features.

	LPCCs	Difference cepstral coefficients	Excitation source features	Combination by sum-rule
EER (%)	16.1	20.2	21.5	13.4

Table 6.8 lists the performance of combination of evidences due to LPCCs, difference cepstral coefficients and excitation source features. Although the error rates due to difference cepstral coefficients and source features are higher compared to that of LPCCs, the combination provides significant improvement. This is due to nature of

speaker-specific information represented by these two features, which is complementary to that of spectral features (LPCCs). The performance of speaker verification for individual features and the result of combination of evidences is indicated in the DET curve in Fig. 6.8.

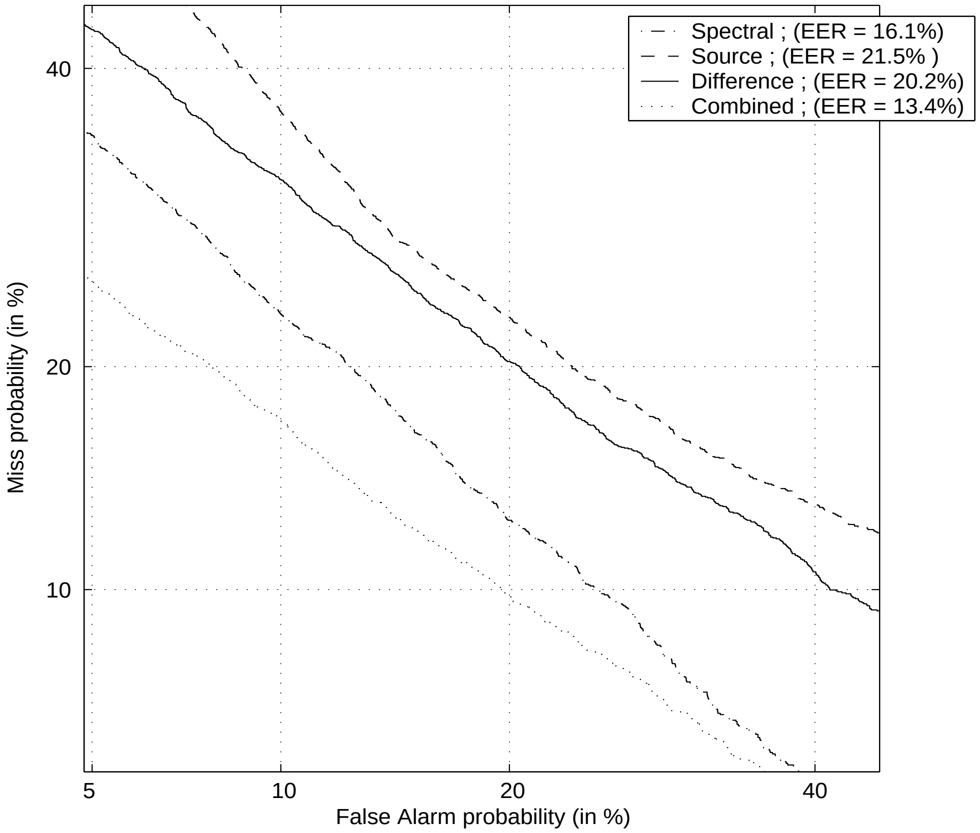


Fig. 6.8: DET curves indicating the performance of speaker verification based on LPCCs, difference cepstral coefficients, excitation source features and combination of evidences.

6.4 PERFORMANCE COMPARISON OF SPEAKER VERIFICATION SYSTEMS

In this section, the speaker verification system discussed in this thesis is compared with certain contemporary speaker verification systems, in terms of the performance achieved on a common dataset, namely, the NIST 2003 dataset. Table 6.9 lists a few systems, along with the features, models and normalization methods used for developing those systems.

Table 6.9: Comparison of performances of speaker verification systems.

System	Features	Channel compensation methods	Models	Normalization methods	EER (%)
IITM	LPCC, ESF, DCC	CMS	AANN	Z-norm, T-norm	13.4
MITLL	MFCC, DC	RASTA, FM	GMM-UBM, SVM	Z-norm, T-norm	6.5
DDRD	MFCC, DC	CMS	PCA, AANN, GMM-UBM	T-norm	8.0
IBM	LPC	-	GMM-UBM	Z-norm, T-norm	7.5
IRISA	LFCC	RASTA	GMM-UBM	Z-norm, Tnorm, D-norm	8.5

A glossary of the abbreviations used in the table is as follows:

LPCC	- Linear prediction cepstral coefficients
ESF	- Excitation source features
DCC	- Difference cepstral coefficients
DC	- Delta cepstrals
MFCC	- Mel frequency cepstral coefficients
LFCC	- Linear filter-bank cepstral coefficients
CMS	- Cepstral mean subtraction
RASTA	- Relative spectral
FM	- Feature mapping
AANN	- Autoassociative neural network
GMM	- Gaussian mixture model
UBM	- Universal background model
SVM	- Support vector machine
PCA	- Principal component analysis
D-norm	- Distance normalization

Details about these systems can be found in [45] and [46]. It is evident that most of these systems use spectral features, especially MFCC and DC, and are based on GMMs. In this sense, the system described in this thesis (IITM) attempts to explore novel features. The best performance of speaker verification obtained for NIST 2003 dataset, as reported in [46], is an EER value of 6.5 %. The main reasons for the better performance of these systems could be the following:

1. Some systems pool data from all types of channels to develop channel-dependent models. For the unknown utterance, the channel is detected and the features are mapped into a channel-independent space. This may reduce the mismatch

between the training and test patterns.

2. Some systems use an automatic speech recognizer to categorize speech into different sound units. Separate models are then developed for the different categories of sound units. Speaker-dependent language models are also developed using the output of the recognizer.
3. Modeling prosodic features such as intonation and duration has been shown to be effective for speaker verification.

6.5 SUMMARY

In this chapter, the issue of score normalization was discussed. Three methods of normalization of scores were proposed. In the model normalization method, model-specific statistics were estimated from example impostor utterances. However, model normalization suffers from the mismatch between the example utterances and the test utterance. Hence, methods of test normalization were investigated. A method based on the rank of claimant scores among the background models was proposed, to exclude non-competitive scores for normalization. Another method was proposed, based on framewise weighting of scores. Evidences due to complementary features were combined to improve the performance of speaker verification.

CHAPTER 7

SUMMARY AND CONCLUSIONS

The objective of automatic speaker verification is to validate a speaker's claim of identity based on the speaker's voice. Speaker verification consists of three steps, namely, feature extraction, modeling and score normalization. In this thesis, we have addressed issues related to feature extraction and score normalization. In feature extraction, significance of the position of analysis window was discussed for accurate estimation of short-time spectral characteristics. A feature for speaker verification was developed based on the difference between fine and gross spectra of speech. Autoassociative neural network models were used to estimate the probability density function of feature vectors in the feature space. Methods of model normalization and test normalization were proposed for calibrating the scores obtained from the models. Evidences were combined from three different features, which represent complementary information for speaker verification.

7.1 CONTRIBUTIONS OF THE WORK

1. Pitch synchronous analysis of speech was studied for accurate estimation of short-time spectral characteristics. Pitch synchronous LPCC features yielded a lower value of within-speaker to across-speaker dissimilarity, as compared to LPCCs obtained by block processing.
2. Difference cepstral coefficients were proposed as a feature for speaker verification. The ability of these features to add complementary evidence for speaker

verification was illustrated.

3. Methods for model normalization and test utterance normalization were proposed.
4. Evidences from three features were combined, namely, LPCC features, difference cepstral coefficients and excitation source features. The features are complementary sources of information and hence, their combination improves the performance of verification.

7.2 SCOPE FOR FUTURE WORK

1. Features that are robust to channel variations need to be extracted from speech signal. This can help reduce the mismatch between training and test utterances caused by channel effects.
2. Certain categories of sounds may be more important for speaker recognition than others. Thus, for each speaker, speech can be classified into a few broad categories of sound units. This may be done in an unsupervised manner rather than explicitly using a speech recognizer. A separate model can be developed for each category, and the evidences due to different models can be combined for speaker verification.
3. The temporal variation of feature vectors may contain useful speaker-specific information. Methods based on modeling the probability density function of feature vectors overlook this aspect. Hence, methods are needed to represent and model the temporal information.

APPENDIX A

BACKPROPAGATION ALGORITHM FOR FEEDFORWARD NEURAL NETWORKS

Multilayer feedforward neural networks are an important class of neutral networks. Typically, a multilayer feedforward neural network consists of a set of sensory units (source nodes) that form the input layer, computation nodes that form one or more hidden layers, and computation nodes that form the output layer. The input signal propagates through the network in a forward direction, on a layer-by-layer basis. The error between the desired pattern and the output pattern is used to update the weights of the network, using a method called backpropagation algorithm. The objective of this appendix is to discuss the backpropagation learning algorithm. A detailed discussion of multilayer feedforward neural networks can be found in [27] and [30].

In multilayer feedforward neural networks, each neuron is characterized by an activation function that could be a linear or a nonlinear function of the inputs to the neuron. Let v_j denote the induced local field (i.e., the weighted sum of all synaptic inputs plus the bias) of neuron j, and let y_j denote the output of the neuron. Then, an example of nonlinear activation function is the sigmoidal nonlinearity defined by the logistic function:

$$y_j = \frac{1}{1 + exp(-v_j)}.$$

The necessary condition here is that the nonlinearity should be smooth, i.e., differentiable everywhere. In the present work, the following nonlinearity has been used:

$$y_j = tanh(\lambda v_j),$$

where $\lambda = 0.66$ has been chosen, based on experiments.

The neurons of the hidden layers are not part of the input or the output layer. However, the hidden neurons enable the network to learn complex tasks by extracting progressively meaningful features from the input patterns. Also, it is important to distinguish between function signals and error signals. A function signal is an input signal that comes in at the input end of the network, propagates forward through the hidden layers of the network, and emerges at the output end of the network as an output signal. An error signal originates at an output neuron of the network and propagates backward, layer by layer, through the network.

In the remaining part of the appendix, derivation of the backpropagation algorithm is presented. In Section A.1, a summary of the notations used in the derivation is presented. Section A.2 discusses the derivation of the algorithm.

A.1 NOTATION

- The indices i, j and k refer to different neurons in the network. The signals propagate through the network from left to right, neuron j lies in a layer to the right of neuron i, and neuron k lies in a layer to the right of neuron j when neuron j is a hidden unit.
- In iteration n, the n^{th} training pattern is presented to the network.
- $E(n)$ refers to the instantaneous sum of error squares at iteration n. The average of $E(n)$ over all values of n is denoted by the average energy E_{av}.
- $e_j(n)$ refers to the error signal at the output of neuron j of iteration n.
- $d_j(n)$ refers to the desired response for neuron j and is used to compute $e_j(n)$.

- $y_j(n)$ denotes the function signal appearing at the output of neuron j at iteration n.
- $w_{ji}(n)$ denotes the synaptic weight connecting the output of neuron i to the input of neuron j at iteration n. The correction applied to this weight at iteration n is denoted by $\Delta w_{ji}(n)$.
- The induced local field of neuron j at iteration n is denoted by $v_j(n)$. It is the signal applied to the activation function associated with neuron j.
- The activation function describing the input-output functional relationship of the nonlinearity associated with neuron j is denoted by $\phi_j(.)$.
- The bias applied to neuron j is denoted by b_j. Its effect is represented by a synapse of weight $w_{j0} = b_j$ connected to a fixed input equal to $+1$.
- The i^{th} element of the input vector (pattern) is denoted by $x_i(n)$.
- The k^{th} element of the overall output vector (pattern) is denoted by $o_k(n)$.
- The learning-rate parameter is denoted by η.
- m_l denotes the number of nodes (size) in layer l of the network where, $l = 0, 1, ..., L$ and L denotes the depth (number of layers) of the network.

A.2 BACKPROPAGATION ALGORITHM

The error signal at the output of neuron j at iteration n is defined by

$$e_j(n) = d_j(n) - y_j(n), \tag{A.1}$$

where neuron j is an output node. The instantaneous sum of error squares over all neurons in the output layer is given by

$$E(n) = \frac{1}{2} \sum_{j \epsilon C} {e_j}^2(n), \tag{A.2}$$

where the set C includes all the neurons in the output layer of the network. The average squared error energy is obtained as

$$E_{av}(n) = \frac{1}{N}\sum_{n=1}^{N} E(n), \tag{A.3}$$

where N denotes the total number of patterns contained in the training set. For a given training set, E_{av} represents a cost function. The objective of the learning process is to adjust the free parameters of the network to minimize E_{av}.

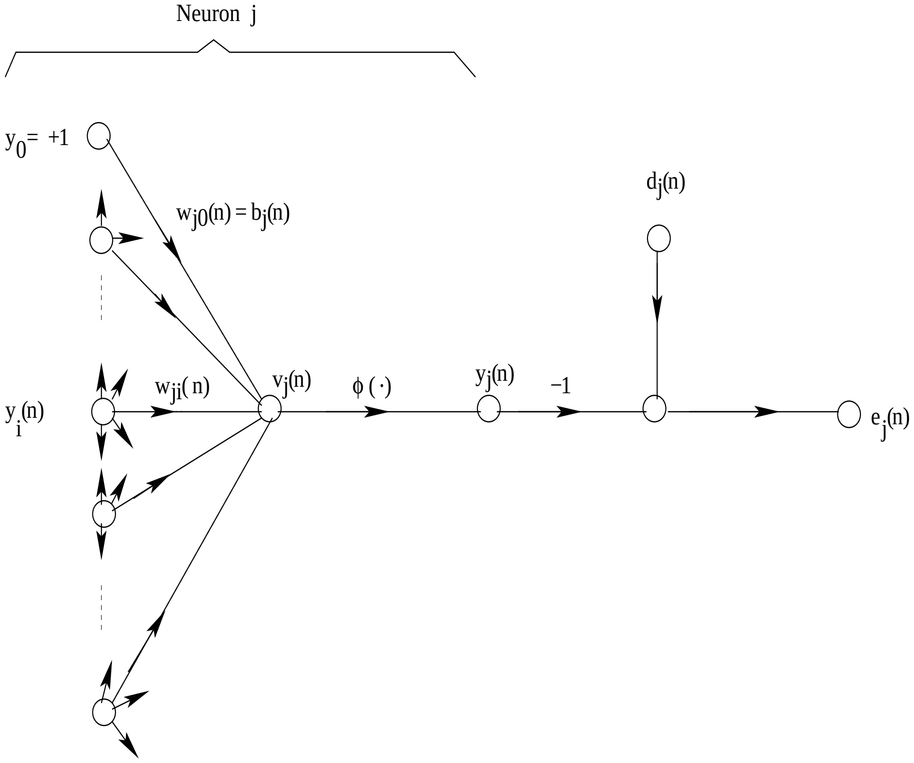


Fig. A.1: Signal-flow graph highlighting the details of output neuron j.

Figure A.1 depicts neuron j being fed by a set of function signals produced by a layer of neurons to its left. The induced local field $v_j(n)$ produced at the input of the activation function associated with neuron j is given by

$$v_j(n) = \sum_{i=0}^{m} w_{ji}(n) y_i(n), \tag{A.4}$$

where m is the total number of inputs applied to neuron j, excluding the bias. Thus, the function signal $y_j(n)$ appearing at the output of neuron j at iteration n is given by

$$y_j(n) = \phi_j(v_j(n)). \tag{A.5}$$

The gradient or the partial derivative $\frac{\partial E(n)}{\partial w_{ji}(n)}$ can be expressed, using the chain rule of calculus, as follows:

$$\frac{\partial E(n)}{\partial w_{ji}(n)} = \frac{\partial E(n)}{\partial e_j(n)} \frac{\partial e_j(n)}{\partial y_j(n)} \frac{\partial y_j(n)}{\partial v_j(n)} \frac{\partial v_j(n)}{\partial w_{ji}(n)}. \tag{A.6}$$

The various partial derivatives in the above equation are obtained as follows:

$$\frac{\partial E(n)}{\partial e_j(n)} = e_j(n). \tag{A.7}$$

$$\frac{\partial e_j(n)}{\partial y_j(n)} = -1. \tag{A.8}$$

$$\frac{\partial y_j(n)}{\partial v_j(n)} = \phi_j^{'}(v_j(n)). \tag{A.9}$$

$$\frac{\partial v_j(n)}{\partial w_{ji}(n)} = y_i(n). \tag{A.10}$$

Substituting for the various partial derivatives in the expression for $\frac{\partial E(n)}{\partial w_{ji}(n)}$, we obtain

$$\frac{\partial E(n)}{\partial w_{ji}(n)} = -e_j(n)\phi_j^{'}(v_j(n))y_i(n). \tag{A.11}$$

The correction $\Delta w_{ji}(n)$ applied to $w_{ji}(n)$ is defined by the delta rule as follows:

$$\Delta w_{ji}(n) = -\eta \frac{\partial E(n)}{\partial w_{ji}(n)}, \tag{A.12}$$

where η is the learning-rate parameter of the backpropagation algorithm, and the use of minus sign accounts for gradient descent in weight space. Thus,

$$\Delta w_{ji}(n) = \eta \delta_j(n) y_i(n), \tag{A.13}$$

where the local gradient $\delta_j(n)$ is defined by

$$\begin{aligned} \delta_j(n) &= -\frac{\partial E(n)}{\partial v_j(n)} \\ &= -\frac{\partial E(n)}{\partial e_j(n)} \frac{\partial e_j(n)}{\partial y_j(n)} \frac{\partial y_j(n)}{\partial v_j(n)} \\ &= -e_j(n) \phi_j^{'}(v_j(n)). \end{aligned} \tag{A.14}$$

It is seen that the local gradient is dependent on the corresponding error signal. For the nodes of the output layer, the computation of the error signal is straightforward, since the desired response is known. The situation for nodes of hidden layers is shown in Figure A.2, which depicts a neuron j as a hidden node of the network.

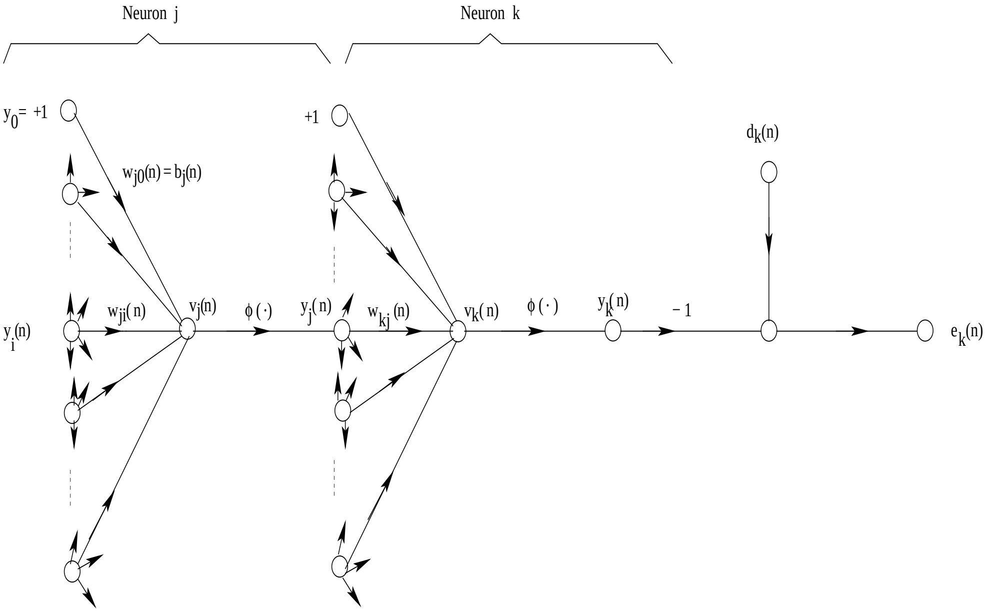


Fig. A.2: Signal-flow graph highlighting the details of output neuron k connected to hidden neuron j.

The local gradient $\delta_j(n)$ for hidden neuron j is redefined as

$$\begin{aligned}\delta_j(n) &= -\frac{\partial E(n)}{\partial y_j(n)}\frac{\partial y_j(n)}{\partial v_j(n)} \\ &= -\frac{\partial E(n)}{\partial y_j(n)}\phi_j^{'}(v_j(n)). \qquad \text{(A.15)}\end{aligned}$$

The instantaneous sum of error squares $E(n)$ is given by

$$E(n) = \frac{1}{2}\sum_{k\epsilon C} e_k^2(n), \qquad \text{(A.16)}$$

where k denotes a neuron in the output node. Now, differentiating the above equation with respect to the function signal $y_j(n)$, we get

$$\frac{\partial E(n)}{\partial y_j(n)} = \sum_k e_k(n)\frac{\partial e_k(n)}{\partial y_j(n)}. \qquad \text{(A.17)}$$

Using the chain rule for the partial derivative $\frac{\partial e_k(n)}{\partial y_j(n)}$, the above equation can be rewritten as

$$\frac{\partial E(n)}{\partial y_j(n)} = \sum_k e_k(n)\frac{\partial e_k(n)}{\partial v_k(n)}\frac{\partial v_k(n)}{\partial y_j(n)}. \qquad \text{(A.18)}$$

Also,

$$\begin{aligned}e_k(n) &= d_k(n) - y_k(n) \\ &= d_k(n) - \phi_k(v_k(n)), \qquad \text{(A.19)}\end{aligned}$$

where neuron k is an output node.

Hence

$$\frac{\partial e_k(n)}{\partial v_k(n)} = -\phi_k^{'}(v_k(n)). \qquad \text{(A.20)}$$

The induced local field for neuron k is given by

$$v_k(n) = \sum_{j=0}^{m} w_{kj}(n)y_j(n), \qquad \text{(A.21)}$$

where m is the total number of inputs (excluding the bias) applied to neuron k.

Differentiating the above equation with respect to $y_j(n)$ yields

$$\frac{\partial v_k(n)}{\partial y_j(n)} = w_{kj}(n). \tag{A.22}$$

Thus, the desired partial derivative of $E(n)$ is obtained as

$$\begin{aligned} \frac{\partial E(n)}{\partial y_j(n)} &= -\sum_k e_k(n)\phi_k^{'}(v_k(n))w_{kj}(n) \\ &= -\sum_k \delta_k(n)w_{kj}(n), \end{aligned} \tag{A.23}$$

where the definition of the local gradient has been used for the nodes of the output layer.

Finally, the backpropagation formula for the local gradient $\delta_j(n)$ is given by

$$\delta_j(n) = \phi_j^{'}(v_j(n)) \sum_k \delta_k(n)w_{kj}(n), \tag{A.24}$$

where neuron j is hidden.

Thus, the local gradients are computed backward, starting from the hidden layer preceding the output layer.

The correction $\Delta w_{ji}(n)$ applied to the weight connecting neuron i to neuron j is defined by the delta rule as follows:

$$\Delta w_{ji}(n) = \eta\delta_j(n)y_i(n). \tag{A.25}$$

To summarize the backpropagation algorithm:

1. If neuron j is an output node, $\delta_j(n)$ equals the product of the derivative $\phi_j^{'}(v_j(n))$ and the error signal $e_j(n)$, both of which are associated with neuron j.

2. If neuron j is a hidden node, $\delta_j(n)$ equals the product of the associated derivative $\phi_j^{'}(v_j(n))$ and the weighted sum of the δs computed for the neurons in the next hidden or output layer that are connected to neuron j.

BIBLIOGRAPHY

[1] D. Lancker, J. Kreiman, and K.Emmorey, "Familiar voice recognition: Patterns and parameters - recognition of backward voices," *Phonetics*, vol. 13, no. 1, pp. 19–38, 1985.

[2] M. Sigmund, "Speaker recognition - Identifying people by their voices," Master's thesis, Institute of radio electronics, Brno University of Technology, Czech Republic, 2000.

[3] J. J. Wolf, "Efficient acoustic parameters for speaker recognition," *Journal of the Acoustical Society of America*, vol. 52, no. 6, pp. 2044–2056, 1972.

[4] B. Yegnanarayana, "Formant extraction from linear prediction phase spectra," *Journal of the Acoustical Society of America*, vol. 63, pp. 1638–1640, May 1978.

[5] H. A. Murthy and B. Yegnanarayana, "Formant extraction from group delay function," *Speech Communication*, vol. 10, pp. 209–221, Aug. 1991.

[6] J. Makhoul, "Linear prediction: A tutorial review," *Proceedings of the IEEE*, vol. 63, pp. 561–580, Apr. 1975.

[7] B. S. Atal, "Effectiveness of linear prediction characteristics of the speech wave for automatic speaker identification and verification," *Journal of the Acoustical Society of America*, vol. 55, pp. 1304–1312, June 1974.

[8] J. Markel, B. T. Oshika, and A. H. Gray, "Long-term feature averaging for speaker recognition," *IEEE Transactions on Acoustics Speech and Signal Processing*, vol. 25, pp. 330–337, Aug. 1977.

[9] K. T. Assaleh and R. J. Mammone, "New LP-derived features for speaker identification," *IEEE Transactions on Speech and Audio Processing*, vol. 2, pp. 630–638, Oct. 1994.

[10] M. R. Sambur, "Speaker recognition using orthogonal linear prediction," *IEEE Transactions on Acoustics Speech and Signal Processing*, vol. 24, pp. 283–289, Aug. 1976.

[11] J. Naik and G. R. Doddington, "High performance speaker verification using principal spectral components," in *Proceedings of the International Conference on Acoustics Speech and Signal Processing*, (Tokyo), pp. 881–884, 1986.

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

[13] F. K. Soong and A. E. Rosenberg, "On the use of instantaneous and transitional spectral information in speaker recognition," in *Proceedings of the International Conference on Acoustics Speech and Signal Processing*, (Tokyo), pp. 877–880, 1986.

[14] D. A. Reynolds and R. C. Rose, "Robust text-independent speaker identification using Gaussian mixture speaker models," *IEEE Transactions on Speech and Audio Processing*, vol. 3, pp. 72–83, Jan. 1995.

[15] L. R. Rabiner and B. H. Juang, *Fundamentals of Speech Recognition.* Prentice-Hall, Englewood Cliffs and N.J., 1993.

[16] P. Thevanaz and H. Hugli, "Usefullness of lpc-residue in text-dependent speaker verification," *Speech Communication*, vol. 17, pp. 145–157, 1995.

[17] M. Faundez-zanuy and D. Rodriguez-Porcheron, "Speaker recognition using residual signal of linear and nonlinear prediction models," in *Proceedings of the International Conference on Spoken Language Processing*, vol. 2, (Australia), pp. 121–124, 1998.

[18] G. Fant, "Glottal flow: models and interaction," *Journal of Phonetics*, vol. 14, pp. 393–399, 1986.

[19] M. D. Plumpe, T. F. Quatieri, and D. A. Reynolds, "Modeling of the glottal flow derivative waveform with application to speaker identification," *IEEE Transactions on Speech and Audio Processing*, vol. 7, pp. 569–586, Sep. 1999.

[20] C. S. Gupta, "Significance of source features for speaker recognition," MS thesis, Dept. of Computer Science and Engg., IIT Madras, Chennai, Apr. 2003.

[21] W. Hess, *Algorithms and Devices.* Berlin, Germany: Springer-Verlag, 1983.

[22] B. S. Atal, "Automatic speaker recognition based on pitch contours," *Journal of the Acoustical Society of America*, vol. 52, no. 6, pp. 1687–1697, 1972.

[23] K. Sonmez, L. Heck, M. Weintraub, and E. Shriberg, "A lognormal tied mixture model of pitch for prosody-cased speaker recognition," in *Proceedings of the European Conference on Speech Processing and Technology*, vol. 3, (Rhodes, Greece), pp. 1391–1394, 1997.

[24] J. M. Zachariah, "Text-dependent speaker verification using segmental, suprasegmental and source features," MS thesis, Dept. of Computer Science and Engg., IIT Madras, Chennai, Mar. 2002.

[25] C. d'Alessadro, V. Darsinos, and B. Yegnanarayana, "Effectiveness of a periodic and aperiodic decomposition method for analysis of voice sources," vol. 6, pp. 12–23, Jan. 1998.

[26] A. Higgins, L. Bahler, and J. Porter, "Voice identification using nearest neighbour distance measure," in *Proceedings of the International Conference on Acoustics Speech and Signal Processing*, (Minneapolis, MN, USA), pp. 375–378, 1985.

[27] Simon Haykin, *Neural networks: A Comprehensive Foundation.* New Jersey: Prentice-Hall International, 1999.

[28] F. K. Soong, A. E. Rosenberg, L. R. Rabiner, and B. H. Juang, "A vector quantization approach in speaker recognition," in *Proceedings of the International Conference on Acoustics Speech and Signal Processing*, (Tampa, FL, USA), pp. 387–390, Mar. 1985.

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

[30] B.Yegnanarayana, *Artificial Neural Networks.* New Delhi: Prentice-Hall of India, 1999.

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

[32] H. Hermansky and M. Narendranath, "Speaker verification using speaker-specific mapping," in *RLA2C*, (Avignon, France), Apr. 1998.

[33] H. Misra, M. S. Ikbal, and B. Yegnanarayana, "Speaker-specific mapping for text-independent speaker recognition," *Speech Communication*, vol. 39, pp. 301–310, Feb. 2003.

[34] H. Misra, "Development of mapping feature for speaker recognition," MS thesis, Dept. of Electrical Engg., IIT Madras, Chennai, May 1999.

[35] B. Yegnanarayana and S. P. Kishore, "AANN: an alternative to GMM for pattern recognition," *Neural Networks*, vol. 15, pp. 459–469, May 2002.

[36] K. S. Reddy, "Source and system features for speaker recognition," MS thesis, Dept. of Computer Science and Engg., IIT Madras, Chennai, Sep. 2001.

[37] Y. Gong and J. P. Haton, "Non-linear vectorial interpolation for speaker recognition," in *Proceedings of the International Conference on Acoustics Speech and Signal Processing*, vol. 2, (San Francisco, CA, USA), pp. II173–II176, 1992.

[38] G. Doddington, "Sheep, goats, lambs and wolves: A statistical analysis of speaker performance in the nist 1998 speaker recognition evaluation," in *Proceedings of the International Conference on Spoken Language Processing*, (Australia), 1998.

[39] M. Carey, E. Parris, and S. Bennett, "Speaker verification," in *Proceedings of the Institute of Acoustics (Speech & Hearing)*, (Windermere, UK), pp. 99–106, 1997.

[40] A. Rosenberg, J. Delong, C. Lee, B. Juang, and F. Soong, "The use of cohort normalized scores for speaker recognition," in *Proceedings of the International Conference on Spoken Language Processing*, (Banf, Alberta, Canada), pp. 599–602, 12-16 Oct. 1992.

[41] K. Fukunaga, *Introduction to Statistical Pattern Recognition.* New York, USA: Academic Press, 1972.

[42] A. Ariyaeeinia and P. Sivakumaran, “Analysis and comparison of score normalization methods for text-dependent speaker verification,” in *Proceedings of the European Conference on Speech Processing and Technology*, (Rhodes, Greece), pp. 1379–1382, 1997.

[43] R. Auckenthaler, M. Carey, and H. L. Thomas, “Score normalization for text-independent speaker verification systems,” *Digital signal processing*, no. 10, pp. 42–54, 2000.

[44] D. Reynolds, “Comparison of background normalization methods for text-independent speaker verification,” in *Proceedings of the European Conference on Speech Processing and Technology*, (Rhodes, Greece), pp. 963–966, 1997.

[45] “Speaker recognition evaluation workshop notebook,” in *Speaker Recognition Workshop*, (Vienna, Virginia, USA), 20-22, May 2002.

[46] “Speaker recognition evaluation workshop notebook,” in *Speaker Recognition Workshop*, (College Park, Maryland USA), 24-25 June 2003.

[47] A. Martin, G. Doddington, T.Kamm, M. Ordowski, and M. Przybocki, “The DET curve in assessment of detection task performance,” in *Proceedings of the European Conference on Speech Processing and Technology*, vol. 4, (Rhodes, Greece), pp. 1895–1898, 1997.

[48] B. Yegnanarayana and R. N. J. Veldhuis, “Extraction of vocal-tract system characteristics from speech signals,” *IEEE Transactions on Speech and Audio Processing*, vol. 6, pp. 313–327, July 1998.

[49] J.R.Deller, Jr., J.G.Proakis and J.H.L.Hansen, *Discrete-Time Processing of Speech Signals.* New York: Macmillan, 1993.

[50] B. Yegnanarayana and R. L. H. M. Smits, “A robust method for determining instants of major excitations in voiced speech,” in *Proceedings of the International Conference on Acoustics Speech and Signal Processing*, (Detroit, USA), pp. 776–779, May 8-12 1995.

[51] P. S. Murthy and B. Yegnanarayana, “Robustness of group-delay-based method for extraction of signification excitation from speech signals,” *IEEE Transactions on Speech and Audio Processing*, vol. 7, pp. 609–619, Nov. 1999.

[52] R. Smits and B. Yegnanarayana, “Determination of instants of significant excitation in speech using group delay function,” *IEEE Transactions on Speech and Audio Processing*, vol. 3, pp. 325–333, Sep. 1995.

[53] B. Yegnanarayana and R. Teunen, “Prosodic manipulation of speech using knowledge of instants of significant excitation,” Tech. Rep. Report No.1029, Institute of Perception Research, IPO, Eindhoven, The Netherlands, 1994.

[54] B. Yegnanarayana, S. R. M. Prasanna, and K. S. Rao, “Speech enhancement using excitation source information,” in *Proceedings of the International Conference on Acoustics Speech and Signal Processing*, vol. 1, (Orlando, FL, USA), pp. 541–544, May 2002.

[55] Dhananjaya. N, “Speaker segmentation using excitation source features,” MS thesis, Dept. of Computer Science and Engg., IIT Madras, Chennai, June 2004. Submitted.

[56] B. S. Atal, “Automatic recognition of speakers from their voices,” *Proceedings of the IEEE*, vol. 64, pp. 460–475, Apr. 1976.

[57] G. E. Peterson and H. L. Barney, “Control methods used in a study of vowels,” *Journal of the Acoustical Society of America*, vol. 24, pp. 175–194, Mar. 1952.

[58] S. P. Kishore, “Speaker Verification Using Autoassociative Neural Network Models,” MS thesis, Dept. of Computer Science and Engg., IIT Madras, Chennai, Dec. 2000.

[59] H. Gish and M. Schmidt, “Text-independent speaker identification,” *IEEE Signal Processing Magazine*, vol. 11, pp. 18–32, Oct. 1994.

[60] S. S. Kajarekar, A. G. Adami, and H. Hermansky, “Novel approaches for one and two-speaker detection,” in *Proceedings of the European Conference on Speech Processing and Technology*, (Geneva, Switzerland), pp. 2661–2664, 2003.

[61] K. Chen, L. Wang, and H. Chi, “Methods of combining multiple classifiers with different features and their application to text-independent speaker identification,” *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 11, no. 3, pp. 417–445, 1997.

[62] D. Genoud, G. Gravier, F. Bimbot, and G. Chollet, “Combining methods to improve speaker verification decision,” Tech. Rep. IDIAP-RR-96-02, IDIAP, Martigny, Switzerland, 1996.

[63] L. Xu, A. Krzyzak, and C. Y. Suen, “Methods for combining multiple classifiers and their spplications to handwriting recognition,” *IEEE Transactions on System and Man and Cybernetics*, vol. 22, no. 3, pp. 418–435, 1992.

[64] M. Mathew, “Combining evidence from different classifiers for text-dependent speaker verification,” MS thesis, Dept. of Computer Science and Engg., IIT Madras, Chennai, June 1999.

[65] F. Kimura and M. Shridhar, “Handwritten numerical recognition based on multiple algorithms,” *Pattern Recognition*, vol. 24, no. 10, pp. 969–983, 1991.

[66] J. Franke and E. Mandler, “A comparison of two approaches for combining the votes of cooperating classifiers,” in *Proceedings of the* 11^th *IAPR International Conference on Pattern Recognition*, vol. 2, pp. 611–614, 1992.

[67] Hashem and B. Schmeiser, “Improving model accuracy using optimal linear combinations of trained neural networks,” *IEEE Transactions on Neural Networks*, vol. 12, no. 3, pp. 792–794, 1995.

[68] G.Shafer, *A mathematical theory of evidence.* New Jersey, USA: Princeton University Press, 1976.

[69] A. P. Dempster, “Upper and lower probabilities induced by a multivalued mapping,” *Annals of Mathematical Statistics*, vol. 38, pp. 325–339, 1967.

[70] J. Kittler, M. Hatef, R. P. W. Duin, and J. Matas, “On combining classifiers,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, pp. 226–239, Mar. 1998.

LIST OF PUBLICATIONS

Presentations in Conferences :

1. Guruprasad. S, Dhananjaya. N and B. Yegnanarayana, "AANN Models for Speaker Recognition Based on Difference Cepstrals," in *Proc. Int. Joint Conf. Neural Networks*, (Portland, OR, USA), pp. 692-697, July 2003.

2. Dhananjaya. N, Guruprasad. S, and B. Yegnanarayana, "Speaker Segmentation Based on Subsegmental Features and Neural Network Models," Accepted for *Int. Conf. Neural Information Processing*, to be held in Science City, Calcutta, during Nov. 22-25, 2004.