LIMITED DATA SPEAKER RECOGNITION

A

Thesis submitted

for the award of the degree of

DOCTOR OF PHILOSOPHY

By

H. S. JAYANNA

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

INDIAN INSTITUTE OF TECHNOLOGY GUWAHATI

GUWAHATI - 781 039, ASSAM, INDIA

March 2009

To my guide **Dr. S. R. Mahadeva Prasanna**

for his help, guidance, inspiration and encouragement

and

To My Parents

for their love and support

Certificate

This is to certify that the thesis entitled "**LIMITED DATA SPEAKER RECOGNITION**", submitted by **H. S. Jayanna** (05610204), a research scholar in the *Department of Electronics and Communication Engineering, Indian Institute of Technology Guwahati*, for the award of the degree of **Doctor of Philosophy**, is a record of an original research work carried out by him under my supervision and guidance. The thesis has fulfilled all requirements as per the regulations of the institute and in my opinion has reached the standard needed for submission. The results embodied in this thesis have not been submitted to any other University or Institute for the award of any degree or diploma.

Dated:

Guwahati.

Dr. S. R. Mahadeva Prasanna
Associate Professor
Dept. of Electronics and Communication Engg.
Indian Institute of Technology Guwahati
Guwahati - 781 039, Assam, India.

Acknowledgements

At the outset, I would like to express my whole hearted and deep sense of gratitude to my guide Dr. S. R. Mahadeva Prasanna for his guidance, help and encouragement throughout my research work. I have been very fortunate to get Dr. S. R. Mahadeva Prasanna as a guide who has extraordinary patience, great enthusiasm and positive thinking. I greatly admire his attitude towards research, creative thinking, hard work, dedication and punctuality in work. He has been a great source of inspiration for me in all my endeavours. I am highly grateful to him for patiently checking all my manuscripts and thesis. This thesis would not have been possible without his bounteous effort. More than a guide, he is my mentor for shaping my personal and professional life, without whom I would not have been where I am today. As a friend and guide his help is immeasurable. I owe my profound gratitude to Dr. S. R. Mahadeva Prasanna for his support in all respects.

My sincere thanks are due to Prof. S. Dandapat for his moral support, encouragement and suggestions rendered during my research work. I am also very thankful to my doctoral committee members Prof. P. K. Bora, Dr. Rohit Sinha and Dr. P. K. Das for their useful advice and for sparing their valuable time to evaluate the progress of my work. I specially thank Dr. Rohit Sinha for sparing his precious time for discussing my research work and giving valuable suggestions.

My sincere thanks to Prof. B. Yegnanarayana, Dr. K. Sreenivasa Rao and Dr. Bhanu Prasad for their kind help on many occasions. I specially thank Dr. Suryakanth V Gangashetty for his help and precious time spent while teaching me support vector machines online.

I thank the Head of the Department, Prof. S. Majhi, and other faculty members for their help and support in carrying out my work. I specially thank Prof. Anil Mahanta for his advice and suggestions. My thanks go to Mr. L. N. Sharma sir for providing an excellent facility and maintaining a good ambience in the Electro Medical and Speech Technology lab to carry out my work. His friendly and helpful nature has made my work easy. I sincerely thank Mr. Sanjib Das sir for his timely help.

I am highly grateful to Siddaganga Institute of Technology (SIT), Tumkur, Karnataka, India for deputing me to study at Indian Institute of Technology Guwahati (IITG), a prestigious institute in India and providing me with financial assistance.

I owe my invaluable thanks to Dr. M. N. Channabasappa, Director, SIT, Tumkur. His inspiration and motivation have helped me to complete the research work on time. I sincerely thank Dr. S. M. Shashidhar, Principal, SIT, Tumkur for his help and support.

I sincerely thank Mr. P. Krishnamoorthy who helped me since the beginning of my research work in all respects. Particularly he has helped me in developing some programmes. I thank Mr. D. Senthil Kumar for his help during my course work. My thanks go to Mr. Kali Charan Gajula and Mr. Kailsh B. Patil whose programs I used in my work.

I sincerely thank Mrs. Nirmala madam for her moral support, help and care rendered to me and my wife. My sincere thanks go to Prof. Jois and Mrs. Ambuja Jois for their parental treatment, moral support and help to me and my wife. I would like to thank Prof. Rao for his help and encouragement.

I thank Mr. M. Sabarimali Manikandan, Mr. K. Narasimha Murthy and his family, Mr. Debadatta Pathi, Ms. Sumitra Sukla, Mr. D. Govind, Ms. Shweta Ghai, Ms. Arphana and Mr. Y. Sunil for their help and support.

Stay at IITG would have been difficult without my wife K. S. Rajeshwari. I thank her for unlimited sacrifices, support and encouragement. I thank my teachers, parents, brothers, sisters, brother-in-laws, nephews for their love, affection and support in my life and making me what I am today.

H. S. Jayanna

Abstract

This work demonstrates some approaches for improving the speaker recognition performance under limited data condition. The performance of the speaker recognition system depends on the techniques employed in the analysis, feature extraction, modelling and testing stages. Existing limited data speaker recognition techniques mostly concentrate on modelling techniques to improve the performance. It is also possible to improve the performance using efficient techniques for speech analysis, feature extraction, modelling and testing. We have developed techniques suitable for each stage of the speaker recognition system to improve the performance. In the analysis stage, speech signal is analyzed using various analysis techniques and a method is proposed. In the feature extraction stage, the different feature extraction techniques are used to extract the features and are finally combined. In the modelling stage, first the different modelling techniques are evaluated and then a subset of them is selected based on their performance and finally combined. We then build integrated systems using proposed analysis technique, different features and modelling techniques. The different combination schemes are finally proposed to combine the evidences from the integrated systems. As a result, we propose a scheme for speaker recognition under limited data condition. The proposed system provides significant improvement in performance.

The major contributions of the work reported in this thesis for speaker recognition under limited data condition includes,

1. Multiple Frame Size and Rate (MFSR) analysis of speech.
2. Combination of features.
3. Combined modelling techniques.
4. Integrated systems, and
5. Combining evidences.

The other contributions are,

1. The use of Vector-Pulse Code Modulation (V-PCM) for speaker modelling from the Linear Prediction Residual (LPR) and Linear Prediction Residual Phase (LPRP) features.
2. Combination of LPR and LPRP information for improving the performance.
3. The combination techniques such as Strength Voting (SV), weighted Ranking (WR), Support Systems (SS) and Hierarchical Combination (HC).

Keywords: Speaker recognition, limited data, speech analysis, frame size, frame shift, combination of features, combined modelling, combination schemes, integrated systems.

Contents

List of Figures

List of Tables

List of Acronyms

AANN	Auto Associative Neural Networks
BC	Borda Count
CVQ	Crisp Vector Quantization
DCT	Discrete Cosine Transform
DFT	Discrete Fourier Transform
DTW	Dynamic Time Warping
DTM	Direct Template Matching
EM	Expectation Maximization
FVQ	Fuzzy Vector Quantization
FFT	Fast Fourier Transform
GC	Glottal Closure
GMM	Gaussian Mixture Models
GMM-UBM-IE	GMM-UBM - Including Evaluation set
GMM-UBM-NIE	GMM-UBM - Not Including Evaluation set
HE	Hilbert Envelope
HMM	Hidden Markov Models
HC	Hierarchical Combination
IE	Including Evaluation set
KEMLLR	Kernel Eigenspace-based Maximum Likelihood Linear Regression (KEMLLR)
LVQ	Learning Vector Quantization
LLR	Log-Likelihood Ratio
LP	Linear Prediction

LPR	Linear Prediction Residual
LPRP	Linear Prediction Residual Phase
LPC	Linear Predictive Coefficients
LPCC	Linear Predictive Cepstral Coefficients
LCFR	Linear Combination of Frame Ratio
MFS	Multiple Frame Size
MFR	Multiple Frame Rate
MFSR	Multiple Frame Size and Rate
MAP	Maximum *a Posteriori* Adaptation
MFCC	Mel-Frequency Cepstral Coefficients
ML	Maximum Likelihood
MLLR	Maximum Likelihood Linear Regression
NIE	Not Including Evaluation set
PDF	Probability Density Function
SFSR	Single Frame Size and Rate
SFS	Single Frame Size
SOM	Self-Organizing Map
SVM	Support Vector Machines
SV	Strength Voting
SS	Supporting Systems
UBM	Universal Background Model
VQ	Vector Quantization
VAD	Voice Activity Detector
V-PCM	Vector-Pulse Code Modulation
WR	Weighted Ranking
WLCFR	Weighted LCFR

List of Symbols

a_k	Linear predictive coefficients
α	Adaptation coefficients
$c(n)$	Cepstral coefficients
C	Number of Mel cepstral coefficients
D	Down sampling factor
Δ	Delta coefficients
$\Delta\Delta$	Delta-Delta coefficients
F	Real frequency
F_{mel}	Perceived frequency
F_s	Sampling frequency
γ	Scale factor in GMM-UBM
H	Frame size or hop
$h(n)$	Hilbert transform of the LP residual
$h(e)$	Hilbert Envelop of LP residual
h	neighborhood parameter
i	Gaussian mixture number
η	Learning rate parameter
λ	Specifies speaker in GMM technique
μ_i	Mean vector
P_i	Performance of the system i
$P(X\|\lambda)$	GMM likelihood
$r(n)$	LP residual

r	Fixed relevance factor for MAP adaptation
R_i	Rank of a speaker in system i
S	Frame size
ε	Splitting parameter
σ	Standard deviation
Σ_i	Covariance matrix
W_j	Weight vector for the class j at t
$W_c(t+1)$	Class associated with the the weight vector W_c at time *t+1*
w_i	Weights of Gaussian mixture
X	Feature vectors
ζ_{w_c}	Class associated with the the weight vector
ζ_X	Class label of the input vector

1

Introduction

Contents

1.1 Objective of the Thesis

State-of-the-art speaker recognition systems assume the availability of sufficient data for training and testing. In the present work, *sufficient data* denotes the case of speech data of few minutes (more than or equal to one minute). Sufficient data may give enough information for recognizing a speaker. As a result, existing speech analysis, feature extraction, modelling and testing techniques for speaker recognition work well under this condition. *Speaker recognition under limited data condition* is defined as the task of recognizing speakers with the constraint that both training and testing data are limited. In this work, *limited data* denotes the case of speech data of few seconds (less than or equal to 15 seconds). The main problem under limited data is the insufficient speaker information for training and testing and hence the use of existing techniques provides poor performance. The objective of the thesis is therefore to develop methods that provide good performance under limited data condition. As a result, this thesis proposes some approaches for speech analysis, feature extraction, modelling and testing for improving the performance.

1.2 Importance of Speaker Recognition

Recently, to enhance the security system, biometrics are most widely used for person recognition that include the properties like physiological, behavioral and combination of both [1]. The biometrics based on physiological properties include finger print, iris, retina, hand geometry, DNA, speech, gait, etc. These offer better security than the behavioral features. This is because unlike behavioral features they do not tender to the voluntary variations. The biometrics based on behavioral properties are speech, signature, gait, etc. Speech is an example of a biometric that combines both physiological and behavioral characteristics. Among the different biometrics speech is used as one of the important biometric in the recent applications. This is because speech production is natural and does not require any special attention from user as contrast to other biometrics [1]. In addition, physical presence of the person is not necessary (remote authentication possible). Moreover, speech is the main modality in telephone

transactions [2].

Speech signal contains both linguistic and nonlinguistic information [3–5]. The linguistic information includes message that can be used for automatic speech recognition. The non-linguistic information includes emotion, dialect and the characteristics of the physical speech production apparatus etc. These information can be used in automatic recognition tasks like speaker, dialect, gender, age, emotion etc. Sources of all these information is present in both physiological and behavioral characteristics. This thesis will focus on automatic speaker recognition task. This aims to analyze, extract, characterize and recognize information about speaker identity [6, 7]. The importance of speaker recognition system is to overcome the problems of traditional authentication system. The traditional system uses password, identity card and badges for person authentication in the secured system [1, 8]. The problem with them are too many of them to manage or somebody may hack them.

1.3 Nature of Speech for Speaker Recognition

Speech is natural and non-stationary signal [9, 10]. Over a short period (quasi-stationary) of time (10-30 ms) its characteristics are assumed to be stationary for practical processing. We can classify speech into voiced and unvoiced depending on the nature of excitation. During voiced speech production vocal cards excite and vibrate. In case of unvoiced speech production, total or partial constriction somewhere along the length of vocal tract takes place. The details of speech production mechanism is given in Appendix A. Though the unvoiced region contains the speaker-specific information, it is less compared to voiced region [1]. Therefore, usually in speaker recognition voiced portion is considered for discriminating the speakers.

In speaker recognition unique speaker-specific features from the speech signal are extracted to recognize a speaker. The uniqueness in the voice of a speaker is due to both physiological and behavioral features. The physiological features include shape and size of the vocal tract, the dynamics of the articulator, the shape and size of the vocal cords and rate of vibration of the vocal cords etc. The behavioral features include accent, speaking rate, prosodic features etc. Since these factors reflect in the speech signal, they can be used for speaker recognition.

Further, these factors vary in the same speaker from time to time [11]. Therefore, the speaker recognition system must cope with these variations.

1.4 Classifications of Speaker Recognition

Depending on the task objective, speaker recognition can be classified as *speaker identification* and *speaker verification.*

1.4.1 Speaker Identification

In this case there is no identity claim and the system identifies who the speaking person is [1,6,11,12]. The identification process compares the test speech data with all the speaker models to decide the speaker of the test speech data. The computational complexity of the speaker identification increases as the population size increases and at the same time performance degrades [6]. Improving speaker recognition performance for limited data and large database is an interesting and challenging task. Speaker identification can be further classified as *closed-set* and *open-set* [11, 13].

1.4.1.1 Closed-set

In this case, test speaker is known *a priori* to be a member of a set of N speakers [11, 13]. The steps involved in closed-set speaker identification is shown as a block diagram in Figure 1.1. In closed-set speaker identification the speaker whose model best matches with the test speech data is declared as speaker of the test data. The closed-set speaker identification system provides the output as identity of a speaker. This system provides high security in the closed environment. The disadvantage of this system is that only *a priori* enrolled speakers can use the system. This type of system may be useful where high security is needed which includes defense, private organization etc.

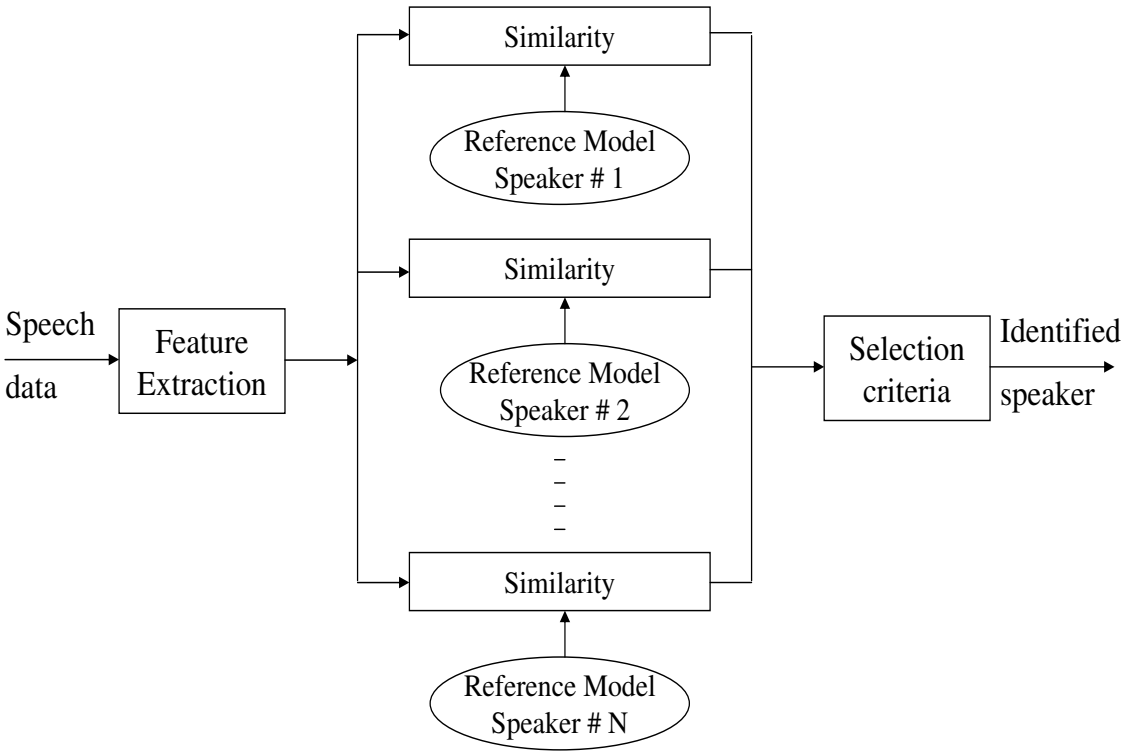

Figure 1.1: Block diagram of closed-set speaker identification system.

1.4.1.2 Open-set

The speaker identification system which is able to identify the speaker, may be from outside the set of N speakers and is known as open-set speaker identification [11,13]. The steps involved in open-set speaker identification is shown as a block diagram in Figure 1.2. In this case, first the closed-set speaker identification system identifies the speaker closest to the test speech data. Then, the verification system is used to compare the distance of this speaker with a threshold to come up with a decision. If the speaker is accepted, then the speaker is the identified speaker for the test data. Otherwise, the system has to generate an error message that the speaker is unknown. The open-set speaker identification system does not impose any restriction in usage as in closed-set. The disadvantage is that the design complexity involved is more and finding the optimum threshold to verify the speaker is a tedious task. This system can be used in public relation services like banking, airport, railways etc.

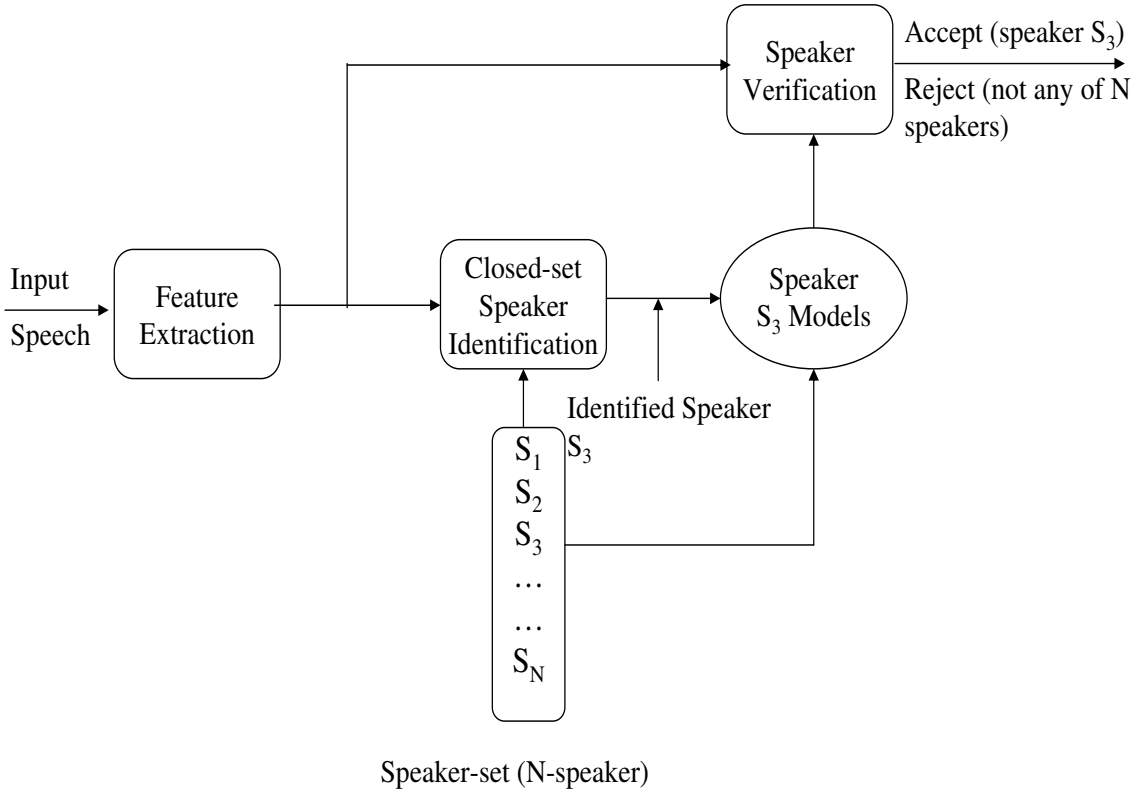

Figure 1.2: Block diagram of open-set speaker identification system.

1.4.2 Speaker Verification

This is used to verify the claimed identity of a person from his/her speech [1, 6, 11, 14]. The steps involved in speaker verification is shown as a block diagram in Figure 1.3. In this case, when an identity claim is made by a speaker, the speech data is compared with respect to the model of the speaker whose identity is claimed. The concept of threshold is used to come up with the decision. If the distance of the test speech data to the target model is below the threshold, then the speaker is accepted as genuine speaker. Unlike in identification, this process involves a binary decision (accept/reject) about the claimed identity regardless of the population size. Hence, the performance of the verification system does not depend on the size of the population.

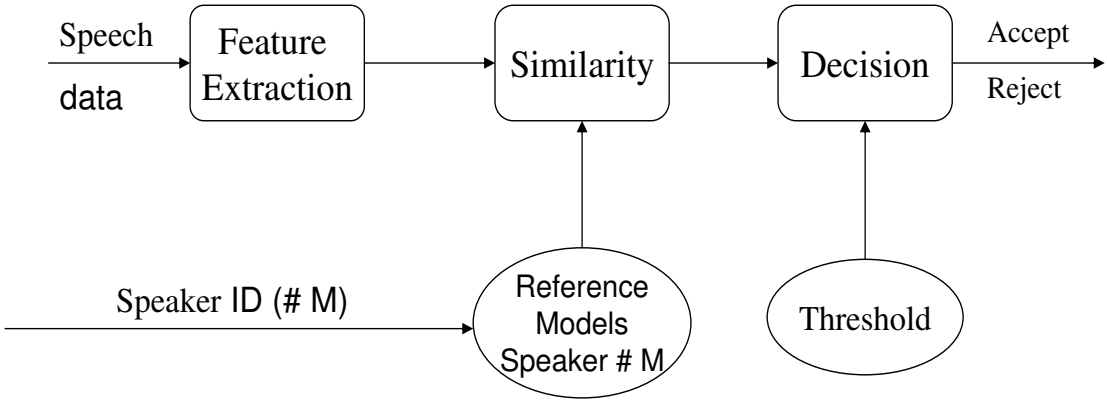

Figure 1.3: Block diagram of speaker verification system.

Depending on the mode of operation, speaker recognition can be classified as *text-dependent* and *text-independent* [1].

1.4.3 Text-dependent

In this mode, speakers require to provide same speech for both training and testing. Moreover, in this mode of recognition the cooperative users speaking fixed digit string passwords or repeating prompted phrases from a small vocabulary are needed [1, 8, 15]. Due to the constraints in the operation, this system can greatly improve the accuracy of recognition. In this case, in order to recognize speaker, template matching is done between training and testing speech data. The technique like Dynamic Time Warping (DTW) is used for template matching.

1.4.4 Text-independent

In this mode, training and testing speech need not be the same [1,8,15]. As we have seen, the text-dependent mode imposes constraint on the speaker. However, there are cases when such constraints can be difficult to impose. This include background verification, forensic, remote biometric person authentication etc. For such cases, a more flexible recognition system which can be able to operate without explicit user cooperation and independent of the spoken speech. Since different speech data is used for training and testing, the performance is not as good

as text-dependent mode. The text-independent recognition is more difficult but also more flexible [1, 8]. The modelling technique like Vector Quantization (VQ) or Gaussian Mixture Models (GMM) is used in this case. Since GMM captures good statistical variations compared to VQ, GMM is used as state-of-the-art modelling technique [16]. Text-independent mode speaker recognition is highly useful if the speaker is noncoperative [8]. *In this thesis, we focus on closed-set, text-independent, speaker identification task.*

1.5 Components of Speaker Recognition

We view the speaker recognition system as four stages, namely, analysis, feature extraction, modelling and testing. In the process of speaker recognition, first speaker-specific features are extracted using suitable analysis technique. Then the speaker reference models are built using the features. Finally, the speaker of the test data is recognized by comparing speaker test data with the reference speaker models. A block diagram of this procedure is shown in Figure 1.4. In this thesis, each stage in the speaker recognition system is dealt in detail in the context of limited data condition. To understand the basic working principle of each block, a brief explanation as an introductory note is given here.

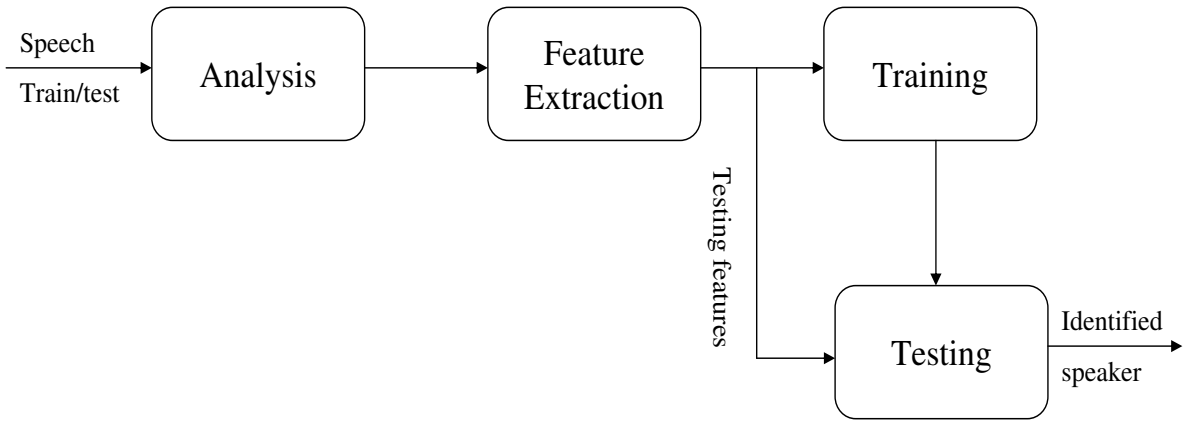

Figure 1.4: Basic block diagram of a speaker recognition system.

1.5.1 Analysis

Speech data contains different levels of speaker information. This mainly include vocal tract, source and suprasegmental levels [17]. The speaker characteristics present at different levels of speech can be captured by various analysis techniques. This include, sub-segmental [18], segmental [19], suprasegmental [20] etc. These analysis techniques provide the means for effectively discriminating the speakers. The significance of the analysis stage is to analyze the speech signal using appropriate frame size and rate to obtain speech segments for feature extraction. During analysis the frame size can be as low as 3-5 ms (sub-segmental) to extract speaker-specific excitation source features [18], medium size like 10-30 ms (segmental) to extract speaker-specific vocal tact information [5, 19] and as high as 100-300 ms (suprasegmental) to extract speaker-specific suprasegmental information [20–22]. The significance of these analysis techniques needs to be verified for speaker recognition under limited data condition.

1.5.2 Feature extraction

Usually the speech production system generates a large amount of data which may not be significant for speaker recognition [19, 23]. Therefore, feature extraction plays an important role in speaker recognition to extract the speaker-specific features at a reduced data rate. The extracted features fill the feature space for effective modelling [6]. The distributions of feature vectors of different speakers overlap and share common feature space, but are distinguishable from each other and hence speaker recognition is possible. In the limited data condition, feature vectors are less due to nonavailability of sufficient data. The insufficient feature vectors lead to poor speaker models and hence degradation in the performance. The selection of features capable of efficiently representing the speaker information in limited data condition from the large variety of features is an issue that needs to be tackled. The features should have some desirable characteristics as follows [6]:

- High inter-speaker and low intra-speaker variability.
- Easily measurable.

- Stable over time.
- Occur naturally and frequently in speech.
- Be robust against noises and distortions.
- Not be subject to mimicry.

It is unlikely that a single feature would fulfill all the above properties. Therefore, a large number of different aspects of speaker information can be extracted and combined to improve the performance.

1.5.3 Modelling

The importance of modelling stage is to create a speaker model using speaker-specific feature vectors. The quality of the speaker model depends on the quality and amount of feature vectors. The desirable attributes of speaker model are better representation of speaker and must consume less time and space complexity [24]. State-of-the-art speaker recognition employs various modelling techniques which will be discussed in the later chapter, that have some or all of these attributes and have been used. The selection of modelling is largely dependent on the type of speech to be used, the expected performance and ease of training and updating. Since the data available is small in limited condition, the selection of modelling technique is also an issue.

1.5.4 Testing

In this stage, speaker models generated during training are used as reference to recognize a speaker of the test speech data. This involves comparison of the feature vectors from the test speech data with the reference speakers models using either template matching or probabilistic modelling techniques [1, 6]. In template matching, test feature vector is compared against the stored reference feature vectors using a suitable distance measurement technique. Probabilistic models involve modelling of speaker by probability distribution and deriving the classification decision based on the probability or likelihood.

1.6 Issues in Speaker Recognition

There are several issues in the speaker recognition area that needs detailed exploration. State-of-the-art speaker recognition technology assumes availability of sufficient data for training and testing [16]. There are practical situations where data available is limited. How to get a reliable and satisfactory performance under limited data? The existing solution in the literature is the use of Gaussian Mixture Model-Universal Background Model (GMM-UBM) modelling technique [25, 26].

The degradation of speech due to sensor, environment and channel conditions is also an important issue in speaker recognition. How to get a reliable and satisfactory performance under degraded condition? The existing solution in the literature is the use of speech enhancement as a preprocessing stage or compensation technique [27, 28].

The use of speaker recognition system in multilingual context is a requirement in a country like India where there is a coexistence of large number of languages. How to get a reliable and satisfactory performance under multilingual context is also another issue? The existing solution in the literature is the use of a language identification system as a preprocessing stage [29].

The use of speaker recognition system in e-transaction needs a robust performance under stressed conditions. This is due to the production of mostly stressed speech during testing speech like loud, fast, angry, and so on. This is also an issue that how to get a reliable and satisfactory performance of speaker recognition system under stressed condition? The existing solution in the literature is the use of a stress compensation system [30].

The robust performance of human speaker recognition infers that there may be some robust features in the speech signal and also different approach followed for recognizing the speakers. This motivates researchers in the speaker recognition area to explore new features and modelling techniques for speaker recognition. In this thesis, among all these issues we considered the limited data issue for the study.

1.7 Motivation for the Present Work

Issues in speaker recognition reveals that the speaker recognition under limited data condition is addressed only from the modelling perspective. As a result, state-of-the-art speaker recognition system uses GMM-UBM for speaker modelling with Mel Frequency Cepstral Coefficients (MFCC) as features. In addition to the modelling, the performance of the speaker recognition system also depends on the techniques employed in the analysis, feature extraction, and testing stages. State-of-the-art speaker recognition employs various analysis techniques, varieties of features and a large number of modelling techniques for speaker recognition in different contexts. Studies are not made to understand the potential of existing techniques other than the MFCC based GMM-UBM modelling in the context of limited data. Therefore, a study is yet to be made to see the strength of the existing techniques. Based on the study one can develop new speech analysis, feature extraction and testing techniques, in addition to existing modelling techniques to improve the performance under limited data condition. Hence the motivation for this work. Also, there are applications where the speech data is limited. Therefore, there is a need for developing techniques for the limited data condition.

1.8 Applications of Limited Data Speaker Recognition

The applications of speaker recognition could be found in physical access control, authentication of remote transactions, shared computer resource access, forensics etc, [8,15]. In practice, there are some applications which require speaker recognition under the constraint of limited training and testing data. This section gives some of such applications.

- *Multi-speaker speech detection, tracking and segmentation*: In multi-speaker scenario it is required that the speakers involved are to be detected, tracked and segment their speech data. Detection aims at verifying whether a particular speaker is present in a speech segment or not? Tracking aims at finding the intervals within a speech segment where a known speaker is speaking. Segmentation aims at determining the intervals within a segment that correspond to each of the unknown speakers present in the segment. In order

to perform these tasks, a model for each of the speakers has to be built using some similar segments of speech data. The speaker models are effective when a large amount of speech data is available. Moreover, the regions of data that are similar have to be identified in the beginning itself. This is little tedious and cumbersome. Therefore, techniques need to be developed which can perform these tasks efficiently with the constraint of limited data [25, 31]. For instance, in spoken document retrieval, it is useful to identify if a speaker within a group appears repeatedly across an audio stream such as tracking television anchors, and separate these speakers from those being interviewed [25].

Studies reported in [32, 33] use statistical dissimilarity measure on the features representing the vocal tract system. A study made in [31] reported that the statistical methods may degrade the performance due to poor parameters estimation under limited data. To overcome this, excitation source is used as feature which does not require long speech data to build good speaker models. However, the existing studies concentrated either on feature extraction or modelling stage. In this work we focus on analysis, feature extraction, modelling and testing stages. Thus, better performance can be obtained compared to the existing techniques for the above mentioned task in multi-speaker environment.

- *Forensic*: Speaker recognition finds increased application in forensic study [34]. This means that speaker recognition can also be used in law enforcement related applications which includes police investigation, judgment in courts etc. [35–38]. In such cases, the amount of speech data available for a speaker training and testing may be as small as few seconds. The speech data may be typically recorded by obtaining access to a telephone channel. This could be an anonymous call or known or expected or tapping channels [15, 31]. For instance, in case of tapped channels, the speaker may be speaking only for few seconds and the next recording, even available, may be from different channel. Further, he/she may not be available for giving enough speech data to train and test the system. In such a scenario a better approach is to develop methods for better modelling and testing of speakers with only the available limited data, rather than taking some more

data which may not yield required improvement in performance.

- *Remote biometric person authentication*: In modern times with the interest of using internet based computer systems, the interest for security issues is an increasing demand. An important issue in this is automatic authentication of persons [1, 11, 15]. This has made remarkable migration from password and token-based authentication to speech based. In case of remote person authentication, person can send the speech through the transmission media for authentication purpose. This means that the physical presence of the person is not required for the authentication. The existing systems typically employ large amount of training data for authentication. This is both labor and computation intensive. Moreover, sending large amount of speech data over the transmission media is not feasible and may increase communication overheads. Therefore, if we have a speaker recognition system which can be able to recognize the speaker using less amount of data is indeed useful. Thus in all these applications we benefit immensely if we have speaker recognition system that provides reliable performance using only limited data and hence the motivation for the work.

1.9 Organization of the Thesis

The contents of the thesis are organized as follows:

In **chapter 2**, a review of the existing methods for the speech analysis, feature extraction, modelling and testing techniques for speaker recognition is given. This chapter also discusses developments in speaker recognition system under limited data condition.

Chapter 3 discusses existing and proposed speech analysis techniques for speaker recognition. A method for analyzing the speech signal under limited data condition is proposed.

In **Chapter 4** efficiency of different feature extraction techniques are studied for speaker recognition under limited data condition. Motivation for combining different information to improve the performance is discussed and a combination of features is demonstrated.

Chapter 5 illustrates the performance of traditional modelling techniques under limited

data. This chapter also discusses the most widely used modelling techniques under limited data condition. A combined modelling technique is proposed to improve the performance under limited data condition.

In **Chapter 6** significance of the integrated systems for speaker recognition under limited data condition is discussed.

Chapter 7 describes the combination schemes for combining the evidences from the integrated systems.

A summary of the present work is given in **Chapter 8** by listing major contributions of the present work and some directions for further research in the area of speaker recognition under limited data condition.

2

Speaker Recognition - A review

Contents

2.1 Introduction

Speaker recognition is a multidisciplinary problem. This means that, domain knowledge of speech, an understanding of pattern recognition and signal processing techniques are necessary to design an efficient recognition system. The overall performance of the speaker recognition system depends on the techniques employed in the analysis, feature extraction, modelling and testing stages. Therefore, to improve the performance, for each stage personalities in the field have put effort since about four decades. To develop a speaker recognition system for limited data condition that provides practically good performance, we need to understand the various techniques developed so far in the speaker recognition field. This chapter will therefore provide a review of some of the approaches developed in the speaker recognition field. The chapter is organized as follows: The approaches developed for speech analysis, feature extraction, modelling and testing stages are given in Section 2.2, 2.3, 2.4, 2.5, respectively. Section 2.6 discusses summary of the literature review and scope for the present work of speaker recognition under limited data condition. In Section 2.7, the organization of the present work is given.

2.2 Speech Analysis Techniques

Speech data contains different levels of information which can be used to convey speaker identity. These include speaker-specific information due to vocal tract, excitation source and suprasegmental features like intonation, duration and accent. In order to obtain good speaker characteristics, speech data is to be analyzed using suitable analysis technique. The analysis technique aims at selecting proper frame size and shift to extract relevant speaker-specific information in the feature extraction stage. State-of-the-art speaker recognition systems mainly analyze the speech using the following analysis techniques:

- *Segmental analysis:* In this case, speech is analyzed using the frame size and shift in the range of 10-30 ms to extract mainly the vocal tract information as speaker cues for speaker recognition. State-of-the-art systems mainly use vocal tract information for

speaker recognition. The speaker-specific vocal tract information is usually slow varying one and hence 10-30 ms frame size and shift capture the required information and also satisfies the stationarity assumption [23]. Studies made in [19, 20, 39–42] used segmental analysis to extract the vocal tract information for speaker recognition. In most of these studies, vocal tract information is captured using either Mel Frequency Cepstral Coefficients (MFCC) or Linear Prediction Cepstral Coefficients (LPCC).

- *Sub-segmental analysis*: Speech analyzed using the frame size and shift in the range of 3-5 ms is known as sub-segmental analysis [18]. Studies made in [17, 22, 43–46] demonstrated that speaker-specific excitation source information can be captured using the sub-segmental analysis of speech. Also, it is demonstrated that such an information can be used for speaker recognition. Since the excitation source information is fast varying one, small frame size and shift are required to capture the speaker-specific information.

- *Suprasegmental analysis*: In this case, speech is analyzed using the frame size and shift in the range of 100-300 ms. Studies made in [17, 20, 21, 47, 48] demonstrated that some high level speaker information can be captured using suprasegmental analysis that can be used for speaker recognition. Since this information is concerned to the behavioral aspect of the speaker, the large frame size and shift is required to capture the speaker-specific information.

- *Pitch synchronous analysis*: In speaker recognition systems, if there is a pitch mismatch between training and testing then it may increase the error rates. Therefore, pitch synchronous technique is required to overcome these drawbacks. A study made in [49] reported that Pitch Synchronous Mel-frequency cepstral coefficients (PSMFCC) give better recognition performance than the MFCC alone for speaker recognition. In another study, pseudo pitch synchronous analysis of speech with applications to speaker recognition is studied [50]. In this paper, a method is proposed to alleviate the truncation of pitch cycles while using constant frame size and rate. It is shown that the speaker recognition performance can be improved by aligning each individual frame to its natural cycle.

All these speech analysis techniques are employed on the speech signal to extract speaker-specific information. The amount of speaker-specific information obtained is proportional to the amount of speech data available. Thus more the speech data, more will be the extracted speaker-specific information for further processing. To obtain good performance, sufficient speech data is collected. However, the present work considers limited data condition for the study. Thus the suitability and efficiency of each of these techniques need to be relooked under the constraint of limited data.

2.3 Feature Extraction Techniques

The purpose of feature extraction techniques is to extract the speaker-specific information at a reduced data rate. To develop a good feature set for speaker recognition, it is necessary to understand the different aspects of speech signal which are responsible to the human perception of voice individuality. This section describes the development of feature extraction techniques for speaker recognition.

Spoken digit recognition conducted by P. Denes *et al.* in 1960 suggested that inter-speaker differences exist in the spectral patterns of speakers [51]. In this work, each utterance of a speaker is converted in the form of a time-frequency spectrum for recognition. S. Pruzansky motivated from the previous study by Denes conducted a first speaker identification study in 1963 by pattern matching procedure. In his study, spectral energy patterns are used as the features and product moment coefficient of correlation is used as similarity measure. It is shown for the text-dependent speaker recognition that the frequency-energy based spectrum yields the good recognition performance compared to the time-frequency spectrum [52]. Further, he reported in [53] the speaker identification study using analysis of variance in 1964. In this work, a subset of features are selected from the analysis of variance which uses F-ratio test.

Speaker verification study was first conducted by Li *et al.* in 1966 using adaptive linear threshold elements [41]. This study used spectral representation of the input speech, obtained from a bank of 15 bandpass filters spanning the frequency range 300-4000Hz. Two stages of adaptive linear threshold elements operate on the rectified and smoothed filter outputs. These

elements are trained with fixed speech utterances. The training process results in a set of weights for the various frequency bands and time segments. The weights characterize the speaker. A study by Glenn *et al.* in 1967 suggested that acoustic parameters produced during the nasal phonation are highly effective for speaker recognition [54]. In this study, average power spectra of nasal phonation are used as the features for speaker recognition and cosine angle is used as the decision criteria. In 1969, Fast Fourier Transform (FFT) based cepstral coefficients are used in the speaker verification study [19]. In this work, a 34-dimensional vector is extracted from speech data. The first 16 components are from FFT spectrum, next 16 are from Log magnitude FFT spectrum and the last two components are related to pitch and duration. Decision was made based on nearest neighbour procedure using Euclidean distance.

A study made in [41] uses only band energies for speaker verification, but study in [55] uses in addition to band energies, pitch and formant information to improve speaker verification performance. A study made by G. R. Doddington in [40] reported different approach for speaker verification than in [41] and [55]. He does not use a filter bank, but converts the speech directly to pitch, intensity and formant frequency values, all sampled 100 times per second. In his study, a procedure is developed by which a sample utterance is time registered with a stored reference data of identity claimed. The second formant is used as the criterion of time registration. The problem with this study is that it is not found to be suitable for commercial application due to the requirement of large computing capability. Alternatively, R. C. Lummis proposed the use of intensity pattern in place of the second formant as the criterion of time registration without spoiling system performance for speaker verification [56].

Most of the above studies used spectrographic pattern of speech for speaker recognition. Atal in 1972 introduced the use of variations of pitch as feature for speaker recognition [20]. In this work, pitch period is obtained by short time correlation analysis on the cubed low pass filtered signal. In addition to pitch the other acoustic parameters such as glottal source spectrum slope, word duration and voice onset time are proposed for speaker recognition by Wolf in 1971 [39]. The concept of linear prediction for speaker recognition is introduced by Atal in 1974 [57]. In this work, it is demonstrated that LPCC are better than the Linear Prediction Coefficients

(LPC), and other features such as pitch and intensity. The Mahalanobis distance criteria is used as similarity measure. In general, the advantages of the cepstral coefficients is that they can be derived from a set of parameters which are invariant to any fixed frequency-response distortion introduced by the recording or transmission system [6].

Earlier studies neglected the features such as formant bandwidth, glottal source poles and higher formant frequencies due to nonavailability of measurement techniques. However, studies introduced after the linear prediction analysis, invented the speaker specific potentials of these features for speaker recognition [58]. A study made by Rosenberg and Sambur suggested that adjacent cepstral coefficients are highly correlated and hence all coefficients may not be necessary for speaker recognition [59]. In 1976, Sambur proposed to use orthogonal linear prediction coefficients as features in speaker identification [60]. In this work, he pointed out that for a speech feature to be effective, it should reflect the unique properties of the speakers vocal apparatus and contain little or no information about the linguistic content of the speech. Also, it is shown that only a small subset of parameters describe the significant variance in the utterance.

In 1977, long-term parameter averaging which includes pitch, gain and reflection coefficients for speaker recognition are studied [61]. In this study, it is shown that the reflection coefficients are highly informative and effective for speaker recognition. Furui introduced the concept of dynamic features, to track the temporal variability in the feature vector to improve the speaker recognition performance in 1981 [62, 63]. A study by Reynolds in 1994 compared the different features like MFCC, Linear Frequency Cepstral Coefficients (LFCC), LPCC and Perceptual Linear Prediction Cepstral Coefficients (PLPCC) for speaker recognition [42]. He reported that, among these features MFCC and LPCC give better performance than the other features. Even today most of state-of-the-art speaker recognition systems use either LPCC or MFCC as main features for speaker recognition [7, 13, 64].

The above discussed most of the studies considered vocal tract information as speaker characteristics for speaker recognition. In [65], it is reported that Linear Prediction (LP) residual also contains speaker-specific source information that can be used for speaker recognition. Also,

it is reported that though the energy of the LP residual alone gives less performance, combining with LPCC improves the performance over the LPCC alone. In the similar line, several studies demonstrated that though the LP residual gives less performance compared to the MFCC for speaker recognition, combining LP residual feature with MFCC improves the performance over MFCC [22,44–46]. Recently, it is reported that LP residual phase also contains speaker-specific source information in [43]. In this study, it is demonstrated that though the LP residual phase gives less performance than that of the MFCC, combining together improves the performance over MFCC [43].

Most of the studies discussed so far have not considered the high level features like word duration, intonation, pitch contour, speaking rate and speaking style etc., for speaker recognition. A study made in [66], demonstrated the significance of pitch and energy information for speaker recognition. In another study, pitch, pitch tracks and local dynamics in pitch are also used in speaker verification [67]. A study in [68] reported that prosodic features like pitch alone even though does not able to provide the good performance but with spectral features provide significant improvement. A study made in [17] demonstrated the use of suprasegmental features like pitch and duration information along with source and spectral features for text-dependent speaker recognition. In this study, suprasegmental information like pith and duration are obtained using DTW algorithm. In [47], suprasegmental features like duration and intonation are used for speaker recognition. In this study, these features are captured with the help of neural networks.

In the direction of new features, Imperl *et al.* analyzed the spectral information captured by harmonic features for speaker identification [69]. In this study, he suggested that these features are better than that of the LP-based features. Plumpe *et al.* developed a technique for estimating and modelling the glottal flow derivative waveform from speech for speaker recognition. In this study, the glottal flow estimate is modeled as coarse and fine glottal features which are captured using different techniques. Also, it is shown that the combined coarse and fine structured parameters give better performance than the individual [70]. In [71], Amplitude Modulation (AM)- Frequency modulation (FM) based parameterization of speech is

proposed for speaker recognition. In this study, it is demonstrated that different instantaneous frequencies due to the presence of formants and harmonics in the speech signal it is possible to discriminate speakers. Also, it is shown that channel normalization is not required, as the (instantaneous) amplitude is used only for identifying the short time frequency estimate within a single band.

Most of the explorations in the feature extraction stage aims to characterize the speaker information due to the vocal tract, excitation source and behavioral (suprasegmental) aspects. Among these speaker-specific vocal tract information is shown to provide best performance. This observation is under the implicit assumption of sufficient data. Further explorations are underway to exploit speaker-specific excitation source and suprasegmental analysis. Attempts are also needed to exploit these different levels of speaker information under limited data condition.

2.4 Speaker Modelling Techniques

The objective of modellng is to generate speaker models using speaker-specific feature vectors. State-of-the speaker recognition systems employ different modelling techniques which are briefly described in this section. Earlier studies for speaker recognition used direct template matching between training and testing data [19, 52, 54, 57–60]. In the direct template matching, training and testing feature vectors are directly compared using similarity measure. For the similarity measure any of the techniques like spectral or Euclidean distance or Mahalanobis distance are used. Furui introduced the concept of Dynamic Time Warping (DTW) for text-dependent speaker recognition [63]. However, it was originally developed for speech recognition [72]. In this approach the sequence of feature vectors of training signal is the text-dependent template model. The DTW finds the match between the template model and the input sequence of feature vectors. The disadvantage of template matching is it is time consuming as the feature vector size increases. For this reason, it is common to reduce the number of test vectors by clustering approach. The cluster centers are known as *codevectors* and a set of codevectros is known as *codebook*.

A study reported in [73], demonstrated text-independent speaker identification with short utterance using the concept of clustering feature space of speakers to build reference models. This study showed that the cluster centers approximate the multi-modal distribution of the speaker's characteristics that can be estimated using the notion of character encoding. Also, it is shown that the speaker recognition performance for a set of 11 speakers using the speaker models built by multi-modal distribution is better than the uni-modal distribution.

The most well-known codebook generation algorithm is the Generalized Lloyd Algorithm (GLA) or the Linde-Buzo-Gray (LBG) or the *K-means* algorithm [74, 75]. In 1985, Soong *et al.* [76] used the LBG algorithm for generating speaker-based Vector Quantization (VQ) codebooks for speaker recognition on the database of isolated word recognition. In this study, it is demonstrated that larger codebook and larger test data gives good recognition performance. Also, the study suggested that VQ codebook can be updated from time to time to alleviate the performance degradation due to different recording conditions and intra-speaker variations [76]. The disadvantage of the VQ classification is, it ignores the possibility that a specific training vector may also belong to another cluster. As an alternate to this, Fuzzy Vector Quantization (FVQ) using the well-known fuzzy *C-means* method was introduced by Dunn and its final form was developed by Bezdek [77, 78]. In [79] and [80], FVQ is used as a classifier for speaker recognition. In these studies, it is demonstrated that FVQ gives better performance than the traditional *k-means* algorithm.

The use of Hidden Markov Model (HMM) for text-dependent speaker recognition is studied in [12, 81, 82]. In these studies, it is demonstrated that HMM efficiently models statistical variations and gives better performance than the DTW. In HMM, time-dependent parameters are observation symbols. Observation symbols are created by VQ codebook labels. Continuous probability measures are created using Gaussian Mixtures Models (GMM). Main assumption of HMM is that the current state depends on the previous state. In training phase, state transition probability distribution, observation symbol probability distribution and initial state probabilities are estimated for each speaker as a speaker model. The probability of observations for given speaker model is calculated for speaker recognition.

Kimbal *et al.* studied the use of HMM for text-dependent speaker recognition under the constraint of limited data and mismatched channel conditions [83]. In this study, MFCC feature is extracted for each speaker and then models are built using the Broad Phonetic Category (BPC) and the HMM based Maximum Likelihood Linear Regression (MLLR) adaptation technique. The BPC modelling is based on phonetic categories identification in an utterance and modelling them separately. In HMM-MLLR, first Speaker-Independent (SI) model is created using HMM and then MLLR technique is used to adapt each speaker to SI model. It is shown that the speaker model built using the adaptation technique gives better performance than the BPC and GMM for cross channel conditions.

The capability of neural networks to discriminate between patterns of different classes is exploited for speaker recognition [84–86]. Neural network has an input layer, one or more hidden layer and an output layer. Each layer consists of processing units, where each unit represents model of an artificial neuron, and the interconnection between two units as a weight associated with it. The concept of Multi-Layer Perception (MLP) is used for speaker recognition in [87]. In this study, it is demonstrated that one hidden layer network with 128 hidden nodes gives same performance as that of the 64 codebook VQ approach. The disadvantage of MLP is that it takes more time for training the network. This problem was alleviated using the Radial Basis Function (RBF) in [88]. In this study, it is shown that the RBF network takes less time than the MLP and outperforms both VQ and MLP.

The Self Organization Map (SOM) and associative memory model are used together as a hybrid model for speaker identification in [89]. In this study, it is shown that the hybrid model gives better recognition performance than the MLP. A text independent speaker recognition system based on SOM neural networks is also studied in [90]. The Learning Vector Quantization (LVQ) is proposed for speaker recognition in [91]. In this study the experiment is conducted for a set of 10 speakers using MFCC and LPCC as feature vectors. It is shown that MFCC gives better result than the LPCC. The speaker recognition study using VQ, LVQ and GVQ (Group Vector Quantization) is demonstrated for YOHO database in [92] for various data lengths. In this study, the experimental results show that LVQ gives better recognition performance when

the data is small compared to the traditional VQ and proposed GVQ, but GVQ yields better recognition performance when the data size is large. Though the neural network classifier performs better than the other classifiers, it needs to be retrained on addition of any new speaker.

Reynolds proposed GMM classifier for speaker recognition task in 1995 [16]. In this study, experiments are conducted for different databases to study the effectiveness of GMM. In GMM, the underlying probability density function of the feature vectors of each speaker is captured using Gaussian mixtures. The complete Gaussian mixture density is parameterized by the mean vector, covariance matrix, and mixture weights for all the components. In another study, Reynolds compared GMM performance of speaker identification with other classifiers like unimodal Gaussian, VQ codebook, tied Gaussian mixture, and radial basis functions [93]. In this study, it is shown that GMM outperforms the other modelling techniques. Therefore, state-of-the-art speaker recognition systems use GMM as classifier due to probabilistic framework, training methods scalable to large data sets and high accuracy recognition [94]. Studies made in [71, 95–98] also used GMM for speaker modelling in speaker recognition. In [99], the feature vectors that discriminate the speakers well have been considered for speaker recognition under limited data. In this study, first speaker models are built for training feature vectors using the conventional GMM. Then the training feature vectors are separated as overlapped and non-overlapped feature vectors by comparing them with trained models. Again, the speaker models are built only for non-overlapped feature vectors and then identification is done. It is shown for a set of 8 speakers, using MFCC feature that non-overlapped GMM performs better than the conventional GMM and Liner Discriminant Analysis (LDA)-GMM.

The disadvantage of GMM is that it requires sufficient data to model well the speaker parameters [16]. To overcome this problem, Reynolds *et al.* introduced Gaussian Mixture Models-Universal Background Model (GMM-UBM) for speaker recognition task [100]. In this system speech data collected from large number of speakers is pooled and the UBM is trained which acts as a speaker independent model. The speaker dependent model is then created by performing maximum *a posteriori* (MAP) adaptation technique from the UBM using speaker-

specific training speech. As a result, the GMM-UBM gives better result than the GMM. The use of GMM-UBM with MAP adaptation has been studied for speaker recognition under limited data condition in [25]. This study used the 19 dimensional MFCC, in addition, appended four dimensional subband spectral gravity center (SSGC) as feature vectors. The SSGCs are computed from four non-overlapped subbands, where 1000 Hz is allocated per subband. Experiment is conducted to identify an Inset/Out of set speaker. Alternatively, a cohort GMM model is built for each speaker by pooling data from acoustically close speakers [26]. It is shown that the performance of the speaker recognition using only MFCC and MAP based adaptation in cohort GMM models is better than the reported in [25]. In [101], Kernel Eigenspace-based Maximum Likelihood Linear Regression (KEMLLR) adaptation technique is proposed for speaker verification with limited enrollment data. It is shown using the MFCC feature that KEMLLR adaptation yields better performance than the MAP adaptation when training data is less than 8 sec. In [102], Vector quantization - Universal Background model (VQ-UBM) is proposed as an alternate to GMM-UBM. It is claimed that only the mean (centroid) adaptation rather than adapting mean, covariance and weights as in GMM-UBM gives same result as that of GMM-UBM.

As an alternate to the GMM, an Auto Associative Neural Network (AANN) is developed for pattern recognition task [86, 103, 104]. AANN is basically a feed forward neural network which tries to map an input vector onto itself. The number of units in the input and output layers are equal to the size of the input vectors. The number of nodes in the middle layer is less than the number of units in the input or output layers. The activation function of the units in the input and output layer is linear, whereas the activation function of the units in the hidden layer can be either linear or nonlinear. The advantage of AANN over GMM is that, it does not impose any distribution. The application of AANN is extensively studied for speaker recognition in [17, 22, 36, 43, 105].

Vincent Wan and Steve Renals in 2002 studied Support Vector Machines (SVM) for speaker recognition [106]. In this study, different kernels like the polynomial, the Fisher, a likelihood ratio and the pair HMM are studied. Authors reported that using these kernels it is indeed

possible to achieve the state-of-the-art speaker recognition performance. Further, the same authors have used score space kernels for speaker verification study in [107]. The score space kernels generalize Fisher kernels and are based on underlying generative models such as GMM. In this study, it is demonstrated that SVM reduces the error rate compared to GMM likelihood ratio system. W. M. Campbell *et al.* proposed Generalized Linear Discriminant Sequence (GLDS) kernel for speaker recognition and language identification task [94].

The studies made include both probabilistic and non-probabilistic speaker modelling techniques. Among these two, probabilistic modelling, in particular, GMM shown to give good recognition performance. However, it has its own advantages and disadvantages. This includes, it gives good recognition performance compared to other modelling techniques provided sufficient data is available for modelling. As an alternate to this, GMM-UBM is developed which requires less speaker dependent data but more speaker independent data for modelling. However, in the limited data condition we may not have sufficient speaker independent data. Therefore, in addition to GMM-UBM system for speaker recognition under limited data condition either the efficiency of other existing non-probabilistic or new modelling techniques are to be explored.

2.5 Speaker Testing and Decision Logic

Testing stage in the speaker recognition system includes matching and decision logic. During testing, usually the test feature vectors are compared with the reference models. Hence, matching gives a score which represents how well the test feature vectors are close to the reference models. Decision will be taken on the basis of matching score which depends on the threshold value. In the speaker verification system the performance is measured in terms of Equal Error Rate (EER), which is defined as the error rate at which False Acceptance (FA) rate is equal to False Rejection (FR) rate. Moreover, the detection probability as a function of false alarm probability known as Receiver Operating Characteristics (ROC) plot which is used for the visualization of speaker verification performance. In order to improve the visualization the Detection Error Trade-offs (DET) plot is used where miss and false alarm probabilities

are plotted according to their corresponding Gaussian deviations [64]. On the other hand, the computation of speaker identification performance is direct and simple. This is measured as a ratio of the total number of correctly identified speakers out of N speakers considered for the study.

In both the speaker verification and identification, for matching test feature vectors to reference model, either we can use distance measurement techniques or probabilistic model. Earlier studies employed spectral or Euclidean or Mahalanobis distance measurement techniques for comparison between the speaker models and testing data for speaker recognition [19, 52, 54, 57–60]. Reynolds used the concept of log likelihood ratio test for speaker recognition [16]. In 2001, H. Jiang and L. Deng studied the Bayesian approach for speaker recognition [108]. In this study, it is demonstrated that Bayesian approach moderately improves the performance compared to well-trained baseline system using the conventional likelihood ratio test.

In order to improve the speaker recognition performance at decision level, combination of multiple classifiers is proposed [109]. In this study, voting method is used for speaker identification based on the results of various resolution filterbanks. A study made in [17] reported that by combining the evidences from source, suprasegmental and spectral features it is indeed possible to improve the performance of the speaker recognition system. In the similar lines, studies in [22, 43] have also demonstrated the combination of evidences from system and source features to improve the performance. In [94], it is reported that the performance of the speaker recognition can be improved by combining the evidences from SVM and GMM classifiers.

Testing technique largely depends on the modelling technique employed in the system. The design of the new modelling technique also leads to new testing technique. In the literature, GMM-UBM is the widely used modelling technique under limited data condition. Therefore, the maximum likelihood ratio test is used as testing strategy and is shown to give good recognition performance. In addition to this, the efficiency of other modelling and distance measurement techniques are to be explored under limited data condition. Moreover, various combination schemes at the decision level are to be explored under limited data condition.

2.6 Summary and Scope for Present Work

In this chapter, we have discussed some techniques developed for each stage of the speaker recognition system. The different analysis, feature extraction, modelling and testing techniques are discussed. In summary,

(i) In the speech analysis stage, though the techniques have been developed to improve the speaker recognition performance, no particular analysis techniques is specially meant for limited data condition. All most all the techniques assume the availability of sufficient data for speaker recognition and hence analyze the speech signal using segmental analysis (fixed frame size and rate) to extract vocal tract information. The use of segmental analysis under limited data condition provides few feature vectors which leads to poor speaker models. As a result, degradation in the performance. Therefore, techniques are to be explored to analyze the speech signal using the suitable analysis technique to improve the performance under limited data condition.

(ii) Few attempts have been made in the feature extraction stage to improve the performance under limited data condition. In those studies also MFCC is used as feature. There are no attempts in using alternate features from the excitation source and suprasegmental information. The limited data provides few speaker-specific features. Therefore, efforts are yet to be made either to develop new features or combining the different features to improve the performance under limited data condition.

(iii) Majority of the studies made in speaker recognition under limited data condition are concentrated in modelling stage. In all most all the studies, GMM-UBM with different adaptation techniques are used as modelling technique. The advantage of the GMM-UBM is it gives good performance under the constraint of limited data. The disadvantage is that a large speaker independent and gender balanced speaker set required for UBM. In practice it is very difficult to meet the requirement. Therefore, a modelling technique which is suitable for limited data condition without such constraints is required to improve

the performance. Moreover, the effectiveness of other modelling techniques like neural networks, crisp and fuzzy vector quantization are to be explored under limited data condition.

(iv) In the testing and decision logic stage, majority of the studies used distance measurement techniques like either Euclidean or Mahalanobis or maximum likelihood ratio test. The effectiveness of other distance measurement techniques are to be explored under limited data condition. In the decision logic, combination of different classifiers or systems output using different combination schemes are not much explored in speaker recognition. Therefore, studies are to be made using the different combination schemes to improve the performance.

2.7 Organization of the Work

Chapter 3 demonstrates the strength of Multiple Frame Size and Rate (MFSR) analysis for speaker recognition under limited data condition. First, studies are made using Multiple Frame Size (MFR) and Multiple Frame Rate (MFR). Then, the experimental results are compared with Single Frame Size and Rate (SFSR) analysis. Finally, to gain the advantages of MFS and MFR, we combine them and call it as MFSR analysis. The experimental results of MFSR is also compared with MFS, MFR and SFSR.

The combination of features for speaker recognition under limited data condition is described in Chapter 4. First, the working principle of different feature extraction techniques which provide features like MFCC, Delta (Δ) MFCC, Delta-Delta ($\Delta\Delta$) MFCC, Linear Prediction Residual (LPR) and Linear Prediction Residual Phase (LPRP) are studied. Then the effectiveness of each feature extraction technique is experimentally explored independently to know the level of speaker information present in them. Finally, all the features are combined to obtain better representation of speaker. The experimental results of combined features are compared with the individual features.

Chapter 5 proposes the combined modelling techniques for speaker recognition under lim-

ited data condition. First, the pattern classification principle involved in different modelling techniques like VQ, FVQ, SOM, LVQ, GMM and GMM-UBM are studied. Then, the performance of each modelling technique is verified through experimental studies. Finally, based on the performance of the individual classifiers we have combined them at the scoring level to see the effectiveness under limited data condition. The experimental results of different combined modelling techniques are compared with the individual modelling techniques.

The techniques we propose for analysis, feature extraction and modelling stages are demonstrated independently. That is, proposed technique used in the respective stage and keeping the existing techniques in the remaining stages. In Chapter 6, proposed techniques are integrated to see the effectiveness of integrated system. The experimental results of the integrated system are compared with the proposed individual systems.

Integrating the techniques lead to different individual integrated systems. In chapter 7, evidences from the different integrated systems are combined using different combination schemes at abstract, rank and measurement levels. The different combination schemes results are compared with the proposed integrated and individual systems.

In chapter 8, summary of the present work is discussed first. Then, the major contributions of the work in developing some approaches for speaker recognition under limited data condition are mentioned. Finally, some possible future directions are mentioned.

3

MFSR Analysis of Speech for Limited Data Speaker Recognition

Contents

State-of-the-art speaker recognition systems assume the availability of sufficient data for modelling and testing. Due to this, speech signals are analyzed with fixed frame size and rate which may be termed as Single Frame Size and Rate (SFSR) analysis. In the limited data condition available training and testing data is small. If we use SFSR analysis, then it may not provide sufficient feature vectors to train and test the speaker. Further, insufficient feature vectors lead to poor speaker modelling during training and may not yield reliable decision during testing. In this chapter, as part of analysis, we demonstrate the use of Multiple Frame Size (MFS), Multiple Frame Rate (MFR) and Multiple Frame Size and Rate (MFSR) analysis techniques for speaker recognition under limited data condition. These techniques produce relatively more number of feature vectors. This helps in better modelling and testing under limited data condition. The experimental results show that use of MFS, MFR and MFSR analysis improves the performance significantly compared to SFSR analysis.

3.1 Introduction

This chapter focuses on exploring alternate speech analysis techniques to extract vocal tract information from the speech signal. In the existing speaker recognition systems the analysis stage uses frame size and shift in the range of 10-30 ms. If the data available is only few seconds, then 10-30 ms choice provides only few feature vectors This will lead to poor speaker modelling and also may not reliably test the speakers [110]. One approach to mitigate this problem is to artificially increase the number of feature vectors. In the existing speaker recognition systems once the frame size and shift are chosen, they are kept constant throughout the experiment, and hence it may be termed as Single Frame Size and Rate (SFSR) analysis.

In this work, the same speech is analyzed using different frame size and rate and hence it is termed as Multiple Frame Size and Rate (MFSR) analysis. The motivation behind varying the frame size is to perform a multi-resolution analysis of the same speech data. It is observed that the feature vectors representing the vocal tract information extracted from the same speech signal by multi-resolution analysis are considerably different [110, 111]. Further, the speaking rate as well as pitch are different for different speakers and also for the same speaker depending

on the contextual information during speech production. This is manifested as rate of change in the spectral information and hence the associated non-stationarity. This may be captured by changing the frame rate. Since the same speech data is analyzed at different frame sizes and rates, the set of speech samples involved in the analysis are different at each frame size and rate and hence the feature vectors representing the vocal tract information may be different. Thus the MFSR analysis technique generates more number of feature vectors and hence may result in better speaker modelling and testing.

The rest of the chapter is organized as follows: MFSR analysis of speech is discussed in Section 3.2. In Section 3.3, we describe the speaker recognition studies using MFSR analysis technique under limited data condition. Section 3.4 presents experimental results and discussion. Finally summary of the work discussed in this chapter is given in Section 3.5.

3.2 MFSR Analysis of Speech

Earlier, speech recognition studies used the Variable Frame Rate (VFR) analysis technique to increase frames around the significant spectral variations in speech [112–115]. Since the speaking rate and pitch also lead to change in spectral information, analyzing the speech signal with VFR seem to provide better performance compared to Single Frame Rate (SFR) analysis. The motivation of all these studies was to increase the speech recognition performance at the cost of reducing the time and space complexity. Nowadays, due to advancement in technology the time and space complexity is not a major problem. In MFSR analysis, the same speech data is analyzed using multiple frame size and rates to increase the performance over single frame size analysis. The main advantage of MFSR over VFR is reducing the burden of identifying the spectral changes in speech. The significance of MFSR analysis technique has already been demonstrated in speech processing tasks like speech recognition, automatic transcription and language identification [111, 116]. It was demonstrated in all these studies that the Multiple Frame Size (MFS) and Multiple Frame Rate (MFR) analysis leads to different feature vectors for the same data and in turn ensures reasonable variance for each Gaussian mixture in the models. Further, the performance compared with Single Frame Size (SFS) analysis was demonstrated to

be better under limited data condition. The significance of combined MFS and MFR analysis for speech recognition in limited data condition has been demonstrated in [117]. It was shown that when the number of training examples are less, combined MFS and MFR analysis provides better performance compared to existing SFS analysis. This motivated us to use this concept in speaker recognition under limited data condition to mitigate the problem of sparseness of limited data. The details of the techniques used for speaker recognition are as follows:

3.2.1 MFS Analysis

In MFS analysis, speech data is analyzed using different frame sizes but with constant frame shift. It is effectively a multi-resolution analysis technique. The magnitude spectra and the resultant feature vectors extracted from the speech signal with different frame sizes are considerably different due to different frequency resolutions [23]. This is because information present in the spectral domain is due to the convolution of true spectrum of speech and spectral domain window. Further, the speech samples in each frame size are slightly different. Both these factors may lead to varied levels of manifestation of speaker information in different feature vectors. Thus by varying frame size we can vary the spectral information manifested and hence feature vectors with different speaker-specific information.

In order to study the SFSR and proposed methods this work uses Mel Frequency Cepstral Coefficients (MFCC) as feature vectors [42]. The MFCC are known to give good performance for speaker recognition [118]. The cepstral analysis is described in Appendix B. The various steps involved in the MFCC feature extraction is shown in Figure 3.1. The brief description of the steps are as follows [3, 5, 9]:

(i) Pre-emphasis: This refers to filtering that emphasizes the higher frequencies. Its purpose is to balance the spectrum of voiced sounds that have a steep roll-off in the high frequency region. The most commonly used pre-emphasis filter is given by the following transfer function

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

where the value of b controls the slope of the filter and is usually between 0.9 to 1.0.

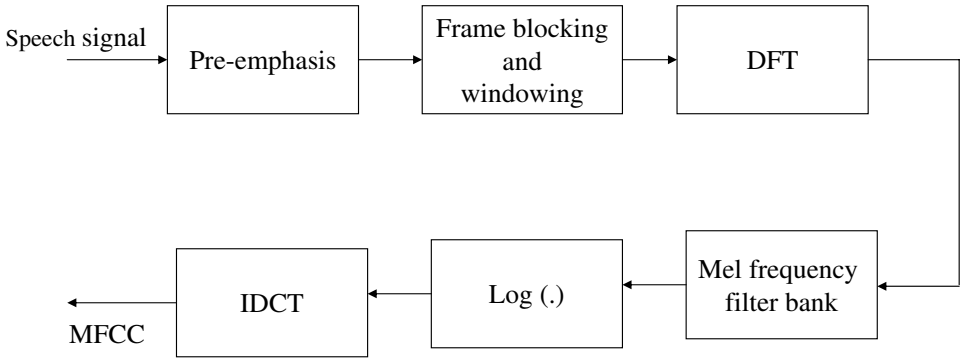

Figure 3.1: MFCC feature extraction process

(ii) Frame blocking and windowing: The speech is slow varying quasi-stationary signal. Therefore, speech analysis must always be carried out on short segments across which the speech signal is assumed to be stationary. Short-term spectral measurements are typically carried out over the range of 10-30 ms frame size and shift [5, 9]. The blocked frames are Hamming windowed. This helps to reduce the edge effect while taking the DFT on the signal.

(iii) DFT spectrum: Each windowed frame is converted into magnitude spectrum by applying DFT.

$$X(k) = \sum_{n=0}^{N-1} x(n) e^{\frac{-j2\pi nk}{N}}; \qquad 0 \leq k \leq N-1 \tag{3.2}$$

where N is the number of points used to compute the DFT.

(iv) Mel-spectrum: This can be computed by passing the Fourier transformed signal through a set of band-pass filters known as mel-filter bank. A mel is a unit of perceived speech frequency or a unit of tone. The mel scale is therefore a mapping between the physical frequency scale (Hz) and the perceived frequency scale (Mels). The approximation of mel from physical frequency can be expressed as [5]:

$$f_{mel} = 2595 \log_{10}\left(1 + \frac{f}{700}\right) \tag{3.3}$$

where f denotes the physical frequency and f_{mel} denotes the perceived frequency.

The mel spectrum values or mel frequency coefficients of the magnitude spectrum $X(k)$ is computed by multiplying the magnitude spectrum by each of the triangular mel weighting filters.

$$S(m) = \sum_{k=0}^{N-1} |X(k)|^2 H_m(k); \qquad 0 \leq m \leq M-1 \tag{3.4}$$

where M is total number of triangular mel weighting filters.

(v) Inverse Discrete Cosine Transform (IDCT): The log operation is performed on the mel frequency coefficients. The IDCT is then applied to obtain cepstral coefficients. This results in a signal in the cepstral domain. MFCC is computed as :

$$c(n) = \sum_{m=0}^{M-1} \log_{10}(S(m)) \cos\left(\frac{\pi n(m-0.5)}{M}\right) \qquad n = 0, 1, 2, ..., C-1 \tag{3.5}$$

where $c(n)$ are the cepstral coefficients and C is the number of MFCCs. The zeroth coefficient is often excluded since it represents the average log-energy of the input signal, which only carries little speaker-specific information.

We consider only the first 13 dimensional feature vectors excluding the coefficient c_o computed using 35 filters in the filter bank. Cepstral Mean Subtraction (CMS) is applied to the MFCC to remove linear channel effect. Silence and low-energy speech parts are removed using an energy-based Voice Activity Detection (VAD) technique [5]. The threshold we used is 0.1 times the average frame energy for selection of speech frames. In conventional speech processing systems using SFSR, feature vectors are extracted by analyzing the speech signal in frame size (S) of 20 ms and shift or hop (H) of 10 ms. Due to VAD, the actual number of feature vectors or frames (N_{act}) vary from speaker to speaker for the same amount of speech data D and is given by

$$N_{act} = D \times \left(\frac{1-S}{H}\right) + 1 - N_{VAD} \tag{3.6}$$

where N_{VAD} is the number of frames removed due to voice activity detection technique. In case of MFS, MFCC feature vectors are computed for each frame of size $S = \{12, 14, 16, 18, 20\}$ ms

with shift of 10 ms. The actual number of frames of speaker can be obtained by

$$N_{act} = \sum_{i=1}^{5} D \times \left(\frac{1 - S_i}{H} \right) + 1 - N_{VAD} \tag{3.7}$$

To illustrate the distribution of feature vectors in the feature space for SFSR and MFS analysis techniques, we have taken 100 ms speech data of a speaker, computed only first four MFCC i.e., c_0, c_1, c_2, and c_3 and plotted excluding c_0 in Figure 3.2. Figure 3.2(a) shows SFSR based feature extraction and Figure 3.2(b) shows MFS based feature extraction. From these two figures it can be observed that MFS analysis results in more number of feature vectors for the same speech data. Further, presence of feature vectors at different places other than the SFSR demonstrate the manifestation of different spectral information.

To understand still better, we have computed the average number of frames (μ) for 30 speakers, each having 3 sec data using SFSR and MFS analysis and is shown in Table 3.1. The μ for MFS is 1005. This value is significantly high compared to the μ for SFSR which is 203. The more number of feature vectors may therefore help in mitigating the sparseness in the distribution of feature vectors during training relatively and provide more number of frames for testing the speaker reliably. As will be demonstrated later through different speaker recognition studies, this is indeed the case and hence the usefulness of MFS analysis for speaker recognition under limited data condition.

3.2.2 MFR Analysis

In MFR analysis, the same speech data is analyzed using different frame shifts (rates) with constant frame size. The speaking rate as well as pitch are different for each speaker. This is because speaking rate is a behavioral aspect of speaker information, which depends on how the speaker is habituated to produce speech. The pitch is an attribute of excitation source endowed with the speaker. Also these two information vary for the same speaker depending on the contextual information during speech production. Since speaking rate and pitch are different, rate of change of spectral information will be different. Thus we may benefit by using different frame shifts. Therefore by analyzing same speech data at different frame rates,

set of speech samples involved in the analysis of speech are different at each rate and hence feature vectors representing the vocal tract information may be different. Thus MFR analysis is effectively a multi-shifting technique. Accordingly, spectral resolution will remain same, but there will be new set of speech samples for each shift. For instance, 20 ms frame size with shift of 2 ms will have different set of speech samples compared to 20 ms frame size with shift of 4.5 ms. The new set of speech samples leads to considerably different magnitude spectra and hence feature vectors.

To analyze the behavior of MFR technique, features are extracted with fixed frame of size 20 ms and for different frame shifts of $H = \{2, 4.5, 6.5, 8.5, 10.5\}$ ms. The 13 dimensional MFCC vector is computed for each case. The actual number of frames of speaker can be computed by

$$N_{act} = \sum_{j=1}^{5} D \times \left(\frac{1-S}{H_j}\right) + 1 - N_{VAD} \tag{3.8}$$

As demonstrated in MFS case, to illustrate distribution of feature vectors in the feature space for the MFR analysis technique, we have taken 100 ms speech data of a speaker, computed only first four MFCC i.e., c_0, c_1, c_2, and c_3 and plotted excluding c_0 in Figure 3.2 (d). Comparing Figure 3.2(d) with Figure 3.2(a) it can be stated that MFR analysis provides significantly more number of feature vectors. Further, presence of feature vectors at different places other than SFSR demonstrate the manifestation of different spectral information. This may help in relatively mitigating the problem due to limited data during training and testing. These aspects are also observed for 30 speakers by computing μ and is shown in Table 3.1. The μ for MFR is 2263. This value is significantly high compared to the μ for SFSR which is 203. Further, it can be observed by comparing Figure 3.2(d) with Figure 3.2(b) that distribution of feature vectors in the feature space is different for MFR and MFS. This may be exploited for combining feature vectors from them to further improve the speaker recognition performance.

3.2.3 MFSR Analysis

In MFSR analysis, same speech data is analyzed using both MFS and MFR analysis techniques. Since MFS is multi-resolution technique and MFR is multi-shifting technique, the

magnitude spectra manifested in each case may be different and hence the resulting feature vectors. Due to this, as it can observed from Figure 3.2(b) and Figure 3.2(d), the distribution of feature vectors is different in each case. Therefore MFS and MFR may be exploited for obtaining combination method termed as MFSR. This combination may lead to further increase in the number of different feature vectors and hence improved performance compared to MFS or MFR alone. Figure 3.2(f) shows that MFSR analysis technique has more number of feature vectors compared to SFSR, MFS and MFR. Further, since they are obtained by MFS and MFR, the distribution of feature vectors may have better speaker-specific information. In the present work feature vectors are extracted using different frame sizes of $S = \{12, 14, 16, 18, 20\}$ ms and each for five different frame shifts of $H = \{2, 4.5, 6.5, 8.5, 10.5\}$ ms. The actual number of frames of a speaker is give by

$$N_{act} = \sum_{i=1}^{5} \sum_{j=1}^{5} D \times \left(\frac{1 - S_i}{H_j} \right) + 1 - N_{VAD} \tag{3.9}$$

It should be noted that mere increase in the number of feature vectors may not ensure better manifestation of speaker information as achieved in the case of MFSR. To show this pictorially, features are extracted for 20 ms frame size and shift of 10, 1 and 0.125 ms. Figure 3.2(a), (c) and (e) show distribution of feature vectors in the feature space for 20 ms frame size and shift of 10, 1 and 0.125 ms, respectively. As it can be observed, number of feature vectors obtained even using 20 ms frame size and 1 ms frame shift appears to be less than the MFSR. The same is also observed for 30 speakers by computing μ and is given in Table 3.1. The μ for 20 ms frame size and 1 ms frame shift is 2025. This value is less than the μ for MFSR which is 11429.

Another important observation that we made in Figure 3.2(e) and Figure 3.2(f) is that the number of feature vectors obtained using 20 ms frame size and 0.125 ms frame shift are more than the MFSR. This is also observed for 30 speakers by computing μ and is given in Table 3.1. The μ for 20 ms frame size and 0.125 ms frame shift is 16204. This value is significantly high compared to the value of the MFSR. Though the value is high, the speaker information manifested in them may not be same as that of the MFSR technique. This is mainly because the feature vectors generated appears to be obtained by mere interpolation. This aspect is also

demonstrated experimentally later. In order to verify the credibility of the proposed methods, we have taken another speaker of 100 ms speech data and plotted in Figure 3.3. The Figure 3.3 resembles Figure 3.2 and hence the motivation for the study.

Table 3.1: Comparison of average number of frames (μ) using different analysis techniques for the first 30 speakers taken from the YOHO database, *each having 3 sec training data*

Speech analysis	Frame size (ms)	Frame shift (ms)	μ
SFSR	20	10	203
MFS	12, 14, 16, 18, 20	10	1005
MFR	20	2, 4.5, 6.5, 8.5, 10.5	2263
SFSR	20	1	2025
SFSR	20	0.125	16204
MFSR	12, 14, 16, 18, 20	2, 4.5, 6.5, 8.5, 10.5	11429

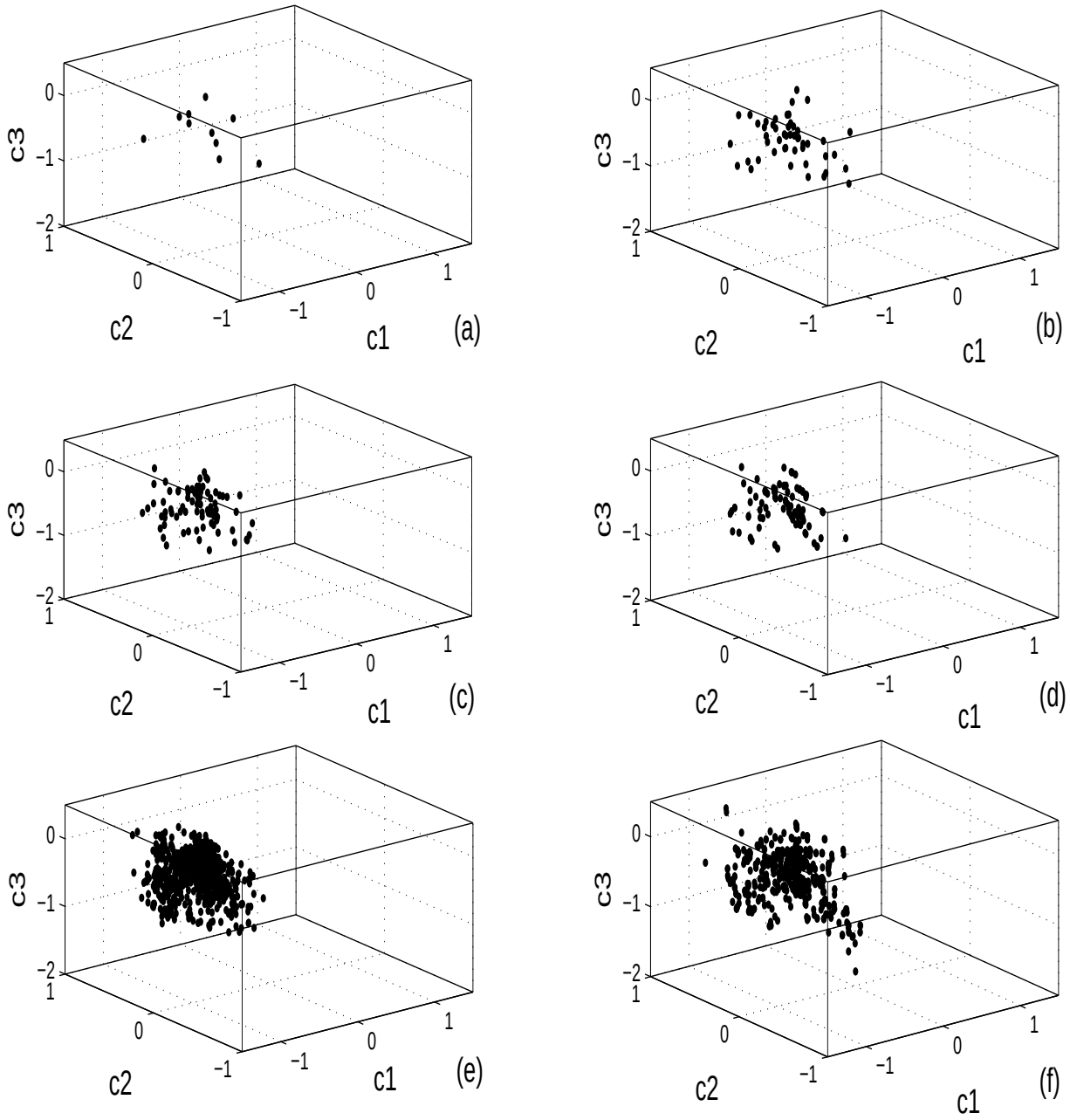

Figure 3.2: Features of a speaker for 100 ms speech data: (a) Features extracted for 20 ms frame size and 10 ms frame shift (b) MFS based feature vectors (c) Features extracted for 20 ms frame size and 1 ms frame shift (d) MFR based feature vectors (e) Features extracted for 20 ms frame size and 0.125 ms frame shift (f) MFSR based feature vectors.

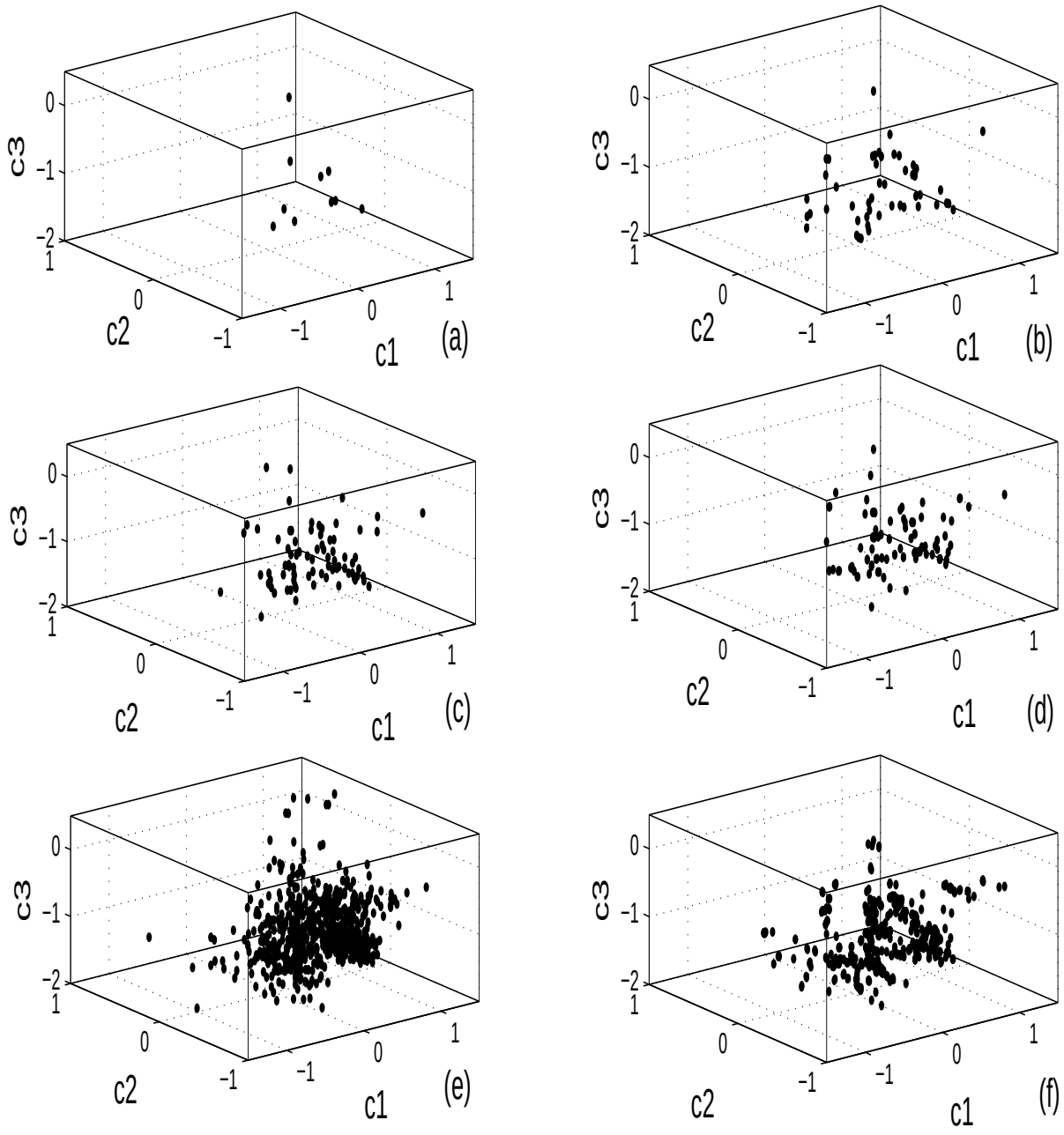

Figure 3.3: Features of another speaker for 100 ms speech data: (a) Features extracted for 20 ms frame size and 10 ms frame shift (b) MFS based feature vectors (c) Features extracted for 20 ms frame size and 1 ms frame shift (d) MFR based feature vectors (e) Features extracted for 20 ms frame size and 0.125 ms frame shift (f) MFSR based feature vectors.

3.3 Limited Data Speaker Recognition Studies using MFSR Analysis

3.3.1 Speech Database

In this work to evaluate the performance of the speaker recognition system, the YOHO [119] and the TIMIT [120] databases are used. The YOHO database consists of speech data from 138 speakers (108 male and 30 female). Each speaker has four training sessions with 24 speech files per session and ten test sessions with four speech files per session. Therefore, the training data for each speaker includes 96 speech files, each of about 3 sec duration. The testing data for each speaker includes 40 speech files, each of about 3 sec duration. The speech files are of type *combination lock phrases* (e.g. 36-24-36). The speech data is collected over telephone handsets, sampled at 8 kHz and stored with 16 bits/sample resolution. Since the database is not meant for limited data condition, we have taken one, two, four and eight speech files of each speaker to create the database.

The TIMIT database consists of speech data from 462 speakers (314 male and 131 female) in the training set and 168 (112 male and 56 female) speakers in the test set of total 630 speakers. The speech data is collected over microphone, sampled at 16 kHz and stored with 16 bits/sample resolution. Since most of the speech information is present up to 4 kHz, the speech database is resampled to 8 kHz. The speech data for each speaker includes 10 speech files, each of about 3 sec duration. The speech files are of *continuous speech* type (e.g. she had your dark suit in greasy wash water all year). In this work, we have used one set of first 30 speakers and another set of first 138 speakers from the test set of the TIMIT database. The first 5 speech files of each speaker are used for training and the remaining for testing. This database is also not meant for limited data condition and hence we have taken one, two, four and five speech files of each speaker to create the database.

The initial studies are conducted using the training data and test data of one file (3 sec) from each of the first 30 speakers of the YOHO database. These studies are later extended to the data of all the 138 speakers from the YOHO database and to the data of first 30 and first 138

speakers from the test set of the TIMIT database. The same databases and the experimental setups are used in the future works of this thesis.

3.3.2 Speaker Modelling and Testing

Although the GMM-UBM is widely used modelling technique, in this study we use VQ as a modelling technique due to the following reasons: VQ is simple and easy to implement and also VQ based method is robust with respect to utterance variations even when only a short utterance is available [93, 98]. The VQ is simple technique in the sense that only mean is estimated from the feature vectors for each cluster and reliable value for the same may be achieved using small number of feature vectors. VQ involves finding a subset of feature vectors termed as *codevectors* from the whole set, which can act as representative vectors. The set of codevectors is known as *codebook*. The objective function of VQ and steps involved in designing the objective function are as follows:

Let x_i are the training vectors of size n and y_j are the codebook vectors (cluster centers) of size M. The objective function J is given by

$$J = \sum_{j=1}^{c} \sum_{i=1}^{n} \|x_i - y_j\|^2 \tag{3.10}$$

Steps required to achieve the objective function are as follows [23]:

(i) Design a 1-vector codebook by computing the mean of x_i

(ii) Double the size of the codebook by splitting each current codebook according to the equations $y_c^+ = y_c\,(1+\varepsilon)$ and $y_c^- = y_c\,(1-\varepsilon)$, where ε is a splitting parameter (typically in the range $0.01 \leq \varepsilon \leq 0.05$).

(iii) Use the *K-means* iterative algorithm to get the best set of centroids for the split codebook.

(iv) Iterate the steps (ii) and (iii) until codebook of size c is designed.

In [76], it is shown that larger codebook size and sufficient testing data yields good recognition performance. Therefore, experiments are conducted with different codebook size N,

to verify recognition performance for different amounts of data. Moreover, the selection of codebook size largely depends on the amount of training data, but we still conducted the experiments for different codebook sizes to observe the significance of codebook size as data size changes. The codebooks of different sizes are built using binary split and *k-means* clustering procedures [75]. Since 1 sec and 2 sec data sizes are too small to conduct the speaker recognition studies, we start from $3 * 2^n$ sec data and subsequently vary according to the value of n, where $n = 0, 1, 2, 3, 4, ...$

During testing, we have followed two strategies namely: 1) Majority voting 2) Minimum average distance. In both strategies, each feature vector of the test speech data is compared with the codebook vectors of each speaker using Euclidean distance computation. In the first approach, for each frame minimum distance is noted for each codebook and the speaker of the codebook who gives minimum distance is the winner of a frame and is assigned to that speaker. The speaker with maximum number of assignments is recognized as the final speaker of the test speech data. In the latter approach, for each frame accumulate the minimum distance for each codebook and the speaker of the codebook who gives the minimum average distance across all frames is recognized as the speaker of the test speech data.

3.3.3 Speaker Recognition using SFSR, MFS, MFR and MFSR Analysis

Speech analyzed using a fixed frame of size 20 ms with a shift 10 ms is termed as SFSR. On the other hand, speech is analyzed using MFS by varying the frame size and keeping frame shift constant or MFR by varying frame shift and keeping frame size constant or MFS and MFR combined known as MFSR. For the chosen analysis, MFCC feature vectors are extracted. The ensemble of feature vectors are used for training and testing the models for different codebook sizes and also for different amounts of data. The steps involved in the SFSR and proposed MFSR based speaker recognition system for limited data condition are shown as block diagram in Figure 3.4. *In the future discussion while comparing with SFSR, unless specified MFSR signifies MFS, MFR and MFSR cases.*

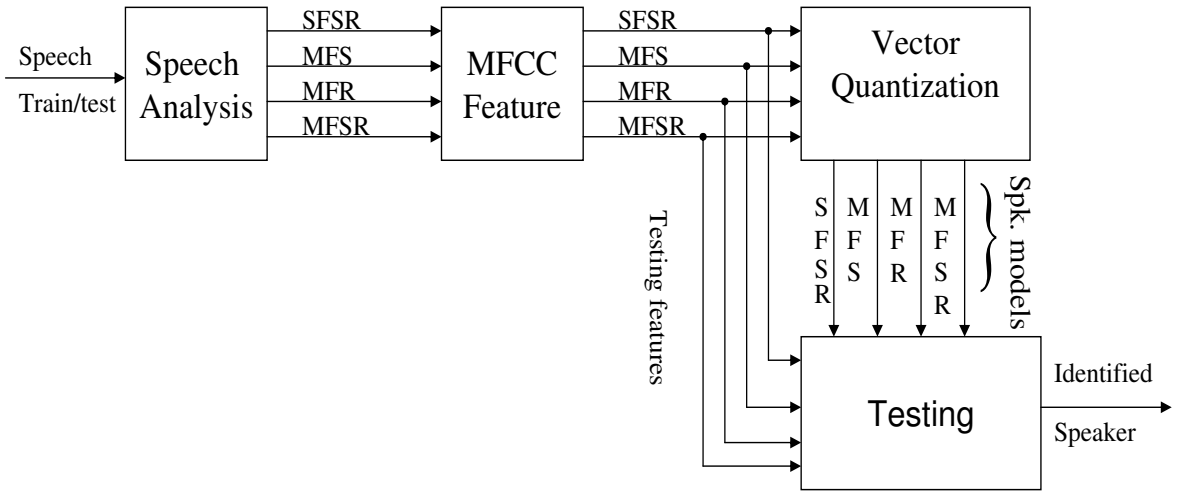

Figure 3.4: The SFSR and proposed MFSR based speaker recognition system for limited data condition

3.4 Experimental Results and Discussions

In this section, we present the results of different experiments performed. The experiments are carried out under the following *three* conditions to verify the speaker recognition performance:

- Limited training and sufficient testing data
- Sufficient training and limited testing data
- Limited training and testing data

Before we discuss the above experiments, let us first demonstrate the following: 1) Effectiveness of testing strategies under limited data condition. 2) Comparison of MFSR testing with SFSR. In both the studies, speaker recognition experiment is conducted for 30 speakers, each having 3 sec training and testing data. *1) Effectiveness of testing strategies*:- As we have explained in Section 3.3.2, the experiments are conducted accordingly using both the strategies for different codebook sizes and the results are shown in Table 3.2. The following observations can be made: i) The majority voting strategy gives the highest performance of 70% for codebook of size 64. ii) Averaging the distances with the constraint of few feature vectors

may not provide reliable decision and hence leads to poor performance. Therefore, in all our experimental studies we use majority voting as the testing strategy.

Table 3.2: Speaker recognition performance (%) for the first 30 speakers taken from the YOHO database, each having *3 sec training and testing data* for different testing strategy.

Testing strategy	Codebook size			
	16	32	64	128
Majority Voting	63.33	66.76	**70**	60
Minimum avg. distance	**46.76**	36.76	43.33	**46.76**

2) Comparison of MFSR testing with SFSR:- In order to exploit proposed MFSR analysis techniques during testing, the strategy employed is shown in Table 3.3. The entries in the table may be viewed in the following way: The SFSR performance for frame size 20 ms and frame shift 10 ms is given by first row and last column first entry. The MFS performance is given by the last column and all rows and the maximum of all these entries is chosen as MFS performance. The MFR performance is given by the first row and all column entries and the maximum of all these entries is chosen as MFR performance. Similarly, the highest performance obtained for all the combinations of frame size and shift of MFS and MFR is chosen as MFSR performance. It should be noted that this testing strategy may be treated as the advantage of using MFSR and may not be available in fixed frame sizes and shifts like 10, 1 and 0.125 ms. Thus MFSR yields highest performance of 90% (27 speakers are identified out of 30) for the frame size 18 ms and shift 2 ms for codebook of size 256. The results are shown in Table 3.4. The MFSR yields better recognition performance in limited data condition compared to the features extracted for 20 ms frame size and shift of 10, 1 and 0.125 ms.

3.4.1 Limited Training and Sufficient Testing Data

The objective behind this experiment is to verify the effect of limited data on training. We have trained the speaker models for different amounts of training data using SFSR and MFSR analysis methods. The trained models are tested against complete testing data of YOHO database which is approximately 80 sec for each speaker after removing non-speech regions.

Table 3.3: Number of speaker identified by the proposed methods using *3 sec training and testing data* for codebook of size 256 for the first 30 speakers taken from the YOHO database.

Frame size (ms)	Frame shift (ms)				
	2	4.5	6.5	8.5	10
20	24	23	23	23	22
18	**27**	25	26	23	24
16	23	24	24	24	24
14	24	24	24	25	23
12	24	23	25	23	23

Table 3.4: Comparison of speaker recognition performance for *20 ms* frame size and shift of *10, 1, and 0.125 ms* with *MFSR* for the first 30 speakers taken from the YOHO database, each having *3 sec training and testing data*.

Speech Analysis	Frame size (ms)	Frame shift (ms)	Codebook size	Performance (%)
SFSR	20	10	32	66.76
SFSR	20	1	128	86.76
SFSR	20	0.125	256	86.76
MFSR	12, 14, 16, 18, 20	2, 4.5, 6.5, 8.5, 10.5	256	**90**

The experimental results for 30 speakers are shown in Figure 3.5. We compared SFSR to MFSR analysis methods for their performance on limited data. We can infer that applying MFSR methods for feature extraction improve recognition performance. The observations also show that the performance of speaker recognition system depends on the amount of data available for training [22]. If the training data available is small, the distribution of feature vectors in the feature space is sparse, and if we use SFSR, recognition performance is poor during testing. Hence, an attempt is made here to increase speaker recognition performance for the same amount of training data using MFS, MFR and MFSR. Due to increase in the number of speaker-specific features in the feature space, better representation of speaker may be possible, and hence these three methods perform better compared to SFSR. Further, in the MFSR methods there is substantial improvement in the recognition performance from one method to another. This is because the analysis technique adopted in each case is different. Hence MFSR analysis for training improves the recognition performance.

Figure 3.5 demonstrates that increase in training data up to 12 sec shows significant improvement in recognition performance using MFSR methods. Beyond this, gain achieved seems

to be less. Further, limited data condition that is of present interest signifies typical data size of less than 15 s. Also, from the computation point of view, for whole database of 138 speakers the experiment was carried only up to 12 sec and the results are shown in Figure 3.6. It shows that for large population also the MFSR techniques give better result than the SFSR. The performance of an approximately 10% difference can be seen from SFSR to MFSR at 3 sec and 6 sec training data. As the data size increases the performance approaches near SFSR. Therefore, from these results we can understand that MFSR is indeed required to improve speaker recognition performance when training data is small.

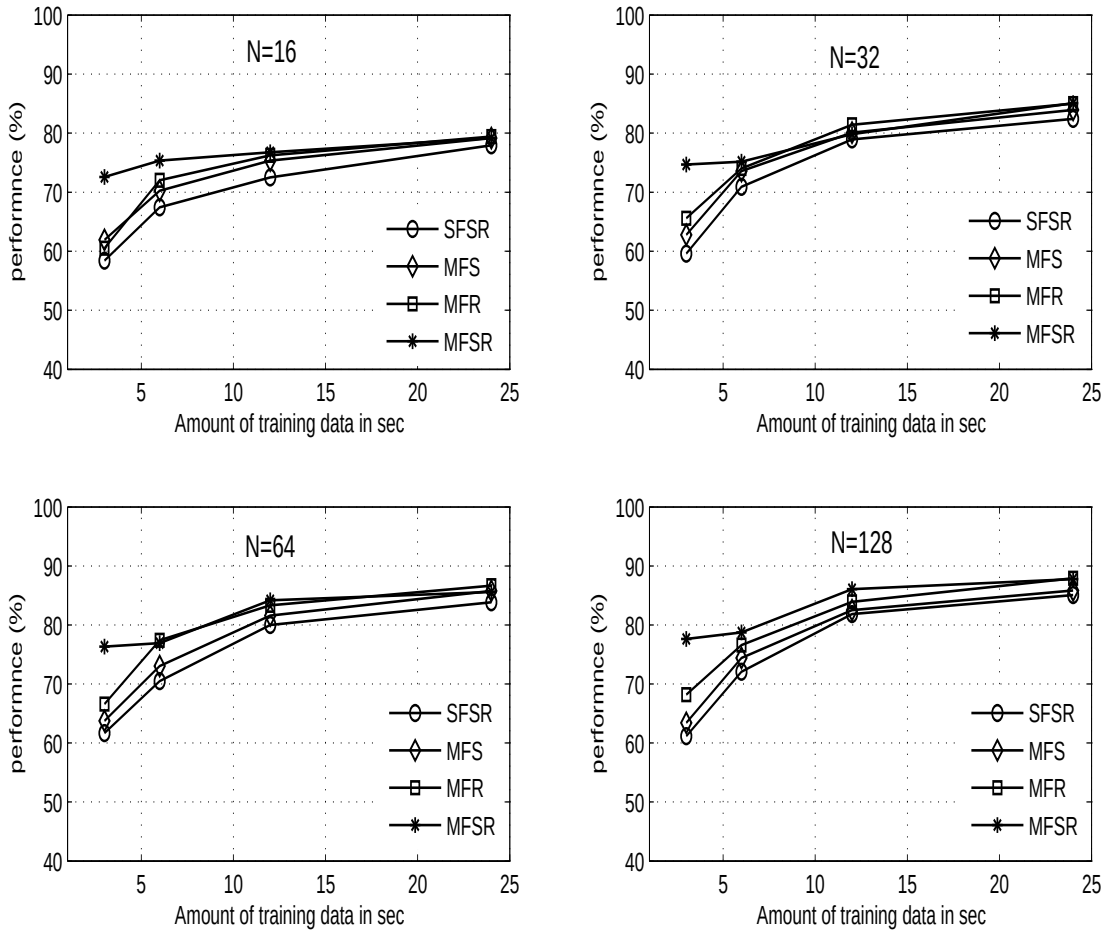

Figure 3.5: Performance of the speaker recognition system based on SFSR and MFSR analysis techniques for different sizes of training data for the first 30 speakers taken from the YOHO database.

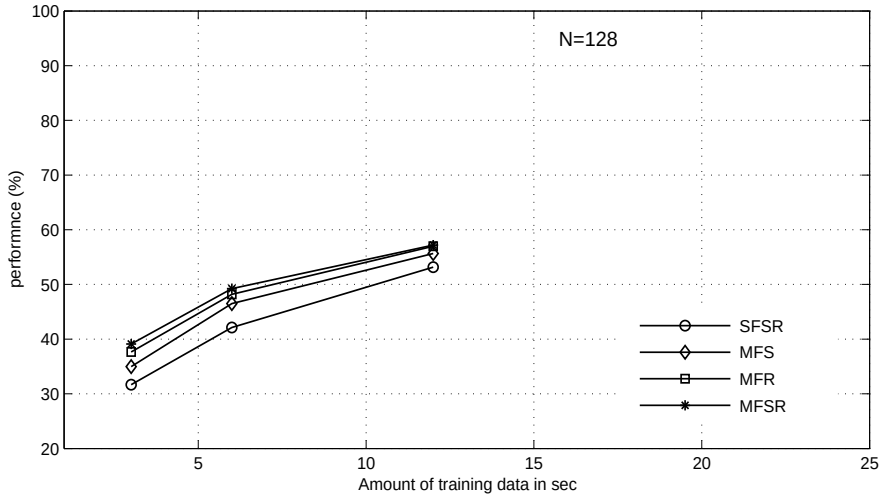

Figure 3.6: Performance of the speaker recognition system based on SFSR and MFSR analysis techniques for 138 speakers taken from the YOHO database for different sizes of training data

3.4.2 Sufficient Training and Limited Testing Data

The objective of this experiment is to verify the effect of limited data on testing. It is evident from Figure 3.5 that about 24 sec of training data may be sufficient to capture the speaker-specific information. Hence, all the models are trained with SFSR based 24 sec of speaker data for this study to examine the effect of SFSR and MFSR methods of analysis on limited testing data. Further, during testing we have taken different amounts of test speech data and applied SFSR and MFSR analysis techniques. The results for 30 speakers are shown in Figure 3.7. From the figure, it can be observed that MFSR methods outperform SFSR and hence MFSR for testing improves recognition performance. This is because MFSR methods increase number of speaker-specific testing feature vectors, depending on the analysis technique adopted. This helps in reliable testing of each speaker. Hence, there is a substantial recognition improvement from one method to another and all the methods outperform the SFSR.

In order to verify recognition performance in this condition for the whole database, we observed from the Figure 3.7 that any increase in testing data size over 12 sec results in almost the same recognition performance in all the methods, and hence the experiment is carried out up to 12 sec for the whole database and the results are shown in Figure 3.8. From the figure it

is observed that though the models are well trained with sufficient data, during testing, MFSR methods give better performance compared to SFSR when testing data is less. The recognition performance of nearly 15% high is achieved for testing data less than 12 sec compared to SFSR. Hence, MFSR methods can also be used for large database to improve speaker recognition performance when testing data is small. From these studies we may conclude that, sufficient data during training of speaker models and MFSR analysis on the test data, results in higher recognition performance in limited data condition.

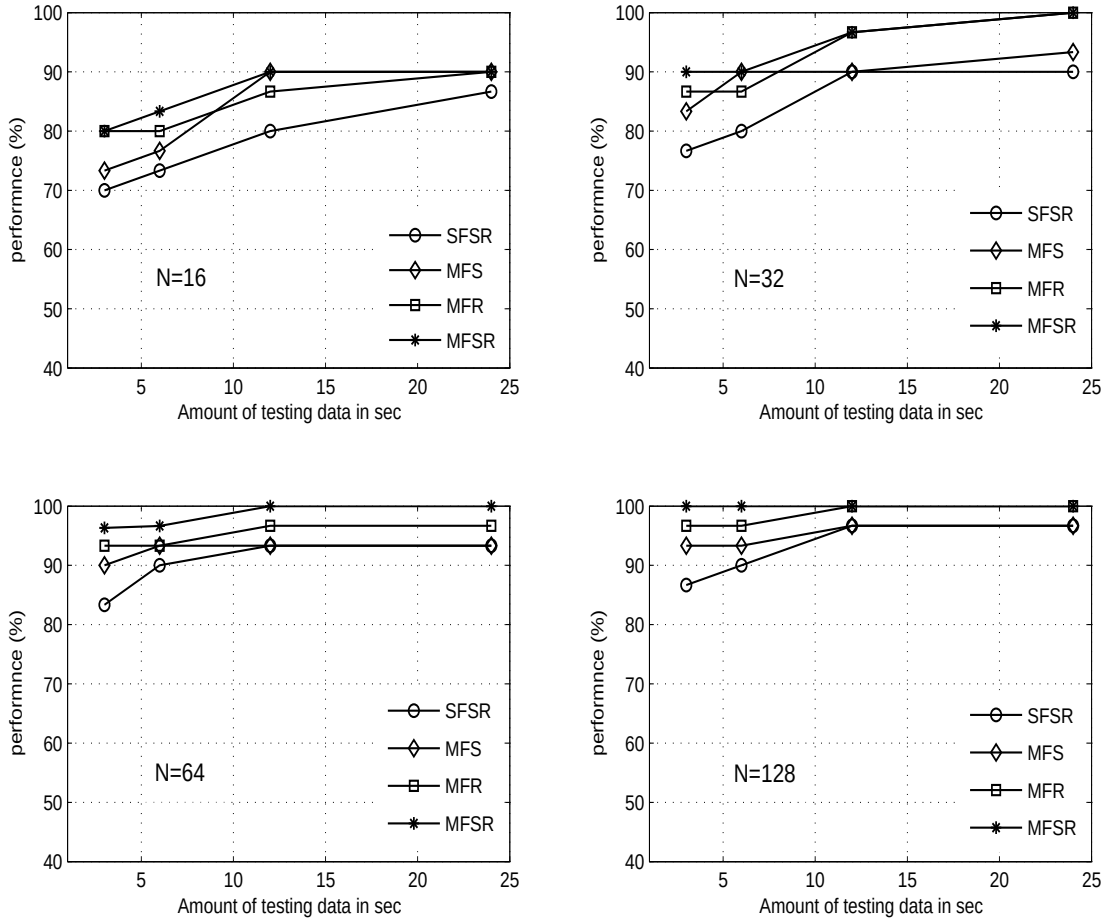

Figure 3.7: Performance of the speaker recognition system based on SFSR and MFSR analysis techniques for different sizes of testing data for the first 30 speakers taken from the YOHO database.

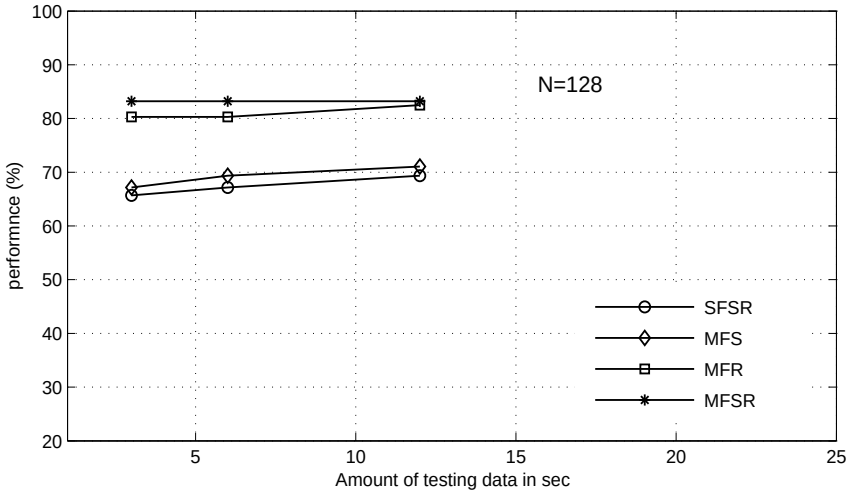

Figure 3.8: Performance of the speaker recognition system based on SFSR and MFSR analysis techniques for 138 speakers taken from the YOHO database for different sizes of testing data.

3.4.3 Limited Training and Test Data

The aforementioned two conditions require either sufficient training or testing data. But in practice data is limited both for training and testing. An attempt is made here to study the speaker recognition performance when both training and test data are limited. We have conducted four experiments to study the effectiveness of SFSR and MFSR analysis methods during training and testing. The experiments conducted are as follows:

- SFSR trained models with SFSR and MFSR methods of testing
- MFS trained models with SFSR and MFSR methods of testing
- MFR trained models with SFSR and MFSR methods of testing
- MFSR trained models with SFSR and MFSR methods of testing

In the first experiment, speaker models are trained using SFSR based analysis. The trained models are individually tested using SFSR and MFSR methods. The experimental results for 30 speakers are shown in Figure 3.9. From the figure it is observed that though the models are

poorly trained using SFSR analysis, during testing MFSR analysis significantly improves the performance. The recognition performance of 83.33% is achieved for 3 sec data using MFSR for codebook of size 128. The performance is higher than the performance of SFSR that provides 70% for codebook of size 64. The comparative differences in performance from SFSR to MFSR for other data sizes of 6 and 12 sec can also be seen in the figure. The figure also shows that the performances of all analysis methods approach nearer at 24 sec of data.

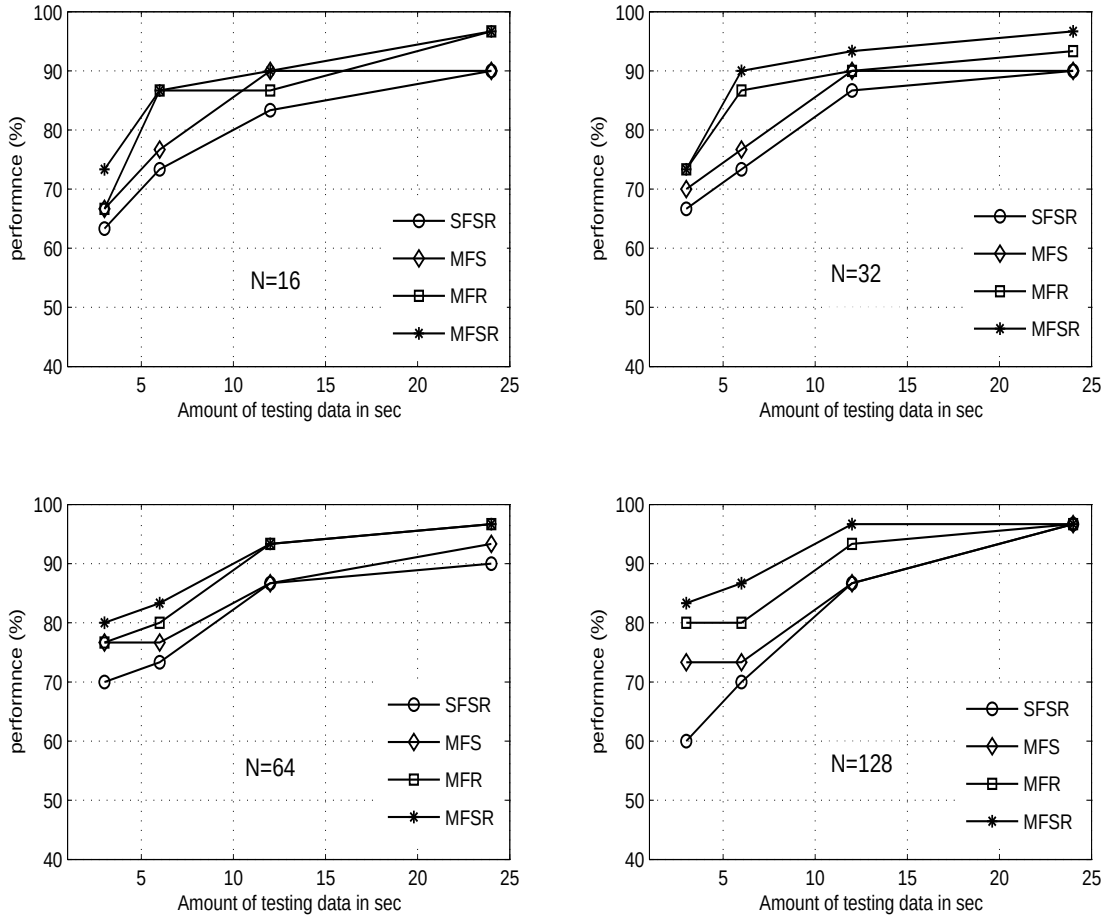

Figure 3.9: Performance of the speaker recognition system based on SFSR and MFSR analysis techniques for different sizes of testing data for the first 30 speakers taken from the YOHO database. SFSR trained model is used for testing.

The speaker models are trained using MFS based analysis technique in the second experiment. The trained models are tested using SFSR and MFSR method of analysis and the results are shown in Figure 3.10. It shows that the recognition performance of 83.33% is achieved for 3 sec data using MFSR for codebook of size 128. The performance is higher than the performance of SFSR that provides 66.76% for codebook of size 64. These results are almost same as that of the previous experiment. Further, we can infer that when both training and testing data are limited the MFS method alone does not improve the performance. For other data sizes of 6 and 12 sec also the performance is almost same as that of the previous experiment.

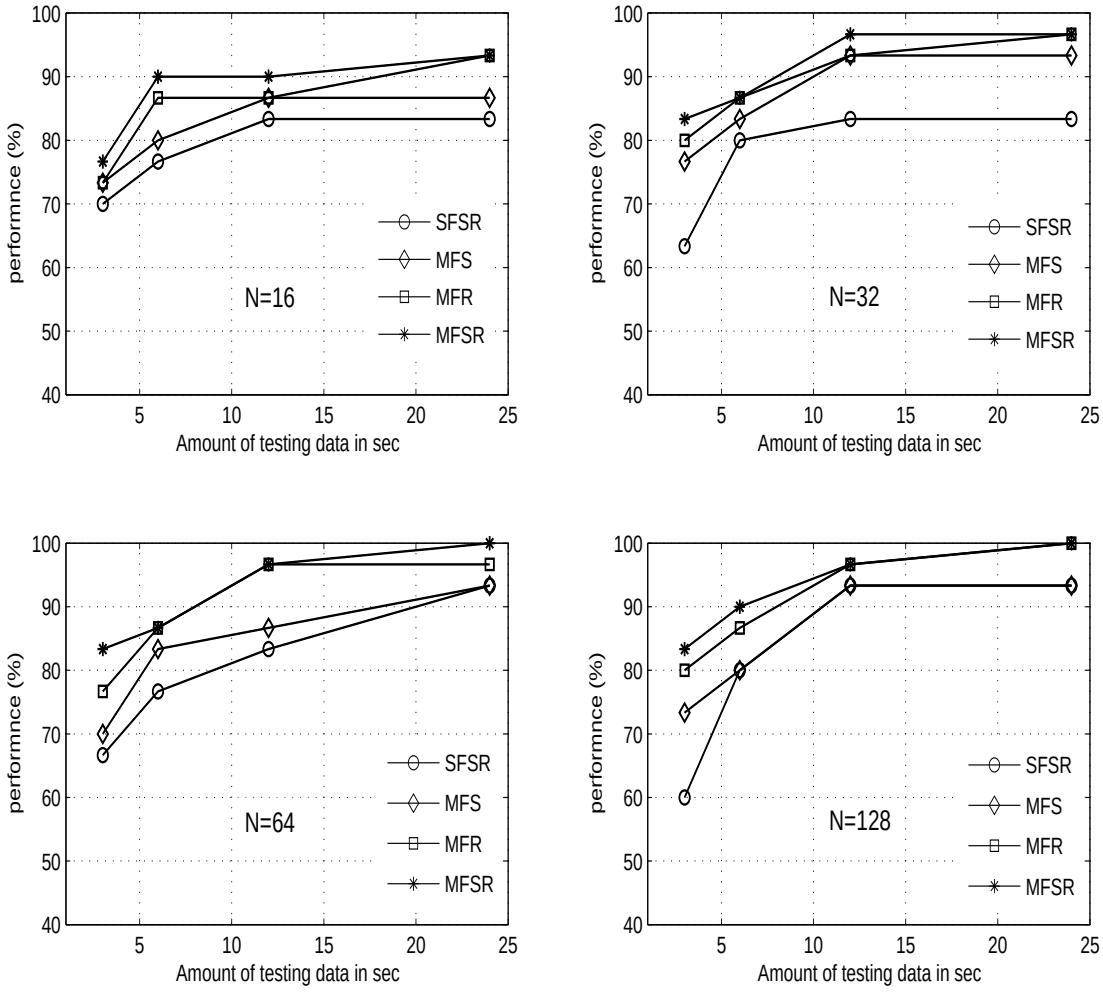

Figure 3.10: Performance of the speaker recognition system based on SFSR and MFSR analysis techniques for different sizes of testing data for 30 speakers taken from the YOHO database. MFS trained model is used for testing.

In the third experiment, speaker models are trained using MFR based analysis technique. The trained models are tested using SFSR and MFSR analysis methods. The experimental results for 30 speakers are shown in Figure 3.11. It shows that the recognition performance of 86.76% is achieved for 3 sec data for codebook of size 128. The performance is higher than the performance of SFSR that provides 83.33% for codebook of size 128 and the performance of previous experiment (83.33%). Further, it can be observed that MFR indeed improves the performance when training and testing data are limited. For other data sizes of 6 and 12 sec also we can see about 7% improvement in MFSR performance over SFSR in the figure.

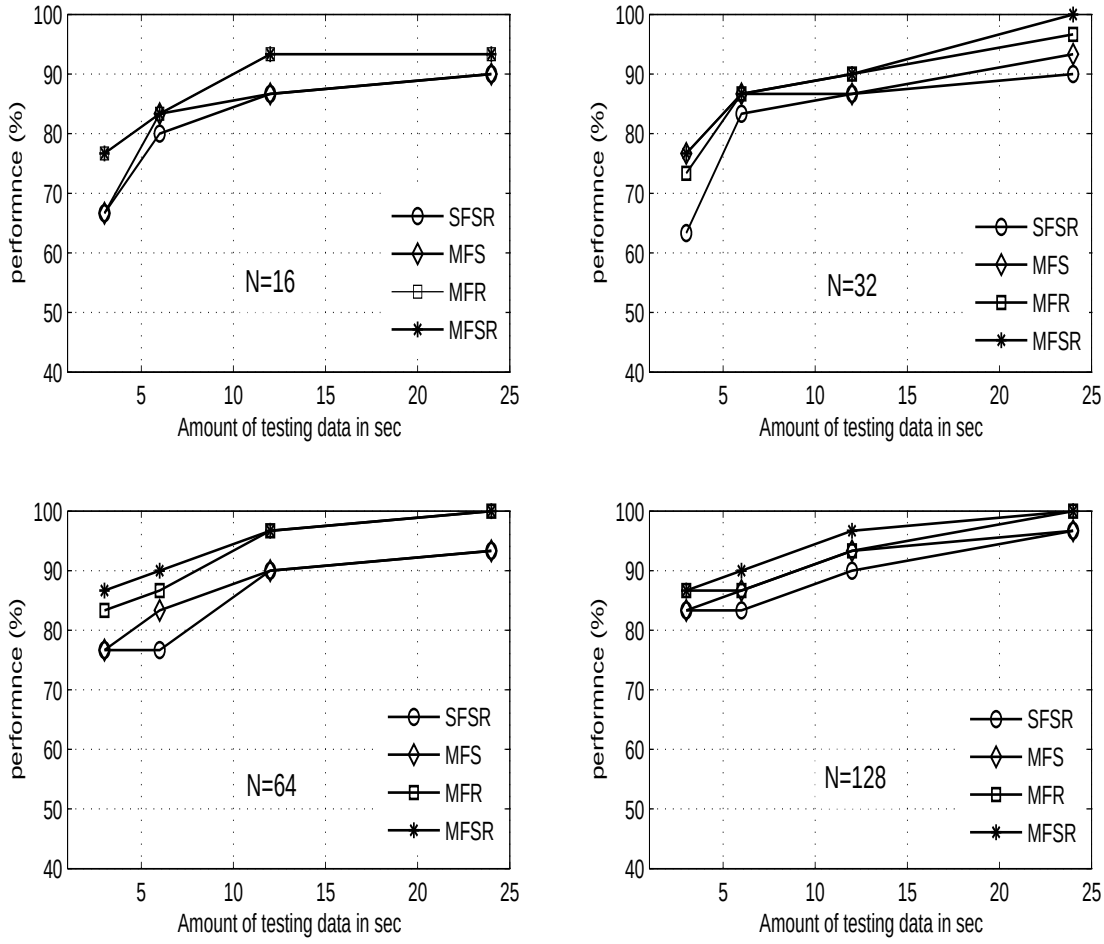

Figure 3.11: Performance of the speaker recognition system based on SFSR and MFSR analysis techniques for different sizes of testing data for 30 speakers taken from the YOHO database. MFR trained model is used for testing.

The above mentioned experiments improve the performance when testing is done with MFSR. In this experiment, speaker models are trained using MFSR based analysis technique and testing is done using SFSR and MFSR methods. The experimental results for 30 speakers are shown in Figure 3.12. The recognition performance of 90% is achieved for 3 sec training and testing data using MFSR for codebook of size 128. The performance is higher than the performance of SFSR that provides 80% for codebook of size 64 and the performance of previous experiment (86.76%). For other data sizes of 6 and 12 sec also MFSR analysis gives better results than the SFSR. Further, the results show that MFSR analysis at both training and testing improves the performance compared to using only MFSR at testing as in the previous experiments.

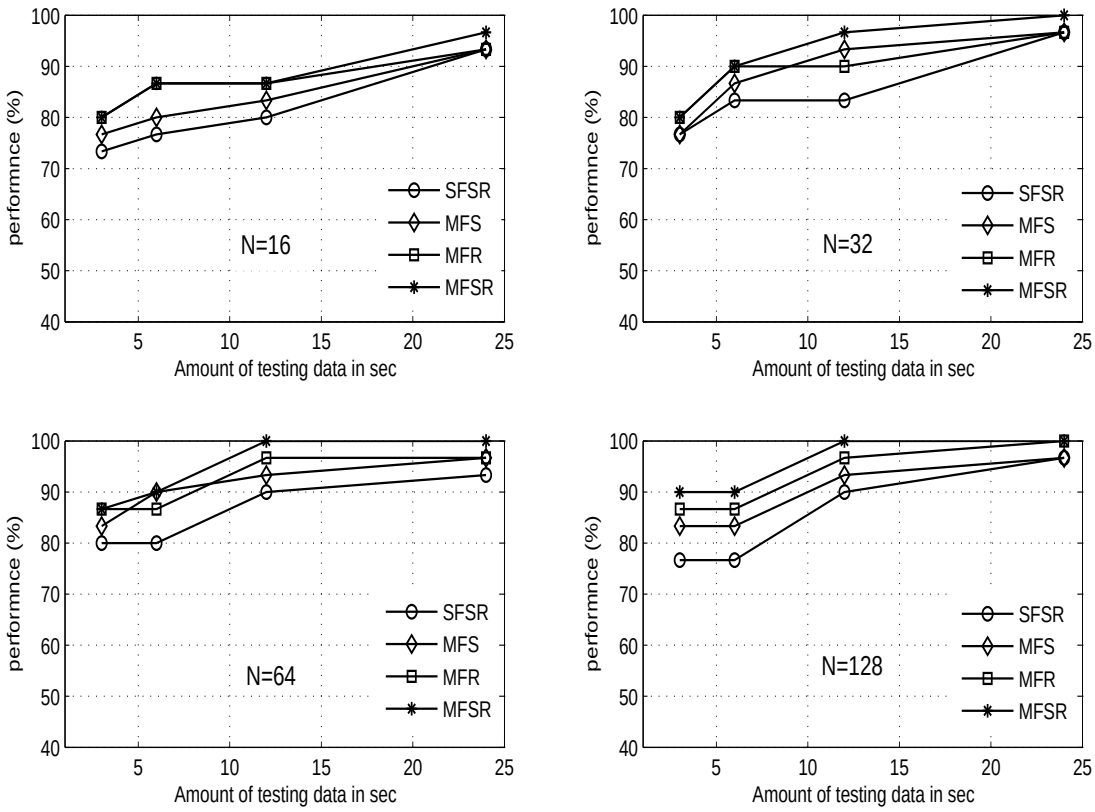

Figure 3.12: Performance of the speaker recognition system based on SFSR and MFSR analysis techniques for different sizes of testing data for the first 30 speakers taken from the YOHO database. MFSR trained model is used for testing.

Aforementioned all the experiments are conducted for the whole database of 138 speakers to verify recognition performance under limited data condition. The results are shown in Figure 3.13. The plots demonstrate that for large population also the MFSR analysis methods results in improved recognition performance compared to SFSR. Hence MFSR methods can also be used for large database having limited training and test data. From this study we can conclude that when both training and testing data is small, MFSR analysis techniques on training and testing data improve the recognition performance compared to SFSR.

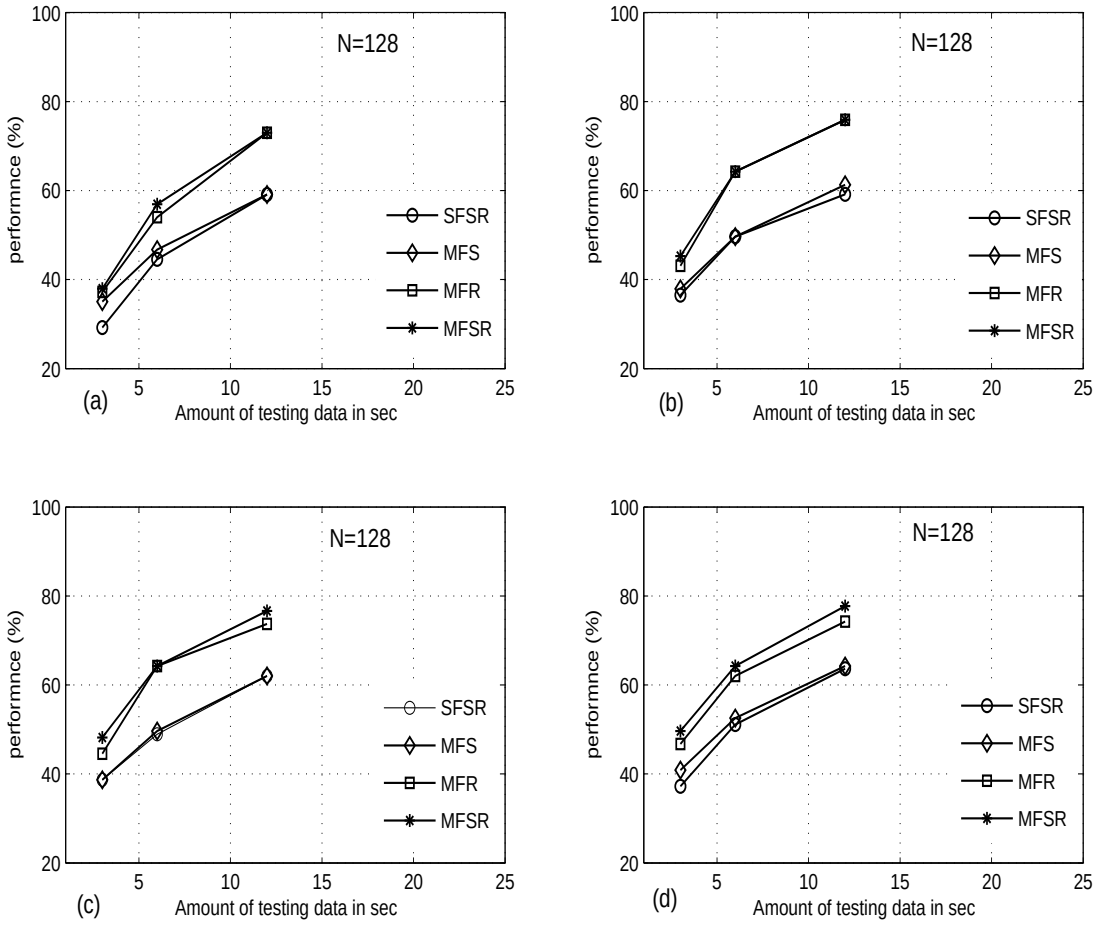

Figure 3.13: Performance of the speaker recognition system based on SFSR and MFSR analysis techniques for different sizes of testing data for 138 speakers. (a) SFSR trained model is used for testing (b) MFS trained model is used for testing (c) MFR trained model is used for testing and (d) MFSR trained model is used for testing.

To verify the effectiveness of the proposed MFSR analysis, we have conducted the experiments on the TIMIT database also. As we have already mentioned that in practice both training and testing data may be limited. Therefore, in this case SFSR and MFSR analysis are studied only when both limited training and testing data are limited. The experimental results are shown in Figure 3.14 and Figure 3.15 for a set of the first 30 and 138 speakers, respectively. The experimental results for the TIMIT database also resemble those for the YOHO database irrespective of speaker population and amount of data. Hence, we can suggest that MFSR analysis can be used for improving the speaker recognition performance when both training and testing data is limited.

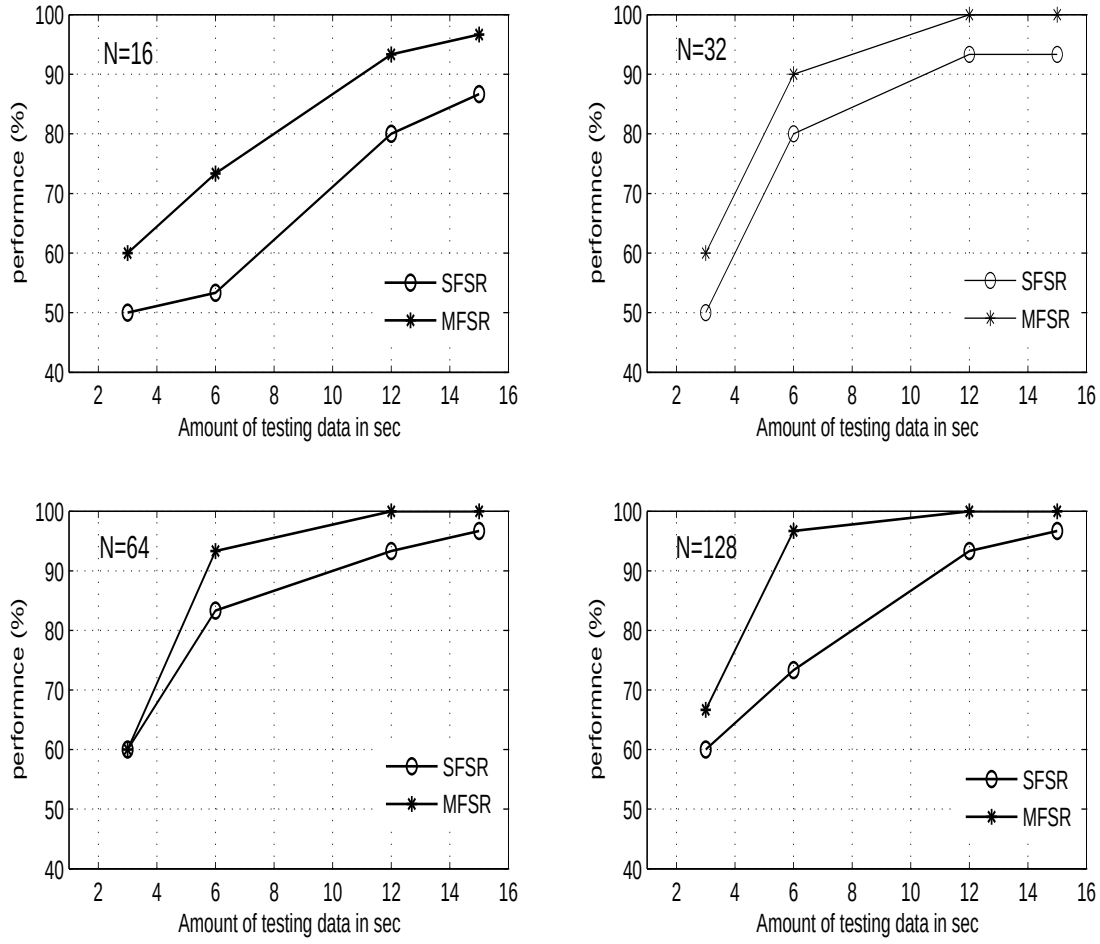

Figure 3.14: Performance of the speaker recognition system based on SFSR and MFSR analysis techniques for different sizes of testing data for the first 30 speakers taken from the TIMIT database.

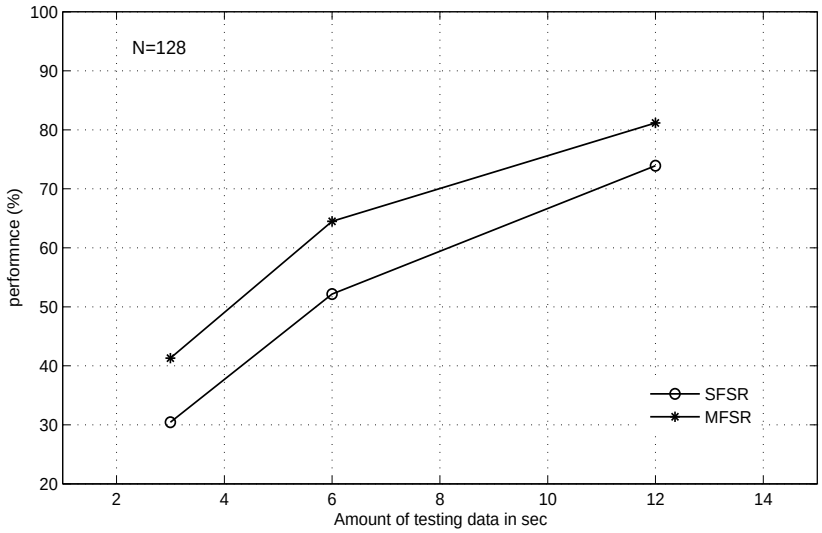

Figure 3.15: Performance of the speaker recognition system based on SFSR and MFSR analysis techniques for different sizes of testing data for the first 138 speakers taken from the TIMIT database.

3.5 Summary

In this chapter, we demonstrated the usefulness of MFS, MFR and MFSR techniques for speaker recognition under limited data condition. First, we discussed the influence of feature vector contribution by SFSR and MFS, MFR and MFSR methods. Then, we analyzed practically to verify the performance for different conditions. Experimental results show that SFSR method is unable to capture more speaker-specific information whereas MFS, MFR and MFSR methods does in all conditions. This shows that the recognition performance can be improved relatively by proper analysis technique. Further, the MFSR methods give better recognition performance in all the conditions compared to SFSR. Moreover, among different analysis methods MFSR yields highest recognition performance.

In the next chapter we discuss the performance of different feature extraction techniques and significance of combining evidences from different levels of speech for speaker recognition under limited condition.

4

Combination of Features for Limited Data Speaker Recognition

Contents

This chapter presents an experimental evaluation of the different speech feature extraction techniques for speaker recognition under limited data condition. Since the amount of data is small in limited data condition, any one feature extraction technique may not provide enough features for modelling and testing. In this work different feature extraction techniques like Mel Frequency Cepstral Coefficients (MFCC), Delta (Δ) MFCC, Delta-Delta ($\Delta\Delta$) MFCC, Linear Prediction Residual (LPR) and Linear Prediction Residual Phase (LPRP) are explored independently to know the level of speaker information present in them. These features are then combined to obtain better representation of speaker. As a result the combination of features MFCC, Δ, $\Delta\Delta$, LPR and LPRP provides better performance compared to the individual and also other combination of features.

4.1 Introduction

Speech data contains different levels of information that can be used to convey speaker information [1]. These can be obtained by using either high or low level features. The high level features include dialect, idiolect, accent, speaking style etc. Though these features aid to discriminate and recognize the speaker well, acquiring and using them is practically difficult. The disadvantages of high level features are 1) Require very large amount of speech data to acquire them 2) More intra-speaker variability. On the other hand, low level features like spectral amplitudes, pitch frequency, formant frequencies and bandwidths are easy to acquire and hence used in practice [8]. Therefore, state-of-the-art speaker recognition systems use speaker-specific information like vocal tract [23], excitation source [22] and suprasegmental feature like intonation, duration and accent [121] for speaker recognition. The problem in limited data condition is that the amount of available data is very small. This leads to poor recognition performance. Therefore, extraction of different levels of information is especially important in speaker recognition under limited data condition to obtain reliable performance. Moreover, it is necessary to have multiple evidences obtained using different levels of speech and combine them to improve the recognition performance. Since the amount of available speech data is limited, it is difficult to use suprasegmental features which represent the behavioral characteristics of a

speaker. Hence, we consider only vocal tract and excitation source information for evaluating speaker recognition performance under limited data condition.

Already, the combination of vocal tract and source information was extensively studied for speaker recognition and demonstrated that the combined system gives better performance compared to individual system [17, 22, 43]. However, these studies are not demonstrated for the system performance under limited data condition. In this work, we use MFCC, ΔMFCC, ΔΔMFCC, LPR and LPRP features to evaluate the performance under limited data condition and then combine them to improve the performance. Since each of these features offer different information, they may combine well to improve the performance [17]. Hence the motivation for the study. The rest of the chapter is organized as follows: In Section 4.2 we describe the speaker recognition studies using different feature extraction techniques. Section 4.3 presents the combination of features for speaker recognition under limited data condition. Finally, summary of the work presented in this chapter is given Section 4.4.

4.2 Limited Data Speaker Recognition Studies using Different Features

In this study also the YOHO [119] and the TIMIT [120] databases are used to evaluate the performance of different feature extraction techniques. The analysis technique we used is SFSR and the modelling technique used is VQ [76]. The steps involved in the combination of features for speaker recognition are shown as block diagram in Figure 4.1. The different feature extraction techniques used in this study are as follows:

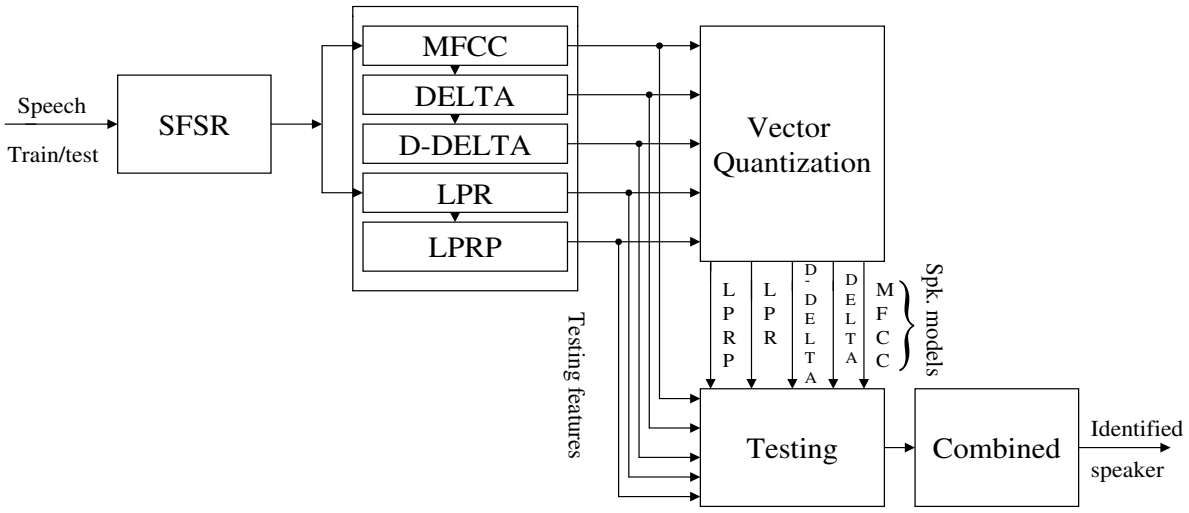

Figure 4.1: Block diagram of combination of features for speaker recognition under limited data condition.

4.2.1 Vocal Tract Features for Speaker Recognition

The dominant features representing predominantly vocal tract information for speaker recognition are MFCC [42] and LPCC [57, 122]. Though the MFCC and LPCC are used to extract the same vocal tract information, in practice these features differ in their performance due to different principle involved in extracting them [23]. That is, MFCC computation first apply DFT on each frame and then weights the DFT spectrum by a mel-scaled filter bank. The filter bank outputs are then converted to cepstral coefficients by applying the IDCT. In case of LPCC, first the LPC are obtained for each frame using Durbins-Recursive method and then these coefficients are converted to cepstral coefficients [23]. Therefore, we have conducted experiments to study the effectiveness of both these features.

4.2.1.1 Speaker Recognition using MFCC

We have already given the working principle of MFCC in Chapter 3 of Section 3.2.1. The SFSR analyzed MFCC experimental results for the 30 speakers of the YOHO database using one speech file (3 sec) for training and testing data and different codebook sizes are given in Section 3.4 of Chapter 3 (Table 3.2). In this chapter also for the comparison purpose, the same

results are given in Table 4.1. The highest performance of 70% is achieved using codebook of size 64. Although the MFCC technique is widely used and has proven to be effective in speaker recognition, it does not provide satisfactory performance in limited data condition. Therefore, we need to extract some other information present in the speech to improve the performance.

MFCC feature vectors that are being computed represent static properties of a given frame of speech. However, these feature vectors does not capture the transitional characteristics of the speech signal which also contains speaker-specific information [62, 123]. The transitional characteristics can be captured by computing the Δ and $\Delta\Delta$ coefficients which are obtained respectively from the MFCC and ΔMFCC by a first-order time-derivative. The commonly used equations to compute Δ and $\Delta\Delta$ coefficients are given by

$$\Delta c_m(n) = \frac{\sum_{i=-T}^{T} k_i c_m(n+i)}{\sum_{i=-T}^{T} |i|} \tag{4.1}$$

$$\Delta\Delta c_m(n) = \frac{\sum_{i=-T}^{T} k_i \Delta c_m(n+i)}{\sum_{i=-T}^{T} |i|} \tag{4.2}$$

where $c_m(n)$ denotes the m^{th} feature for the n^{th} time frame, k_i is the i^{th} weight and T is the number of successive frames. We have taken T as 2. In order to verify the effectiveness of these features, we have conducted two experiments: 1) Concatenated the derived cepstrals to MFCC which implies 26 and 39 dimensional feature vectors for ΔMFCC and $\Delta\Delta$MFCC, respectively [123]. 2) We first treat the derived cepstrals as individual features, conducted the experiments and finally combined them with MFCC [124]. In the combined system we used Linear Combination of Frame Ratio (LCFR) which can be described as follows:

Let $x_{i1}, x_{i2}, ..., x_{iN}$ be the normalized frame scores of the test data with respect to the total number of frames of a speaker for the different feature systems i and N is the number of speaker models. The frame scores are linearly added which results in $z_1, z_2, ..., z_N$, where $z_j = \sum_{i=1}^{3} x_{ij}$. The combination is done at the scoring level in which the highest performed codebook size of

each feature system is considered and their frame ratios are linearly combined.

The experimental results using the Δ and ΔΔMFCC features for the 30 speakers of the YOHO database using one speech file (3 sec) for training and testing data and different codebook sizes are also given in Table 4.1. The concatenated (MFCC + ΔMFCC) and (MFCC + ΔMFCC + ΔΔMFCC) features provide the highest performance of 70% for codebook of size 32 and 60% for both codebook of sizes 32 and 128, respectively. It can be observed that the concatenated features does not show improvement in the performance as compared to MFCC alone of 70% for codebook of size 64. This may be due to the limited data that may not represent substantially the feature vectors at high dimensionality feature space. Moreover, the number of features are very less due to limited data, when features are concatenated, this problem becomes severe and hence poor performance. To take care of this, first individually speaker-specific information is extracted from the spectral change and then combined.

The Table 4.1 also shows that the highest performance of 36.67% using ΔMFCC features and 33.33% using ΔΔMFCC features is achieved for the codebook size of 64. From these results we can understand that the dynamic features contain less information compared to the MFCC, but they may contain different speaker information. This may be exploited by combining evidences from Δ and ΔΔ with MFCC to obtain improved performance. The combined (MFCC, ΔMFCC and ΔΔMFCC) system gives 76.67% which is higher than the individual and the concatenated features. Therefore, in future in all the experimental studies we use combination of features.

Table 4.1: Speaker recognition performance (%) for the first 30 speakers of the YOHO database using *3 sec training and testing data* for MFCC feature and its derivatives.

Features	Dimensions	Codebook size			
		16	32	64	128
MFCC	13	63.33	66.76	**70**	60
MFCC + ΔMFCC	26	66.76	**70**	63.33	43.33
MFCC + ΔMFCC + ΔΔMFCC	39	43.33	**60**	56.76	**60**
ΔMFCC	13	30	23.33	**36.76**	26.76
ΔΔMFCC	13	20	20	**33.33**	23.33
(MFCC, ΔMFCC and ΔΔMFCC)	**76.67**				

4.2.1.2 Speaker Recognition using LPCC

The LPCC are also widely used for speaker recognition [57,122]. The computation of LPCC follows mainly two steps:

(i) Computation of LPC :- The LPC are computed by the LP analysis of speech. The LP analysis is described in Appendix C. The LP analysis assumes that the current sample of the signal is the linear combination of the previous samples. Therefore, it can be predictable if the previous samples are known which depends on the order of prediction p. If $s(n)$ is the present sample, then it is predicted by the past p samples as

$$\hat{s}(n) = -\sum_{k=1}^{p} a_k s(n-k) \tag{4.3}$$

where a_k are the LPCs and are obtained by solving the set of p normal equations

$$\sum_{k=1}^{p} a_k R(n-k) = -R(n), \quad n = 1, 2, ..., p \tag{4.4}$$

where R(k) is the auto-correlation sequence given by

$$R(k) = \sum_{n=0}^{N-(p-1)} s(n)s(n-k), \quad k = 0, 1, ..., p \tag{4.5}$$

and $s(n)$ are the speech samples.

(*ii*) LPC to LPCC transformation :- The LPCC are computed from LPC using the following recursive relationships [23]

$$c_1 = a_1, \tag{4.6}$$

$$c_n = \sum_{i=1}^{n-1}(1 - i/n)a_i c_{n-i} + a_i, \quad 1 < n \leq p, \tag{4.7}$$

$$c_n = \sum_{i=1}^{n-1}(1 - i/n)a_i c_{n-i}, \quad n > p \tag{4.8}$$

$$\tag{4.9}$$

In this study, we computed 13 dimensional LPCC feature vector for every frame of size 20 ms with shift of 10 ms using 10^{th} order LP analysis [122]. The reason for using 10^{th} order is that for speech sampled at 8 kHz, the LP order in the range 8-12 is shown to be appropriate to compute LPCC [22, 122]. The experimental results for LPCC and its derivatives for different codebook sizes are given in Table 4.2. The highest performance of 66.67%, 30% and 13.33% is obtained for codebook of size 32 for LPCC, ΔLPCC and ΔΔLPCC, respectively. The results show that LPCC feature performance is less than the MFCC. We then combined the evidences from LPCC, ΔLPCC and ΔΔLPCC to see the effectiveness. The combined system gives 70% recognition performance. Though the combined system gives higher performance than the individual systems, the performance is still less than the combined (MFCC, ΔMFCC and ΔΔMFCC) system. From this study we can conclude that the MFCC feature perform better than the LPCC feature under limited data condition also. We suggest that the MFCC feature can be used for speaker recognition under limited data condition. In order to improve further the speaker recognition performance over the combined (MFCC, ΔMFCC and ΔΔMFCC) system, some other aspect of speech also needs to be considered.

Table 4.2: Speaker recognition performance (%) for the first 30 speakers of the YOHO database using *3 sec training and testing data* for LPCC feature and its derivatives.

Features	Codebook size			
	16	32	64	128
LPCC	60.00	**66.67**	63.33	60.00
ΔLPCC	16.67	**30.00**	26.67	13.33
ΔΔLPCC	13.33	**13.33**	13.33	10.00
(LPCC, ΔLPCC and ΔΔLPCC)	**70.00**			

4.2.2 Excitation Source Features for Speaker Recognition

4.2.2.1 Speaker Recognition using LPR

Earlier studies using neural network models demonstrated that the LP residual contains speaker-specific excitation information that can be used for speaker recognition [17, 43]. In order to obtain the LP residual, first the vocal tract information is predicted from the speech signal by LP analysis [22]. The estimated LPC represent the vocal tract information and are suppressed from the speech signal using inverse filter formulation to obtain LP residual [22]. Mathematically LP residual is the difference between actual and predicted value, which is given by

$$r(n) = s(n) - \hat{s}(n) = s(n) + \sum_{k=1}^{p} a_k s(n-k) \tag{4.10}$$

The LP residual is obtained by passing the speech signal through the inverse LP filter *A(z)*, given by

$$A(z) = 1 + \sum_{k=1}^{p} a_k z^{-k} \tag{4.11}$$

The LP spectrum $|H(jw)|^2$ is given by

$$|H(jw)|^2 = \left| \frac{G}{1 + \sum_{k=1}^{p} a_k e^{-jwk}} \right|^2 \tag{4.12}$$

where G is the gain parameter given by the minimum mean squared error

$$G^2 = \min_{a_k} \left\{ \sum_n e^2(n) = \sum_{k=0}^{p} a_k R(k) \right\} \tag{4.13}$$

To study the effectiveness of speaker-specific information from LP residual for limited data

condition, we conducted speaker recognition studies using Vector-Pulse Code Modulation (V-PCM) concept [125]. The V-PCM is a simple extension of scalar quantization, which tries to quantize a group of samples in the given signal together [126, 127]. Since LP residual is noise-like signal in which relations up to second order have been removed, conventional feature extraction based on first and second order relations among the samples may not bring out speaker information present in the LP residual [17]. The approach to handle this is to use non-linear modelling techniques like neural network as demonstrated earlier [22]. Alternatively, the simple V-PCM concept can be used and hence the motivation for the present work. Before conducting the study using V-PCM, it is essential to understand the difference between this and the VQ technique. In V-PCM we are directly quantizing the group of signal values. In case of VQ, first the signal values are converted into parameter vectors using feature extraction techniques and then quantization of parameter vectors is performed. Since we cannot convert the LP residual signal into parameter vectors, as mentioned earlier, we restrict to V-PCM. In simple sense what is done in V-PCM is, group similar looking signal waveform segments, and are represented by a representative waveform as vector. By observation also we can find such temporal patterns in the LP residual to cluster them together and represent using a centroid vector. Apart from this, the process of obtaining the centroid vectors remains same in both cases.

In V-PCM study, blocks of LP residual samples are used for quantization. To have better matching among different blocks, it may be required to smooth the LP residual slightly. This can be done by decimating the same. Too high a decimation value may remove the speaker information. Hence in this work decimation factor of 2 and 4 are used for experimental study. Thus the LP residuals with and without decimation are applied to V-PCM using blocks of different size and shift. The block size is chosen to be 3 and 5 ms with the objective of keeping its size less than pitch period to avoid the dominance of pitch information [17]. The representation of LP residual in terms of the quantized vectors using V-PCM concept leads to speaker-specific modelling. During testing, LP residual is processed in blocks of 3 and 5 ms with decimation factor by 2 and 4 and compared with reference centroid vectors by Euclidean

distance computation. The speaker of the model with the best match is recognized as the speaker of the test speech data.

The experimental results obtained for different cases are shown in Table 4.3. The first striking observation is the very poor performance obtained for less number of centroid vectors (codebook size 16 and 32). Since the variability among different blocks of residual samples is more, 16 and 32 may be insufficient to have enough unique sequences. Alternatively, 64 and 128 codebook sizes provide better performance. Any further increase in codebook size may improve the performance but this increases the computational complexity and hence it is a trade-off. Thus in all other experiments we have taken 128 as the maximum codebook size. In 128 codebook size, performance is better in most of the cases for 3 ms frame size. This may be due to better uniqueness over smaller sequence lengths, compared to longer sequence lengths. Compared to the original LP residual, decimation by a factor of 2 provides better performance for 128 codebook size with 3 ms block size. This infers that decimation by a factor 2 improves matching among different sequences. Decimation by a factor of 4 provides significant reduction in performance. This is due to the removal of most of the speaker information. Thus this experiment demonstrates that the V-PCM based approach indeed provides highest performance of 46.67% for speaker recognition using excitation source information. Even though this performance is poor compared to what is obtained using MFCC, it may be possible that the combination of the two may provide improved performance due to its origin from the excitation source component.

Table 4.3: Speaker recognition performance (%) for the first 30 speakers of the YOHO database using *3 sec training and testing data* for LPR.

Decimation	Block size (ms)	Codebook size			
		16	32	64	128
Nil	5/2.5	6.6	16.67	23.33	30.00
	3/1.5	13.33	26.67	33.33	40.00
2	5/2.5	26.67	23.33	36.67	40.00
	3/1.5	13.33	23.33	30.00	**46.67**
4	5/2.5	26.67	26.67	40.00	30.00
	3/1.5	10.00	16.67	23.33	16.67

4.2.2.2 Speaker Recognition using LPRP

Recent study shows that the LP residual phase also contains speaker-specific information which can be used for speaker recognition [43]. The LPRP is obtained by dividing the LP residual using its Hilbert envelope [43]. Hilbert envelope is the magnitude of the analytic signal of a given real signal. LPRP computation is as follows:

The analytic signal $r_a(n)$ corresponding to the LP residual *r(n)* is given by

$$r_a(n) = r(n) + jr_h(n) \tag{4.14}$$

where $r_h(n)$ is the Hilbert transform of *r(n)* and is given by

$$r_h(n) = IFT\left[R_h(\omega)\right] \tag{4.15}$$

where

$$R_h(w) = \begin{cases} -jR(\omega), & 0 \leq \omega < \pi \\ jR(\omega), & 0 > \omega \geq -\pi \end{cases} \tag{4.16}$$

Here $R(\omega)$ is the Fourier transform of *r(n)*, and IFT denotes the inverse Fourier transform. The magnitude of the analytical signal $r_a(n)$ which is the Hilbert envelope $h_e(n)$ given by

$$h_e(n) = |r_a(n)| = \sqrt{r^2(n) + r_h^2(n)} \tag{4.17}$$

and the cosine of the phase of the analytic signal $r_a(n)$ is given by

$$\cos(\Theta(n)) = \frac{Re(r_a(n))}{|r_a(n)|} = \frac{r(n)}{h_e(n)} \tag{4.18}$$

The difference between LPR and LPRP is shown in Figure 4.2. From the figure we can understand that the LPRP contains speaker-specific sequence information whereas the LPR contains excitation source information related mainly to Glottal Closure Instants (GCIs) due to dominance of energy around GCIs. It may be possible that these two features may have different aspect of speaker-specific excitation information. To study the effectiveness of LPRP, the speaker information is estimated from the residual phase for different block size and shift,

for 30 speakers each having 3 sec training and testing data. The speaker modelling is done using V-PCM as in the LPR case. The testing is also done as in the LPR case.

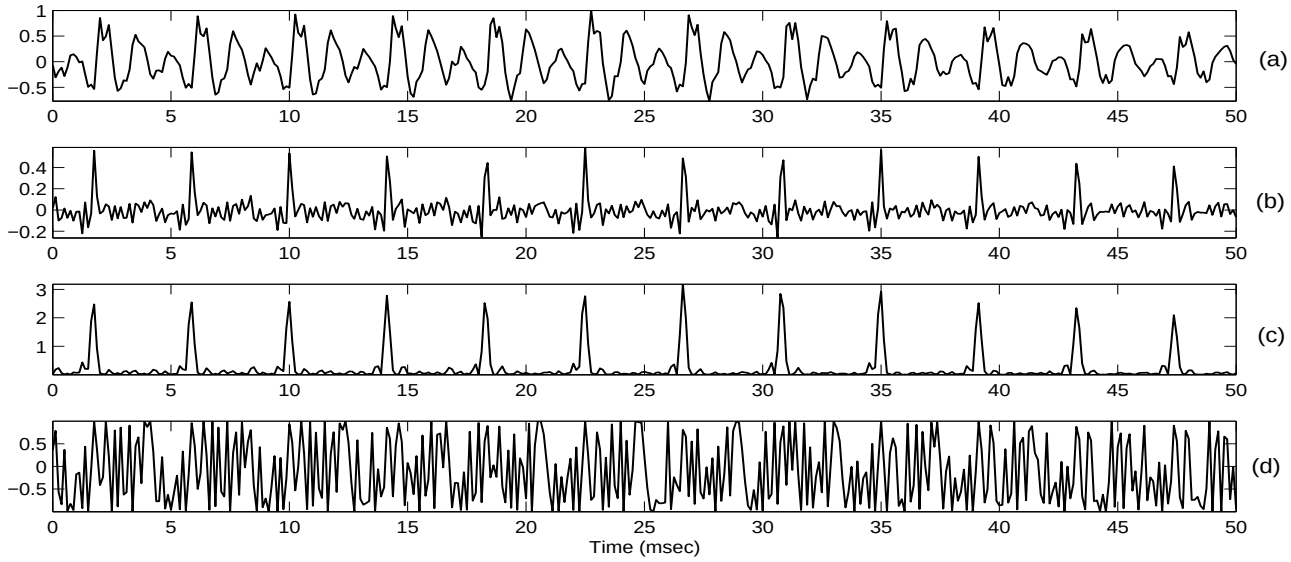

Figure 4.2: Difference between LPR and LPRP: (a) a segment of speech signal, (b) corresponding LP residual, (c) Hilbert envelop (HE) of LP residual and (d) corresponding LP residual phase.

The performance of the LPRP system is given in Table 4.4. The results show that the information extracted for block size 3 ms and shift 1.5 ms from the original LP residual at codebook size 64 yields highest recognition performance of 46.67%. The high performance in the original LP residual phase is due to sequence information belonging to high frequency component. On the other hand, decimating the LPRP removes sequence information due to high frequency component and hence degradation in performance. Though the performance of LPRP is same as that of LPR, speakers identified are different in both cases and are shown in Table 4.5. The speaker numbers 7, 8, 9, 18, 26, 28 and 29 are not identified by LPRP, but identified by LPR. Similarly, speaker numbers 4, 10, 11, 19 and 20 are not identified by LPR, but identified by LPRP. As a result the combined LPR and LPRP system yields 56.67% performance. This is higher than that of both LPR and LPRP performance. This implies that the excitation source aspect of LPRP is different as compared to LPR. Hence LPRP can also be used as independent evidence with MFCC and LPR to improve the performance.

Table 4.4: Speaker recognition performance (%) for the first 30 speakers of the YOHO database using *3 sec training and testing data* for LPRP.

Decimation	Block size (ms)	Codebook size			
		16	32	64	128
Nil	5/2.5	20.00	23.33	43.33	40.00
	3/1.5	30.00	30.00	**46.67**	23.33
2	5/2.5	20.00	20.00	23.33	20.00
	3/1.5	16.67	23.33	10.00	20.00
4	5/2.5	10.00	20.00	16.67	23.33
	3/1.3	6.67	3.33	10.00	16.67

Table 4.5: Number of speakers identified by the LPR, LPRP and (LPR and LPRP) systems for 30 speakers.

Testing speakers

Features	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	No.
LPR	✓	✓			✓	✓	✓	✓	✓									✓				✓		✓	✓	✓		✓	✓		14
LPRP	✓	✓		✓	✓	✓			✓	✓	✓								✓	✓		✓		✓	✓					✓	14
LPR+ LPRP	✓	✓			✓	✓		✓	✓		✓		✓					✓	✓	✓		✓	✓	✓	✓			✓		✓	17

4.3 Limited Data Speaker Recognition using Combination of Features

As we before mentioned, the evidences obtained by each of the features are different and hence they may be combined to further improve the recognition performance. To verify this we use the LCFR described earlier. The best performance of MFCC, its derivatives, LPR, LPRP and the performance of different combined evidences are given in Table 4.6. The combined (MFCC, ΔMFCC, ΔΔMFCC, LPR and LPRP) system yields the highest recognition performance of 86.67%. This is higher than the individual and other combined systems performance. To understand better, the details of speakers identified by each technique for their highest performance quoted in Table 4.6 is shown in Table 4.7. The combined (MFCC, ΔMFCC, ΔΔMFCC, LPR and LPRP) system identifies more number of speakers compared to all individual techniques and hence improvement in the performance. The improvement in performance is due to the different information provided by each feature.

Table 4.6: The MFCC, LPR and LPRP based individual and combined systems performance (%) for the first 30 speakers of the YOHO database using *3 sec training and testing data.*

Features	Performance (%)
MFCC	70.00
ΔMFCC	36.67
ΔΔMFCC	33.33
LPR	46.67
LPRP	46.67
(MFCC, ΔMFCC and ΔΔMFCC)	76.67
(MFCC, ΔMFCC, ΔΔMFCC and LPR)	80.00
(MFCC, ΔMFCC, ΔΔMFCC and LPRP)	80.00
(MFCC, ΔMFCC, ΔΔMFCC, LPR and LPRP)	**86.67**

Table 4.7: Number of speakers identified by the individual and combined systems for the first 30 speakers of the YOHO database.

Testing speakers

Features	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	No.
MFCC	✓	✓	✓		✓	✓		✓	✓	✓	✓		✓		✓		✓	✓	✓	✓	✓			✓	✓		✓		✓	✓	**21**
ΔMFCC			✓		✓		✓				✓		✓			✓		✓			✓	✓	✓							✓	**11**
ΔΔMFCC					✓				✓	✓						✓		✓		✓			✓		✓			✓		✓	**10**
LPR	✓	✓			✓	✓	✓	✓	✓									✓				✓		✓	✓	✓		✓	✓		**14**
LPRP	✓	✓		✓	✓	✓			✓	✓	✓								✓	✓		✓		✓	✓					✓	**14**
Combined	✓	✓	✓	✓	✓	✓		✓	✓	✓	✓		✓		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓		✓	✓	✓	✓	**26**

Earlier we have mentioned that the combined MFCC and its derivatives outperform the combined LPCC and its derivatives. Further, to verify the effectiveness of combination of LPCC and its derivatives with LPR and LPRP, we have combined the evidences from them at scoring level using the LCFR. The best performance of LPCC, its derivatives, LPR, LPRP and the performance of different combined systems are given in Table 4.8. The combined (LPCC, ΔLPCC, ΔΔLPCC, LPR and LPRP) system gives the highest recognition performance of 83.33%. Though this performance is higher than that of the individual and other combined systems, it less than than that of the combined (MFCC, ΔMFCC, ΔΔMFCC, LPR and LPRP) system which provides 86.67%. Therefore, in future studies we use MFCC, its derivatives, LPR and LPRP as features for speaker recognition.

The same study is also conducted for other data sizes of 6, 12 and 24 sec to verify the

Table 4.8: LPCC, LPR and LPRP based individual and combined systems performance (%) for the first 30 speakers of the YOHO database using *3 sec training and testing data.*

Features	Performance(%)
LPCC	66.67
ΔLPCC	30.00
ΔΔLPCC	13.33
LPR	46.67
LPRP	46.67
(LPCC, ΔLPCC and ΔΔLPCC)	70.00
(LPCC, ΔLPCC, ΔΔLPCC and LPR)	76.67
(LPCC, ΔLPCC, ΔΔLPCC and LPRP)	80.00
(LPCC, ΔLPCC, ΔΔLPCC, LPR and LPRP)	**83.33**

effectiveness of the combined system. The experimental results are shown in Figure 4.3. The performance of different combined systems are higher compared to MFCC, particularly, the combined (MFCC, ΔMFCC, ΔΔMFCC, LPR and LPRP) system gives highest performance. Further, the proposed combined system shows significant improvement in the performance up to 12 sec and above 12 sec the performance of different combined systems approach one another. Therefore, in order to verify the recognition performance for the whole database, the experiment is carried out up to 12 sec training and testing data and the results are shown in Figure 4.4. The trend in the experimental results shown in Figure 4.4 resemble that in Figure 4.3. This implies that the proposed combined feature system also show similar behavior for large database also.

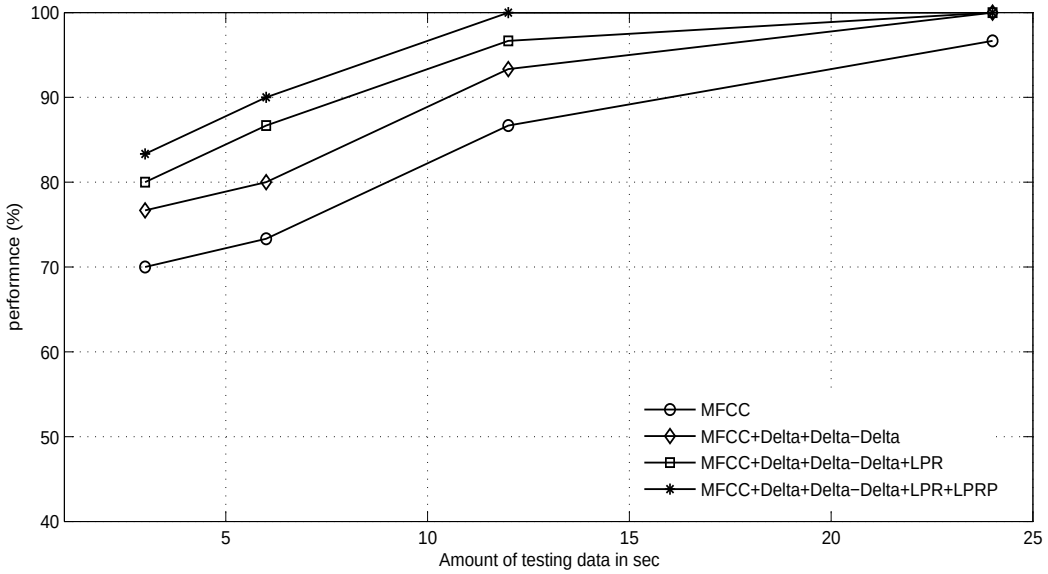

Figure 4.3: Performance of the speaker recognition system for MFCC and different combined system using one wave file (3 sec) training and testing data for the first 30 speakers taken from the YOHO database.

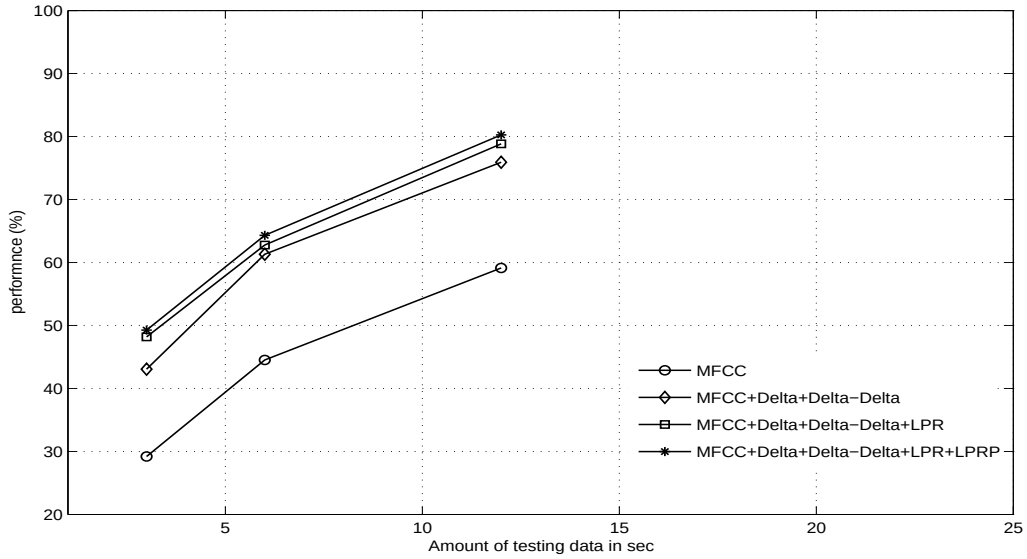

Figure 4.4: Performance of the speaker recognition system for MFCC and different combined system using one wave file (3 sec) training and testing data for 138 speakers taken from the YOHO database.

To verify the robustness of the proposed combination of feature system, we conducted the experiments on the TIMIT database also. The speaker recognition experiments are conducted on the TIMIT test set and the results are shown in Figure 4.5 and 4.6 for a set of first 30 and 138 speakers, respectively. The experimental results of the TIMIT database also resemble those for the YOHO database irrespective of speaker population and amount of data. Hence, we suggest that the combination of features can be used for speaker recognition under limited data condition.

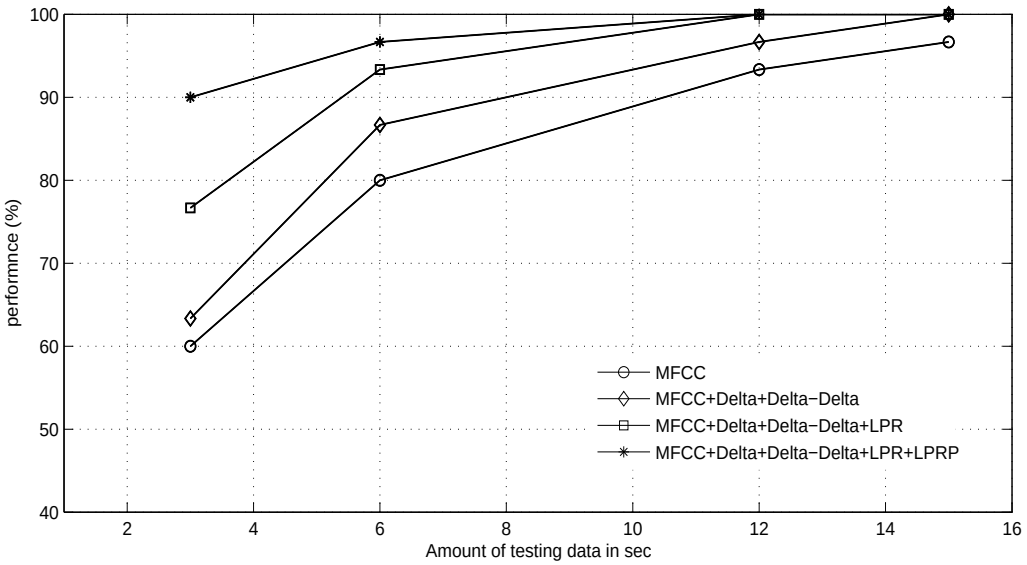

Figure 4.5: Performance of the speaker recognition system for MFCC and different combined system using one wave file (3 sec) training and testing data for the first 30 speakers taken from the TIMIT database.

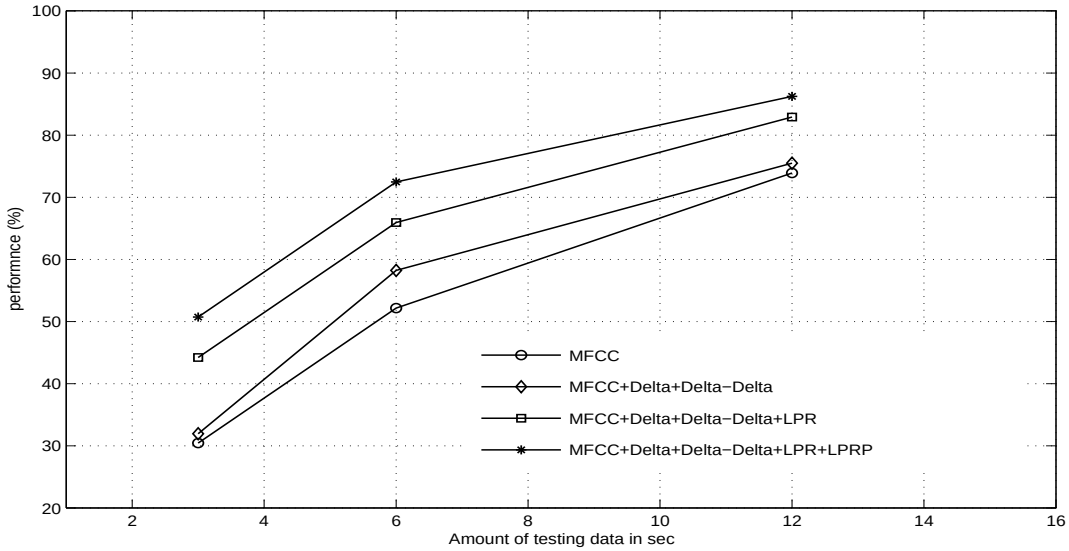

Figure 4.6: Performance of the speaker recognition system for MFCC and different combined system using one wave file (3 sec) training and testing data for the first 138 speakers taken from the TIMIT database.

4.4 Summary

In this chapter we demonstrated that the multiple evidences obtained from different levels of speech indeed improve the speaker recognition performance under limited data condition. First, we studied the working principles of different feature extraction techniques. Then, we compared the performance of MFCC and its derivatives against LPCC and its derivatives and we found that the first one outperforms the second one. Also, the use of V-PCM for speaker modelling using LPR and LPRP features are studied and found that the combination of these two improved the performance over the individual features. Finally, we made an attempt to combine MFCC and its derivatives with LPR and LPRP. As a result, we found that the combined (MFCC, ΔMFCC, $\Delta\Delta$MFCC, LPR and LPRP) system yields good performance compared to the individual features system for speaker recognition under limited data condition.

In the next chapter we study the performance of different modelling techniques and implication of combining different modelling techniques for speaker recognition under limited data condition.

5

Combined Modelling Techniques for Limited Data Speaker Recognition

Contents

Most of the existing modelling techniques for the speaker recognition task make an implicit assumption of sufficient data for speaker modelling and hence may lead to poor modelling under limited data condition. This chapter presents an experimental evaluation of the modelling techniques like Crisp Vector Quantization (CVQ), Fuzzy Vector Quantization (FVQ), Self-Organizing Map (SOM), Learning Vector Quantization (LVQ), and Gaussian Mixture Model (GMM) classifiers. An experimental evaluation of the most widely used Gaussian Mixture Model-Universal Background Model (GMM-UBM) is also made. The experimental knowledge is then used to select a subset of classifiers for obtaining the combined classifiers. It is proposed that the combined LVQ and GMM-UBM classifier provides relatively better performance compared to all the individual as well as combined classifiers.

5.1 Introduction

Modelling techniques aim at generating good representative vectors for features of the speaker. The significance of the amount of speech data for speaker modelling and testing has been studied earlier using Auto Associative Neural Networks (AANN) [22]. In this study, it is reported that when the speech data for training is less, then the performance is poor due to poor speaker modelling. Also, insufficient test speech leads to unreliable decision during testing. Therefore, the task objective of speaker recognition under limited data condition is to obtain as good and reliable performance as possible. State-of-the-art speaker recognition systems employ various modelling techniques like CVQ, FVQ, SOM, LVQ and GMM. The success of each of the modelling techniques depends on the principle employed for clustering. Among these modelling techniques, the widely used one is GMM [16]. The success of GMM is due to the availability of sufficient data for speaker modelling [25]. Recently, some attempts have been made to recognize the speakers under limited data condition using the concept of GMM-UBM [25, 26].

In this work, we first evaluate the performance of various modelling techniques and then combine some of them to improve the performance. The modelling techniques may offer different information about the patterns to be classified due to the difference in the working

principle employed and hence could be used to improve the performance in a combined modelling system [128]. This is the motivation for combining the different models. For instance, in case of CVQ, the feature vectors are clustered into non-overlapping clusters, whereas in case of FVQ, the feature vectors are clustered into overlapping clusters. Thus since the principle of clustering is different, it may be possible to combine these modelling techniques to obtain a combined modelling technique. The rest of the chapter is organized as follows: In section 5.2, the speaker recognition studies using different modelling techniques under limited data are discussed. Section 5.3 presents the proposed combined modelling techniques for speaker recognition. Summary of the work presented in this chapter is mentioned in Section 5.4.

5.2 Limited Data Speaker Recognition Studies using Different Modelling Techniques

In this study also the YOHO [119] and TIMIT [120] databases are used to evaluate the performance of different modelling techniques. The analysis technique we used is SFSR and MFCC is used as feature. The steps involved in the combined modelling technique for speaker recognition under limited data condition are shown as block diagram in Figure 5.1. The different modelling techniques we studied are as follows:

5.2.1 Speaker modelling by Direct Template Matching (DTM)

When the amount of available data is small, the number of feature vectors is also small. For instance, assuming about 80% speech frames, for 3 sec of speech signal there are about 240 feature vectors. Since the number of feature vectors are insufficient, we can use direct template matching to find the speaker recognition performance. In DTM technique, during the identification phase, the test feature vector of an unknown speaker is compared with all the reference training feature vectors to identify tentative speaker of the speech frame. This process is repeated for all the testing frames. The speaker with maximum number frames is identified as the speaker of the test speech data. In the 30 speakers case of the YOHO database, the recognition performance of 63.33% is obtained for one speech file (3 sec) of training and testing

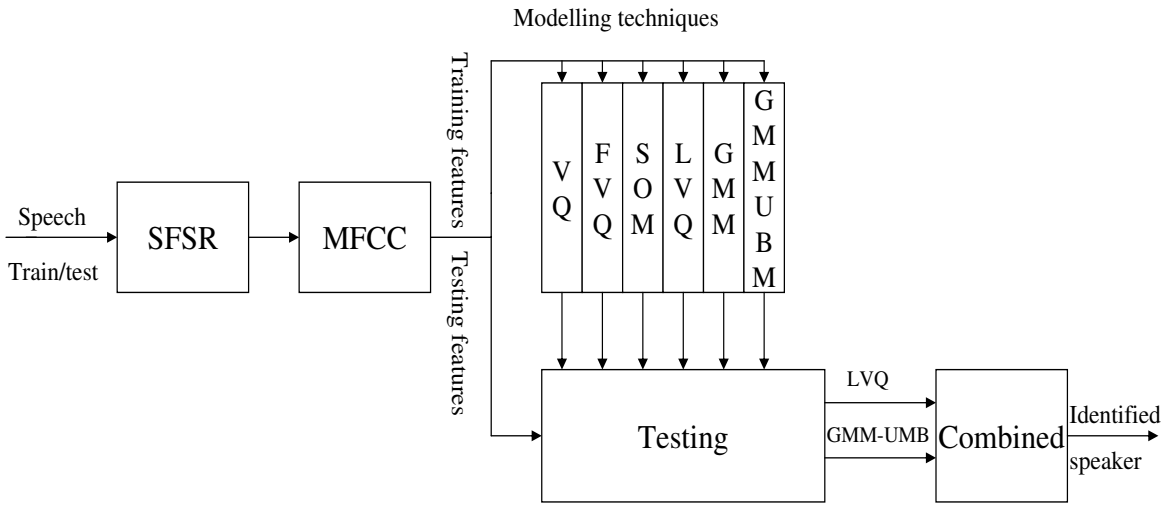

Figure 5.1: Block diagram of proposed combined modelling technique for speaker recognition under limited data condition.

data. Though this technique is simple and easy to perform, the performance is poor. The poor performance is due to large intraspeaker and inter speaker variability. This shortcoming may therefore be reduced using different modelling techniques. The objective of modelling technique is to better cluster or capture the distribution of the feature vectors according to the speaker information. Speaker models built contain the feature vectors from different sound units, but from the same speakers. This may enable dominance of speaker information over speech information. This aspect is verified in limited data condition using the following speaker modelling techniques.

5.2.2 Speaker Modelling using CVQ

CVQ is also termed as VQ [75]. The brief introduction and working principle of VQ has already been given in Section 3.3.2 of Chapter 3. Moreover, to find the codevectors for a given speaker, CVQ clusters all the feature vectors in the feature space into non-overlapping clusters with crisp boundaries and hence the name. The experimental results of CVQ for the 30 speakers of the YOHO database using one speech file (3 sec) for training and testing data are given in Section 3.4 of Chapter 3 (Table 3.2). In this chapter also for the comparison purpose, the same

experimental results are given in Table 5.1. The highest recognition performance of 70% is achieved using a codebook size of 64.

Table 5.1: Speaker recognition performance (%) for the first 30 speakers of the YOHO database using *3 sec training and testing data* for CVQ modelling technique.

Modelling technique	Codebook size			
	16	32	64	128
CVQ	63.33	66.67	**70.00**	60.00

As we have already mentioned 3 sec data provides about 240 frames. This number is too small for forming a CVQ codebook of size 64, since as a thumb rule, the number of feature vectors should be about 10 times the number of non-overlapping clusters [23]. Accordingly, CVQ with codebook of size 16 seems to be optimum. However, as per the result obtained, even larger codebook sizes like 32 and 64 give higher recognition performance. This implies that for limited data condition about 5 times the codebook size may be kept as thumb rule, while deciding the codebook size. Accordingly, further increase in the codebook size to 128 gives poor performance. Better performance of CVQ compared to DTM implies that, it may be better to use some techniques for modelling. A maximum performance of only 70% is due to the limited training and testing data and also the modelling technique employed. Therefore, for given training and test data, to increase the performance, we can explore alternate modelling techniques.

5.2.3 Speaker Modelling using FVQ

FVQ is an alternative to CVQ and employs fuzzy logic principle for clustering. The basic principle of fuzzy logic is that a given feature vector can be assigned to more than one cluster with certain degree of association to find the codevectors for a given speaker. FVQ clusters all the feature vectors in the feature space into overlapping clusters with fuzzy boundaries and hence the name [77, 78, 129]. In FVQ each feature vector is assigned to all the clusters, but with different degrees of association, as dictated by the membership function. The merit of FVQ compared to CVQ is that, since all the feature vectors are associated with all the clusters,

there are relatively more number of feature vectors for each cluster and hence the codevectors may be more reliable. The codebooks of different sizes are built using binary split and fuzzy *C-means* clustering procedures during training [77, 129]. The fuzzy *C-means* clustering method involves finding the membership values of each of the feature vectors with the centroids and recomputing the centroids using the new membership values which is as follows:

Let x_i be the training feature vectors of size n. Each feature vector is to be assigned to a given cluster w_i, where the number of possible clusters is C, i.e. $i = 1, 2, ..., C$. The FVQ finds membership grade μ_{ik} for each x_n and updates the cluster centers V_i (mean) according to the following equations:

$$\mu_{ik} = \frac{\left(\frac{1}{\|x_k - V_i\|_G^2}\right)^{1/(m-1)}}{\sum_{j=1}^{c}\left(\frac{1}{\|x_k - V_i\|_G^2}\right)^{1/(m-1)}} \qquad i = 1, ..., C \quad k = 1, ..., n \tag{5.1}$$

$$V_i = \frac{1}{\sum_{k=1}^{n}(\mu_{ik})^m}\sum_{k=1}^{n}(\mu_{ik})^m x_k \qquad i = 1, ..., C \tag{5.2}$$

The fuzziness of the clustering procedure is controlled by the parameter m known as exponential weight (learning rate), which is always greater than one. When m tends to one, the clustering tends to the one provided by the crisp procedure. V_i is the mean of the x_k. The x_k with high degrees of membership have a higher influence on V_i than those with low degrees of membership. The m reduces the influence of small μ_{ik} (points further away from V_i) compared to that of large μ_{ik} (points close to V_i).

The steps involved in the binary split technique for building a FVQ codebook are as follows [77]:

(i) Design a 1-vector codebook by computing the mean of x_i.

(ii) Double the size of the codebook by splitting each current codebook according to the equations $y_c^+ = y_c\,(1 + \varepsilon)$ and $y_c^- = y_c\,(1 - \varepsilon)$, where ε is a splitting parameter (typically in the range $0.01 \leq \varepsilon \leq 0.05$).

(iii) Use fuzzy *C-means* clustering method to get best set of centroids for the split codebook.

(iv) Iterate the steps (ii) and (iii) until codebook of size C is designed.

The nature of clustering depends strongly on the learning rate parameter and hence it needs to be tuned for better performance. The feature vectors of the test speech data are compared with the codebooks of different speakers as in the case of CVQ. The experimental results using FVQ for the 30 speakers of the YOHO database using one speech file (3 sec) for training and testing data and different learning rate parameter are given in Table 5.2. The highest performance of 76.67% is achieved for the codebook size of 32 using the learning rate parameter of 1.39, 1.40 and 1.50. For the same amount of speech data (3 sec), we are able to further increase the performance from 70% of CVQ to 76.67%. This can be attributed to the fuzzy *c-means* clustering employed in FVQ.

Table 5.2: Speaker recognition performance (%) for the first 30 speakers of the YOHO database using *3 sec training and testing data* for FVQ modelling technique.

Learning rate	Codebook size			
	16	32	64	128
1.30	70.00	60.00	60.00	60.00
1.31	70.00	63.33	66.67	56.67
1.32	70.00	63.33	60.00	56.67
1.33	70.00	73.33	66.67	60.00
1.34	63.33	70.00	66.67	60.00
1.35	63.33	70.00	63.33	60.00
1.38	63.33	70.00	70.00	60.00
1.39	60.00	**76.67**	66.67	60.00
1.40	60.00	**76.67**	70.00	66.67
1.45	66.67	63.33	66.67	63.33
1.50	60.00	**76.67**	73.33	70.00
1.55	63.33	73.33	73.33	66.67
1.60	56.67	66.67	60.00	60.00

The better recognition performance by FVQ suggests that by increasing the number of elements for clustering, the performance also increases. This is achieved by associating the same set of feature vectors to different clusters, of course, with different membership functions. This improvement in performance is at the cost of increased computational complexity of tuning the learning rate parameter. However, it is still preferable due to the small amount of data. We can also explore other VQ modelling techniques based on neural networks on similar lines.

5.2.4 Speaker Modelling using SOM

A neural network counter part of VQ, but with unsupervised learning can be realized using SOM. The approach for identifying the codevectors is by learning and also in an unsupervised way. The clustering is therefore influenced by the actual distribution of feature vectors and hence the modelling may be different. SOMs are a special class of neural networks based on competitive learning [130]. In case of competitive learning the output neuron of the network compete among themselves to be activated, with the result that only one output neuron per group is on at a time. An output neuron that wins the competition is called a winning neuron. In the synaptic adaptation step the winning neurons are made to increase their individual values of discriminant function in relation to the input pattern through suitable adjustments applied to their synaptic weights. The adjustments are made such that the response of the winning neuron to the subsequent application of a similar input pattern is enhanced [131]. The steps involved in the weight vector tuning (codebook) using SOM are as follows:

Let $X = [x_1, x_2, ..., x_k]^T$ be the training feature vectors of size N of each k dimensions. The SOM consists of one input layer and one output layer. The output layer includes L number of the output neurons $u_j, j = 1, 2, ..., L$, which are typically organized in a planar two dimensional lattice. The weight from the input layer neuron to the output layer neuron are $W_{ij}, i = 1, 2, ..., N, j = 1, 2, ..., L$. The weight vector of each neuron has the same dimension as the input pattern. The following steps are used to fine tune the weight vectors [131]:

(i) Initialize the training process with selecting weight vectors randomly from the input vectors.

(ii) Find the distance between the input vector X and the weight vector W_j of each u_j by using the Euclidean distance measurement:

$$d_j = \|X - W_j\| = \sqrt{\sum_{i=1}^{n}(X_i - w_{ij})^2} \tag{5.3}$$

(iii) Find the winning neuron based on smallest distance.

(iv) Adjust the weights of the winning neuron and its topological neighborhood neurons in the direction of the input vector by using the following equations:

$$h_j = \exp\left(-\frac{\left\|u_j - u_j^*\right\|^2}{2\sigma^2}\right) \tag{5.4}$$

Where h_j is the topological neighbourhood, σ is the *effective width* of the topological neighbourhood, and u_j^* is the winning neuron. The change to the weight vector W_j can be obtained as

$$\Delta W_j = \eta h_j (X - W_j) \tag{5.5}$$

Where η is the learning rate parameter of the algorithm. Hence, the updating weight vector $W_j(t+1)$ at time *t+1* is defined by

$$W_j(t+1) = W_j(t) + \eta(t)h_j(t)(X - W_j(t)) \tag{5.6}$$

Where $\eta(t)$ and $h_j(t)$ are the learning rate parameter and the topological neighborhood at time t

(v) Repeat from step (ii) to (iv) until no noticeable change in the weight vectors.

According to the algorithm the performance of SOM depends on the parameters like neighbourhood (h), learning rate (η) and number of iterations. The performance for the 30 speakers of the YOHO database using one speech file (3 sec) for training and testing data and different

h, η and iterations are given in Table 5.3. The SOM gives the highest performance of 73.33% for codebook sizes of 16, 32 and 128.

Table 5.3: Speaker recognition performance (%) for the first 30 speakers of the YOHO database using *3 sec training and testing data* for SOM modelling technique.

Iterations	h	η	Codebook Size (CS)			
			16	32	64	128
500*CS	1	0.01	60.00	60.00	53.33	66.67
500*CS	1	0.02	53.33	70.00	70.00	66.67
500*CS	1	0.03	60.00	70.00	66.67	66.67
500*CS	1	0.04	56.67	60.00	63.33	70.00
500*CS	1	0.05	70.00	60.00	60.00	63.33
500*CS	1	0.06	66.67	70.00	70.00	**73.33**
500*CS	1	0.07	70.00	56.67	63.33	53.33
500*CS	1	0.08	66.67	63.33	63.33	53.33
500*CS	1.1	0.06	66.67	70.00	56.67	56.67
500*CS	1.2	0.06	70.00	66.67	56.67	66.67
500*CS	1.4	0.06	66.67	60.00	66.67	56.67
550*CS	1	0.06	**73.33**	**73.33**	63.33	56.67
600*CS	1	0.06	53.33	70.00	53.33	70.00
650*CS	1	0.06	63.33	66.67	70.00	60.00

The recognition performance of 73.33% by SOM using unsupervised learning implies that, even the feature vectors from limited data provide speaker information in the feature space. Further, each speaker has a unique distribution of feature vectors which is learnt by SOM. It may be further possible to improve the performance of SOM by using the LVQ.

5.2.5 Speaker Modelling using LVQ

LVQ developed by Kohonen [130] is used to globally optimize the codebooks after they are generated with unsupervised learning algorithm like SOM. LVQ is a supervised learning technique that uses class information to optimize the positions of codevectors obtained by SOM, so as to improve the quality of the classifier decision regions. An input vector is picked at random from the input space. If the class label of the input vector and the codevector agree, then the codevector is moved in the direction of the input vector. Otherwise the codevector is moved away from the input vector. Therefore, the fine tuning may improve the performance

compared to SOM. The LVQ initially follows the steps involved in SOM for tuning weight vectors (codebooks). These weight vectors are further fine tuned by LVQ which is as follows:

Suppose X_t be an input vector at time t, and W_j weight vector for the class j at time t. Let ζ_{w_c} denote the class associated with the the weight vector W_c and ζ_X denote the class label of the input vector X. The weight vector W_c is adjusted as follows [131]:

- If $\zeta_{w_c} = \zeta_X$, then

  $W_c(t+1) = W_c(t) + \eta(t)\,[X - W_c(t)]$

  Where $0 < \eta(t) < 1$.

- If, on the other hand, $\zeta_{w_c} \neq \zeta_X$, then

  $W_c(t+1) = W_c(t) - \eta(t)\,[X - W_c(t)]$

- The other weight vectors are not modified.

The algorithm shows that LVQ performance depend on the parameters like η and iterations. The performance for the 30 speakers of the YOHO database using one speech file (3 sec) for training and testing data and different η and iterations are given in Table 5.4. The LVQ gives the best recognition performance of 80% which is better than that of all other VQ techniques discussed so far.

The improvement in the performance compared to SOM implies that employing supervised learning over initially obtained unsupervised codevectors indeed improves the performance. Thus the fine tuning by LVQ is beneficial under limited data condition also. The aforementioned modelling techniques are based on non-parametric clustering approach. Speaker modelling by parametric probabilistic approach like GMM and GMM-UBM can also be explored.

Table 5.4: Speaker recognition performance (%) for the first 30 speakers of the YOHO database using *3 sec training and testing data* for LVQ modelling technique.

Iterations	η	Codebook Size (CS)			
		16	32	64	128
500*CS	0.01	63.33	60.00	66.67	60.00
500*CS	0.02	60.00	66.67	63.33	53.33
500*CS	0.03	60.00	66.67	73.33	60.00
500*CS	0.04	53.33	63.33	66.67	63.33
500*CS	0.05	63.33	76.67	66.67	66.67
500*CS	0.06	70.00	60.00	70.00	60.00
500*CS	0.08	63.33	70.00	63.33	60.00
550*CS	0.05	63.33	53.33	70.00	63.33
550*CS	0.06	73.33	**80.00**	60.00	60.00
575*CS	0.06	56.67	66.67	66.67	53.33
600*CS	0.06	63.33	70.00	66.67	63.33
700*CS	0.06	60.00	63.33	73.33	60.00

5.2.6 Speaker Modelling using GMM

The GMM is the most widely used probabilistic modelling technique in speaker recognition. The GMM needs sufficient data (at least one minute) to model the speaker well to yield good recognition performance [132]. Unlike the centroids design, as we discussed in the above modelling techniques, in GMM system the distribution of feature vectors extracted from speaker speech is modeled by the parameters like weight, mean and covariance [132]. Since in our experimental conditions training and testing data are limited, GMM may not be the best choice. However, we conducted the experiment using VQ initialized GMM to see its effectiveness under limited data condition. The GMM approach for speaker modelling is described as follows [16]:

For a D-dimensional feature vector denoted as x_t, the mixture density for speaker λ is defined as weighted sum of M component Gaussian densities given by

$$P(x_t|\lambda) = \sum_{i=1}^{M} w_i P_i(x_t) \tag{5.7}$$

where w_i are the weights and $P_i(x_t)$ are the component densities. Each component density is

a D-variate Gaussian function of the form

$$P_i(x_t) = \frac{1}{(2\pi)^{D/2}\left|\Sigma_i\right|^{\frac{1}{2}}} e^{-\frac{1}{2}\left[(x_t-\mu_i)'\Sigma_i^{-1}(x_t-\mu_i)\right]} \tag{5.8}$$

where μ_i is the mean vector and Σ_i covariance matrix for i^{th} component. The mixture weights have to satisfy the constraint

$$\sum_{i=1}^{M} w_i = 1. \tag{5.9}$$

The complete Gaussian mixture density is parameterized by the mean vector, the covariance matrix and the mixture weight from all component densities. These parameters are collectively represented by

$$\lambda = \{w_i, \mu_i, \Sigma_i\}\,; \qquad i = 1, 2, ...M. \tag{5.10}$$

During training, to maximize the likelihood value, we used the most popular iterative expectation maximization (EM) algorithm. The steps involved are as follows:

(i) Initialization: An initial estimate of the parameters is obtained using *k-means* algorithm.

(ii) Likelihood Computation: In each iteration the posterior probabilities for the i^{th} mixture is computed as,

$$\Pr(i|x_t) = \frac{w_i P_i(x_t)}{\sum_{j=1}^{M} w_j P_j(x_t)}. \tag{5.11}$$

(iii) Parameter Update: Having the posterior probabilities, the model parameters are updated according to the following expressions.

Mixture weight update:

$$\overline{w_i} = \frac{\sum_{i=1}^{T} \Pr(i|x_t)}{T}. \tag{5.12}$$

Mean vector update:

$$\overline{\mu_i} = \frac{\sum_{i=1}^{T} \Pr(i|x_t) x_t}{\sum_{i=1}^{T} \Pr(i|x_t)}. \tag{5.13}$$

Covariance matrix update:

$$\overline{\sigma}_i^2 = \frac{\sum_{i=1}^{T} \Pr(i|x_t) \, |x_t - \overline{\mu_i}|^2}{\sum_{i=1}^{T} \Pr(i|x_t)}. \tag{5.14}$$

During testing, mixture densities are calculated for every feature vector of all speakers and speaker with maximum likelihood is selected as identified speaker. Assuming equal probability of all speakers and the statistical independence of the observations, the decision rule for the most probable speaker is defined as:

$$\hat{s} = \max_{1 \leq s \leq S} \sum_{t=1}^{T} \log P(x_t|\lambda_s) \tag{5.15}$$

where T is the number of feature vectors of the test speech data.

The experimental results for the 30 speakers of the YOHO database using one speech file (3 sec) for training and testing data and different Gaussian mixtures are given in Table 5.5. The GMM yields the highest recognition performance of 73.33% using 16 Gaussian mixtures.

Table 5.5: Speaker recognition performance (%) for the first 30 speakers of the YOHO database using *3 sec training and testing data* for GMM modelling technique.

Modelling technique	Gaussian mixtures			
	16	32	64	128
GMM	**73.33**	40.00	36.67	13.33

The performance of GMM based system is better than that of the CVQ, but poor compared to all other VQ modelling techniques. This implies that the data may be too sparse to model by the Gaussian mixtures. To alleviate this problem to some extent the concept of Universal Background Model (UBM) can be used along with GMM.

5.2.7 Speaker Modelling using GMM-UBM

The concept of GMM-UBM [100] is widely used for speaker recognition where the availability of training data is sparse [25, 26]. We also conducted speaker recognition study using the GMM-UBM to study the effectiveness under limited data condition. In GMM-UBM system, speech data collected from large number of speakers is pooled and the UBM is trained which acts as a speaker independent model. The speaker dependent model can be created by performing maximum *a posteriori* (MAP) adaptation technique from the UBM using speaker-specific training speech. The MAP adaptation steps are as follows:

For each mixture i from the background model, $Pr(i|x_t)$ is given by

$$\Pr(i|x_t) = \frac{w_i P_i(x_t)}{\sum_{j=1}^{M} w_j P_j(x_t)}. \tag{5.16}$$

Using $Pr(i|x_t)$, the statistics of the weight, mean and variance are computed as,

$$n_i = \sum_{i=1}^{T} \Pr(i|x_t) \tag{5.17}$$

$$E_i(x_t) = \frac{\sum_{i=1}^{T} \Pr(i|x_t) x_t}{n_i} \tag{5.18}$$

$$E_i(x_t^2) = \frac{\sum_{i=1}^{T} \Pr(i|x_t) x_t^2}{n_i}. \tag{5.19}$$

These new statistics calculated from the training data are then used to adapt the background model, and the new weights ($\hat{w}_i$), means ($\hat{\mu}_i$) and variances ($\hat{\sigma}_i^2$) are given by

$$\hat{w}_i = \left[\frac{\alpha_i n_i}{T} + (1-\alpha_i) w_i\right] \gamma \tag{5.20}$$

$$\hat{\mu}_i = \alpha_i E_i(x_t) + (1-\alpha_i)\mu_i \tag{5.21}$$

$$\hat{\sigma}_i^2 = \alpha_i E_i(x_t^2) + (1-\alpha_i)(\sigma_i^2 + \mu_i^2) - \hat{\mu}_i^2. \tag{5.22}$$

A scale factor γ ensures that all the new mixture weights sum to 1. α_i is the adaptation coefficient which controls the balance between the old and new model parameter estimates. α_i

is defined as:

$$\alpha_i = \frac{n_i}{n_i + r} \tag{5.23}$$

where r is a fixed relevance factor, which determines the extent of mixing of the old and new estimates of the parameters. Generally only mean values are adapted [100]. It is experimentally shown that mean adaptation gives slightly higher performance than adapting all three parameters .

The UBM training can be done in two ways: 1) Speech data pooled from the other database, not used for the speaker recognition study, provided speech data is collected from the same environment. 2) Same speech data for both UBM training and evaluation, provided the speakers set used for recognition is not included in UBM training [25, 26]. In [100], it is mentioned that there are no criteria to select number of speakers and amount of data to train the UBM. We trained UBM with roughly two hours of data. Also, it is mentioned that the ratio of gender must be equal to avoid the bias towards one side. We conducted the study using the YOHO database for both UBM training and evaluation. Since the database has imbalanced gender set, we have conducted two sets of experiments namely: 1) Using only the male speakers 2) Using the first 30 speakers (28 male, 2 female). First set uses first 30 male speakers for speaker recognition study and the remaining 76 male speakers of each 30 utterances for UBM training. In second set, the first 30 speakers are used for speaker recognition study and the remaining 108 speakers of each 21 utterances for UBM training.

Our experimental studies consider evaluation set of first 30 and 138 speakers. Therefore, experiments are conducted with Not Including Evaluation set (NIE) and Including Evaluation set (IE) in UBM training. The experimental results for the 30 speakers of the YOHO database using one speech file (3 sec) for training and testing data and different Gaussian mixtures are given in Table 5.6. The GMM-UBM for the male speakers set yields the performance of 73.33% and 80% using 128 Gaussian mixtures for NIE and IE, respectively. Though there are few female speakers in the first 30 speakers, GMM-UBM yields the highest performance of 76.67% and 83.33% using 128 Gaussian mixtures for NIE and IE, respectively. This means that the GMM-UBM inclination is towards the majority of the speakers. The high performance of

the first 30 speakers compared to the male speakers is due to availability of more number of background speakers. In future studies with GMM-UBM we use the first 30 speakers set for experimental studies.

Table 5.6: Speaker recognition performance (%) for the first 30 speakers of the YOHO database using *3 sec training and testing data* for GMM-UBM modelling technique.

Speakers set	Modelling technique	Gaussian mixtures			
		16	32	64	128
Male	GMM-UBM-NIE	60.00	63.33	60.00	**73.33**
Male	GMM-UBM-IE	60.00	70.00	73.33	**80.00**
First 30	GMM-UBM-NIE	60.00	60.00	63.33	**76.67**
First 30	GMM-UBM-IE	60.00	66.67	73.33	**83.33**

The recognition performance by not including the speakers in building UBM i.e. GMM-UBM-NIE is the actual result for GMM-UBM case. The recognition performance of 76.67% by the same implies that even by using UBM, does not seem to provide any benefit in terms of improving the performance. The high performance of 83.33% for GMM-UBM-IE is due to the data of each of the speakers used in building the UBM. Hence there is bias in the UBM towards each of the speakers [100].

5.3 Limited Data Speaker Modelling using Combined Modelling Techniques

The modelling techniques discussed so far are different with respect to their working principle and hence may be combined to further improve the performance. To verify this we have used the Linear Combination of Frame Ratio (LCFR) which described in Section 4.2.1.1 Chapter 4. In this study also the combination is done at the scoring level in which the highest performed codebook size of modelling techniques are considered. The speaker with the combined highest frame score is recognized as the final speaker of the test speech data. The best performance of individual models and the performance of different combined models are given in Table 5.7. Among the combined modelling techniques, the LVQ-GMM-UBM-NIE and LVQ-GMM-UBM-IE systems yield the highest performance of 83.33% and 86.67%, respectively. The

improvement in the performance is due to the different working principle employed in LVQ and GMM-UBM. That is, the supervised learning over unsupervised learning involved in LVQ and other speakers data used as UBM in GMM-UBM. Moreover, LVQ modelling technique is based on non-parametric approach, whereas GMM-UBM based on parametric approach and hence this combination gives the best recognition performance. Further, in the other combined techniques like LVQ-FVQ and LVQ-GMM also the working principle are different. However, the FVQ and LVQ are fine tuned using only the speaker specific speech data which may not be optimum and hence the combination techniques using these modelling techniques do not yield higher performance compared to LVQ-GMM-UBM.

Table 5.7: Best individual and combined modelling speaker recognition performance (%) for the first 30 speakers of the YOHO database using *3 sec training and test data* for different modelling techniques.

Modelling techniques	Codebook size/Gaussian mixtures			
	16	32	64	128
CVQ	63.33	66.67	**70.00**	60.00
FVQ	70.00	**76.67**	73.33	70.00
SOM	**73.33**	**73.33**	70.00	**73.33**
LVQ	73.33	**80.00**	73.33	66.67
GMM	**73.33**	40.00	36.67	13.33
GMM-UBM-NIE	60.00	60.00	63.33	**73.33**
GMM-UBM-IE	60.00	66.67	73.33	**83.33**
LVQ-FVQ	80.00			
LVQ-GMM	80.00			
LVQ-GMM-UBM-NIE	**83.33**			
LVQ-GMM-UBM-IE	**86.67**			

For the other data sizes of 6, 12 and 24 sec we conducted the study only with LVQ, GMM-UBM and the combined LVQ-GMM-UBM modelling techniques. The experimental results are shown in Figure 5.2. It is evident from the figure that the performance of GMM-UBM-NIE below 10 sec of training and testing data is less than that of LVQ and GMM-UBM-IE. This means that the available training data is insufficient to train the speaker dependent model in GMM-UBM. Under such conditions the combined system gives better recognition performance than the individual systems. Also, GMM-UBM-IE performance is higher than that of the other

individual techniques even for data less than 10 sec duration. This is due to the availability of speaker-specific sufficient data while training the UBM model. The proposed combined modelling technique shows significant improvement in the performance up to 12 sec and above 12 sec the performance of all modelling techniques approach one another. Therefore, in order to verify the performance for the whole database, the experiment is carried out up to 12 sec training and testing data and the results are shown in Figure 5.3. The trend in the experimental results shown in Figure 5.3 resemble that in Figure 5.2 which implies that the proposed combined modelling technique shows a similar behavior for the large database also.

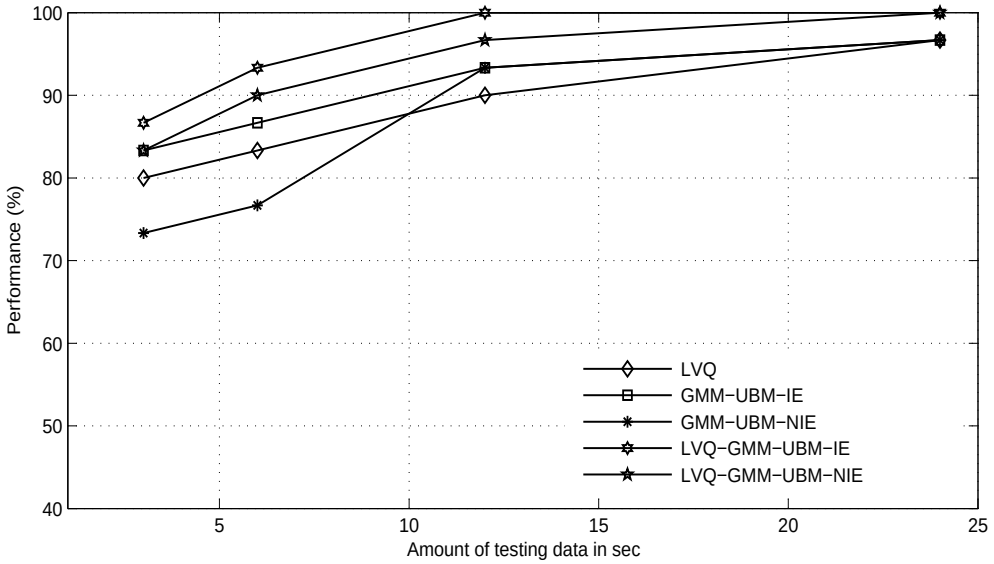

Figure 5.2: Speaker recognition performance based on LVQ, GMM-UBM and LVQ-GMM-UBM modelling for different sizes of training and testing data for the first 30 speakers taken from the YOHO database.

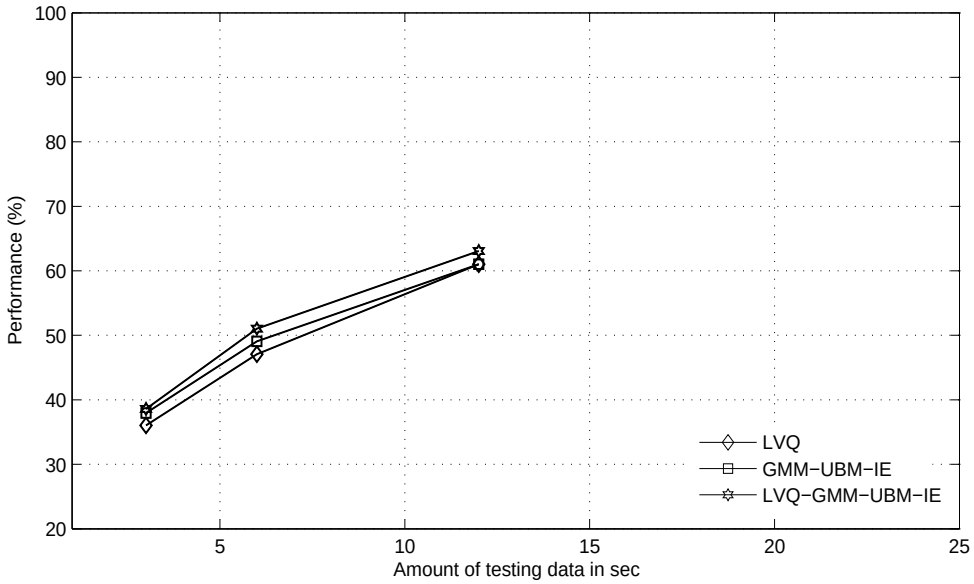

Figure 5.3: Speaker recognition performance based on LVQ, GMM-UBM and LVQ-GMM-UBM modelling for different sizes of training and testing data for 138 speakers taken from the YOHO database.

To verify the robustness of the proposed combined modelling technique, we conducted the experiments on the TIMIT database also. In the GMM-UBM modelling technique, we used the TIMIT training set to train the UBM roughly for 2 hours of data. The speaker recognition experiments are conducted on the TIMIT test set. Experimental studies are conducted as the YOHO database setup and the results are shown in Figure 5.4 and Figure 5.5 for a set of first 30 and 138 speakers, respectively. The experimental results for the TIMIT database also resemble those for the YOHO database irrespective of speaker population and amount of data. Hence, the LVQ-GMM-UBM can be used as a modelling technique for speaker recognition under limited data condition.

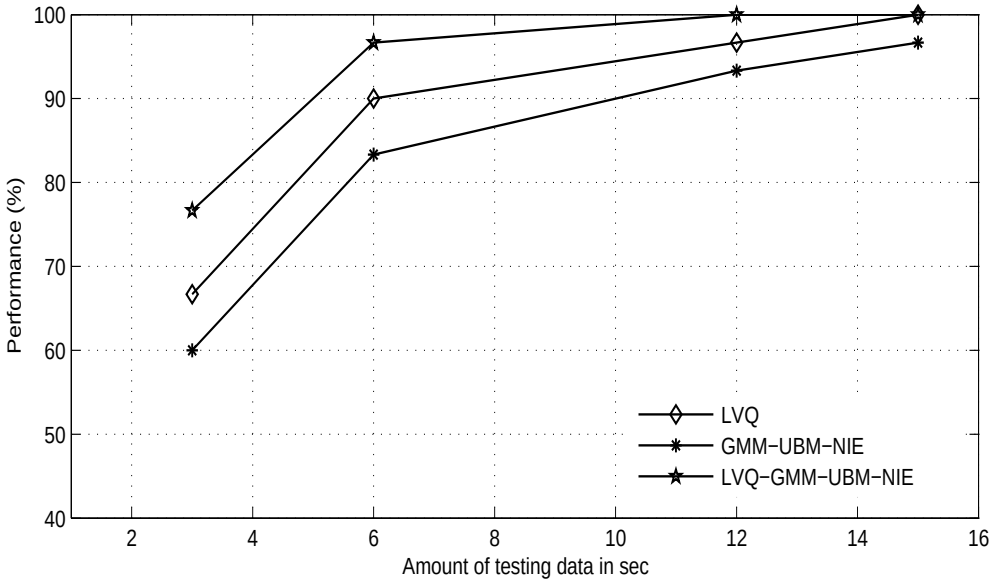

Figure 5.4: Speaker recognition performance based on LVQ, GMM-UBM and LVQ-GMM-UBM modelling for different sizes of training and testing data for the first 30 speakers taken from the TIMIT database.

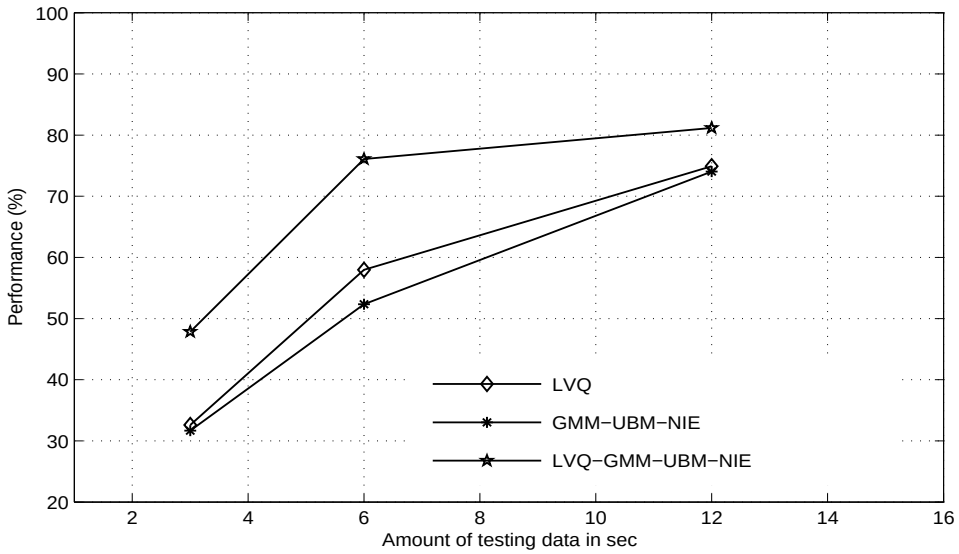

Figure 5.5: Speaker recognition performance based on LVQ, GMM-UBM and LVQ-GMM-UBM modelling for different sizes of training and testing data for the first 138 speakers taken from the TIMIT database.

5.4 Summary

In this chapter, we explored the different modelling techniques and then proposed combined LVQ-GMM-UBM modelling technique for speaker recognition under limited data condition. First, we discussed the working principle and the efficiency of different modelling techniques. Then, we combined different modelling techniques to see the effectiveness. As a result, we found that the combined LVQ-GMM-UBM model gives better recognition performance than the individual and other combined modelling techniques. Therefore, LVQ-GMM-UBM model can be used for speaker modelling under limited data condition.

In the next chapter we integrate the MFSR analysis, combination of features and the combined LVQ-GMM-UBM modelling in the analysis, feature extraction and modelling stages, respectively to study the effectiveness under limited data condition.

6

Integrated Systems for Limited Data Speaker Recognition

Contents

In the previous chapters, we demonstrated that the MFSR analysis provides improved performance in the analysis stage. The combination of features like Mel Frequency Cepstral Coefficients (MFCC), its temporal derivatives (ΔMFCC, $\Delta\Delta$MFCC), Linear Prediction Residual (LPR) and Linear Prediction Residual Phase (LPRP) provides improved performance in the feature extraction stage. The combination of Learning Vector Quantization (LVQ) and Gaussian Mixture Model - Universal Background Model (GMM-UBM) provides improved performance in the modelling stage. The studies are made individually. That is, proposed technique is used in the respective stage and the existing techniques in the remaining stages. In this chapter, we present the *integrated systems* using MFSR analysis, different features and LVQ and GMM-UBM modelling techniques in the analysis, feature extraction and modelling stages, respectively to study the effectiveness under limited data condition. The experimental results show that the integrated systems give better performance compared to the individual systems.

6.1 Introduction

To achieve good performance, especially under limited data, efficient techniques are essential for each stage of the speaker recognition system. The different stages include analysis, feature extraction, modelling and testing. The efficient strategy should be from the perspective of alleviating the difficulty arising out of the limited data. The efficient techniques can be initially explored independently to observe their potential in improving the speaker recognition performance. Later, these techniques can be integrated to obtain a speaker recognition system that provides significantly improved performance under limited data.

The literature shows that the studies made so far treat the problem of limited data for speaker recognition mainly either in feature extraction or modelling stage. Studies made in [36,99] concentrated on feature extraction stage to improve the performance with the constraint of limited data. In [99], the fixed frame size and rate analysis is used to extract the MFCC and the GMM is used as a modelling technique. In this study, the feature vectors that discriminate the speakers well have been considered for speaker recognition under limited data. In [36], the fixed frame size and rate analysis is used to obtain the Glottal Flow Derivative (GFD) signal

which represents the excitation source information. The GFD is obtained by passing the LP residual signal through an integrator. In this study, the correlation-based similarity between two GFD signals is used for speaker verification under the constraint of limited data.

Studies made in [25, 26, 73, 83, 101] concentrated on modelling stage to improve the performance under limited data. These studies used the fixed frame size and rate as analysis technique and MFCC as feature. The GMM-UBM is used as a modelling technique in all the studies except the one in [73]. In this study, the notion of character encoding is used as a modelling technique. From these studies, we can understand that state-of-the-art speaker recognition systems use MFCC as feature and GMM-UBM for speaker modelling. This direction for alleviating the effect of limited data is obvious. This is because, the limited data leads to the sparse distribution for clustering and insufficient number of features for statistical modelling.

Apart from improved modelling it may be possible to improve the speaker recognition performance by developing new speech analysis, feature extraction and testing techniques. In this direction, we have already made an attempt to demonstrate that the MFSR analysis in the analysis stage improves the performance over the SFSR analysis. In the feature extraction stage, the combination of features like MFCC, its temporal derivatives (ΔMFCC, $\Delta\Delta$MFCC), LPR and LPRP improves the performance over the individual and other combination of features. The combined LVQ-GMM-UBM modelling improves the performance over the individual and other combined modelling in the modelling stage.

In this study, an attempt is made to see the significance of the *integrated systems* using MFSR analysis, different features and LVQ and GMM-UBM modelling techniques in the analysis, feature extraction and modelling stages, respectively under limited data condition. The rest of the chapter is organized as follows: Section 6.2 briefly describes our earlier studies for speaker recognition under limited data condition. In Section 6.3, different techniques involved in integrated systems are presented. Section 6.4 presents experimental studies using the integrated systems. Finally, summary of the work demonstrated in this chapter is given Section 6.5.

6.2 Individual Systems for Limited Data Speaker Recognition

In this section, we briefly describe the studies we earlier conducted for speaker recognition and present the important experimental results for comparison purposes with the integrated systems. To study the effectiveness of the integrated systems, we use the YOHO [119] and TIMIT [120] databases which are also used for evaluating the individual systems.

6.2.1 Limited Data Speaker Recognition using MFSR Analysis

In this study, we explored the significance of MFSR analysis of speech for speaker recognition. The feature extraction technique we used was MFCC and VQ as a modelling technique. The experimental results obtained for different codebook sizes for a set of the first 30 speakers taken from the YOHO database, each having 3 sec training and testing data are shown in Table 6.1. The results show that MFSR yields the highest performance of 90% for codebook of size 128. The performance is higher than the performance of SFSR which provides 70% for codebook of size 64. The same trend was observed for the cases of different data sizes of 6, 12 and 24 sec, 30 speakers, whole database of YOHO and also for the TIMIT database.

Table 6.1: Speaker recognition performance (%) for the frist 30 speakers taken from the YOHO databse, each having *3 sec training and test data* using SFSR and MFSR analysis.

Analysis	Feature	Modelling	Codebook size			
			16	32	64	128
SFSR	MFCC	VQ	63.33	66.67	**70.00**	60.00
MFSR	MFCC	VQ	80.00	80.00	86.67	**90.00**

6.2.2 Limited Data Speaker Recognition using Combination of Features

In this study we evaluated different feature extraction techniques. For this study we used SFSR as analysis technique and VQ as modelling technique. The different feature extraction techniques evaluated are MFCC, ΔMFCC, $\Delta\Delta$MFCC, LPR and LPRP to know the amount of speaker information present in them. The experimental results obtained for a set of the first

30 speakers taken from the YOHO database, each having 3 sec training and testing data are shown in Table 6.2. The results show that the combined features yield the highest performance of 86.67%. This performance is better than any of the individual features. The same trend was observed for the cases of different data sizes of 6, 12 and 24 sec, 30 speakers, whole database of YOHO and also for the TIMIT database.

Table 6.2: Speaker recognition performance (%) using different feature extraction techniques for the first 30 speakers taken from the YOHO databse, *each having 3 sec training and testing data* .

Analysis	Features	Modelling	Codebook size			
			16	32	64	128
SFSR	MFCC	VQ	63.33	66.67	**70.00**	60.00
SFSR	Δ	VQ	30.00	23.33	**36.67**	26.67
SFSR	$\Delta\Delta$	VQ	20.00	20.00	**33.33**	23.33
SFSR	LPR	VQ	13.33	23.33	30.00	**46.67**
SFSR	LPRP	VQ	30.00	30.00	**46.67**	23.33
	MFCC+Δ+$\Delta\Delta$+LPR+LPRP				**86.67**	

6.2.3 Limited Data Speaker Recognition using LVQ-GMM-UBM Modelling

In this study, SFSR is used as analysis technique and MFCC as feature. In the modelling stage, first the performance of different modelling techniques like CVQ, FVQ, SOM, LVQ, GMM and GMM-UBM (NIE and IE) are studied individually. The modelling techniques are then combined based on their performance. The best performance of individual models and the performance of different combined models are given in Table 6.3. Among the different combined modelling techniques we found that the combined (LVQ-GMM-UBM-NIE) and (LVQ-GMM-UBM-IE) systems yield the highest performance of 83.33% and 86.67%, respectively. The same trend was observed for the cases of different data sizes of 6, 12 and 24 sec, 30 speakers, whole database of YOHO and also for the TIMIT database.

Table 6.3: Individual and combined modelling based speaker recognition performance (%) for the first 30 speakers taken from the YOHO database, each having *3 sec training and test data.*

Analysis	Feature	Modelling	Codebook size			
			16	32	64	128
SFSR	MFCC	CVQ	63.33	66.67	**70.00**	60.00
SFSR	MFCC	FVQ	70.00	**76.67**	73.33	70.00
SFSR	MFCC	SOM	**73.33**	**73.33**	70.00	**73.33**
SFSR	MFCC	LVQ	73.33	**80.00**	73.33	66.67
SFSR	MFCC	GMM	**73.33**	40.00	36.67	13.33
SFSR	MFCC	GMM-UBM-NIE	60.00	60.00	63.33	**73.33**
SFSR	MFCC	GMM-UBM-IE	60.00	66.67	73.33	**83.33**
		LVQ-FVQ	80.00			
		LVQ-GMM	80.00			
		LVQ-GMM-UBM-NIE	**83.33**			
		LVQ-GMM-UBM-IE	**86.67**			

6.3 Integrated Systems for Limited Data Speaker Recognition

The aforementioned studies have shown improved performance under limited data condition. As we have already mentioned that the studies are made individually. However, though the proposed individual systems show the improved performance over the existing systems, the performance is not satisfactory for large database. Therefore, improving the performance for such requirement is of great interest and a challenging task. To improve the performance we have integrated the techniques demonstrated individually. This includes MFSR in the analysis stage, different features in the feature extraction stage and LVQ and GMM-UBM models in the modelling stage. Since these techniques have already shown improved performance, integrating them may further improve the performance. This is the motivation for this study. The scheme for the development of the integrated systems are shown as block diagram in Figure 6.1. The speaker recognition stages in the integrated systems are as follows:

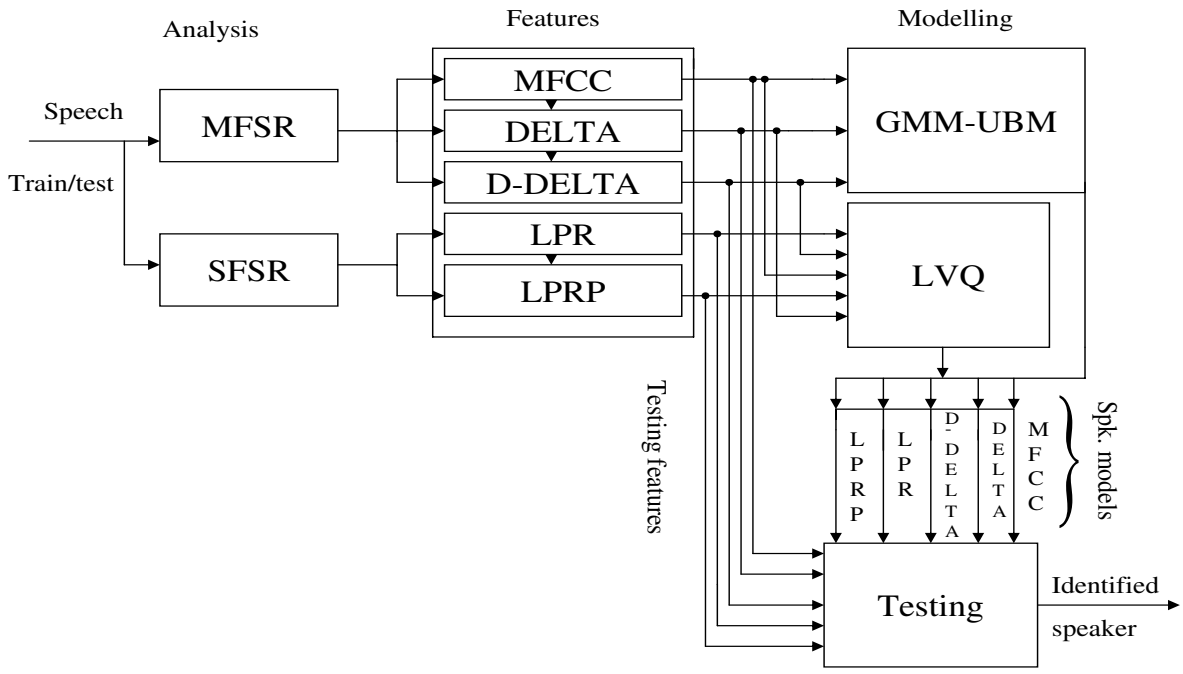

Figure 6.1: Block diagram of integrated systems based speaker recognition for limited data condition.

6.3.1 Analysis Stage

In this stage, speech is mainly analyzed using MFSR analysis for speaker recognition. Since the MFSR analysis is developed from the frequency domain perspective, SFSR analysis is also used in this stage to analyze the speech which is to be compared in the time domain.

6.3.2 Feature Extraction Stage

In this stage, the features like MFCC, ΔMFCC, ΔΔMFCC, LPR and LPRP are extracted from the speech signal. The MFCC and its derivatives are extracted using MFSR analysis. The LPR and LPRP features are extracted using SFSR analysis. Since in LPR and LPRP cases residual samples are compared in the time domain, the MFSR analysis may not be effective.

6.3.3 Modelling Stage

In this stage, both LVQ and GMM-UBM modelling techniques are used for speaker modelling. The MFCC, ΔMFCC and ΔΔMFCC features are modeled individually both by LVQ and

GMM-UBM modelling. The LPR and LPRP features are modeled by using only LVQ. Since we are not doing any parameterization for extracting LPR and LPRP, the use of GMM-UBM for modelling them may not be effective [100].

6.3.4 Testing Stage

In this stage, the speaker test data is analyzed using MFSR analysis to extract MFCC and its derivatives and SFSR analysis to extract LPR and LPRP features. These features are compared with the respective modelling techniques using either the Euclidean distance or maximum likelihood ratio test to identify the speaker of the test data.

6.4 Limited Data Speaker Recognition Studies using Integrated Systems

The modules in the block diagram of Figure 6.1 show that there are 8 independent integrated systems possible. These are termed as S_1 to S_8 and are shown in Table 6.4. The details of these systems are as follows:

Table 6.4: Integrated systems.

Integrated systems	Analysis	Features	Modelling
S_1	MFSR	MFCC	LVQ
S_2	MFSR	MFCC	GMM-UBM-NIE
S_3	MFSR	ΔMFCC	LVQ
S_4	MFSR	$\Delta\Delta$MFCC	LVQ
S_5	MFSR	ΔMFCC	GMM-UBM-NIE
S_6	MFSR	$\Delta\Delta$MFCC	GMM-UBM-NIE
S_7	SFSR	LPR	LVQ
S_8	SFSR	LPRP	LVQ

- S_1 *Integrated System*: This system uses MFSR analysis to extract MFCC features and LVQ for speaker modelling (MFSR-MFCC-LVQ). Since the performance of the LVQ depends on the parameters such as learning rate (η) and number of iterations, we have fine tuned them to avail the best performance. The performance of the system S_1 for the first

30 speakers of the YOHO database using 3 sec training and testing data for different parameter values are given in Tables 6.5. It provides the highest performance of 93.33% for iterations of 600*128 and η of 0.05. This result is higher than that of MFSR-MFCC-VQ which provides 90% and given in Table 6.1.

Table 6.5: Speaker recognition performance (%) for the first 30 speakers taken from the YOHO database using *3 sec training and testing data* for integrated system S_1 i.e., MFSR-MFCC-LVQ technique.

Iterations	η	Codebook Size (CS)			
		16	32	64	128
550*CS	0.06	73.33	86.67	83.33	83.33
600*CS	0.06	83.33	83.33	90.00	83.33
700*CS	0.05	76.67	80.00	90.00	90.00
650*CS	0.05	80.00	83.33	86.67	86.67
600*CS	0.05	80.00	83.33	86.67	**93.33**
600*CS	0.04	76.67	80.00	80.00	83.33
550*CS	0.05	80.00	80.00	83.33	86.67
500*CS	0.05	73.33	76.67	80.00	83.33

- S_2 *Integrated System*: This system uses MFSR analysis to extract MFCC features and GMM-UBM for speaker modelling. In case of GMM-UBM, speakers used for UBM training are not used for speaker recognition study (MFSR-MFCC-GMM-UBM-NIE). The system S_2 gives the recognition performance of 80% for the 30 speakers of the YOHO database using 3 sec training and testing data. This performance is higher than that of SFSR-MFCC-GMM-UBM-NIE which provides 73.33% and given in Table 6.3.

- S_3 *Integrated System*: This system uses MFSR analysis to extract ΔMFCC features and LVQ for speaker modelling (MFSR-ΔMFCC-LVQ). The performance of system S_3 for the 30 speakers of the YOHO database using 3 sec training and testing data for different LVQ parameter values are given in Table 6.6. It provides the highest performance of 56.67% for iterations of 600*128 and η of 0.06. This result is higher than that of SFSR-ΔMFCC-VQ which provides 36.67% and given in Table 6.2.

- S_4 *Integrated System*: This system uses MFSR analysis to extract $\Delta\Delta$MFCC features and

Table 6.6: Speaker recognition performance (%) for the first 30 speakers taken from the YOHO database using *3 sec training and testing data* for S_3 integrated system i.e., MFSR-ΔMFCC-LVQ technique.

Iterations	η	Codebook Size (CS)			
		16	32	64	128
600*CS	0.05	36.67	40.00	50.00	43.33
600*CS	0.06	30.00	33.33	50.00	**56.67**
600*CS	0.07	26.67	30.00	43.33	46.67
600*CS	0.04	33.33	40.00	46.67	46.67
550*CS	0.05	33.33	43.33	43.33	43.33
550*CS	0.06	30.00	30.00	40.00	46.67
500*CS	0.06	23.33	26.67	33.33	43.33

LVQ for speaker modelling (MFSR-ΔΔMFCC-LVQ). The performance of system S_4 for the 30 speakers of the YOHO database using 3 sec training and testing data for different LVQ parameter values are given in Table 6.7. It provides the highest performance of 43.33% for iterations of 600*128 and η of 0.05. This result is higher than that of SFSR-ΔΔMFCC-VQ which provides 33.33% and given in Table 6.2.

Table 6.7: Speaker recognition performance (%) for the first 30 speakers taken from the YOHO database using *3 sec training and testing data* for integrated system S_4 i.e., MFSR-ΔΔMFCC-LVQ technique.

Iterations	η	Codebook Size (CS)			
		16	32	64	128
500*CS	0.04	20.00	26.67	30.00	33.33
550*CS	0.05	23.33	26.67	36.67	40.00
600*CS	0.05	30.00	33.33	30.00	**43.33**
600*CS	0.06	30.00	33.33	40.00	33.33
600*CS	0.04	26.33	26.33	33.33	40.00
450*CS	0.06	23.33	30.00	40.00	40.00
400*CS	0.06	23.33	23.33	30.00	36.67

- S_5 *Integrated System*: This system uses MFSR analysis to extract ΔMFCC features and GMM-UBM for speaker modelling (MFSR-MFCC-GMM-UBM-NIE). This system gives the recognition performance of 26.67% for the 30 speakers of the YOHO database using 3 sec training and testing data. Since we have not used GMM-UBM in the individual combination of features system, the performance of S_5 is not compared with SFSR-ΔMFCC-GMM-UBM-NIE.

- S_6 *Integrated System*: This system uses MFSR analysis to extract ΔΔMFCC features and GMM-UBM for speaker modelling (MFSR-ΔΔMFCC-GMM-UBM-NIE). This system provides the recognition performance of 33.33% for the 30 speakers of the YOHO database using 3 sec training and testing data. Since we have not used GMM-UBM in the individual combination of feature system, the performance of S_6 is not compared with SFSR-ΔΔMFCC-GMM-UBM-NIE.

- S_7 *Integrated System*: This system uses SFSR analysis to extract LPR features and LVQ for speaker modelling (SFSR-LPR-LVQ). The performance of system S_7 for the 30 speakers of the YOHO database using 3 sec training and testing data for different LVQ parameter values are given in Tables 6.8. It provides the highest performance of 46.67% for iterations of 600*64 and η of 0.06. This result is same as that of SFSR-LPR-VQ which provides 46.67% and given in Table 6.2.

Table 6.8: Speaker recognition performance (%) for the first 30 speakers taken from the YOHO database using *3 sec training and testing data* for integrated system S_7 i.e., SFSR-LPR-LVQ technique.

Iterations	η	Codebook Size (CS)			
		16	32	64	128
800*CS	0.06	20.00	20.00	26.67	23.33
500*CS	0.05	23.33	23.33	30.00	26.67
550*CS	0.05	16.67	26.67	30.00	36.67
600*CS	0.06	26.67	33.33	**46.67**	40.00
600*CS	0.05	20.00	33.33	36.67	36.67
550*CS	0.06	20.00	23.33	40.00	33.33
600*CS	0.04	13.33	26.67	26.67	33.33

- S_8 *Integrated System*: This system uses SFSR analysis to extract LPRP features and LVQ for speaker modelling (SFSR-LPRP-LVQ). The performance of system S_8 for the 30 speakers of the YOHO database using 3 sec training and testing data for different LVQ parameter values are given in Tables 6.9. It provides the highest performance of 46.67% for iterations of 600*64 and η of 0.05. This result is also same as that of SFSR-LPRP-VQ which provides 46.67% and given in Table 6.2.

Table 6.9: Speaker recognition performance (%) for the first 30 speakers taken from the YOHO database using *3 sec training and testing data* for integrated system S_8 i.e., SFSR-LPRP-LVQ technique.

Iterations	η	Codebook Size (CS)			
		16	32	64	128
600*CS	0.06	26.67	43.33	36.67	36.67
600*CS	0.05	23.33	33.33	**46.67**	26.67
600*CS	0.04	26.67	33.33	23.33	40.00
550*CS	0.06	23.33	30.00	40.00	33.33
500*CS	0.06	20.00	26.67	30.00	33.33
700*CS	0.05	33.33	36.67	30.00	40.00

The results of all the integrated and individual systems are given in Table 6.10. In summary, the integrated systems S_1, S_2, S_3, and S_4 improved the performance over the respective individual systems. Since the GMM-UBM modelling is not used in the individual feature extraction system, the performance of the integrated systems S_5 and S_6 are not compared with the respective individual systems. Though the fine tuning of the LVQ parameters, the performance of the integrated systems S_7 and S_8 is same as that of the respective individual systems in our experimental condition. From these results we can understand that majority of the integrated systems improve the performance over individual systems.

Table 6.10: Performance of the integrated and individual systems for the first 30 speakers taken from the YOHO database using *3 sec training and testing data.* In the table Per. indicates the performance.

Integrated systems	Per. (%)	Individual systems	Per. (%)
S_1 (MFSR-MFCC-LVQ)	93.33	MFSR-MFCC-VQ	90.00
S_2 (MFSR-MFCC-GMM-UBM-NIE)	80.00	SFSR-MFCC-GMM-UBM-NIE	73.33
S_3 (MFSR-ΔMFCC-LVQ)	56.67	SFSR-ΔMFCC-VQ	36.67
S_4 (MFSR-$\Delta\Delta$MFCC-LVQ)	43.33	SFSR-$\Delta\Delta$MFCC-VQ	33.33
S_5 (MFSR-ΔMFCC-GMM-UBM-NIE)	26.67	SFSR-ΔMFCC-GMM-UBM-NIE	-
S_6 (MFSR-$\Delta\Delta$MFCC-GMM-UBM-NIE)	33.33	SFSR-$\Delta\Delta$MFCC-GMM-UBM-NIE	-
S_7 (SFSR-LPR-LVQ)	46.67	SFSR-LPR-VQ	46.67
S_8 (SFSR-LPRP-LVQ)	46.67	SFSR-LPRP-VQ	46.67

The experimental results, so far we demonstrated are pertaining to the 30 speakers case, each having 3 sec training and testing data. To verify the effectiveness of the integrated systems for the other data sizes of 6, 12 and 24 sec, we have conducted the same experiments using only the best performed integrated systems S_1 and S_2. The experimental results of the same are shown with the baseline system i.e., SFSR-MFCC-VQ and the best performed individual systems MFSR-MFCC-VQ and SFSR-MFCC-GMM-UBM-NIE in Figure 6.2. It is evidenced from the figure that the individual systems outperform the baseline system. The integrated systems give further improved performance over the individual systems. The integrated systems show significant improvement in the performance up to 12 sec and above 12 sec the performance of baseline, individual and integrated systems approach one another. Therefore, in order to ascertain the effectiveness of the integrated systems for the whole database, the experiment is carried up to 12 sec training and testing data and the results are shown in Figure6.3. The trend in the experimental results shown in Figure 6.3 resemble that in Figure 6.2 which implies that the proposed integrated systems show a similar behavior for the large database also.

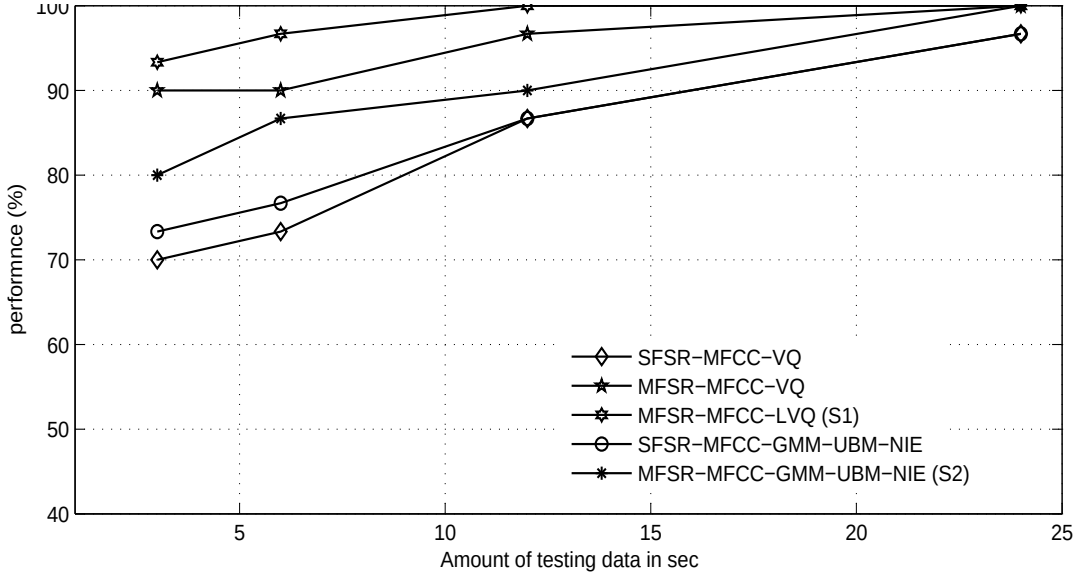

Figure 6.2: Speaker recognition performance for the baseline, individual and integrated systems for different sizes of training and testing data for the first 30 speakers taken from the YOHO database.

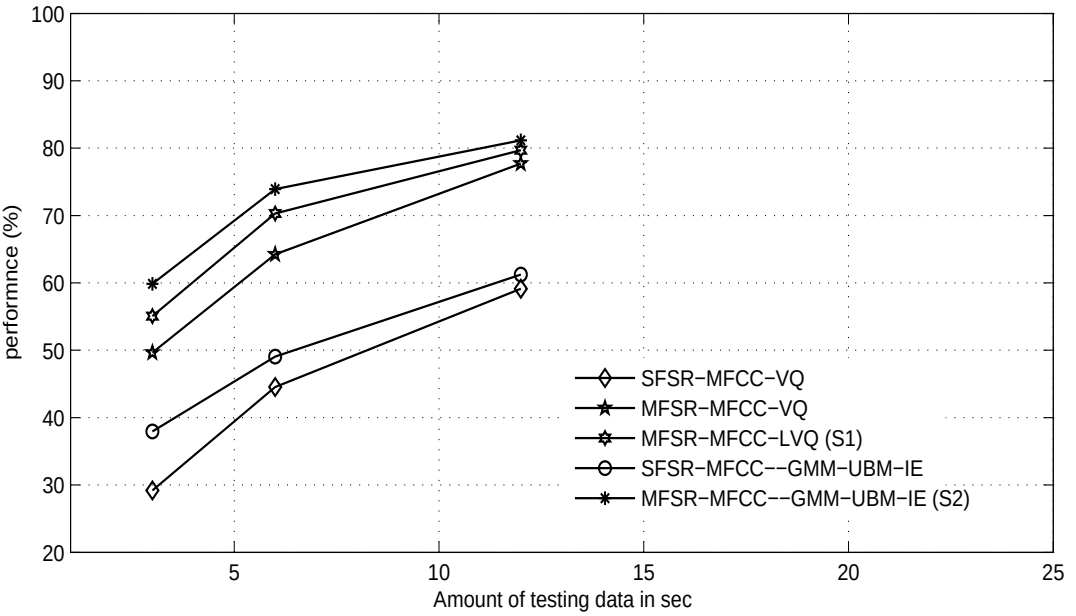

Figure 6.3: Speaker recognition performance for the baseline, individual and integrated systems for different sizes of training and testing data for 138 speakers taken from the YOHO database.

To verify the robustness of the proposed integrated systems, we conducted the experiments on the TIMIT database also. Experimental studies are conducted as that of the YOHO database setup and the results are shown in Figure 6.4 and 6.5 for a set of first 30 and 138 speakers, respectively. The experimental results of the TIMIT database also resemble those for the YOHO database irrespective of speaker population and amount of data. Hence, we suggest that the integrated systems can be used for improving the speaker recognition performance under limited data condition.

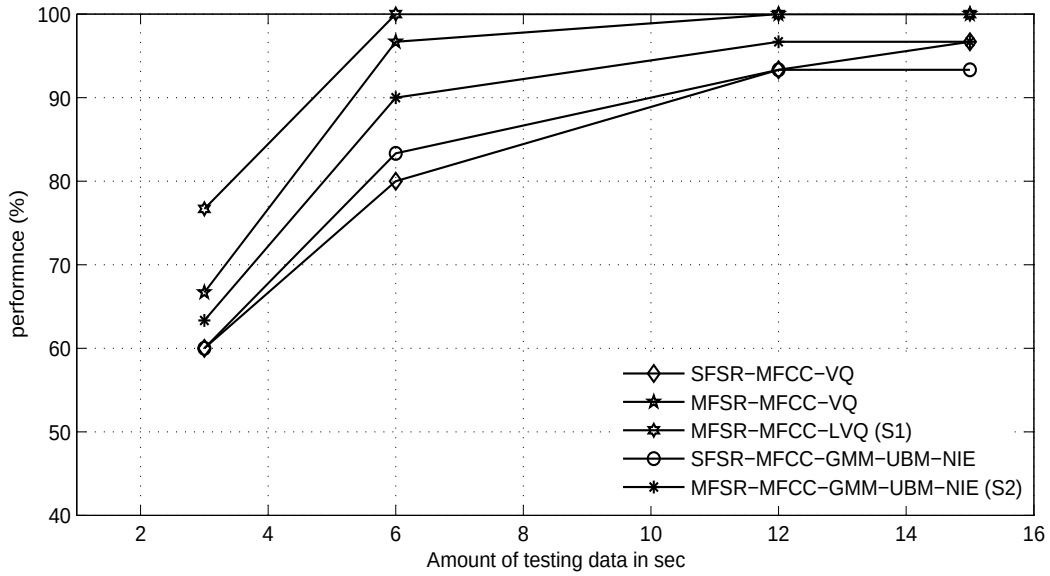

Figure 6.4: Speaker recognition performance for the baseline, individual and integrated systems for different sizes of training and testing data for the first 30 speakers taken from the TIMIT database.

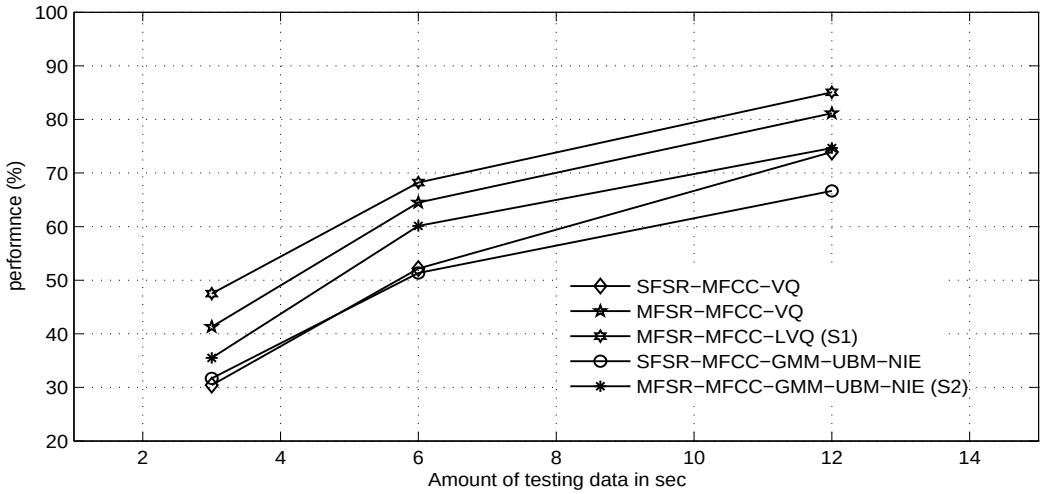

Figure 6.5: Speaker recognition performance for the baseline, individual and integrated systems for different sizes of training and testing data for the first 138 speakers taken from the TIMIT database.

6.5 Summary

In this chapter, first we briefly discussed our previous studies for speaker recognition under limited data condition. The first study presented that the MFSR analysis of speech is better than the SFSR analysis. The second study expressed that the combined features (MFCC, Δ, $\Delta\Delta$, LPR and LPRP) system performs better than the individual feature systems. The third study evidenced that the combined LVQ-GMM-UBM modelling technique gives better performance than the individual modelling techniques. We then integrated the techniques for speaker recognition under limited data condition. In the integrated systems, analysis techniques used are MFSR and SFSR. These techniques are used to extract the features like MFCC, Δ, $\Delta\Delta$, LPR and LPRP in the feature extraction stage. These features are modeled both by LVQ and GMM-UBM in the modelling stage. As a result, the integrated systems show the improved performance over the individual systems.

In the next chapter we study the effectiveness of combining the evidences from different integrated systems using different combination schemes at the abstract, rank and measurement levels.

7

Combining Evidences for Limited Data Speaker Recognition

Contents

In the previous chapter, we studied the integrated systems for speaker recognition. We found that the integrated systems improve the performance over the individual systems. In this chapter, we demonstrate the significance of combining evidences from the integrated systems for speaker recognition under limited data condition. For combining evidences, first different conventional combination techniques are explored at the abstract, rank and measurement levels. This work then proposes Strength Voting (SV), Weighted Ranking (WR) and Supporting Systems (SS) as combining techniques at the abstract, rank and measurement levels, respectively. Finally, using the proposed techniques Hierarchical Combination (HC) is also proposed. The experimental results show that the proposed combination techniques improve the performance over the integrated systems.

7.1 Introduction

The work so far in the thesis concentrated first on selecting efficient techniques suitable for the analysis, feature extraction and modelling stages. This includes MFSR analysis for the analysis stage. The combination of features like MFCC, its temporal derivatives (Δ, $\Delta\Delta$), LPR and LPRP for the feature extraction stage. The combination of LVQ and GMM-UBM modelling techniques for the modelling stage. The work then concentrated on building integrated systems using these techniques. As a result, we have seen that there are 8 integrated systems possible and most of them improved the performance over the individual systems. Further, combining the evidences from the integrated systems may improve the performance over the integrated systems. This is the motivation for the work.

The majority of the studies made for speaker recognition under limited data condition have concentrated on improving the performance by better modelling the speaker [25, 26, 83, 99, 101]. However, there are several techniques in the literature for combining the decision of multiple classifiers to improve the performance [109, 133–135]. In [133], attempts have been made to combine individual classifiers using methods like Bayesian formalism, voting method and Dempster-Shafer (D-S) theory for handwriting recognition. In [134], to make a combined decision using multiple classifier outputs for machine printed word and character recognition,

ranked voting and logistic regression methods are proposed. In [109], voting method is used for speaker identification based on the results of various resolution filterbanks. The combination of decision by majority voting and divide and conquer is proposed for pattern classification in [135]. It is shown that the combined framework gives better performance than the individual method.

In this work, we try with some of the existing combination techniques to combine the integrated systems decision to study the effectiveness of the methods under limited data condition. In addition, new combination techniques will be proposed to improve the performance. The present work proposes Strength Voting (SV), Weighted Ranking (WR), Supporting Systems (SS) and Hierarchical Combination (HC) techniques for combining the evidences. The SV technique ensures more voting power to the best performing integrated system. The performance of the integrated systems is also used to obtain the rank of speaker in WR technique. The integrated systems which are supporting the speakers are considered for combination in SS technique. In HC technique, the combination of SV, WR and SS is used hierarchically to select the speaker of the test data. The rest of the chapter is organized as follows: In Section 7.2, previous study based on the integrated systems is briefly presented. Section 7.3 proposes combination techniques for speaker recognition under limited data condition. Finally, summary of the work presented in this chapter is given in Section 7.4.

7.2 Integrated Systems for Limited Data Speaker Recognition

In this section, we briefly describe our previous study based on the integrated systems for speaker recognition and present the important experimental results for comparison purposes with combination schemes. In the integrated systems, analysis techniques used are MFSR and SFSR. These techniques are used to extract the features like MFCC, Δ, $\Delta\Delta$, LPR and LPRP in the feature extraction stage. These features are modeled both by LVQ and GMM-UBM in the modelling stage. As a result, there are 8 integrated systems possible and are shown in Table 7.1. The experimental results for 30 speakers, each of having 3 sec training and testing data using

these integrated systems are also shown in Table 7.1. We observed in the previous chapter that most of the integrated systems give better results than that of the individual systems.

Table 7.1: The integrated systems and their performance.

Integrated systems	Analysis	Features	Modelling	Performance (%)
S_1	MFSR	MFCC	LVQ	93.33
S_2	MFSR	MFCC	GMM-UBM	80.00
S_3	MFSR	Δ	LVQ	56.67
S_4	MFSR	$\Delta\Delta$	LVQ	43.33
S_5	MFSR	Δ	GMM-UBM	26.67
S_6	MFSR	$\Delta\Delta$	GMM-UBM	33.33
S_7	SFSR	LPR	LVQ	46.67
S_8	SFSR	LPRP	LVQ	46.67

7.3 Combination Techniques for Limited Data Speaker Recognition

Most of the existing combination schemes may be broadly grouped into three categories, namely, abstract level, rank level and measurement level [133]. The abstract level combination will have decision as the output of individual classifiers. The decision from the individual classifiers are further combined to obtain improved decision. In the rank level combination the output of the individual classifiers is in terms of ranks and this information is used for combination and decision. The measurement level combination scheme uses the output of the individual classifiers available as measurement values for combining and making decision. The maximum benefit can be obtained during combination using measurement level scheme, but the limitation is the susceptibility of the combination result for the poor performing individual system measurement. Alternatively, the abstract level provides robustness, but gain achieved is relatively less. The rank level combination performance is better than the abstract level, but poorer than measurement level. The scheme for combining the evidences from the integrated systems are shown as block diagram in Figure 7.1. To study the effectiveness of the combination techniques, we use the YOHO [119] and TIMIT [120] databases which are also used for evaluating the integrated systems.

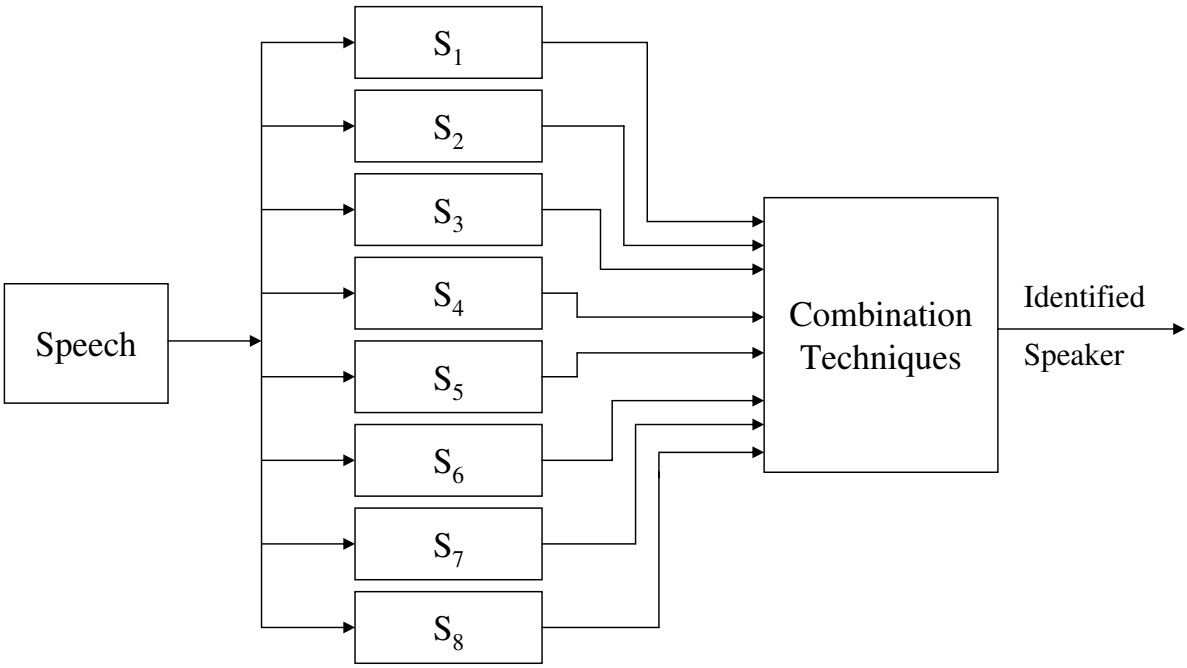

Figure 7.1: Block diagram of combining evidences from the integrated systems for speaker recognition under limited data condition.

7.3.1 Abstract Level Combination

In this technique, each system outputs the most likely speaker of the test data. The abstract level combination is therefore done at the decision level of each system to identify the speaker of the test data. The different abstract level schemes we explored are as follows:

7.3.1.1 Voting

Voting is the most common method used to combine more than one decision at the abstract level. The majority voting method goes with the decision when there is an agreement for it or at least more than half of the classifiers agree on it. Moreover, the decision depends on the type of application and requirements. In the present work, since some of the individual systems provide very poor performance, the most likely speaker decision may not be reliable. Hence achieving half way mark for voting is not possible. Therefore, if the speaker is identified by at least two systems, then the speaker is considered as the identified speaker. As a result, this combination scheme yields the recognition performance of 86.67% for a set of first 30 speakers.

The identified speakers and the performance are shown in Table 7.4. The advantages of this method are robustness, simple and easy to perform. However, the recognition performance is less than that of the integrated system S_1 alone which provides 93.33% and given in Table 7.1. The poor performance may be due to the poor performance of some integrated systems. To improve the recognition performance, we have proposed *strength voting* as a variant to the voting method.

7.3.1.2 Strength Voting (SV)

In the case of majority voting, the decision is accepted as true if it is agreed by all or at least majority i.e., more than half way mark. This is justifiable if all the integrated systems are equally competent. In the present work since some of the individual systems provide poor performance, they are not equally competent like the best performing systems. To take care of this, the threshold for voting was lowered from half way mark (4 systems) to just two systems. In spite of this, the combined system in the voting is unable to provide better performance than the best performing integrated system. This implies that simple lowering the threshold for voting is insufficient. Alternatively, we may benefit by taking into account about the performance of the integrated systems. One approach is to provide more voting power to the best performing system and accordingly lower voting power to the poorly performing systems. However, it also needs to be ensured that every system will have a voting power, irrespective of their performance.

To take care of both the issues, the following scheme is proposed: Among all the integrated systems, the system with the lowest performance is given a voting power of one vote. This ensures voting power to all the systems. Further, the other systems are given voting power of more than one vote based on their performance. In particular, the number of votes to a system is derived based on the ratio of its performance to the lowest performance system and rounded the ratio to the nearest next higher integer value. Thus the system with higher performance is given more voting power and hence this combination scheme is termed as Strength Voting (SV). The number of votes from the strength voting scheme for each of the system are shown

in Table 7.2. The SV works as follows: For the given test data, all the integrated systems will output the most likely speakers. Each of these output speakers are voted according to the voting power of the integrated systems. The speaker who gets the total maximum votes from different integrated systems is declared as the identified speaker. The identified speakers and the performance of SV are shown in Table 7.4. It gives 96.67% performance for a set of first 30 speakers. The performance is higher than the best performing integrated systems which are given in Table 7.1. Although this combined system gives good recognition performance, vote tie problem may occur. We have broken the tie using a strict linear ordering [134].

Table 7.2: Number of votes for the integrated systems based on their performance.

Integrated systems	Performance (%)	No. of Votes
S_1	93.33	4
S_2	80.00	3
S_3	56.67	2
S_4	43.33	2
S_5	26.67	1
S_6	33.33	2
S_7	46.67	2
S_8	46.67	2

7.3.2 Rank Level Combination

In the rank level combination, each system produces a rank list of speakers based on the frame scores for the test data of a speaker. The combination is done at the rank level to identify the speaker of the test data. The different rank level techniques we explored are as follows:

7.3.2.1 Borda count (BC)

This method is originally a voting method in which each system expresses the ranking of the speaker. The speaker recognition system expects that the genuine speaker should get the highest rank to yield good performance. However, due to the methods employed in the system the genuine speaker may slide down in the ranking from the first position. Under such situation the rank of the genuine speaker can be improved by considering the speaker rank from other

systems. In this work, each system identifies the order of the frame ratio of the speakers for the given test data. The speaker who scores the highest frame ratio will be assigned top rank and each subsequent speaker gets one vote less. The ranking from all the systems are averaged first and then the speakers are reranked using averaged ranking. The speaker who gets the top rank is chosen as the speaker of the test data. As a result, this combination system gives the recognition performance of 83.33% for a set of first 30 speakers. The identified speakers and the performance of BC are shown in Table 7.4. This performance is less than that of the integrated system S_1 alone which provides 93.33% and given in Table 7.1. In the recognition process, this technique does not consider the effectiveness of each system and hence degradation in performance. As an alternate to this method, we have proposed *weighted ranking* to enhance the robustness and to improve the performance.

7.3.2.2 Weighted Ranking (WR)

In this technique, the ranking of speaker is same as that of Borda count method. However, the decision is not only based on the rank of speaker, but also based on the performance of the integrated systems. We computed the Weighted Rank (WR) of each speaker R_{xw} considering all the systems and their performance using the equation.

$$R_{xw} = \sum_{i=1}^{N} R_{xi} \frac{P_i}{\sum_{j=1}^{N} P_j} \qquad x = 1...M \tag{7.1}$$

where M is the number of speakers, R_{xi} is the rank of speaker x in the system i, P_i is the performance of the system i and N is the number of systems. The speaker who gets the highest R_{xw} is chosen as the speaker of the test data. As a result, this combination scheme gives the recognition performance of 96.67% for a set of first 30 speakers. The identified speakers and the performance are shown in Table 7.4. The performance is higher than that of the best performing integrated systems which are given in Table 7.1.

7.3.3 Measurement Level Combination

Each of the 8 integrated systems i. e., S_1 to S_8 generate the frame scores of the speakers for the same test data. The frame scores by the systems are considered as the measurement level information of the test data. The different measurement level techniques we attempted are as follows:

7.3.3.1 Linear Combination of Frame Ratio (LCFR)

This combination technique considers all the systems equally and combines the frame scores of the respective speakers of all the systems. The method of combination is as follows: Let $x_{i1}, x_{i2}, ..., x_{iN}$ be the normalized frame scores of the test data with respect to the total number of frames of a speaker for the different integrated systems i and N is the number of speaker models. The frame scores are linearly added which results in $z_1, z_2, ..., z_N$, where

$$z_j = \sum_{i=1}^{8} x_{ij} \tag{7.2}$$

The speaker with the highest number of combined frame score is recognized as the final speaker of the test speech data. The identified speakers and the performance of the LCFR for a set of the first 30 speakers are shown in Table 7.4. The LCFR yields the recognition performance of 86.67% which is less than that of the integrated system S_1 alone which provides 93.33% and given in Table 7.1. It seems that the reason of performance degradation is the poor performance of some integrated systems. On the other hand, the same combination scheme improved the performance in our earlier studies, may be due to the moderate performance of most of the systems. Therefore, in order to improve the performance of the speaker recognition system, alternate approaches are required where the performance of the systems are also considered for the decision.

7.3.3.2 Weighted LCFR (WLCFR)

In this technique, weighting factor of each system is computed based on the total performance of all the systems. The weighting factor of each system is given by the ratio of its performance to the total sum performance of all the systems and is shown in Table 7.3 for each of the eight systems S_1 to S_8. The frame ratio of each system are now multiplied with the corresponding weighting factors and then the speaker identification is performed as in LCFR. Hence it is termed as Weighted LCFR (WLCFR) scheme. The identified speakers and the performance of the WLCFR are shown in Table 7.4. The WLCFR yields the recognition performance of 86.67%. Despite the use of the weighting factor, the WLCFR approach gives the same performance as that of the LCFR system. The poor combination performance may be due to the reason quoted in the LCFR. Hence, other combination schemes suitable for highly varying individual performance systems are required. As a result, we are proposing a new measurement level scheme as a variant to WLCFR. This scheme is termed as *supporting systems* combination.

Table 7.3: Weighting factor for the integrated systems based on their performance.

Integrated systems	Performance (%)	Weighting factor
S_1	93.33	0.23
S_2	80.00	0.19
S_3	56.67	0.13
S_4	43.33	0.10
S_5	26.67	0.06
S_6	33.33	0.07
S_7	46.67	0.10
S_8	46.67	0.10

7.3.3.3 Supporting Systems (SS)

The development of SS combination scheme is motivated from the observations made in the LCFR and WLCFR schemes. The LCFR scheme is the simplest measurement level combination scheme works well when the performance of all the integrated systems are high or nearly same. In the present work some of the integrated systems are very good and some of them are equally bad. In such a case LCFR is bound to give poor combination and hence no improvement in the performance. The main drawback of the LCFR scheme is the equal weightage given to the measurements of all the systems. We may gain in combination if we use the additional knowledge of the integrated systems output. Taking this thought process, first the weight factors are derived for each of the integrated systems based on their performance as explained earlier. Now the measurement of the good systems are given more weightage and those of poor are penalized accordingly. This ensures the reduction of the damage caused by the poor performing systems. This lead to the WLCFR scheme. However, as observed in the results, the WLCFR scheme also did not improve the performance. This implies that the weighting scheme may not be able to reduce the damage due to the poor performance systems to minimum or negligible level. The reasons may be the weighting factor for each of these systems is higher and also there are more number of such systems. One of the schemes to take care of this is described next which we are calling as *Supporting Systems* (SS) scheme.

For each speaker in the population, we can identify the subset of the eight systems that support him/her. This depends on the nature of the speaker-specific information and hence modelling by the respective systems. For instance, to some of the speakers, the dominant speaker characteristics might have been manifested in the MFCC, Δ and $\Delta\Delta$ features. Similarly, for some other speakers, the speakers characteristics might have been manifested in the MFCC, LPR and LPRP features. In that condition we benefit more by combining only those integrated systems for the combination. The steps involved in the supporting systems combination are as follows:

(i) Identify the integrated systems which are supporting the speaker.

(ii) Apply the WLCFR combination technique on the supporting systems.

(iii) Normalize the frame score of each speaker by total number of the supporting systems.

(iv) Verify the speaker who provides the highest score as the identified speaker of the test data.

(v) Repeat the steps (i) to (iv) for each speaker in the population.

For instance, as shown in Table 7.4 the systems S_1, S_2, S_3, S_4 and S_7 support the *speaker-2*. Therefore, for *speaker-2* only the WLCFR of S_1, S_2, S_3, S_4 and S_7 are combined and normalized and then identify the speaker. Similarly, the systems S_1, S_2, S_3 and S_4 support the *speaker-3*. In this approach only the systems which are supporting the speaker are combined to identify the speaker of the test data and hence it is termed as Supporting Systems (SS) combination scheme.

It should be noted that, there are as many supporting systems combinations as that of the number of speakers in the population. The identified speakers and the performance of SS are shown in Table 7.4. It gives 96.67% performance for a set of 30 speakers. This performance is better than the best integrated system as shown in Table 7.1. As illustrated, the advantage of this technique is the intelligent choice and combination of subset of available systems that leads to the improved performance as compared to WLCFR. However, SS performance is also same as that of the SV and WR techniques. Even though the performance is same in all the three techniques, the set of speakers identified are different. Of course, the SV and WR identified the same set of speakers, but the approach used to achieve the same are different. Hence, to further gain the advantages of these three systems, we have combined them hierarchically.

7.3.4 Hierarchical Combination (HC)

The proposed combination schemes in the abstract, rank and measurement level are hierarchically combined to improve the performance. The proposed combination schemes are Strength Voting (SV), Weighted Ranking (WR) and Supporting Systems (SS), respectively. For the test data of a speaker, this technique first picks up only the top four speakers who get the highest votes based on the SV method. Then, the WR is computed only for the selected speakers and 50% of the speakers who get less WR are pruned out. Next, out of the two speakers, the speaker who gets the highest frame ratio in the SS would be the recognized speaker of the test data. The identified speakers and the performance of HC are shown in Table 7.4. It gives 100% performance for a set of first 30 speakers. The performance is higher than that of the integrated systems in Table 7.1 and the combination techniques in Table 7.4.

Table 7.4: Performance of the integrated and combination techniques speaker recognition systems with the identified speakers for the first 30 speakers of the YOHO database. In the table, $\surd$ indicates speaker identified, * indicates speaker not identified and P indicates the performance.

	Testing speakers																													
Systems	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**	**17**	**18**	**19**	**20**	**21**	**22**	**23**	**24**	**25**	**26**	**27**	**28**	**29**	**30**	***P (%)***
S1	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	*	✓	✓	✓	✓	✓	✓	✓	✓	✓	*	✓	✓	✓	✓	✓	✓	✓	✓	**93.33**
S2	*	✓	✓	✓	✓	✓	✓	✓	✓	*	*	✓	✓	✓	✓	*	✓	✓	✓	✓	✓	*	*	✓	✓	✓	✓	✓	✓	✓	**80**
S3	*	✓	✓	✓	✓	*	*	✓	✓	✓	*	*	✓	✓	✓	*	*	✓	✓	*	*	✓	*	✓	✓	*	✓	✓	*	*	**56.67**
S4	*	✓	✓	*	✓	✓	*	✓	*	✓	*	*	*	*	✓	✓	✓	✓	*	✓	*	✓	*	*	✓	*	*	*	*	*	**43.33**
S5	*	*	*	*	✓	✓	*	✓	*	*	*	*	*	*	*	*	*	✓	*	*	✓	*	*	*	*	*	✓	*	✓	✓	**26.67**
S6	*	*	*	*	✓	*	✓	✓	*	✓	*	*	✓	*	*	*	*	✓	*	✓	*	✓	*	*	✓	*	*	*	✓	*	**33.33**
S7	*	✓	*	*	✓	*	✓	*	*	✓	*	*	*	*	✓	*	*	✓	✓	✓	✓	✓	*	✓	✓	*	✓	*	✓	*	**46.67**
S8	*	*	*	✓	✓	*	*	✓	✓	✓	*	*	*	*	*	*	✓	✓	*	✓	*	✓	*	✓	✓	*	✓	*	✓	✓	**46.67**
Voting	*	✓	✓	✓	✓	✓	✓	✓	✓	✓	*	*	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	*	✓	✓	✓	✓	✓	✓	✓	**86.67**
SV	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	*	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	**96.67**
BC	✓	✓	✓	✓	✓	✓	✓	✓	*	✓	*	*	✓	✓	✓	✓	✓	✓	✓	✓	✓	*	✓	✓	✓	*	✓	✓	✓	✓	**83.33**
WR	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	*	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	**96.67**
LCFR	*	✓	✓	✓	✓	✓	✓	✓	✓	✓	*	✓	✓	✓	✓	✓	✓	✓	*	✓	✓	✓	*	✓	✓	✓	✓	✓	✓	✓	**86.67**
WLCFR	*	✓	✓	✓	✓	✓	✓	✓	✓	✓	*	✓	✓	✓	✓	✓	✓	✓	*	✓	✓	✓	*	✓	✓	✓	✓	✓	✓	✓	**86.67**
SS	✓	✓	✓	✓	✓	✓	✓	✓	*	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	**96.67**
HC	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	**100**

Most of the combination schemes give equal performance for the 30 speakers case and hence to verify the credibility of the proposed combination schemes, we also conducted the experiments for the whole database of 138 speakers. The experimental results for 3 sec training and testing data are shown in Table 7.5. The results show that the schemes used for the hierarchical combination in the 30 speakers study give significantly better result than that of the other combined schemes for large database also and hence improved result in the HC. It gives the recognition performance of 70.07% that is higher than the integrated system MFSR-MFCC-GMM-UBM-IE which provides 59.85%. The other combined techniques like SV, WR and SS also give relative improvement in the performance compared to the integrated system MFSR-MFCC-GMM-UBM-IE. Note that in the 138 speakers study the integrated system MFSR-MFCC-GMM-UBM-IE gives better results than the integrated system MFSR-MFCC-LVQ unlike in 30 speakers study. This is because the speaker considered for speaker recognition study are also included in the UBM training.

Table 7.5: Speaker recognition performance for 138 speakers taken from the YOHO database using the integrated and different combination techniques.

Integrated system and combination techniques	Performance (%)
MFSR-MFCC-LVQ	55.07
MFSR-MFCC-GMM-UBM-IE	59.85
Voting	48.33
Borda count (BC)	47.33
Linear Combination of Frame Ratio (LCFR)	44.20
Weighted LCFR (WLCFR)	44.20
Strength Voting (SV)	66.42
Weighted Ranking (WR)	62.32
Supporting Systems (SS)	67.15
Hierarchical Combination (HC)	70.07

To further verify the credibility of the proposed combination techniques, the experiments are also conducted for different data sizes of 6, 12 and 24 sec for the first 30 speakers. The experimental results of the same are shown with the best performed integrated systems S_1 i.e., MFSR-MFCC-LVQ and S_2 i.e., MFSR-MFCC-GMM-UBM-NIE in Figure 7.2. It is evidenced from the figure that the proposed combination techniques give improved performance over the integrated systems. Among the proposed combination schemes the HC gives the highest performance. The proposed combination techniques show significant improvement in the performance up to 12 sec and above 12 sec the performance of the integrated systems and the combination techniques approach one another. Therefore, for 138 speakers further experiments are conducted only for 6 and 12 sec to see the effectiveness of the combination techniques. The experimental results are shown in Figure 7.3. The trend in the experimental results shown in Figure 7.3 resemble that in Figure 7.2. This means that the proposed combination techniques show a similar behavior for the large database also.

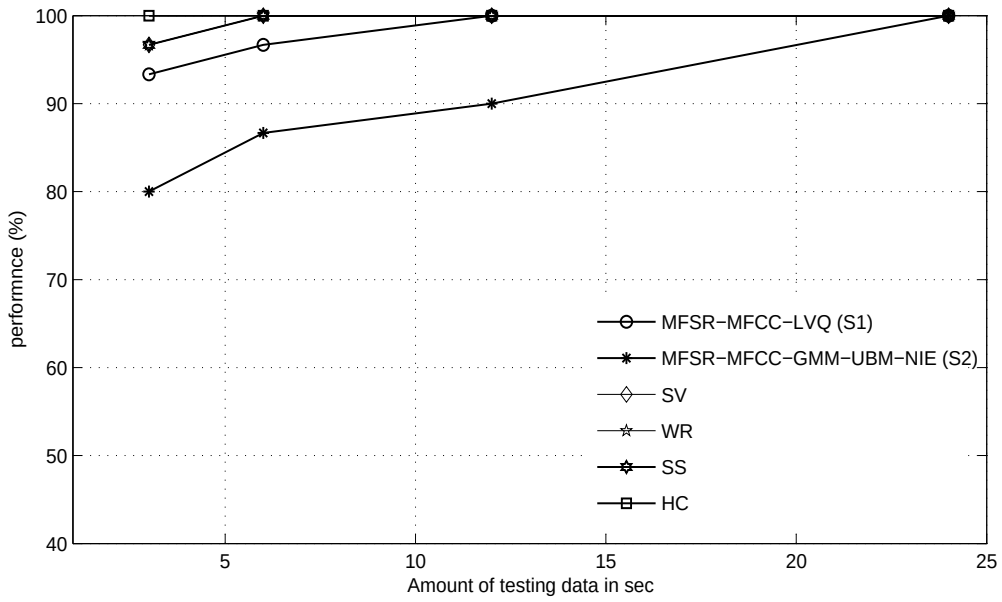

Figure 7.2: Performance of the integrated and combination techniques speaker recognition systems for the first 30 speakers taken from the YOHO database.

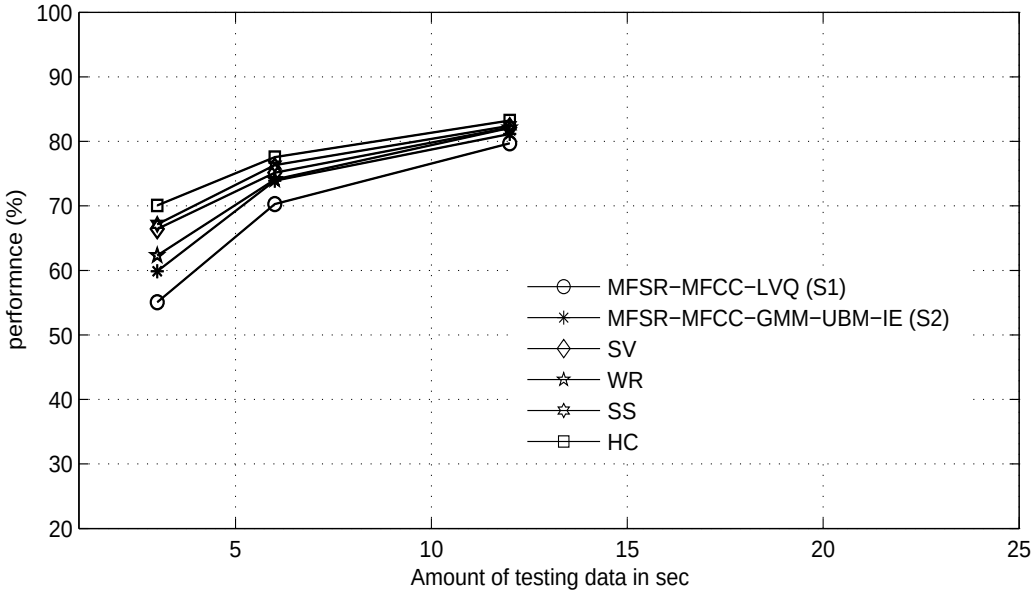

Figure 7.3: Performance of the integrated and combination techniques speaker recognition systems for 138 speakers taken from the YOHO database.

In order to further validate the credibility of proposed combination technique, the experiments are also conducted on the TIMIT database. Experimental studies are conducted as that of the YOHO database setup and the results are shown in Figure 7.4 and Figure 7.5 for a set of first 30 and 138 speakers, respectively. The experimental results for the TIMIT database also resemble those for the YOHO database irrespective of speaker population and amount of data. Hence, the combination technique can be used for speaker recognition under limited data condition.

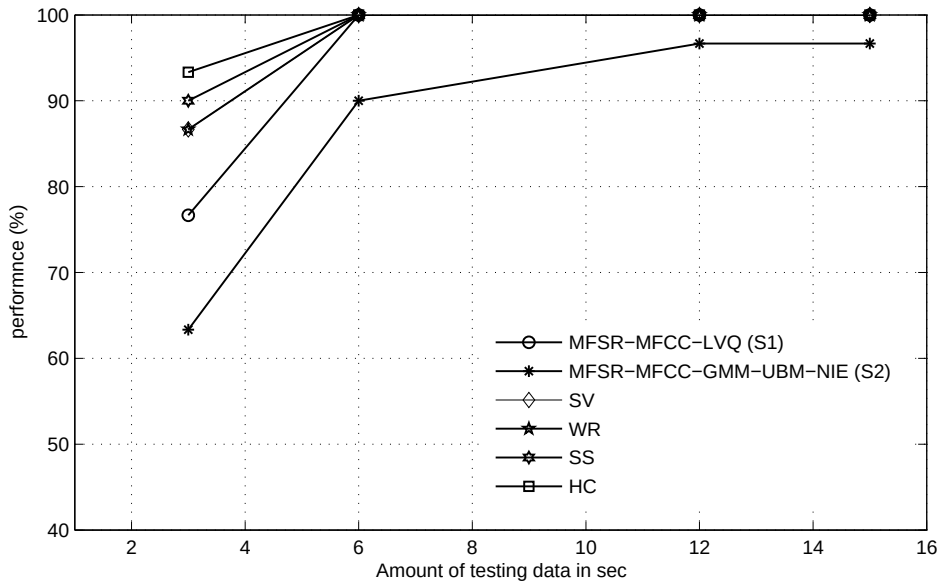

Figure 7.4: Performance of the integrated and combination techniques speaker recognition systems for the first 30 speakers taken from the TIMIT database.

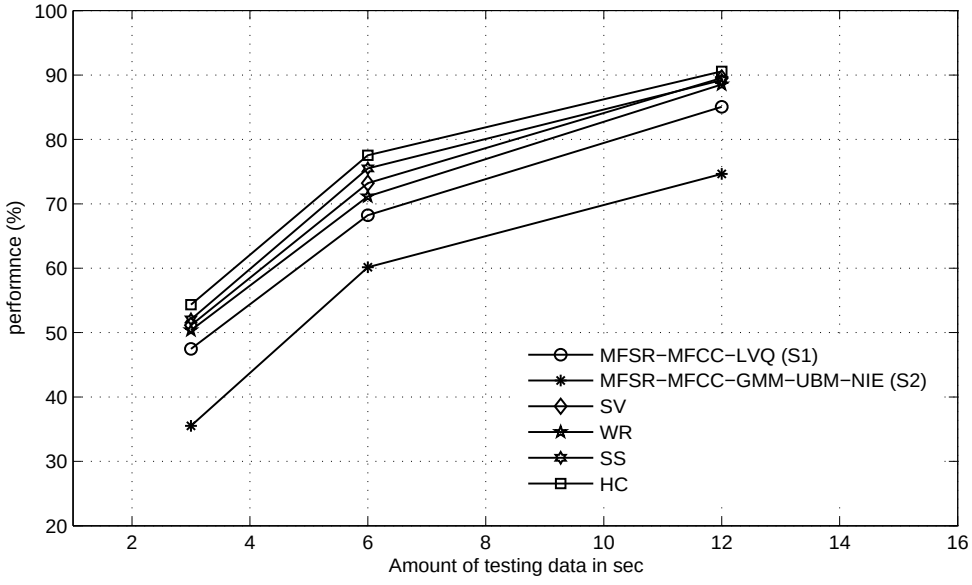

Figure 7.5: Performance of the integrated and combination techniques speaker recognition systems for the first 138 speakers taken from the TIMIT database.

7.4 Summary

In this chapter, first we briefly discussed our previous study based on the integrated systems and their performance. Then we combined the evidences from the integrated systems using different combination techniques at the abstract, rank and measurement levels. In the abstract level, the Strength Voting (SV) gives better recognition performance than the voting and the integrated systems. In the rank level, the Weighted Ranking (WR) gives better recognition performance than the Borda Count (BC) and the integrated systems. In the measurement level, the Supporting Systems (SS) gives better recognition performance than the Linear Combination of Frame Ratio (LCFR), Weighted LCFR (WLCFR) and the integrated systems. The combination schemes like SV, WR, SS are then further combined and proposed Hierarchical Combination (HC) scheme. It is found that the hierarchical combination provides the best results compared to the SV, WR and SS under limited data condition.

8

Summary and Conclusions

Contents

8.1 Summary of the Work

In this thesis some approaches for speaker recognition under limited data condition are proposed. These include approaches for speech analysis, feature extraction, modelling and combination techniques.

In the analysis stage, first we studied the performance of SFSR analysis under limited data condition. The experimental results show that the performance is not satisfactory under limited data. Alternatively, we analyzed speech signal using MFS, MFR and MFSR analysis techniques. To study the effectiveness of all the analysis techniques the feature we used is MFCC and VQ as a modelling technique. In order to verify the strength of different analysis techniques we conducted the experiments in three conditions, namely, 1) Limited training and sufficient testing data 2) Sufficient training and limited testing data 3) Limited training and testing data. The experimental results in all the conditions show that SFSR method is unable to produce more feature vectors whereas MFS, MFR and MFSR methods did in all conditions and hence an improved result over SFSR. Further, we found that among the analysis techniques MFSR gives the best performance. This shows that the recognition performance can be improved relatively by MFSR analysis technique under limited data condition.

The above study used only the MFCC as feature for speaker recognition. The study in the feature extraction stage demonstrated that the combination of evidences from different aspects of speech indeed improves the speaker recognition performance under limited data condition. In this study, using SFSR analysis, first we studied the performance of MFCC and its derivatives (Δ, $\Delta\Delta$) and LPCC and its derivatives (Δ, $\Delta\Delta$) under limited data. The experimental results show that the MFCC and its derivatives give better performance than the LPCC and its derivatives. Then we combined the evidences from LPR and LPRP with MFCC and its derivatives and LPCC and its derivatives individually. As a result, we found that the combined (MFCC, ΔMFCC, $\Delta\Delta$MFCC, LPR and LPRP) system gives better performance than the combined (LPCC, ΔLPCC, $\Delta\Delta$LPCC, LPR and LPRP) system. Therefore, we suggest that the combination of (MFCC, ΔMFCC, $\Delta\Delta$MFCC, LPR and LPRP) features can be used

for improving the performance under limited data condition.

The above studies used VQ as a modelling technique. State-of-the-art speaker recognition, in addition to VQ uses different modelling techniques like FVQ, SOM, LVQ, GMM and GMM-UBM. In the modelling stage study, using SFSR analyzed MFCC features, first we demonstrated the efficiency of different modelling techniques under limited data condition. The experimental results show that the fine tuned LVQ modelling technique gives better performance than that of the most widely used GMM-UBM modelling technique. Next, we combined different modelling techniques based on their performance which includes LVQ-FVQ, LVQ-GMM and LVQ-GMM-UBM to see the effectiveness under limited data. As a result, we found that the combined LVQ-GMM-UBM model gives better performance than the individual and other combined modelling techniques. Hence we suggest that the combined LVQ-GMM-UBM can be used as a modelling technique under limited data condition.

The aforementioned studies are demonstrated individually. This means that the proposed technique is used in the respective stage and the existing techniques in the remaining stages. The proposed individual techniques include MFSR for analysis stage, combination of features like MFCC, its temporal derivatives (Δ, $\Delta\Delta$), LPR and LPRP for feature extraction stage and combined LVQ-GMM-UBM modelling technique for modelling stage. In order to gain the advantages of these techniques, we integrated them at the respective stages. As a result, we found that there are 8 integrated systems possible and each one has different analysis, feature extraction and modelling techniques. The experimental results show that the majority of the integrated systems improved the performance over the individual systems. Therefore we suggest that the integrated systems can be used for improving the performance under limited data condition.

In order to further improve the performance over the integrated systems, we combined evidences from the integrated systems using different combination techniques at abstract, rank and measurement levels. We found that in the abstract level, the SV gives better recognition performance compared to the voting and the integrated systems. In the rank level, the WR gives better recognition performance compared to the BC and the integrated systems. In

the measurement level, the SS gives better recognition performance compared to the LCFR, WLCFR and the integrated systems. The combination schemes like SV, WR, SS are then further combined and proposed hierarchical combination scheme. It is found that the hierarchical combination provides the best results compared to the SV, WR and SS under limited data condition. Therefore, we suggest that the HC can be used for improving the speaker recognition performance under limited data condition.

Table 8.1 and 8.2 shows the summary of the results obtained using the different approaches for the YOHO and TIMIT database, respectively. There is a significant improvement in the performance of speaker recognition system starting from the system based on SFSR-MFCC-VQ to the proposed systems with Hierarchical Combination (HC). Hence, the speaker recognition system that we propose for limited data condition is shown in Figure 8.1. Thus as proposed initially in the proposal of thesis, it is indeed possible to improve the speaker recognition system performance by improving methods in each stage and by properly integrating them.

Table 8.1: Speaker recognition performance (%) for the first 30 and 138 speakers of the YOHO database using different approaches.

Systems	30 speakers			138 speakers		
	3 sec	6 sec	12 sec	3 sec	6 sec	12 sec
SFSR-MFCC-VQ	70.00	73.33	86.67	29.20	44.51	59.12
SFSR-MFCC-GMM-UBM	73.33	76.67	93.33	37.95	49.46	61.03
MFSR-MFCC-VQ	90.00	90.00	100	49.64	64.23	77.72
Combination of features	83.33	90.00	100	50.72	72.46	80.23
Combined LVQ-GMM-UBM	83.33	90.00	96.67	38.59	51.03	63.06
Integrated system (S_1)	93.33	96.67	100	59.85	73.91	81.16
Hierarchical Combination (HC)	100	100	100	70.07	77.54	83.23

Table 8.2: Speaker recognition performance (%) for the first 30 and 138 speakers of the TIMIT database using different approaches.

Systems	30 speakers			138 speakers		
	3 sec	6 sec	12 sec	3 sec	6 sec	12 sec
SFSR-MFCC-VQ	60.00	83.33	93.33	30.43	52.17	73.91
SFSR-MFCC-GMM-UBM	60.00	83.33	93.33	31.07	52.34	74.06
MFSR-MFCC-VQ	66.67	96.67	100	41.30	64.49	81.16
Combination of features	90.00	96.67	100	49.26	64.31	80.26
Combined LVQ-GMM-UBM	76.67	96.67	100	47.83	76.09	81.16
Integrated system (S_1)	76.67	100	100	47.48	68.22	85.06
Hierarchical Combination (HC)	93.33	100	100	54.35	77.54	90.58

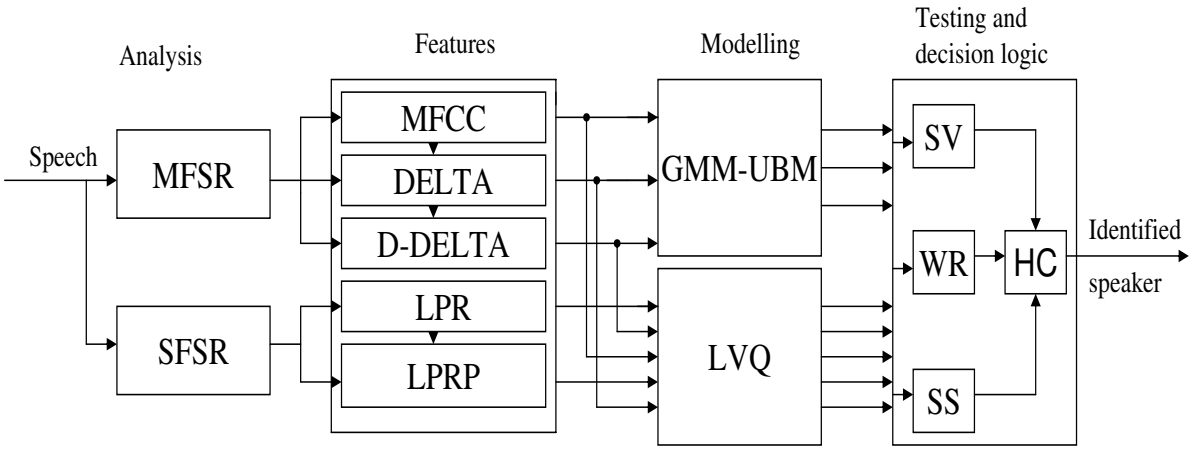

Figure 8.1: Block diagram of proposed speaker recognition system for limited data condition.

8.2 Contributions of the Work

The major contributions of the work reported in this thesis for speaker recognition under limited data condition includes,

- MFSR analysis of speech.
- Combination of features.
- Combined modelling techniques.
- Integrated systems, and
- Combining evidences.

The other contributions are,

- The use of Vector-Pulse Code Modulation (V-PCM) for speaker modelling from the LPR and LPRP features.
- Combination of LPR and LPRP information for improving the performance.
- The combination schemes such as SV, WR, SS and HC.

8.3 Scope for the Future Work

- We made an attempt to increase the number of feature vectors using MFSR analysis of speech. This includes multiresolution concept as MFS and multishifting concept as MFR analysis in which either the frame size or shift is changed. We have not made attempt to use signal processing tools like wavelets with inbuilt multiresolution properties. This can be explored for speech analysis.

- Other techniques for increasing the number of feature vectors like controlled noise addition can be explored.

- In this work existing features are used for improving the speaker recognition performance in limited data condition. Alternate to the existing features, other features for discriminating speakers under the constraint of limited data needs to be found out. For instance, voice quality features could be investigated.

- The combination of LVQ-GMM-UBM modelling is proposed in this work. The maximum *a posteriori* (MAP) adaptation technique is used in the GMM-UBM modelling. The different adaptation techniques need to be explored to improve the performance of GMM-UBM. For instance, the effectiveness of the Kernel Eigenspace-based Maximum Likelihood Linear Regression (KEMLLR) needs to be verified. In addition, an efficient classifier is to be built which can model the speaker considerably without expecting much speech data for foreground or background modelling.

- In this work the modelling techniques we considered are all generative. The efficiency of discriminative modelling technique like Support Vector Machines (SVM) based classifier need to be verified under limited data.

- We used either the Euclidean distance measurement or maximum likelihood ratio test as testing techniques. The efficiency of other measurement techniques like Mahalanobis, Kullback-Leibler divergence etc., are to be explored under limited data.

- In the decision logic though we have explored the techniques in abstract, rank and measurement levels, sophisticated decision techniques need to be explored for better decision about the speaker. For instance, speaker pruning algorithms need to be explored.
- In this work effectiveness of the proposed techniques are verified using clean speech data. The effectiveness of the techniques needs to be verified on noisy speech data.

A

Speech Production Mechanism

Contents

A.1 Speech Production Mechanism

A schematic diagram of the human vocal mechanism is shown in Figure A.1. It consists of the following speech production organs [5]:

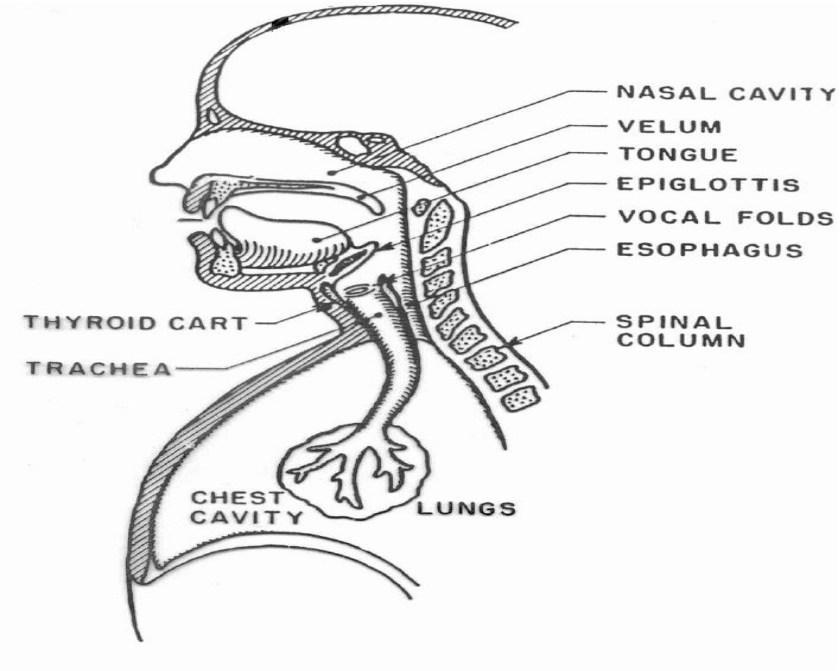

Figure A.1: The representation of the speech production mechanism.

- Lungs: Air enters the lungs through normal breathing mechanism. This acts as an air reservoir and energy source.
- Larynx and Vocal Cords: The larynx contains the vocal folds. As air is expelled from lungs through *trachea*, the tensed vocal cords within the *larynx* are caused to vibrate by the air flow. The space between the vocal folds called the glottis. When the vocal cords are tensed, the air flow causes them to vibrate, producing voiced speech sound. When the vocal cords are relaxed, in order to produce a sound, the air flow either must

pass through a constriction in the vocal tract and thereby become turbulent, producing unvoiced sounds. The rate of vibration of the vocal cords is determined primarily by their mass and tension, though air pressure and velocity can contribute in a smaller way. During a normal mode of vibration, the vocal cords open and close completely during the cycle and generate puffs of air roughly triangular in shape when air flow is plotted against time. A breathy voice quality is produced during an open phase mode of vibration, such that the folds never completely stop the air flow through them. A minimum of air passes through the folds, in short puffs, when producing a creaky voice.

- Vocal tract:

  The vocal tract is extended from the vocal folds to the lips, with a side branch leading to the nasal cavity. Normally, the length of the vocal tract is around about 17 centimeters. The pharynx connects the larynx (as well as the esophagus) with the oral cavity. The oral cavity is the most important component of the vocal tract because its size and shape can be varied by adjusting the relative positions of the palate, the tongue, the lips, and the teeth.

Speech signals are produced by exciting a system with a source. From signal processing point of view, the speech production mechanism can be represented as shown in Figure A.2. Vibration of the vocal folds, powered by air coming from the lungs during exhalation, is the sound source for speech. It sets up a pulse wave in which the pulses are roughly triangular. The amplitude, fundamental frequency, and shape of the waveforms can be modified by the action of the laryngeal muscles. The sound generated in the larynx does not transmit linguistic information. It acts as the source for the information which is imposed upon it by modifications introduced by the vocal tract. Therefore, glottal excitation forms the source, and the vocal tract forms the system. Speech is produced by exciting the vocal tract by the glottal excitation. The vocal tract is replaced with filter, and the filter coefficients depend on the physical dimensions of the vocal tract. Speech is produced by exciting the vocal tract by the glottal excitation. The vocal tract is replaced with filter, and the filter coefficients is replaced with two types of

signal generators, impulse train generator for voiced sounds and random number generator for unvoiced and fricative sounds [5].

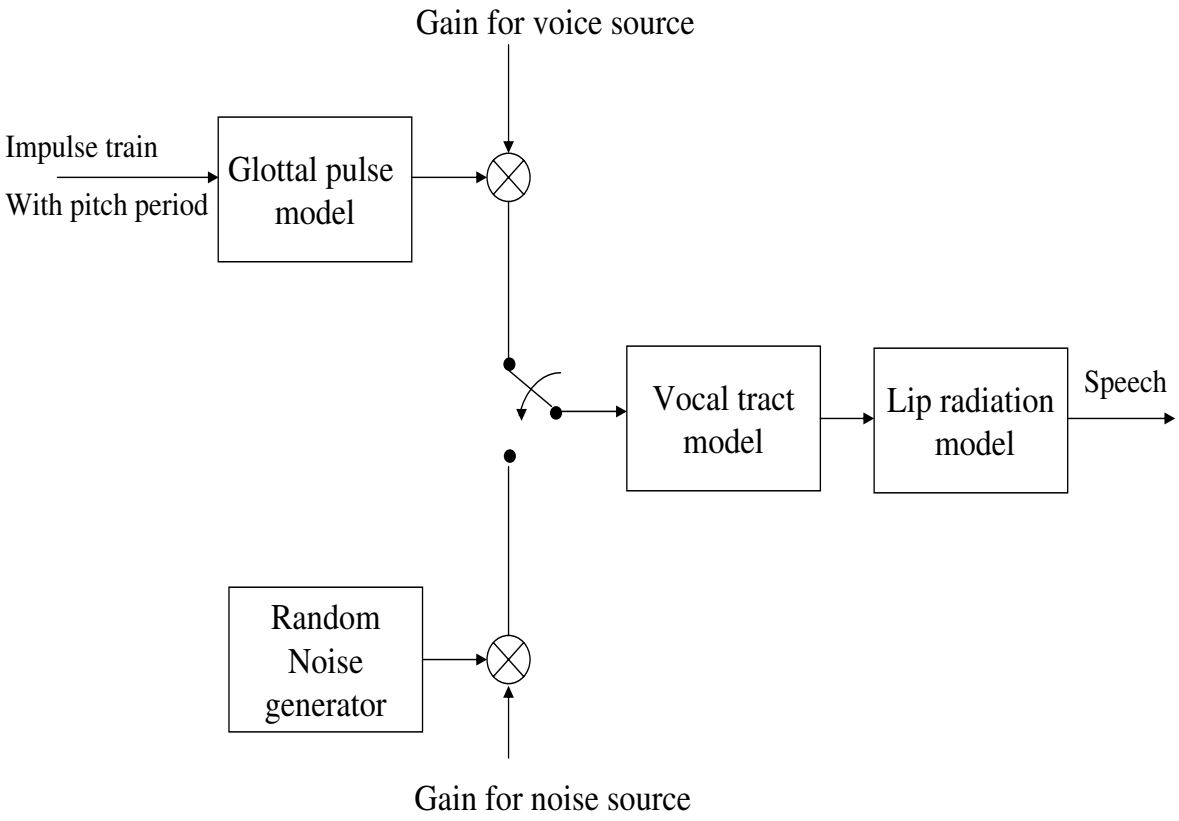

Figure A.2: The representation of the speech production mechanism.

B

Cepstral Analysis

Contents

B.1 Cepstral Analysis

In cepstral analysis, the slow varying vocal tract information and fast varying excitation information in the frequency domain are separated into low quefrency and high quefrency regions, respectively [5, 23]. This can be achieved by performing logarithmic operation on the magnitude spectrum and then taking the inverse Fourier transform. Thus the cepstral values in the low quefrency region starting from zero up to certain values represent vocal tract information. These values are used as feature vectors for speaker recognition task. Let $s(n)$ be the speech signal which can be represented mathematically in different domains as follows: In time domain,

$$s(n) = h(n) * e(n) \tag{B.1}$$

where, $h(n)$ is the impulse response of the vocal tract system and $e(n)$ is the excitation signal. In frequency domain,

$$S(e^{jw}) = H(e^{jw})E(e^{jw}) \tag{B.2}$$

where, $S(e^{jw})$, $H(e^{jw})$ and $E(e^{jw})$ are the Fourier transforms of $s(n)$, $h(n)$ and $e(n)$, respectively.

The magnitude spectrum of the above equation is given by

$$\left|S(e^{jw})\right| = \left|H(e^{jw})\right| \left|E(e^{jw})\right| \tag{B.3}$$

In order to separate the vocal tract and excitation information, we have to linearly combine corresponding magnitude spectrum. For this purposes, we have to take *log* operation. Therefore the above equation can be represented as,

$$\log\left|S(e^{jw})\right| = \log\left|H(e^{jw})\right| + \log\left|E(e^{jw})\right| \tag{B.4}$$

Take Inverse Fourier Transform (IFT) on the above equation to separate the vocal tract and excitation information.

$$c(n) = \frac{1}{2\pi} \int_{-\pi}^{\pi} \log\left|S(e^{jw})\right| e^{jwn} dw \tag{B.5}$$

By substituting Eqn.B.4 to Eqn.B.5 we get

$$c(n) = \frac{1}{2\pi} \int_{-\pi}^{\pi} \log \left| H(e^{jw}) \right| e^{jwn} dw + c(n) = \frac{1}{2\pi} \int_{-\pi}^{\pi} \log \left| E(e^{jw}) \right| e^{jwn} dw \tag{B.6}$$

C

Linear Prediction Coefficients Computation

Contents

C.1 Linear Prediction Coefficients (LPC)

The basic idea behind the LPC model is that a given speech sample *s(n)* at time n can be approximated as a linear combination of the past p speech samples, such that

$$s(n) = a_1 s(n-1) + a_2 s(n-2) + ... + a_p s(n-p) \tag{C.1}$$

where the coefficients a_1, a_2,...,a_p are assumed constant over speech analysis frame.
The above equation can be written as

$$s(n) = \sum_{i=1}^{p} a_i s(n-i) + Gu(n) \tag{C.2}$$

where G is the gain term.
The *z-domain* representation of the above equation is

$$S(z) = \sum_{i=1}^{p} a_i z^{-i} S(z) + GU(n) \tag{C.3}$$

Leading to the trasfer function

$$H(z) = \frac{S(z)}{GU(z)} = \frac{1}{1 - \sum_{i=1}^{p} a_i z^{-i}} = \frac{1}{A(z)} \tag{C.4}$$

There are two methods for estimating LPC:

- Autocorrelation
- Covariance

Both methods choose the short term filter coefficients a_k in such a way that the energy in the error signal (residual) is minimized. For speech processing tasks, the autocorrelation method is almost exclusively used because of its computational efficiency and inherent stability whereas the covariance method does not guarantee the stability of the all-pole LP synthesis filter [136]. The autocorrelation method of computing LPC is described below:

First, speech signal $s(n)$ is multiplied by a window $w(n)$ to get the windowed speech segment $s_w(n)$. Normally, a Hamming or Hanning window is used. The windowed speech signal is expressed as:

$$s_w(n) = s(n)w(n). \tag{C.5}$$

The next step is to minimize the energy in the residual signal. The residual energy E_p is defined as [122]:

$$E_p = \sum_{n=-\infty}^{\infty} e^2(n) = \sum_{n=-\infty}^{\infty} \left(s_w(n) + \sum_{k=1}^{p} a_k s_w(n-k) \right)^2. \tag{C.6}$$

The values of a_k that minimize E_p are found by setting the partial derivatives of the energy E_p with respect to the LPC parameters equal to zero.

$$\frac{\partial E_p}{\partial a_k} = 0, \quad 1 \leq k \leq p. \tag{C.7}$$

This results in the following p linear equations for the p unknown parameters $a_1, \ldots, a_p$

$$\sum_{k=1}^{p} a_k \sum_{n=-\infty}^{\infty} s_w(n-i)s_w(n-k) = -\sum_{n=-\infty}^{\infty} s_w(n-i)s_w(n), \quad 1 \leq i \leq p. \tag{C.8}$$

This linear equations can be expressed in terms of the autocorrelation function. This is because the autocorrelation function of the windowed segment $s_w(n)$ is defined as:

$$R_s(i) = \sum_{n=-\infty}^{\infty} s_w(n)s_w(n+i), \quad 1 \leq i \leq p. \tag{C.9}$$

Exploiting the fact that the autocorrelation function is an even function i.e., $R_s(i) = R_s(-i)$. By substituting the values from Eqn. (C.9) in Eqn. (C.8), we get

$$\sum_{k=1}^{p} R_s\left(|i-k|\right) a_k = -R_s(i), \quad 1 \leq i \leq p. \tag{C.10}$$

These set of p linear equations can be represented in the following matrix form as [136]:

$$\begin{bmatrix} R_s(0) & R_s(1) & \cdots & R_s(p-1) \\ R_s(1) & R_s(0) & \cdots & R_s(p-2) \\ \vdots & \vdots & \ddots & \vdots \\ R_s(p-1) & R_s(p-2) & \cdots & R_s(0) \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_p \end{bmatrix} = - \begin{bmatrix} R_s(1) \\ R_s(2) \\ \vdots \\ R_s(p) \end{bmatrix} \tag{C.11}$$

This can be summarized using vector-matrix notation as:

$$\mathbf{R_s a} = -\mathbf{r_s} \tag{C.12}$$

where the $p \times p$ matrix $\mathbf{R_s}$ is known as the autocorrelation matrix. The resulting matrix is a Toeplitz matrix where all elements along a given diagonal are equal. This allows the linear equations to be solved by the Levinson-Durbin algorithm. Because of the Toeplitz structure of $\mathbf{R_s}$, $A(z)$ is minimum phase [136]. At the synthesis filter $H(z) = 1/A(z)$, the zeros of $A(z)$ become the poles of $H(z)$. Thus, the minimum phase of $A(z)$ guarantees the stability of $H(z)$.

Bibliography

[1] J. P. Campbell, Jr., "Speaker recognition: A tutorial," *Proc. IEEE*, vol. 85(9), pp. 1437–1462, Sep. 1997.

[2] L. R. Rabinar,, "Applications of voice processing to telecommunications," *Proc. IEEE,*, vol. 82(2), p. 199 228, Feb. 1994.

[3] D. O'Shaughnessy, *Speech Communications: Human and Machine*, 2nd ed. IEEE Press, 1999.

[4] S. Furui, *Digital Speech Processing, Synthesis, and Recognition.* Marcel Dekker, 1989.

[5] J. Deller, J. Hansen and J. Proakis, *Discrete Time Processing of Speech Signals*, 1st ed. IEEE Press, 1993.

[6] B. S. Atal, "Automatic recognition of speakers from their voices," *Proc. IEEE*, vol. 64(4), pp. 460–475, Apr. 1976.

[7] R. J. Mammone, X. Zhang, and R. P. Ramachandran, "Robust speaker recognition a feature-based approach," *IEEE Signal Process. Mag.*, vol. 13(5), pp. 58–71, Sep. 1996.

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

[9] L. R. Rabiner and R. W. Schafer, *Digital Processing of Speech Signals*, 1st ed. Prentice Hall Inc., 1978.

[10] T. F. Quatieri, *Discrete-Time Speech Signal Processing.* Prentice Hall, 2004.

[11] A. E. Rosenberg, "Automatic speaker verification: A review," *Proce. IEEE*, vol. 64(4), pp. 475–487, Apr. 1976.

[12] J. M. Naik, L. P. Nestch, G. R. Doddington, "Speaker verification using long distance telephone lines," in *proc. Int. Conf. Acoust., Speech, Signal Process.*, Glasgow, UK, May 1989, pp. 524–527.

[13] H. Gish and M. Schmidt, "Text-independent speaker identification," *IEEE Signal Process. Mag.*, vol. 18, pp. 18–32, Oct. 2002.

[14] J. M. Naik, "Speaker verification: a tutorial ," *IEEE Communication Mag.*, vol. 28, pp. 42 – 48, Jan. 1990.

[15] F. Bimbot, J. F. Bonastre, C. Fredouille, G. Gravier, I. M. Chagnolleau, S. Meignier, T. Merlin, J. O. Garcya, D. P. Delacretaz, and D. A. Reynolds, "A tutorial on text-independent speaker verification," *EURASIP Journal of Applied Signal Process.*, vol. 4, pp. 430 – 451, 2004.

[16] D. A. Reynolds, "Speaker identification and verification using Gaussian mixture speaker models," *Speech Communication*, vol. 17, pp. 91–108, 1995.

[17] B. Yegnanarayana, S. R. M. Prasanna, J.M. Zachariah, and C. S. Gupta,, "Combining evidence from source, suprasegmental and spectral features for a fixed-text speaker verification system," *IEEE Trans. Speech Audio Process.*, vol. 13(4), pp. 575–582, July 2005.

[18] P. Satyanarayana, "Short segment analysis of speech for enhancement," Ph.D. dissertation, Indian Institute of Technology Madras, Dept. of Computer Science and Engg., Chennai, India, Feb. 1999.

[19] James E. Luck, "Automatic speaker verification using cepstral measurements," *J. Acoust. Soc. Amer.*, vol. 46(2), pp. 1026–1032, Nov. 1969.

[20] B. S. Atal, "Automatic speaker recognition based on pitch contours ," *J. Acoust. Soc. Amer.*, vol. 52, no. 6(part 2), pp. 1687–1697, 1972.

[21] F. Farahani, P. G. Georgiou and S. S. Narayanan, "Speaker identification using supra-segmental pitch pattern dynamics ," in *proc. Int. Conf. Acoust., Speech, Signal Process.*, Montreal, Canada, May 2004, pp. 89–92.

[22] S. R. M. Prasanna, C. S. Gupta and B. Yegnanarayana, "Extraction of speaker-specific excitation information from linear prediction residual of speech ," *Speech Communication*, vol. 48, pp. 1243–1261, 2006.

[23] L. Rabiner and B. H. Juang, *Fundamentals of Speech Recognition.* Singapore: Pearson Education, 1993.

[24] D. A. Reynolds, "An overview of automatic speaker recognition technology," *in Proc. IEEE Int. Conf. Acoust., Speech Signal Process*, vol. 4, pp. 4072–4075, May 2002.

[25] P. Angkititrakul and J. H. L. Hansen, "Discriminative In-Set/Out-of-Set speaker recognition," *IEEE Trans. Audio Speech Language Process.*, vol. 15(2), pp. 498–508, Feb. 2007.

[26] V. Prakash and J. H. L. Hansen, "In-Set/Out-of-Set speaker recognition under sparse enrollment ," *IEEE Trans. Audio Speech Language Process.*, vol. 15(7), pp. 2044–2051, Sep. 2007.

[27] S. R. Mahadeva Prasanna, "Event based analysis of speech," Ph.D. dissertation, Indian Institute of Technology Madras, Dept. of Computer Science, Chennai, India, March 2004.

[28] P. Krishnamoorthy, "Combined temporal and spectral processing methods for speech enhancement," Ph.D. dissertation, Indian Institute of Technology Guwahati, Dept. of Electronics and Communication Engg., Guwahati, India, Oct. 2008.

[29] P. H. Arjun, "Speaker recognition in indian languages: A feature based approach," Ph.D. dissertation, Indian Institute of Technology Kharagpur, Dept. of Electrical Engg., Kharagpur, India, Jul. 2005.

[30] G. Senthil Raja, "Feature analysis and compensation for speaker recognition under stressed condition," Ph.D. dissertation, Indian Institute of Technology Guwahati, Dept. of Electronics and Communication Engg., Guwahati, India, Jul. 2007.

[31] N. Dhananjaya and B. Yegnanarayana, "Speaker change detection in casual conversations using excitation source features ," *Speech Communication*, vol. 50, p. 1153 1161, 2008.

[32] L. Lu and H. Zhang, "Speaker change detection and tracking in rea ltime news broadcasting analysis ," in *proc. 10th ACM Multimedia.*, Juan-les-pins, France, 2002, pp. 602–610.

[33] J. Makhoul, F. Kubala F, T. Leek, D. Liu D, L. Nguyen R. Schwartz and A. Srivastava, "Speech and language technologies for audio indexing and retrieval ," *Proc. IEEE*, vol. 88(8), p. 1338 1353, Apr. 2000.

[34] W. M. Campbell, D. A. Reynolds, J. P. Campbell and K. J. Brady, "Estimating and evaluating confidence for forensic speaker recognition," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, vol. 1, Philadelphia, USA, 2005, pp. 717–720.

[35] J-F. Bonastre, F. Bimbot, L-J Boe, J. P. Campbell, D. A. Reynolds and I. Magrin-Chagnolleau, "Person authentication by voice: a need for caution ," in *Eurospeech*, Geneva, Switzerland, Sep. 2003.

[36] N. Dhananjaya and B. Yegnanarayana , "Correlation-based similarity between signals for speaker verification with limited amount of speech data ," in *proc. International Workshop, MRCS 2006*, Istanbul, Turkey, Sep. 2006.

[37] C. Champod and D. Meuwly, "The inference of identity in forensic speaker recognition ," *Speech Communication,*, vol. 31, p. 193 203, 2000.

[38] F. Botti, A. Alexander and A. Drygajlo, "An interpretation framework for the evaluation of evidence in forensic automatic speaker recognition with limited suspect data ," in *proc. The speaker and Language Recognition Workshop - Odyssey-04*, Toledo, Spain, June 2004, pp. 63–68.

[39] J. J. Wolf, "Efficient acoustic parameters for speaker recognition ," *J. Acoust. Soc. Amer.*, vol. 51, no. 6(part 2), pp. 2044–2056, 1971.

[40] G. R. Doddington, "A method of speaker verification," *J. Acoust. Soc. Amer.*, vol. 49, p. 139(A), 1971.

[41] K. P. Li, J. E. Dammann and W. D. Chapman, "Experimental studies in speaker verification using an adaptive system ," *J. Acoust. Soc. Amer.*, vol. 40(5), pp. 966–978, Nov. 1966.

[42] D. A. Reynolds, "Experimental evaluation of features for robust speaker identification," *IEEE Trans. Speech Audio Process.*, vol. 2(4), pp. 639–643, Oct. 1994.

[43] K. S. R. Murthy and B. Yegnanarayana, "Combining evidence from residual phase and MFCC features for speaker recognition ," *IEEE Signal Process. Lett.*, vol. 13(1), pp. 52–56, Jan. 2006.

[44] B. Yegnanarayana, K. Sharat Reddy and S. P. Kishore , "Source and system features for speaker recognition using AANN models ," in *proc. Int. Conf. Acoust., Speech, Signal Process.*, Utah, USA, Apr. 2001.

[45] K. Sharat Reddy, "Source and system features for speaker recognition," Master's thesis, Indian Institute of Technology Madras, Dept. of Computer Science and Engg., Chennai, India, 2001.

[46] C. S. Gupta, "Significance of source features for speaker recognition," Master's thesis, Indian Institute of Technology Madras, Dept. of Computer Science and Engg., Chennai, India, 2003.

[47] L. Mary, K. S. Rao, S. V. Gangashetty and B. Yegnanarayana, "Neural network models for capturing duration and intonation knowledge for language and speaker identification," in *Proc. Int. Conf. Cognitive Neural Systems*, Boston, Massachusetts, May 2004.

[48] F. Weber, L. Manganaro, B. Peskin, and E. Shriberg, "Using prosodic and lexical information for speaker identification," in *proc. Int. Conf. Acoust., Speech, Signal Process.*, vol. 1, London, UK, April. 2002, pp. 141–144.

[49] K. Samuel T. Eriksson, H. Kang and D. H. Youn, "A pitch synchronous feature extraction method for speaker recognition," in *proc. Int. Conf. Acoust., Speech, Signal Process.*, Baltimore, MD, USA, May 2004.

[50] R. D. Zilca, B. Kingsbury, J. Navratil, and G. N. Ramaswamy, "Pseudo pitch synchronous analysis of speech with application to speaker recognition," *IEEE Trans. Speech Audio Process.*, vol. 14(2), pp. 467–478, Mar. 2006.

[51] P. Denes and M. V. Mathews, "Spoken digit recognition using time-frequency pattern matching," *J. Acoust. Soc. Amer.*, vol. 32(11), pp. 1450–1455, Nov. 1960.

[52] S. Pruzansky, "Pattern-matching procedure for automatic talker recognition," *J. Acoust. Soc. Amer.*, vol. 35(3), pp. 354–358, March. 1963.

[53] S. Pruzansky and M. V. Mathews, "Talker-recognition procedure based on analysis of variance," *J. Acoust. Soc. Amer.*, vol. 36(11), pp. 2041–2047, Nov. 1964.

[54] J. W. Glenn and N. Kleiner, "Speaker identification based on nasal phonation ," *J. Acoust. Soc. Amer.*, vol. 43(2), pp. 368–372, June 1967.

[55] S. K. Das, W. S. Mohn and S. L. Saleeby, "Speaker verification experiments ," *J. Acoust. Soc. Amer.*, vol. 49, p. 138(A), 1971.

[56] R. C. Lummis, "Speaker verification by computer using speech intensity for temporal registration," *IEEE Trans. Audio Electro acoustics*, vol. AU-21(2), pp. 80–89, Apr. 1973.

[57] B. S. Atal, "Effectiveness of linear prediction characteristics of the speech wave for automatic speaker identification and verification," *J. Acoust. Soc. Amer.*, vol. 55, pp. 1304–1312, 1974.

[58] M. R. Sambur, "Selection of acoustic features for speaker identification," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-23(2), pp. 176–182, Apr. 1975.

[59] A. E. Rosenberg and M. R. Sambur, "New techniques for automatic speaker verification," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-23(2), pp. 169–176, Apr. 1975.

[60] M. R. Sambur, "Speaker recognition using orthogonal linear prediction," *IEEE Trans. Acoust., Speech, Signal Process..*, vol. ASSP-24(4), pp. 283–289, Aug. 1976.

[61] J. D. Markel, B. T. Oshika and A. H. Gray, Jr., "Long-term feature averaging for speaker recognition," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-25(4), pp. 330–337, Aug. 1977.

[62] S. Furui, "Speaker-independent isolated word recognition using dynamic features of speech spectrum," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-34, pp. 52–59, Feb. 1986.

[63] ——, "Cepstral analysis technique for automatic speaker verification," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 29(2), pp. 254–272, Apr. 1981.

[64] G. R. Doddington, M. A. Przybocki, A. F. Martin and D. A. Reynolds, "The NIST speaker recognition evaluation overview, methodology, systems, results, perspective," *Speech Communication*, vol. 31, pp. 225–254, 2000.

[65] P. Thevenaz and H. Hugli, "Usefulness of the LPC-residue in text-independent speaker verification," *Speech Communication*, vol. 17, pp. 145–157, 1995.

[66] M. J. Carey, E. S. Parris, H. Lloyd-Thomas, and S. Bennett, "Robust prosodic features for speaker identification," in *proc. Int. Spoken Language Process.*, Philadelphia, PA, USA, Oct. 1996.

[67] M. K. Sonmez, E. Shriberg, L. Heck, and M. Weintraub, "Modeling dynamic prosodic variation for speaker verification," in *proc. Int. Spoken Language Process.*, Sydney, NSW, Australia, Nov-Dec. 1998.

[68] B. Peskin, J. Navratil, J. Abramson, D. Jones, D. Klusacek, D. A. Reynolds, and B. Xiang, "Using prosodic and conversational features for high-performance speaker recognition," in *Int. Conf. Acoust., Speech, Signal Process.*, vol. IV, Hong Kong, Apr. 2003, pp. 784–787.

[69] B. Imperl, Z. Kacic, and B. Horvat, "A study of harmonic features for the speaker recognition," *Speech Communication*, vol. 22, pp. 385–402, 1997.

[70] M. D. Plumpe, T. F. Quatieri and D. A. Reynolds, "Modeling of the glottal flow derivative waveform with application to speaker identification," *IEEE Trans. Speech Audio Process.*, vol. 7(5), pp. 569–585, 1999.

[71] M. Grimaldi and F. Cummins, "Speaker identification using instantaneous frequencies," *IEEE Trans. Audio, Speech, Language Process.*, vol. 16(6), pp. 1097–1111, Aug. 2008.

[72] H. Sakoe and S. Chiba, "Dynamic programming algorithm optimization for spoken word recognition ," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 26, pp. 43–49, Feb. 1978.

[73] K. P. Li, "An approach to text-independent speaker recognition with short utterances ," in *Proc. IEEE Int. Conf. Acoust., Speech, signal Process.*, San Diego, California, Mar. 1985, pp. 555–558.

[74] Y. Linde, A. Buzo, and R. M. Gray, "An algorithm for vector quantizer design," *IEEE Trans. Communications*, vol. COM-28(1), pp. 84–96, Jan. 1980.

[75] R. Gray, "Vector quantization," *IEEE Acoust., Speech, Signal Process. Mag.*, vol. 1, pp. 4–29, Apr. 1984.

[76] F. K. Soong, A. E. Rosenberg, L. R. Rabiner and B. H. Juang, "A Vector quantization approach to speaker recognition," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, vol. 10, Detroit, Michingon, Apr. 1985, pp. 387–390.

[77] J. C. Bezdek and J. D. Harris, "Fuzzy portions and relations;an axiomatic basis for clustering," *Fuzzy Sets and Systems*, vol. 1, pp. 111–127, 1978.

[78] H. J. Zimmermann, *Fuzzy set theory and its applications*, 1st ed. Kluwer academic, 1996.

[79] L. Lin and S. Wang, "A Kernel method for speaker recognition with little data," in *Int. Conf. signal Process.*, Budapest, Hungery, May, 2006.

[80] V. Chatzis, A. G. Bors and I. Pitas, "Multimodal decision-level fusion for person authentication," *IEEE Trans. Man CyberneticsPart A: Systems and Humans*, vol. 29, pp. 674–681, Nov. 1999.

[81] A. E. Rosenberg and S. Parthasarathy, "Speaker background models for connected digit password speaker verification," in *proc. Int. Conf. Acoust., Speech, Signal Process.*, Atlanta Georgia, May 1996, pp. 81–84.

[82] T. Matsui and S. Furui, "Comparison of text-independent speaker recognition methods using VQ-distortion and Discrete/continuous HMMs," *IEEE Trans. Speech Audio Process.*, vol. 2(3), pp. 456–459, July 1994.

[83] O. Kimball, M. Schmidt, H. Gish and J. Waterman, "Speaker verification with limited enrollment data ," in *proc. European Conf. Speech Commun. and Tech. (EUROSPEECH'97)*, Rhodes, Greecee, Sep. 1997, pp. 967–970.

[84] R. P. Lipmann, "An introduction to computing with neural nets," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 4, pp. 4–22, Apr. 1989.

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

[86] B. Yegnanarayana, *Artificial neural networks*. New Delhi: Prentice-Hall, 1999.

[87] J. Oglesby and J. S. Mason, "Optimization of neural models for speaker identification ," in *proc. Int. Conf. Acoust., Speech, Signal Process.*, Albuquerque, NM, May 1990, pp. 261–264.

[88] ——, "Radial basis function for speaker recognition ," in *proc. Int. Conf. Acoust., Speech, Signal Process.*, Toronto, Canada, May 1991, pp. 393–396.

[89] M. Inal and Y. S. Fatihoglu, "Self organizing map and associative memory model hybrid classifier for speaker recognition," in *proc. Neu., Net., App., Elec., Engg. (NEUREL'02)*, Belgrade, Yugoslavia, Sep. 2002, pp. 71–74.

[90] A. T. Mafra and M. G. Simoes, "Text independent automatic speaker recognition using self-organizing maps," in *proc. Ind. App. Society conf.*, vol. 3, Victoria, British Columbia, Oct. 2004, pp. 1503–1510.

[91] G. Bannani, F. Fogelman and P. Gallinari, "A connectionist approach for speaker identification ," in *proc. Int. Conf. Acoust., Speech, Signal Process.*, Albuquerque, NM, May 1990, pp. 265–268.

[92] J. He, L. Liu and G. Palm, "A discriminative training algorithm for VQ-based speaker identification," *IEEE Trans. Speech Audio Process.*, vol. 7, pp. 353–356, May 1999.

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

[94] W. M. Campbell, J. P. Campbell, D. A. Reynolds, E. Singer and P. A. Torres-Carrasquillo, "Support vector machines for speaker and language recognition," *Computer Speech and Language*, vol. 20, pp. 210–229, 2006.

[95] H. Aronowitz, D. Burshtein, "Efficient speaker recognition using approximated cross entropy ," *IEEE Trans. Audio, Speech, Language Process.*, vol. 15(7), pp. 2033 – 2043, Sep. 2007.

[96] L. Burget, P. Matejka, P. Schwarz, O. Glembek, and J. Cernocky, "Analysis of feature extraction and channel compensation in a GMM speaker recognition system ," *IEEE Trans. Audio, Speech, Language Process.*, vol. 15(7), pp. 1979 – 1986, Sep. 2007.

[97] H. Rongqing and J. H. L. Hansen, "Unsupervised discriminative training with application to dialect classification ," *IEEE Trans. Audio, Speech, Language Process.*, vol. 15(8), pp. 2444 – 2453, Nov. 2007.

[98] T. Kinnunen, E. Karpov, and P. Franti, "Real-time speaker identification and verification," *IEEE Trans. Audio, Speech, Language Process.*, vol. 14(1), pp. 277–288, Jan. 2006.

[99] S. Kwon and S. Narayanan, "Robust speaker identification based on selective use of feature vectors," *Pattern Recognitiion. Lett.*, vol. 28, pp. 85–89, 2007.

[100] D. A. Reynolds, T. F. Quateri and R. B. Dunn , "Speaker verification using adapted Gaussian mixture models," *Digital Signal Processing*, vol. 10, pp. 19–41, 2000.

[101] Man-Wai Mak, R. Hsiao and B. Mak, "A Comparison of various adaptation methods for speaker verification with limited enrollment data ," in *Proc. IEEE Int. Conf. Acoust., Speech, signal Process.*, Toulouse, France, May 2006, pp. 1–4.

[102] V. Hautamaki, T. Kinnunen, I. Karkkainen, J. Saastamoinen, M. Tuononen and P. Franti, "Maximum *a Posteriori* adaptation of the centroid model for speaker verification ," *IEEE Signal Process. Lett.*, vol. 15, pp. 162–165, 2008.

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

[104] M. Shajith Iqbal, Hemanth Misra and B. Yegnanarayana, "Analysis of auto associative neural networks," in *Int. Joint Conf. Neural Networks*, Washington, USA, 1999.

[105] B. Yegnanarayana, K. S. Redddy and S. P. Kishore, "Source and system features for speaker recognition using AANN models," in *Int. Conf. Acoust., Speech, Signal Process.*, Salt Lake City, Utah, USA, Apr. 2001, pp. 409–412.

[106] V. Wan and S. Renals, "Evaluation of kernel methods for speaker verification and identification," *in Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, vol. 1, pp. 1–669 – 1–672, 2002.

[107] ——, "Speaker verification using sequence discriminant support vector machines," *IEEE Trans. Speech Audio Process.*, vol. 13, pp. 203–210, 2005.

[108] H. Jiang and L. Deng, "A Bayesian approach to the verification problem: Applications to speaker verification," *IEEE Trans. Speech, Audio Process.*, vol. 9(8), pp. 874–975, 2001.

[109] B-J Lee, S-W Yoon, H-G Kang and D H Youn, "On the use of voting methods for speaker identification based on various resolution filterbanks ," in *proc. Int. Conf. Acoust., Speech, Signal Process.*, vol. I, Toulouse, France, May 2006, pp. 917–920.

[110] T. Nagarajan, "Implicit systems for spoken language identification," Ph.D. dissertation, Indian Institute of Technology Madras, Dept. of Computer Science, Chennai, India, 2004.

[111] G. L. Sarada, T. Nagarajan and H. A. Murthy, "Multiple frame size and multiple frame rate feature extraction for speech recognition," in *proc. Int. Conf. Signal Process. Communication*, Bangalore, India, Oct. 2004.

[112] S. M. Peeling and K.M. Ponting, "Variable frame rate analysis in the ARM continuous speech recognition system," *Speech Communication*, vol. 10, pp. 169–179, 1996.

[113] K. Ponting and S. Peeling, "The use of variable frame rate analysis in speech recognition," *Comp., Speech, Language*, vol. 5, pp. 169–179, 1991.

[114] P. L. Cerf and D. V. Compernolle, "A new variable frame rate analysis method for speech recognition," *IEEE Signal Process. Lett.*, vol. 1, pp. 185–187, Dec. 1994.

[115] Q. Zhu and A. Alwan, "On the use of variable frame rate analysis in speech recognition," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, vol. 3, Istanbul, Turkey, 2000, pp. 1783–1786.

[116] T. Nagarajan and H.A. Murthy, "Language identification using acoustic log-likelihoods of syllable-like units," *Speech Communication*, vol. 48(8), Aug. 2006.

[117] G. L. Sarada, N. Hemalatha, T. Nagarajan and H. A. Murthy, "Automatic transcription of continuous speech using unsupervised and incremental training," in *proc. Int. Conf. Spoken Language Process.*, vol. 18(2), Jeju Island, Korea, Oct. 2004.

[118] S. B. Davis and P. Mermelstein, "Comparison of parametric representation for monosyllabic word recognition in continuously spoken sentences ," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSAP-28(4), pp. 357–366, Aug. 1980.

[119] J. P. Campbell, Jr., "Testing with the YOHO CD-ROM voice verification corpus," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Detroit, Michigon, May 1995, pp. 341–344.

[120] S. S. V. Zue and J. Glass, "Speech database development at MIT: TIMIT and beyond," *Speech Communication*, vol. 9, no. 4, pp. 351–356, 1990.

[121] G. Doddington, "Speaker recognition based on idiolectal differences between speakers," *Eurospeech*, pp. 2521–2524, Sept. 2001.

[122] J. Makhoul, "Linear prediction: a tutorial review," *Proc. IEEE*, vol. 63(4), pp. 561–580, Apr. 1975.

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

[124] T. Kinnunen, V. Hautamaki and P Franti , "On the fusion of dissimilarity-based classifiers for speaker identification," in *proc. EUROSPEECH*, Geneva, Switzerland, Sep. 2003, pp. 58–63.

[125] V. Cuperman and A. Gersho, "Vector predictive coding of speech at 16 kbits/s ," *IEEE Trans. Communications*, vol. 33(7), pp. 685–696, July 1985.

[126] H. Abut R. Gray and G. Rebolledo, "Vector quantization of speech and speech-like waveforms ," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 30(3), pp. 423–435, June 1982.

[127] A. Buzo , A. Gray Jr., R. Gray and J. Markel , "Speech coding based upon vector quantization ," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 28(5), pp. 562–574, Oct. 1980.

[128] J. Kittler, M. Hatef, R. P.W. Duin and J. Matas, "On combining classifiers," *IEEE Trans. Patt., Anly, Machine Intelligence*, vol. 20(3), pp. 226–239, March 1998.

[129] G. E. Tsekouras, "A fuzzy vector quantization approach to image compression," *Applied mathematics and computation*, vol. 167, pp. 539–560, 2005.

[130] T. Kohonen, "The self-organizing map," *Proce. IEEE*, vol. 78(9), pp. 1464–1480, Sep. 1990.

[131] Simon Haykin, *Neural Networks: A Comprehensive Foundation.* New Jersy: Prentice-Hall Inc., 1999.

[132] D. A. Reynolds and R. C. Rose, "Robust text-independent speaker identification using Gaussian mixture speaker models," *IEEE Trans. Speech Audio Process.*, vol. 3(1), pp. 72–83, Jan. 1995.

[133] L. Xu, A. Krzyzak, C. Y. Suen, "Methods of combining multiple classifiers and their applications to handwriting recognition," *IEEE Trans. Syst., Man Cybern.*, vol. 22(3), pp. 412–435, May/June 1992.

[134] T. K. Ho, J. J. Hull and S. N. Srihari, "Decision combination in multiple classifier systems," *IEEE Trans. Pattern Analysis Machine Intelligence*, vol. 16(1), pp. 66–76, Jan. 1994.

[135] A. Rahman and M. Fairhurst , "Decision combination of multiple classifiers for pattern classification: hybridisation of majority voting and divide and conquer techniques ," in *proc. IEEE-Workshop Appl., Computer Vision*, California, Oct. 2000, pp. 58–63.

[136] L. R. Rabiner and R. W. Schafer, *Digital Processing of Speech Signals*, 1st ed. Pearson Education, 1978.

LIST OF PUBLICATIONS

Refereed Journals:

1. H. S. Jayanna and S. R. M. Prasanna, "Multiple frame size and rate analysis for speaker recognition under limited data condition," Accepted for publication in *IET Signal Processing*.

2. H. S. Jayanna, S. R. M. Prasanna and Bhanu Prasad, "Speaker recognition using artificial intelligence tools: Review and scope," *International Journal of Tomography and Statistics (IJTS)*, vol. 5(AI207), pp. 5–10, Apr. 2007.

3. H. S. Jayanna and S. R. M. Prasanna, "Modelling techniques for speaker recognition under limited data condition," Dec. 2008, Revised and Resubmitted to *Sadhana* Academy proceedings in engineering sciences.

4. H. S. Jayanna and S. R. M. Prasanna, "Analysis, Feature Extraction, Modelling and Testing Techniques for Speaker Recognition - A Review," Revised and Resubmitted to *J. IETE Technical Review*, Mar. 2009.

5. H. S. Jayanna and S. R. M. Prasanna, "Speaker recognition under limited data condition," to be communicated to *IEEE Speech, Audio and Language Processing*, Mar. 2009.

Refereed International Conferences:

1. H. S. Jayanna and S. R. M. Prasanna, "Variable segmental analysis based speaker recognition in limited data condition," in *IEEE Proc. Int. Conf. Signal and Image processing* , Hubli, Karnataka, India, Dec. 2006.

2. H. S. Jayanna and S. R. M. Prasanna, "Speaker recognition in limited data condition using self-organizing map," in *Proc. Indian Int. Conf. Artificial Intelligence*, Pune, India, Dec. 2007.

3. H. S. Jayanna and S. R. M. Prasanna, "Fuzzy vector quantization for speaker recognition under limited data condition," in *IEEE Proc. TENCON 2008* , Hyderabad, India, Nov. 2008.

4. H. S. Jayanna and S. R. M. Prasanna, "Learning vector quantization for speaker recognition under limited data condition," *to be appeared in proc. Thirteenth Int. Conf. Cognitive Neural Systems*, Boston University, (Boston, MA, USA), May 2009.

Refereed National Conferences/Workshops:

1. H. S. Jayanna, Kali Charan Gujala, S. R. M. Prasanna, "Comparison of fuzzy and crisp vector quantization technique for speaker recognition," in *Proc. National Workshop on Artificial Intelligence*, NCST Mumbai, India, July 2006.

2. H. S. Jayanna and S. R. M. Prasanna, "Speaker recognition in limited data condition for variable frame rate analysis," in *Proc. National conference on VLSI design, Signal Processing and Communication*, Tumkur, Karnataka, India, Feb. 2007.

3. H. S. Jayanna and S. R. M. Prasanna, "Speaker recognition in limited data condition," in *Proc. Workshop on Image and Signal processing* , IIT Guwahati, India, Dec. 2007.

4. C. Vijaya Kumar, H. S. Jayanna and S. R. M. Prasanna, "Variable frame size and rate analysis for speaker recognition," in *Proc. International Symposium Frontiers of Research on Speech and Music* , Kolkata, India, Feb. 2008.