Automatic Speaker Recognition: Modelling, Feature Extraction and Effects of Clinical Environment

A thesis submitted in fulfillment of the requirements for the degree of Doctor of Philosophy

Sheeraz Memon
B.Eng. (Computer Systems)
M.Eng. (Communication Systems and Networks)

School of Electrical and Computer Engineering
Science, Engineering and Technology Portfolio

RMIT University

June 2010

Declaration

I certify that except where due acknowledgement has been made, the work is that of the author alone; the work has not been submitted previously, in whole or in part, to qualify for any other academic award; the content of the thesis is the result of work which has been carried out since the official commencement date of the approved research program; and, any editorial work, paid or unpaid, carried out by a third party is acknowledged.

Sheeraz Memon

2010

Dedication…

I dedicate my work

to my Parents….. for their Years of love and care

to my wife ……for her support and encouragement

to my daughter …… for making my life full of colors

Acknowledgements

This thesis would not have been possible without the support and encouragement of many people. First and foremost to my supervisors Dr Margaret Lech and Dr Namunu Maddage, thank you for all your support and encouragement throughout the past three years. It has been both an honor and pleasure to work with you and learn from you.

To my parents, and sweet sisters Shazia and Maria, thank you for always believing in me and encouraging me to follow my dreams. I could not have achieved any of this without the support and encouragement that you have always given me.

To my wife Samreen, you have come into my life last year and have turned everything in my life beautiful. Your support and care has helped me, specially the time when we were recently married and I had to fly to Australia for continuity of my studies. Your presence in Australia made it possible to finish this thesis. I know that I have been so selfish in spending time on this thesis but you always supported me. Thank you for your love, support and care, it is something which I will always treasure.

To my colleagues and friends at RMIT University, I thank you all for the encouragement and support you have given me during this period. My starting time in Australia was so difficult but only because of friends like you this journey became comfortable. I will never forget the days of the tea room and Oporto, my love and best wishes are with all of you.

Abstract

Speaker recognition is the task of establishing identity of an individual based on his/her voice. It has a significant potential as a convenient biometric method for telephony applications and does not require sophisticated or dedicated hardware.

The Speaker Recognition task is typically achieved by two-stage signal processing: training and testing. The training process calculates speaker-specific feature parameters from the speech. The features are used to generate statistical models of different speakers. In the testing phase, speech samples from unknown speakers are compared with the models and classified.

Current state of the art speaker recognition systems use the Gaussian mixture model (GMM) technique in combination with the Expectation Maximization (EM) algorithm to build the speaker models. The most frequently used features are the Mel Frequency Cepstral Coefficients (MFCC).

This thesis investigated areas of possible improvements in the field of speaker recognition. The identified drawbacks of the current speaker recognition systems included: slow convergence rates of the modelling techniques and feature's sensitivity to changes due aging of speakers, use of alcohol and drugs, changing health conditions and mental state.

The thesis proposed a new method of deriving the Gaussian mixture model (GMM) parameters called the EM-ITVQ algorithm. The EM-ITVQ showed a significant improvement of the equal error rates and higher convergence rates when compared to the classical GMM based on the expectation maximization (EM) method.

It was demonstrated that features based on the nonlinear model of speech production (TEO based features) provided better performance compare to the conventional MFCCs features.

For the first time the effect of clinical depression on the speaker verification rates was tested. It was demonstrated that the speaker verification results deteriorate if the speakers are clinically depressed. The deterioration process was demonstrated using conventional (MFCC) features.

The thesis also showed that when replacing the MFCC features with features based on the nonlinear model of speech production (TEO based features), the detrimental effect of the clinical depression on speaker verification rates can be reduced.

Publications

Book Chapters

1. **Memon S,** Lech M, "Speaker Verification Based on Information Theoretic Vector Quantization", CCIS, Springer-Verlag, Berlin Heidelberg, 2008, Vol. 20, pp.391-399.

2. **Memon S**, Lech M, Maddage N, He L, "Application of the Vector Quantization Methods and the Fused MFCC-IMFCC Features in the GMM Based Speaker Recognition", Book: Recent Advances in Signal Processing", ISBN 978-953-7619-41-1, Sep 2009, INTECH Publishing.

Refereed Journals

3. **Memon S**, Lech M, "Using Mutual Information as a classification error measure paradigm for speaker verification system" GESTS International Transactions on Computer Science and Engineering, vol 42, No. 1, Sep 2007.

4. He. L., Lech. M., **Memon. S**., Allen. N., "Detection of stress in speech using perceptual wavelet packet analysis", GESTS International Transactions on Computer Science and Engineering, Vol.45, No.01, March 30, 2008.

Refereed Conferences

5. **Memon, S.;** Lech, M.; "EM-IT based GMM for speaker verification", International Conference on Pattern Recognition, AUG23-26, 2010, Turkey (Accepted, 23 May 2010).

6. **Memon, S.;** Lech, M.; Namunu, M.; "Speaker Verification based on Different Vector Quantization Techniques with Gaussian Mixture Models", IEEE 3rd

international conference on network and system security and International workshop on frontiers of information assurance and security 2009, October 19-21, Gold coast Australia.

7. **Memon, S.,** Maddage, N., Lech, M., Allen, N., "Effect of Clinical Depression on Automatic Speaker Identification" IEEE 3rd International Conference on Bioinformatics and Biomedical Engineering, China, Page(s): 1-4, 11-13 June 2009.

8. **Memon, S.;** Lech, M.; Ling He; "Using Information Theoretic Vector Quantization for Inverted MFCC based Speaker Verification", IEEE 2nd International Conference on Computer, Communication and Control, 2009, IC4 2009, 17-18 Feb. 2009 Page(s):1– 5.

9. **Memon, S.,** and Lech, M., "Using information theoretic vector quantization for GMM based speaker verification", EUSIPCO 2008, Lausanne, Switzerland.

10. He, L., **Memon, S.;** Lech, M.; "Emotion Recognition in Speech of Parents of Depressed Adolescents", IEEE 3rd International Conference on Bioinformatics and Biomedical Engineering, China, Page(s): 1-4, 11-13 June 2009.

11. He, L., **Memon, S.;** Lech, M.; Namunu, M.; Nicholas, A.; "Recognition of Stress in Speech using Wavelet Analysis and Teager Energy Operator",. Proceedings of Interspeech 2008, Brisbane Australia.

Contents

List of Tables

List of Figures

List of Acronyms and Abbreviations

ACW	Adaptive Component Weighing
ANN	Artificial Neural Network
ASR	Automatic Speech Recognition
CEL-EM	Constraint-Based Evolutionary Learning-Expectation Maximization
DCE	Delta Cepstral Energy
DCF	Decision Cost Function
DCT	Discrete Cosine Transform
DDCE	Delta-Delta Cepstral Energy
DET	Detection Error TradeOff
DFE	Discriminative Feature Extraction
DTW	Dynamic Time Warping
DWT	Discrete Wavelet Transform
EA	Evolutionary Algorithm
EER	Equal Error Rate
EM	Expectation Maximization
FVQ	Fuzzy Vector Quantization
GLDS	Generalized Linear Discriminate Sequence
GMM	Gaussian Mixture Model
GVQ	Group Vector Quantization
HMM	Hidden Markov Models
ICA	Independent Component Analysis

ITGMM	Information Theoretic Gaussian Mixture Modeling
ITVQ	Information Theoretic Vector Quantization
LBG	Linde Buzo Gray
LP	Linear Prediction
LPC	Linear Prediction Coefficients
LPCC	Linear Prediction Cepstral Coefficients
LFCC	Linear Frequency Cepstral Coefficients
LLR	Log-Likelihood Ratio
LSP	Line Spectral Pairs
LVQ	Linea Vector Quantization
MAP	Maximum a Posteriori
MFCC	Mel Frequency Cepstral Coefficients
ML	Maximum Likelihood
MLP	Multi-Layer Perceptron
MSE	Mean Squared Error
NIST	National Institute of Standards and Technologies
ODCF	Optimal Decision Cost Function
PCA	Principal Component Analysis
PDF	Probability Density Function
PLP	Perceptual Linear Prediction
PLPCC	Perceptual Linear Prediction Cepstral Coefficients
PNN	Probabilistic Neural Network
PSC	Principal Spectral Components

RBF	Radial Basis Function
RCC	Real Cepstral Coefficients
ROC	Receiver Operating Characteristics
SAD	Speech Activity Detection
SOM	Self Organizing Map
SVM	Support Vector Machines
TDNN	Time Delay Neural Networks
UBM	Universal Background Model
VQ	Vector Quantization
VQG	Vector Quantization Gaussian
WPT	Wavelet Packet Transform

CHAPTER 1

INTRODUCTION

This chapter provides the thesis problem statement, specifies the thesis aims and the scope. This is followed by a short summary of the major contributions and the outline of each chapter.

1.1 Problem Definition

Speaker recognition techniques alongside with facial image recognition, fingerprints and retina scan recognition represent some of the major biometric tools for identification of a person.

Each of these techniques carries its advantages and drawbacks. The question to what degree each of these techniques provides unique person identification remains largely unanswered.

If these methods can provide unique identification then, it is still not clear what kind of parametric representations contain information which is essential for the identification process, and for how long and under what conditions, this representation remains valid? As long as these questions are unanswered, there is a scope for research and improvements.

This thesis investigates areas of possible improvements in the field of speaker recognition. The following drawbacks of the current speaker recognition systems have been identified as having a scope for potentials improvements:

1. The classical Gaussian mixture model (GMM) modelling and classification method uses the expectation maximization (EM) procedure to derive the probabilistic models of speakers. However it has been reported that EM suffers from slow convergence rates [36] and a tendency to end up at sub-optimal solutions. Various improving methods have been recently proposed [37]. This area of research has been currently very active due to the large interest in efficient modelling algorithms allowing real-time applications of the speaker recognition methodology.

2. The current state of art MFCC feature extraction method makes use of the using human auditory perception properties, which is believed to contribute largely its power to extract speaker specific attributes from voice. However it has been recently reported [32,33] that a fusion of MFCCs with other complimentary features has a potential to provide additional speaker-specific information and lead to better results. Current laryngological studies [272,273] revealed new nonlinear mechanisms underlying the speech production process. This lead to the definition of new types of features which have the potential to improve the speaker identification rates, however these features have not been yet sufficiently studied in speaker recognition applications.

3. Current speaker recognition systems face the challenge of performance degradation due to the speaker's aging, use of alcohol and drugs, changing health conditions and mental state. The exact effects of these factors on speaker recognition are not known. In this thesis we turned our attention towards effects of the depressive disorders on the speaker recognition rates, which has been known to have an effect on the acoustic properties of speech [235,236,237].

The depressive disorder affects approximately 18.8 million American adults or about 9.5% of the U.S. above 18 years of age [38]. Similar statistics have been reported in Australia and other developed nations.

1.2 Thesis Aims

The thesis aimed to investigate the advantages and drawbacks of the existing methodologies of the text-independent speaker verification, and to propose methods that could lead to an improved performance.

In particular the thesis aimed to:

- ➔ Propose an improved modelling and classification methodology for speaker recognition.
- ➔ Determine the usefulness of features derived from nonlinear models of speech production for speaker recognition.
- ➔ Determine the effects of a clinical environment containing clinically depressed speakers on speaker recognition rates.
- ➔ Investigate if the features based on nonlinear models of speech production have the potential to counteract the inverse effects of the clinically depressed environment.

1.3 Thesis Scope

- ➔ The study was limited to the text-independent speaker verification task.
- ➔ The modelling and classification methods used techniques such as: K-means, Linde Buzo Gray (LBG), ITVQ and Gaussian Mixture Models (GMM).
- ➔ The feature extraction was based on data driven techniques (i.e. techniques which calculate parametric features directly from the speech data) including: Mel Frequency Cepstral Coefficients (MFCCs), Inverse Mel Frequency Cepstral Coefficients (IMFCCs) and dynamic features such as delta (first derivative), double delta (second derivative), energy (E) and number of zero crossings (ZC). It also includes feature extraction methodologies based on the Teager Energy Operator (TEO).
- ➔ The algorithm's performance was tested using commercial speech corpora: NIST 2001, NIST 2002 and NIST2004 as well as TIMIT and YOHO.
- ➔ The effect of clinical environment on speaker verification was determined using speakers suffering from the clinical depression. The clinical speech data was obtained from the Oregon Research Institute (ORI), U.S.A.

1.4 Thesis Contributions

The major contributions of the thesis can be summarized as follows.

- ➔ A new method of deriving the Gaussian mixture model (GMM) parameters called the EM-ITVQ algorithm was proposed. The EM-ITVQ showed a significant improvement of the equal error rates and higher convergence rates when compared to the classical GMM based on the expectation maximization (EM) method.

- ➔ It was demonstrated that features based on the nonlinear model of speech production (TEO based features) provided better performance compare to the conventional MFCCs features.
- ➔ For the first time the effect of clinical depression on the speaker verification rates was tested. It was demonstrated that the speaker verification results deteriorate if the speakers are clinically depressed. The deterioration process was demonstrated using conventional (MFCC) features.
- ➔ It was demonstrated that when replacing the MFCC features with features based on the nonlinear model of speech production (TEO based features), the detrimental effect of the clinical depression on speaker verification rates can be reduced.

1.5 Thesis Outline

This thesis is divided into seven chapters,

Chapter 2 defines the speaker recognition task, describes briefly possible applications and summarizes conventional methods of speaker recognition. A general framework of the speaker recognition methodology comprising the training and testing stages is presented. Conventional methods used at each stage of the speaker recognition process are explained. These methods include pre-processing, feature extraction, speaker modeling, classification decision making and methods of assessing the speaker recognition performance. The final section includes a brief review of speech corpora most often used in the speaker recognition research.

Chapter 3 investigates the Vector Quantization (VQ) modeling for the speaker verification task. A relatively new vector quantization method based on the Information Theoretic principles (ITVQ) is for the first time used in the task of speaker verification and compared with two classical VQ approaches: the K-means algorithm and the Linde-

Buzo-Gray (LBG) algorithm. The chapter provides a brief theoretical background of the vector quantization techniques, which is followed by experimental results illustrating their performance. The results demonstrated that the ITVQ provided the best performance in terms of classification rates, equal error rates (EER) and the mean squared error (MSE) compare to K-means and the LBG algorithms. The outstanding performance of the ITVQ algorithm can be attributed to the fact that the Information Theoretic (IT) criteria used by this algorithm provide superior matching between distribution of the original data vectors and the codewords.

Chapter 4 introduces a new algorithm for the calculation of Gaussian Mixture Model parameters called Information Theoretic Expectation Maximization (ITEM). The proposed algorithm improves upon the classical Expectation Maximization (EM) approach widely used with the Gaussian mixture model (GMM) as a state-of-art statistical modeling technique. Like the classical EM method, the ITEM algorithm adapts means, covariances and weights, however this process is not conducted directly on feature vectors but on a set of centroids derived by the information theoretic vector quantization (ITVQ) procedure, which simultaneously minimizes the divergence between the Parzen estimates of the feature vector's distribution within a given class and the centroids distribution within the same class. The ITEM algorithm was applied to the speaker verification problem using NIST 2001, NIST 2002 and NIST 2004 corpora and MFCC with delta features. The results showed an improvement of the equal error rate over the classical EM approach. The EM-ITVQ also showed higher convergence rates compared to the EM.

Chapter 5 compares the classical features based on linear models of speech production with recently introduced features based on the nonlinear model. A number of linear and nonlinear feature extraction techniques that have not been previously tested in the task of speaker verification are tested. New fusions of features carrying complimentary speaker-dependent information are proposed. The tested features are used in conjunction with the

new ITEM-GMM speaker modeling method described in Chapter 4, which provided an additional evaluation of the new method. The speaker verification experiments presented in this chapter demonstrated significant improvement of performance when the conventional MFCC features were replaced by a fusion of the MFCCs with complimentary linear features such as the inverse MFCCs (IMFCCs), or nonlinear features such as the TMFCCs and TEO-PWP-Auto-Env. Higher overall performance of the nonlinear features when compared to the linear features was observed.

Chapter 6 for the first time investigates the effects of a clinical environment on the speaker verification. Speaker verification within a homogeneous environment consisting of the clinically depressed speakers was compared with the speaker verification within a neutral (control) environment containing of non-depressed speakers. Experiments based on mixed environments containing different ratios of depressed/non-depressed speakers were also conducted in order to determine how the depressed/non-depressed ratio relates to the speaker verification rates. The experiments used a clinical speech corpus consisting of 68 clinically depressed and 71 non-depressed speakers. Speaker models were built using the new ITEM-GMM method introduced in Chapter 4. Two types of feature vectors were tested, the classical ΔMFCC coefficients and the TEO-PWP-Auto-Env features. Experiments conducted within homogeneous environments showed a significant decrease of the equal error rates (EER) by 5.1% for the clinically depressed environment when compared with the non-depressed environment. Experiments conducted within mixed environments showed that an increasing number of depressed speakers lead to a logarithmic increase of the EER values; where the increase of the percentage of depressed speakers from 0% to 30% has the most profound effect on the increase of the EER. It was also demonstrated that the TEO-PWP-Auto-Env provided more robust performance in the clinical environments compare to ΔMFCC, lowering the EER from 24.1% (for ΔMFCC) to 17.1% (for TEO-PWP-Auto-Env).

Chapter 7 summarizes the key observations and presents the main conclusions of the thesis. Areas for future exploration based on the work reported in this thesis are also summarized in this chapter.

CHAPTER 2

SPEAKER RECOGNITION METHODS

This chapter defines the speaker recognition task, describes briefly the possible applications and summarizes the conventional methods of speaker recognition. A general framework of the speaker recognition methodology comprising the training and testing stages is presented. Conventional methods used at each stage of the speaker recognition process are explained. These methods include pre-processing methods, feature extraction techniques, speaker modeling methods, classification decision making methods and methods of assessing the speaker recognition performance. The final section includes a brief review of speech corpora most often used in the speaker recognition research.

2.1 Defining Speaker Recognition Task

Speaker recognition can be defined as the task of establishing the identity of speakers from their voices. The ability of recognizing voices of those familiar to us is a vital part of oral communication between humans. Research has considered automatic computer-based speaker recognition since the early 1970's taking advantage of advances in the related field of speech recognition.

The speaker recognition task is often divided into two related applications: speaker identification and speaker verification. Speaker identification establishes the identity of an individual speaker out of a list of potential candidates. Speaker verification, on the other hand, accepts or rejects a claim of identity from a speaker.

Speaker recognition may be categorized into closed set and open set recognition depending on whether the recognition task assumes the possibility that the speaker being identified may not be included on the list of potential candidates.

Speaker recognition may be further categorized into text-independent and text-dependent recognition. If the text must be the same for development of the speaker's template (enrolment) and recognition (testing) this is called text-dependent recognition. In a text-dependent system, the text can either be common across all speakers (e.g.: a common pass phrase) or unique. Text-independent systems are most often used for speaker identification. In this case the text during enrolment and identification can be different.

2.2 Applications of Speaker Recognition

In the recent years commercial applications of speaker recognition systems have become a reality. Speaker verification is starting to gain increasing acceptance in both government and financial sectors as a method to facilitate quick and secure authentication of individuals. For example, the Australian Government organization Centrelink already uses speaker verification for the authentication of Welfare recipients using telephone transactions [267].

Potential applications of speaker recognition include forensics [251], access security, phone banking, web services [268], personalization of services and customer relationship management (CRM) [11]. When combined with speech recognition, speaker recognition has the potential to offer most natural to human-computer means of communication.

Biometric applications of speaker recognition provide very attractive alternatives to biometrics based on finger prints, retina scans and face recognition [2,3]. The advantages of speaker recognition over these techniques include: low costs and non-invasive

character of speech acquisition, no need for expensive equipment, possibility of acquiring the data without speaker's active participation or even awareness of the acquisition process. As an access security tool, speaker recognition can potentially eliminate the need for remembering PIN numbers and passwords for bank accounts and security locks and various online services [12,13].

Moreover, speaker identification and verification is the only biometric technique that can be viably used over the telephone without the user having dedicated hardware. The key importance of speech as a biometric in commercial applications is probably more profoundly expressed by a patent held by IBM for the use of speech biometrics in telephony applications as well as the ongoing intense research in this area [270,271] carried by the IBM researchers.

The drawbacks of using speech as a biometric measure are in the fact that the available methodology is not yet reliable for stand-alone security, and it is used as a complimentary security measure. Due to the data-driven methodology, the performance of current speaker recognition systems is susceptible to changes in speaker characteristics due to the aging process, health problems and environment from which the user calls. Another disadvantage is the possibility of deception by using voice recordings instead of the actual voice of a speaker.

Speaker recognition methodology has been also widely adopted as a supporting measure complimentary to other biometric systems such as face recognition or retina scanning [1,45,46].

With rapidly increasing reliability of speaker recognition technology, speaker verification and identification is becoming a commercial reality and part of everyday consumers life. This thesis proposes a number of improvements to the existing speaker recognition technology. The proposed improvements include:

- A novel classification algorithm;
- a study of effects of clinical environment (a population of speakers that includes speakers suffering from clinical depression) on speaker recognition rates and
- testing of features that were not previously used in speaker recognition, and showed improved recognition rates not only in the neutral but also in the clinical environment.

2.3 Previous Studies of Speaker Recognition

Speaker recognition systems became the topic of research in the early 1970's [227] closely following the advancement in the related topic of speech recognition. Some of the first studies of speaker recognition were published in 1971 [14,15].

The advancements in speaker recognition were due to systematic improvements of the feature extraction and classification (or modeling) methods.

Early text-dependent speaker recognition used Dynamic Time-Warping (DTW) and template matching techniques for text-dependent speaker recognition. Some of the first text-independent approaches employed are linear classifiers [16] and statistical techniques [15].

The early used feature extraction technique included: pitch contours [151], Linear Prediction (LP) [74,76,162], cepstral analysis, linear prediction error energy and autocorrelation coefficients [16].

Current speaker recognition applications are focused almost exclusively on the text-independent tasks and therefore explicit template matching techniques are no longer used.

Modern feature extraction approaches are typically based on the analysis of short frames of speech over which the signal is assumed to be quasi-stationary with frame lengths ranging between 8-30 ms for speech sampled at the rates ranging between 8 kHz and 16 kHz.

The Cepstral analysis [77,167,206,207,218] and the Mel Frequency Cepstral Coefficients (MFCC) [30,31,32,52] are the most common short-time feature extraction approaches. Linear Prediction is not commonly used on its own, although sometimes applied as an intermediate technique to derive the MFCC [77]. Modifications of LP such as the Perceptual Linear Prediction (PLP) have been proposed [166] however PLP have not been widely used. Other suggested approaches which also have not been widely used include Line Spectral Pairs (LSP) [219], and Principal Spectral Components (PSC) [219].

A number of studies provided an extensive comparison of various feature extraction methods for speaker recognition. In [219] the PSC based on a critical 14 band filter bank and Principal Component Analysis (PCA) was found to provide very good performance. It was also observed that Linear Frequency Cepstral Coefficients (LFCC) and MFCC provided good performance. The LFCC marginally outperformed the MFCC due to the fact that LFCC provided better spectral resolution at high frequencies than MFCC. In a study by Reynolds [152], the PLP, MFCC and LFCC approaches were compared. It was again observed that LFCC provided the best performance but marginally outperforming the MFCC features.

It is reported in [32] that combining source features (supra-segmental features) and spectral features such as MFCC leads to better results. The results reported by Murty [33] and Prasanna [34] also pointed to the benefits of fusing MFCC with features providing complementary information.

A number of non-frame based feature extraction techniques including multi-resolution time-frequency approaches have been applied to speaker recognition. These methods include: Discrete Wavelet Transform (DWT) and Wavelet Packet Transform (WPT) [198,199,200,220,221,222,223,224]. The DWT and WPT allow the speech to be analyzed within multiple frequency bands representing different time-frequency and space-scale resolution. Although these methods have been recognized as having a great potential for extracting speaker-specific information, no effective method of using the combined temporal and spectral information has been developed.

As demonstrated in the speech recognition research [126,146,147,165,195,201,202], the feature selection process; that is selection of an optimal subset of features from an initially large set, can provide a significant improvement of the classification results. Magrin-Chagnolleau *et al.* [123], applied the Principal Component Analysis (PCA) as a feature selection method to speaker recognition. Kotani *et. al.* [124] applied a numerical optimization to the feature extraction and Lee *et al.* [121] used the Independent Component Analysis (ICA). In [115], Discriminative Feature Extraction Method (DFE) was also successfully applied as a feature selection method in speaker recognition.

A literature survey of studies concerning the speaker recognition task shows that the majority of research is focused on finding the best performing features. The modeling and classification methodology is also of inertest but plays a secondary role compare to the feature extraction.

The modern classifiers used in speaker recognition technology include Gaussian Mixture Models (GMM) [19], Hidden Markov Models (HMM) [17], Support Vector Machines (SVM) [101] Vector Quantization (VQ) [18], and Artificial Neural Networks (ANN) [20].

The HMMs are mostly used for text-prompted speaker verification, whereas GMM, SVM, VQ approaches are widely used for text independent speaker recognition applications. The GMM is currently recognized as the state of art modeling and classification technique for speaker recognition [19]. The GMM models the Probability Density Function (PDF) of a feature set as a weighted sum of multivariate Gaussian PDFs. It is equivalent to a single state continuous HMM, and may also be interpreted as a form of soft VQ [22].

The Support Vector Machines (SVM) has been used in speaker recognition applications in the past decade; however the improvements of performance over the GMM were only marginal [101,110]. A combined classification approach including SVM and GMM was reported to provide significant improvement over GMM [21].

Various forms of the Vector Quantization (VQ) methods have been also used as classification methods in speaker recognition [87,116]. The most common approach to the use of VQ for speaker recognition is to create a separate codebook for each speaker using the speaker's training data [116]. The speaker recognition rates based on the VQ were found to be lower than those provided by the GMM [242].

The GMM and VQ techniques are closely related, as GMM may be interpreted as a "soft form" of VQ [24]. Making use of that similarity, a combination of the VQ algorithm and a Gaussian interpretation of the VQ speaker model were described in [23]. In [24,25], the Vector Quantization was combined with the GMM method providing significant reduction of the computational complexity over the GMM method.

Matusi *et al.* [87], compared the performance of the VQ classification techniques with various HMM configurations. It was found that continuous HMM outperformed discrete HMM and that VQ based techniques become most effective in the case of minimal training data. Moreover, the study found that the state transition information in HMM

architectures was not important for text-independent speaker recognition. This study provided a strong case supporting the use of the GMM classifier since a GMM classifier can be interpreted as a HMM with only a single state. The Matsui *et. al.* findings were further supported by Zhu *et. al.* [22] who found that HMM based speaker recognition performance was highly correlated with the total number of Gaussian mixtures in the model. This means that the total number of Gaussian mixtures and not the state transitions are important for text-independent speaker recognition.

The ANN techniques have numerous architectures and a variety of forms have been used in the speaker recognition [117] task. The several ANN forms include Multi-Layer Perceptron (MLP) Networks, Radial Basis Function (RBF) Networks [127], Gamma Networks [20], and Time-Delay Neural Networks (TDNN) [118].

Fredrickson [119] and Finan [120] conducted separate studies comparing the classification performance of RBF and MLP networks. In both studies, the RBF networks were found to be superior. The RBF network was found to be more robust in the presence of imperfect training conditions due to its more rigid form. In other words, the RBF network was found to be less susceptible over training than the MLP network.

It was shown that some of the neural network configurations can provide results comparable with the GMM [233], however due to significant structural differences between neural networks and GMM, it is not possible to draw general conclusions as to which architecture is superior.

The above comparisons strongly indicate that the GMM provides the best performing classifier for speaker recognition tasks. For that reason, a number of most recent studies have been focused on the improvements of the classical GMM algorithm [23,24,243,244]. More details can be found in Chapter 4 (Section 4.3).

Any direct comparison of conventional speaker recognition architectures is difficult due to variation in the training and testing conditions, computational complexity of classifiers and feature extraction methods and types of speech data. The quality and number of speech samples used in the training and testing can have a significant impact on the performance of speaker recognition systems.

The only viable approach for comparison of speaker recognition architectures is a study directly comparing different architectures under the same training and testing conditions and using the same set of speech data. This approach has been undertaken in this thesis; a novel approach to the classification process described in Chapter 4, as well as the testing of different feature extraction methods in Chapter 5 were performed in parallel with the conventional state of art speaker recognition techniques and compared.

The literature survey strongly indicated that, to date, the MFCC feature extraction combined with the GMM modeling and classification procedure are widely recognized as the state of art methods providing the best speaker recognition results. For that reason the experiments described in this thesis use the MFCC's and the GMM classifier as the baseline method providing a reference point for the assessment of the new ITGMM classifier described in Chapter 4 and a number of feature extraction methods tested in Chapter 5.

2.4 Conventional Methods of Speaker Recognition

2.4.1 General Framework of the Speaker Recognition System

The existing speaker recognition methodology is based on so called data-driven techniques, where the recognition process relies on the parameters derived directly from

the experimental data and statistical models of these parameters build out of a large population of representative data samples.

The main advantage of the data-driven techniques is that there is no need for an analytic description of a processes being modeled. Thus, very complex biological, psychological or physiological processes can be modeled and classified without mathematical descriptions or knowledge of the underlying processes.

The major drawback of the data driven techniques is that the validity of such systems depends on the quality of the data used to derive the models. If the representative data changes in time or due to different environmental or noise factors, the enrolment process for speaker verification needs to be repeated to update the speaker's models.

A conventional speaker recognition system illustrated in Figure 2.1 is comprised of two stages: the first stage is called the enrolment or training process; the second stage is called the recognition or testing process.

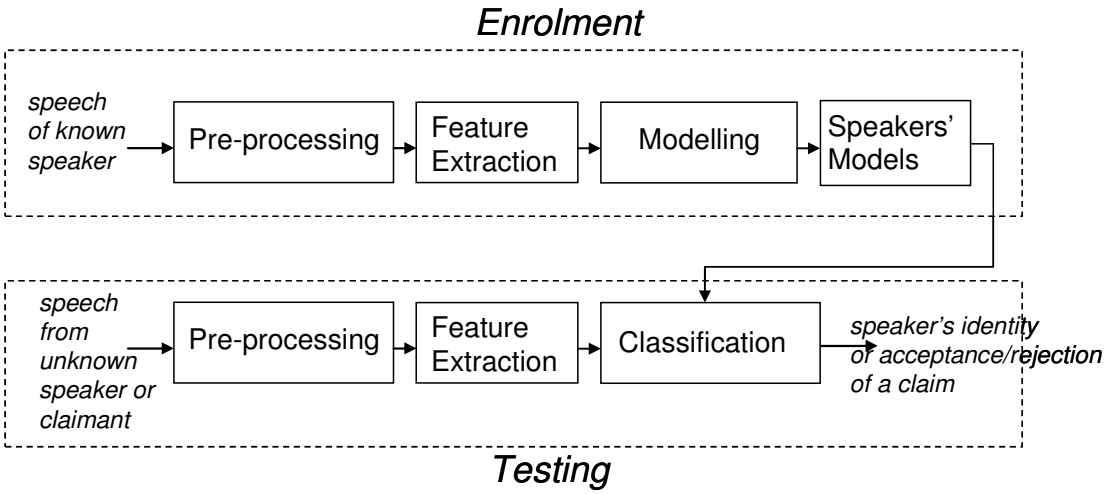


Figure 2.1 Major components of a conventional speaker recognition system.

During the enrolment (or training) stage speech samples from known speakers are used to calculate vectors of parameters called the characteristic features [48,49]. The feature vectors are then used to generate stochastic models (or templates) for each speaker. Since the generation of model parameters is usually based on some kind of optimization procedure iteratively deriving the best values of the model parameters, the enrolment process is usually time-consuming. For that reason, the enrollment procedure is usually performed off line and repeated only if the models are no longer valid. Figure 2.2 shows a typical functional diagram of the training process.

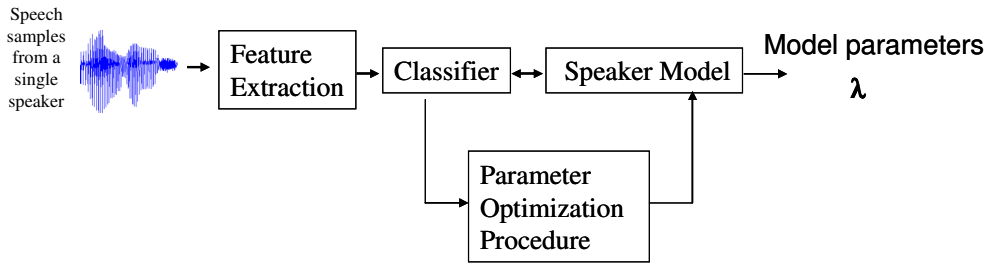


Figure 2.2 Enrollment (or training) phase for a speaker recognition system.

The testing phase is conducted after training; this is when the stochastic models for each class (speaker) have been already built. During the testing (or recognition) phase, the speaker recognition system is exposed to speech data not seen during the training phase [48,49]. Speech samples from an unknown speaker or from a claimant are used to calculate feature vectors using the same methodology as in the enrolment process. These vectors are then passed to the classifier which performs a pattern matching task determining the closest-matching speaker model. This process results in a decision making process which determines either the speaker identity (in speaker identification) or accepts/rejects the claimant identity (in speaker verification) [8,19,41,42,43,47]. The testing stage is usually relatively fast and can be done online in the real time conditions. Figure 2.3 shows a typical block diagram of the testing phase for speaker identification, whereas Figure 2.4 shows the testing phase for speaker verification.

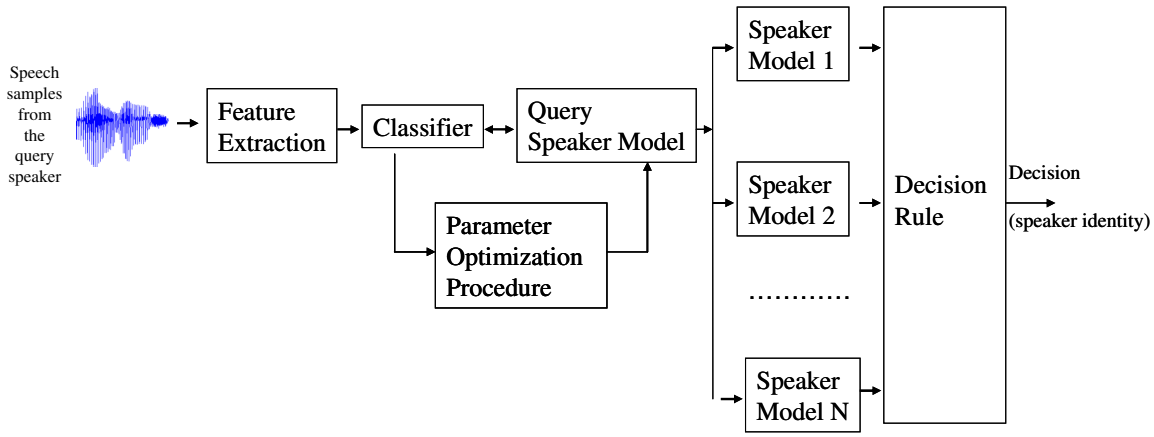


Figure 2.3 Testing phase for a speaker identification system.

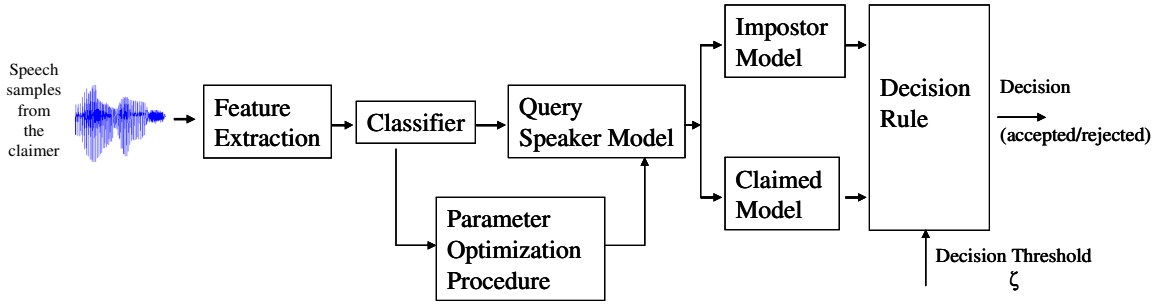


Figure 2.4 Testing phase for a speaker verification system.

2.4.2 Bayesian Decision Theory

The performance of a speaker recognition system is usually determined by the recognition rate or conversely by the error rate. Typical classifiers used in the speaker recognition systems employ the Bayesian minimum error decision rule theory providing

optimal recognition rates. Moreover, it is generalized such that the errors can have associated weights indicating their relative importance.

The generalized Bayesian rule defines a partition of a sample space to minimize the total cost due to classification errors [114] and can be seen as a technique for designing an optimal classifier.

Consider a classifier function $a()$ defined such that $j = a(Y)$ represents a decision $Y \in C_j$, where Y is an observation in the sample space and C_j is the j[th] class (speaker).

The value c_{ij} is then defined as a cost associated with classifying an observation $Y \in C_i$ when in fact $Y \in C_j$. From this definition the conditional risk function is defined as:

$$R(a(Y) = j \mid Y)) = \sum_{i=1}^{M} c_{ij} P(C_k \mid Y) \qquad \textbf{(2.1)}$$

Where $P(C_k \mid Y)$ is the posteriori probability of class C_k given the observation sample Y. The conditional risk $R(a(Y) = j \mid Y))$ is the expected cost given an observation at a particular point in the sample space.

The overall expected cost or risk of the classifier is then defined as the expected cost over the entire sample space and can be calculated by using the following function:

$$L = \sum_{i=1}^{M} \int_{\Omega_r} R(a(Y) = i \mid Y) P(a(Y) = i \mid Y) P(Y) dY \qquad \textbf{(2.2)}$$

Where $P(Y)$ is the probability of making the observation Y in the sample space Ω_r.

The Bayesian decision theory defines an optimal classifier as a classifier which minimizes the overall expected cost L given in Eq. (2.2).

Assuming that

- the classifier correctly models the *a posteriori* probability of each class in the sample space;
- the *a priori* probabilities of each class are known;
- the distribution of observations in the sample space are known;

Then the classifier which minimizes L will also minimize the cost due to the misclassification errors [99,100,102,114].

Such a classifier is commonly known as the Bayesian minimum risk classifier and it is defined by the following decision rule [102,114]:

$$a(Y) = j \text{ if } j = \arg\min_{k \neq j}(R(a(Y) = k \mid Y)) \quad \textbf{(2.3)}$$

If the costs associated with all misclassifications are equal, then the decision rule can be simplified to:

$$a(Y) = j \text{ if } j = \arg\min_{k \neq j}(P(C_k \mid Y)) \quad \textbf{(2.4)}$$

And such a classifier is commonly known as Bayes minimum error or Maximum a Posteriori (MAP) decision rule [102,114].

In the case when the *a posteriori* probability of each class is not available directly, the following formula can be used [19,40]:

$$P(C_k \mid Y) = \frac{P(C_j)P(Y \mid C_j)}{P(Y)} \quad \textbf{(2.5)}$$

Then, the Eq. (2.4) becomes:

$$a(Y) = j \text{ if } \quad j = \arg\min_{k \neq j}(P(C_k, Y)) \quad \textbf{(2.6)}$$

The rule in Eq. (2.6) is most commonly used rule in speaker recognition applications.

The Bayes minimum error decision rule is commonly used in speaker recognition based on the Gaussian Mixture Model (GMM) and Hidden Markov Model (HMM) classifiers.

In summary, a given classifier can be called Bayesian if:

- the architecture is capable of modeling the conditional probability density of each class;
- the estimation of the model parameters within that architecture has correctly modeled the class conditional probability density for each class.

The above two conditions become of particular concern when

- the data is incomplete
-observations are made in the presence of noise
-no definite description of the distribution of the multi-variate sample observations or class conditional densities are available.

Bayesian decision theory designs a classifier which is bounded by the separability of the classes in the feature space. Although, the Bayesian decision theory does not address

feature extraction design, it highlights the need for a design that maximizes the separability of classes in the feature space [114]

Source-filter model of speech production

Majority of current feature extraction methods in speaker recognition use parameters derived from the classical source-filter model. The classical source-filter theory of voice production assumes that the air flow through the vocal folds (source) and the vocal tract (filter) is unidirectional. During phonation, the vocal folds vibrate. One vibration cycle includes the opening and closing phases in which the vocal folds are moving apart or together, respectively. The number of cycles per second determines the frequency of the vibration, which is subjectively perceived as pitch or objectively measured as the fundamental frequency F_0. The sound is then modulated by the vocal tract configuration and the resonant frequencies of the vocal tract are known as formants. Finally the speech signal is passed through the low-pass lip radiation filter which reduces the signal energies with frequency by about 6 dB/octave [239].

The uniqueness of the speaker specific information may be attributed to several factors such as the shape and size of the vocal tract, dynamics of the articulators, rate of vibration of the vocal folds, accent imposed by the speaker and speaking rate. All these factors are reflected in the speech signal, and hence are useful for speaker recognition.

Pre-processing

The pre-processing stage used in speaker recognition [19,40] can include speech processing for noise removal and enhancement; it can also include compensation for the channel distortion, pre-emphasis filtering to remove effects of lip radiation as well as removal of silence and in some cases unvoiced speech intervals.

Each of these approaches provides improvements to speaker verification performance over telephony channels.

In this study it is attempted to compare the performance of proposed classification method (see Chapter 4) or different feature extraction methods (see Chapter 5) and thus no explicit channel compensation or noise removal is used. However, the silence/voiced removal is used as required by some of the feature extraction techniques.

Short-time analysis

During the pre-processing stage speech is usually divided into short-time frames using a windowing process and the subsequent feature extraction is performed on the frame-by-frame basis.

The reason for a short-time approach to the feature extraction is based on the fact that a speech signal can be viewed as a piecewise stationary signal or a short-time stationary signal. In a short-time (e.g., 10-30 milliseconds)), speech can be approximated as a stationary process [50,148,149,150,151,152]. Feature vectors extracted from speech on the frame-by-frame basis can therefore be used to generate stochastic models using approaches such as the Gaussian Mixture Model (GMM) or the Hidden Markov Model (HMM).

The determination of the analyzing window length depends on whether the analysis aims to extract the speech source, vocal tract characteristics or long-term characteristics (e.g. word duration, intonation, speaking rate or accent) [44].

To obtain the information embedded in the vocal tract, speech is analyzed using segmental analysis with frames of length 10-30ms. In the range of 10-30ms few pitch intervals can be captured providing information about the vocal tract characteristics [50].

The segmental analysis is the most widely used method to perform feature extraction for speaker recognition [51,148,149,150,151,152].

To obtain the information embedded in the excitation source sub-segmental analysis is used with speech frames of length 3-5ms [153]. The sub-segmental analysis is designed for capturing information within a single pitch period. Examples of the sub-segmental speech analysis are described in [32,33,154,155,156,157].

For supra-segmental analysis the speech is analyzed using the frames and overlap in between 100-300ms. This analysis method is appropriate to extract the information due to behavioral traits. It includes word duration, intonation, speaking rate, accent, etc. The information varying is relatively slower for behavioral traits thus large sized frames would serve the purpose. The supra-segmental analysis for speech frames is used in [32,151,158,159,160] demonstrating that some behavioral traits can be captured with this analysis of speech.

Speech activity detection (SAD)

An energy based approach proposed by Reynolds [144] was used for the detection of speech activity. This approach has been applied a number of times by Reynolds in the state of art speaker recognition configuration including the MFCC as features and the Gaussian Mixture Model (GMM) as the classifier.

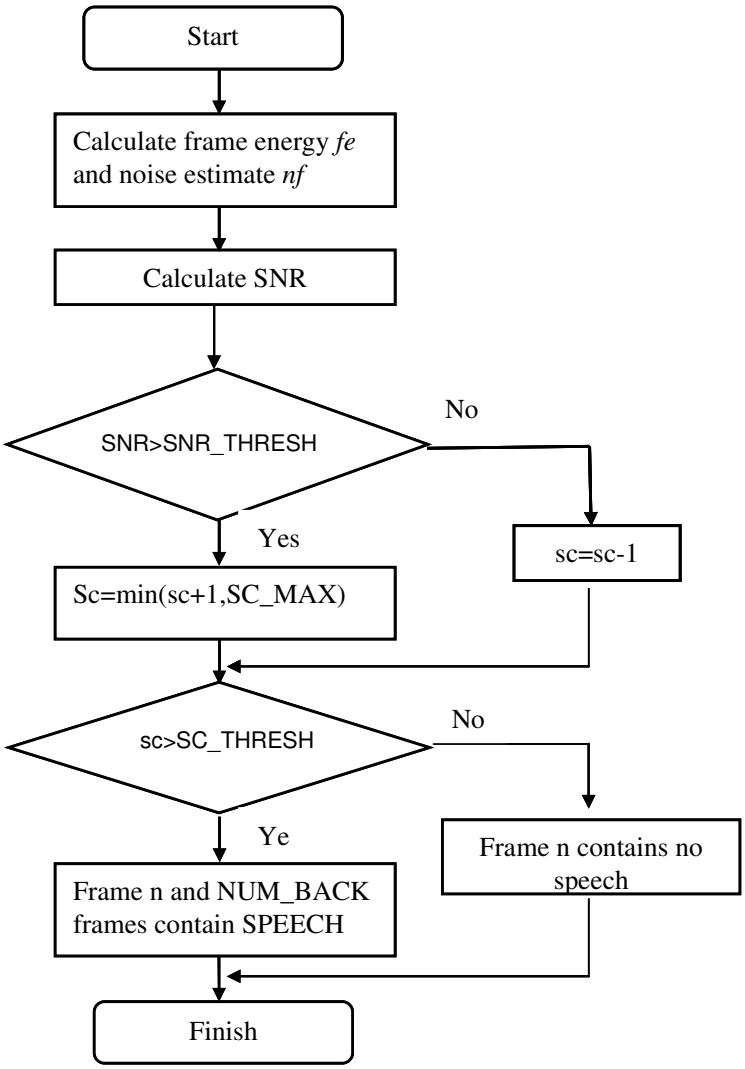


Figure 2.5 Speech Activity Detection Procedure. Adapted from Reynolds [144].

The SAD algorithm is a typical energy based speech activity detection method and it uses an adaptive estimate of the noise energy. The estimate of the noise floor energy adaptively tracks the minimum value of the smoothed energy contour for each frame of the speech signal.

The SAD procedure is performed in the following three major steps [144]:

STEP 1: Raise the noise floor nf[]:

$$nf[n] = 1.01 nf[n-1]$$

STEP 2: Track the lower value nf[] of the smoothed energy contour se[]:

$$se[n] = \frac{fe[n] + fe[n-1]}{2}$$

$$nf[n] = \min(nf[n], se[n])$$

STEP 3: Step down control for transition from speech to silence:

If $(nf[n] > 2\, fe[n])$

$$nf[n] = \frac{fe[n]}{2}$$

End

As illustrated in the algorithm flowchart in Figure 2.5, the estimated values of nf[] and se[] are used to calculate the signal to noise ratio (SNR) for a given frame. Frames with SNR lower than a given constant threshold (SNR_THRESH) are assumed to contain no speech activity. The procedure can be adapted to distinguish between voiced and unvoiced segments of speech.

Figure 2.5 Shows that the SAD algorithm uses several design parameters including:

SNR_THRESH - threshold to which current SNR estimate is compared to determine whether to increment or decrement counter SC.

SC_THRESH - threshold of counter variable SC to determine whether a frame is speech

SC_MAX - maximum value of counter variable SC to limit transition duration from speech to silence.

NUM_BACK - number of frames back to classify as speech in a silence to speech transition.

The values of these parameters need to be determined experimentally for a specific application. In [144] Reynolds provided values used in his speaker recognition experiments. These values are summarized in Table 2.1.

Table 2.1 An example of SAD parameters used by Reynolds in [144].

Parameter	***Reynolds Value [144]***
SNR_THRESH	5dB
SC_THRESH	10
SC_MAX	20
NUM_BACK	10

2.4.3 Features Extraction Methods Used in Speaker Recognition

The process of converting a raw speech signal into a sequence of acoustic feature vectors carrying characteristics information about the speaker is called feature extraction.

The attributes of an ideal feature extraction strategy described in [6,47,51] include:

a) The features should be resistant to an environmental noise and channel distortion
b) Variations in voice caused by speaker's health or aging should not degrade the performance of feature extraction methodology.
c) Feature extractor should maintain high inter-speaker discrimination and as little as possible of intra-speaker variability.
d) The speaker-characteristic features extracted from speech should be relatively easy to calculate.
e) The feature extraction method should be difficult to imitate or mimic using speech of imposters.

The above attributes are difficult to achieve in a single feature extraction procedure. This is because some of the attributes listed above have an inverse relationship; if one is improved the other deteriorates. For example, a large value of the inter-speaker variability (high discrimination) can be obtained with short term spectral method [47,52], however this approach can be easily corrupted when transmitted over a noisy channel. Features such as fundamental frequency F0 are noise robust but requires long speech segments which leads to the reduction of the speaker discrimination capability.

Despite numerous studies examining the source and extent of variability in speech signal [31,55,60,69,70,78], there has been no conclusion from the linguistic, acoustic or forensic point of view, as to what constitutes a "voice print". As a result a variety of parameters representing speaker-characteristic features have been proposed and successfully applied in speaker recognition tasks.

Although, no unique features distinguishing between all speakers are known, it appears that the inter-speaker variability can be observed within speakers on many levels, including temporal and spectral variability.

Speech features used in the classical applications of speaker recognition can be divided using different criteria. Based on the domain in which the analysis is conducted [45,53], the characteristic features can be divided into:

- *spectral features* - descriptors of the short-term speech spectrum, the spectral features represent entirely or partially the physical characteristics of the vocal tract;
- *dynamic features* - time variations of other features such as spectral features;
- *prosodic features* - refer to the fundamental frequency F_0 and energy contours.

Based on the time duration of the analyzed speech segment, the prosodic features can be divided into the following categories:

- *source features* – prosodic features within a single glottal period;
- *suprasegmental features* – prosodic features spanning a few glottal periods;
- *high-level features* – long time features spanning the time duration of a word or utterance.

Table 2.2 shows typical examples of different type of features.

Table 2.2 Types of features and examples.

Type of features	***Examples***
Spectral features	MFCC LPCC LFCC
Dynamic features	Velocity/acceleration features Feature fusion multivariate auto-regression (MAR)
Prosodic features	Pitch and energy contours
Source features	Glottal pulse shape
Suprasegmental features	F0 contours Intensity contours
High level features	Pronunciation Word duration

Spectral features

The spectral features have been the main focus in the speaker recognition studies. The proposed methods include: Real Cepstral Coefficients (RCC) introduced in [164], Linear Prediction Coefficients (LPC) proposed in [161], Linear Predictive Cepstral Coefficients

(LPCC) derived by Atal in [162], and Mel Frequency Cepstral Coefficients (MFCC) derived by Davis and Mermelstein in [163].

The Mel-frequency cepstral coefficients (MFCC) are the most widely used acoustic features for speaker modeling and recognition. The MFCC are the cosine transform coefficients calculated for the log power spectrum mapped onto the Mel-frequency scale. The Mel-frequency bands are equally spaced on the logarithmic scale, which approximates the human auditory system's response. For each frame of a speech signal, the Fourier transform and the power spectrum was calculated. The powers were then mapped onto the Mel scale and the logs of the powers were estimated at each of the Mel-frequencies. Usually, the first 12 coefficients of the discrete cosine transform DCT applied to the Mel log powers provided the MFCC features.

More details about the calculation process and the properties of the MFCC can be found in Chapter 5.

Spectral features such as the Linear Frequency Cepstral Coefficients (LFCC) [163], are similar to the MFCC however instead of the logarithmic Mel frequency spectral subdivision, a linear scale is used providing equally spaced filters on the linear rather than logarithmic scale covering the entire signal bandwidth.

Other types of spectral features include: Perceptual Linear Prediction (PLP) coefficients [166] and the Adaptive Component Weighting (ACW) cepstral coefficients [167,168].

A study by Reynolds in 1994 [152] compared the different features like MFCC, LFCC, LPCC and perceptual linear prediction cepstral coefficients (PLPCCs) for speaker recognition. It was observed that the MFCCs and LPCCs gave significantly higher correct recognition rates than the other features.

From a perceptual point of view, MFCC bear resemblance to the human auditory system, since these features account for the nonlinear nature of pitch perception. This is the primary reason of performance supremacy of MFCC features. This success of MFCC, combined with their robust and cost-effective computation, turned MFCC into a reality in the speech/speaker recognition applications. Recently a number of modifiers of MFCC are introduced and have shown better performance. A number of MFCC variants are described in Chapter 5.

The Perceptual Linear Predictive (PLP) speech analysis technique is based on the short-term spectrum of speech. The short-term spectrum of speech is subsequently modified by several psychophysically based spectral transformations, the PLP technique like most other short-term spectrum based techniques, is vulnerable when the short-term spectral values are modified by the frequency response of the communication channel. Human speech perception seems to be less sensitive to such steady-state spectral factors.

Dynamic features

The features which represent time derivatives of the spectrum-based features are referred to as the dynamic features. Dynamic cepstral features such as delta (first derivative of cepstral features) and double-delta (second derivative of cepstral features) have been shown to play an essential role in capturing the transitional characteristics of the speech signal [169]. A set of new dynamic features for speaker verification system was introduced in [170]. These new features, known as Delta Cepstral Energy (DCE) and Delta-Delta Cepstral Energy (DDCE), can compactly represent the time varying cepstral information. Dynamic features based on MFCC and LPCC were used in [171,173] respectively. Shifted Delta Cepstrum was used in [174] for the speaker recognition and have shown promising results. A method using statistical dynamic features has recently been proposed. In this method, a multivariate auto-regression (MAR) model is applied to the time series of cepstral vectors and used to characterize speakers [172].

The fusion of the cepstra and delta cepstra features have been shown to provide relatively good results for the task of speaker recognition [169,170]. It has been demonstrated that the speaker recognition system performance may be enhanced by adding time derivatives to the static parameters. The first order derivatives referred as delta features [175] can be calculated using the following general formula:

$$d_t = \frac{\sum_{\theta=1}^{\Theta} \theta (c_{i+\theta} - c_{i-\theta})}{2\sum_{\theta=1}^{\Theta} \theta^2} \quad \textbf{(2.7)}$$

Where d_t is the delta coefficient at time t, computed in terms of the corresponding static coefficients $c_{t-\theta}$ to $c_{t+\theta}$ and Θ is the size of delta window. The delta and double-delta cepstra are evaluated based on MFCC and performance improvements are observed, the details are given in Chapter 5.

Prosodic features

Prosodic speech features, are often used to extract the information about the speaking style of a person. The fundamental frequency, formants and the frame energy are the most commonly known prosodic features. These features are also often appended to their logarithmically compressed values and added to the spectrum-based speech parameters in order to obtain the better performance. The use of the temporal derivatives of the fundamental frequency and the frame energy has also remained in practice. A set of statistical parameters evaluated based on the temporal parameters has also established better performance for the speaker recognition systems. The feature extraction methodology proposed in [176] introduces a number of improvements to the estimation of the fundamental frequency and accent. These improvements include the re-synthesis of

the pitch contour which removes the doubling/halving that occurs during the calculation process of the fundamental frequency.

The drawbacks of the prosodic features include the fact that they can be easily mimic or imitated. A combination of prosodic information with the spectrum-based features could lead to a better performance and eliminate the possibility of features being imitated.

Fusion of features

The MFCC have appeared as a performance superior feature extraction method for speaker recognition. Dynamic features or features extracted from prosodic information could be helpful when fused with spectrum-based features, but could not lead to a state of the art design individually. Much more efficient results could be obtained when using combinations (or fusions) of features.

The linear prediction (LP) residual also contains speaker-specific source information [33] which can enhance the performance of speaker recognition systems. It has been reported [54] that a combination of the LP residual with LPCC or MFCC improves the performance as compared to that of MFCC or LPCC alone [34,155,156,157].

Plumpe *et al.* [55] developed a technique for estimating and modeling the glottal flow derivative waveform from speech for speaker recognition. In his study, the glottal flow estimate was modeled as coarse and fine glottal features, which were captured using different techniques. Also, it was shown that the combined coarse and fine structured parameters gave better performance than the individual parameter alone [32,55]. In [145] methods are proposed to extract the speaker specific information from high-level features.

In the past few years an increasing interest has been observed on using several information fusion methods in speaker recognition [47,62,63,64,65,66,67,68]. The feature information fusion can be seen in several forms, such as multi-feature fusion and multi-sample fusion [47]. A target speaker might be conditioned to utter same phrase for a number of times and the decision is thus based on combining the scores [67], this is called multi-sample fusion. In multi-feature fusion approach, same speech utterance is used to extract different features. The example is the use of MFCC cepstra with its delta cepstra. This approach is also used to develop an improved representation of features which is detailed in Chapter 5.

Classifier fusion strategies have also been used to obtain improved recognition results. The fusion at classifier levels combines the match scores to obtain the final decision [62]. The feature extraction strategy is same for the multiple classifiers. Thus the fusion can appear in one of the two forms. By combining the features at the frame level into a vector for which a single model is trained, or by modelling each feature set using a separate classifier. In this thesis the fusion at the feature level is used, detail based on the experiments is given in Chapter 5. The fusion of the static spectral features with their corresponding time derivatives to capture complementary feature information has remained a common practice. The use of single classifier and allowing better discrimination between the speakers are few of the advantages of information fusion at feature level.

2.4.4 Speaker Modeling and Classification Techniques

The modelling techniques transform the voice features of a speaker to an identical representation. The objective of modelling technique is to generate speaker models using speaker-specific feature vectors.

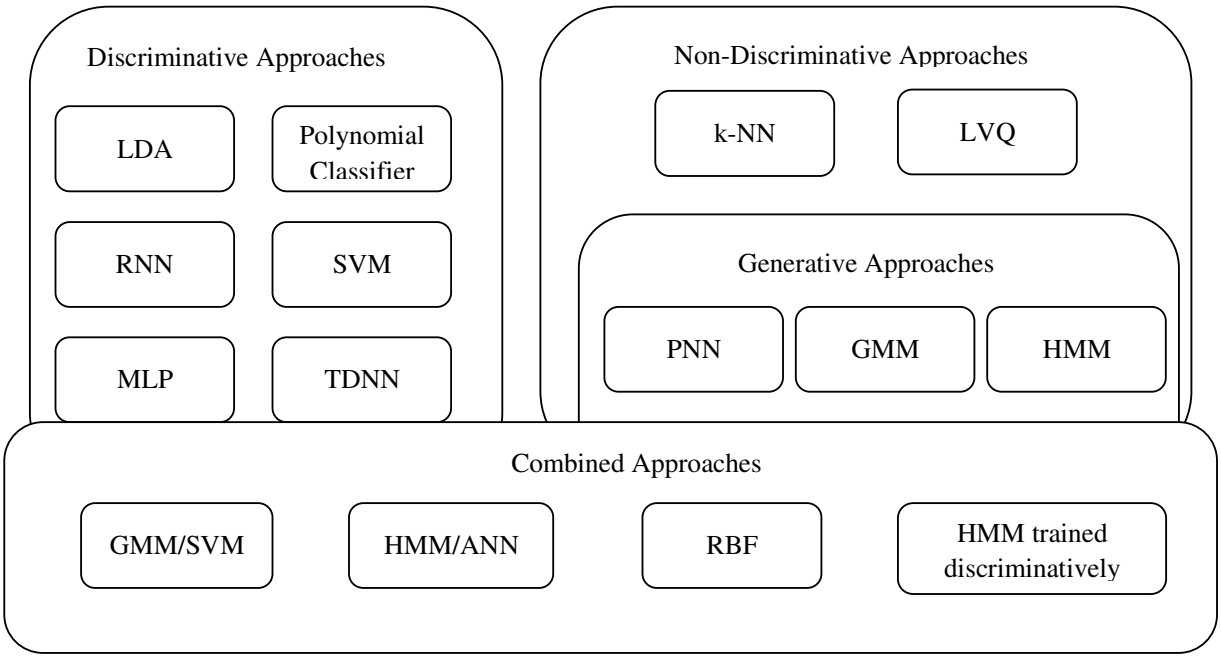


Figure 2.6 Major Modelling Approaches for Speaker Recognition [53].

The Modelling techniques can be classified as generative or discriminative as shown in Figure 2.6. The template matching techniques [71,72,74,75,76,148,162] were the most widely used techniques for speaker recognition at the early stages of this technology. In this approach training and testing feature vectors are directly compared using similarity measure. For the similarity measure, any of the techniques like spectral distance or Euclidean distance or Mahalanobis distance is used.

Dynamic time warping (DTW) [77] for text-dependent speaker recognition was first used by Furui. In this approach, the sequence of feature vectors of the training-speech signal is the text-dependent template model. The DTW finds the match between the template model and the input sequence of feature vectors from the testing-speech signal. The disadvantage of template matching is that it is time consuming, as the number of feature vectors increases. For this reason, it is common to reduce the number of training feature vectors by some modelling technique like clustering.

K-means algorithm

The *K-means* algorithm [79] is one of the most widely used classifiers based on the vector quantization techniques.

The K-Nearest Neighbors (K-NN) method is a classification algorithm where the input feature vector is classified based on the class represented by the majority of the K nearest feature vectors obtained during the training process. Given an input feature vector, the algorithm finds K closest feature vectors representing different classes (speakers). The class represented by the majority of the K nearest feature vectors is assigned to the input vector.

The major drawback of the KNN classification is that the classes with the more frequent examples tend to dominate the prediction of the new vector, as they tend to come up in the K nearest neighbors when the neighbors are computed due to their large number. One way to overcome this problem is to weight the classification taking into account the distance from the test point to each of its K nearest neighbors.

A number of enhanced versions of the original K-means algorithm have been proposed.

Linde-Buzo-Gray (LBG) clustering technique was used in [18] for speaker recognition. It was demonstrated in [18] that a larger codebook gives better performance and how using a VQ quantizer can handle the performance degradation due to different recording conditions and intra-speaker variations.

Fuzzy vector quantization (FVQ) using the well-known fuzzy *C-means* method was introduced by Dunn, and its final form was developed by Bezdek [81, 82]. FVQ was used to classify the speaker models in [83, 84]. It was demonstrated that FVQ gives better performance than the traditional *K-means* algorithm because the feature vectors are associated with all the clusters and there are relatively more number of feature vectors for each cluster.

In Chapter 3 information theoretic vector quantization is investigated against K-means and the LBG algorithms.

Hidden Markov Model (HMM)

The Hidden Markov Model (HMM) is created using continuous probability measures of Gaussian mixture models (GMM) [85,86,87]. HMM's are used for text-dependent speaker recognition in [85,86,87]. In HMM, time-dependent parameters are observation symbols which are created by VQ codebook labels. The 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 a given speaker model is calculated for speaker recognition. The use of HMM for text-independent speaker recognition under the constraint of limited data and mismatched channel conditions was demonstrated in [88].

Neural Networks (NN)

Neural networks have been widely used for pattern recognition problems; the strength of neural networks to discriminate between patterns of different classes is exploited for speaker recognition [89,90,91]. Neural network has an input layer, one or more hidden layers and an output layer. Each layer consists of processing units, where each unit represents model of an artificial neuron, and the interconnection between the two units as a weight associated with it.

The concept of the *Multi-Layer Perception Neural Network* (MLPNN) was used for speaker recognition in [92]. In this work, a comparative analysis between MLPNN and VQ methods is given. Another form of neural networks called radial basis function (RBF) was used for speaker recognition task in [93]. In this work the performance superiority of the RBF to the VQ and MLP is demonstrated.

The *Self-Organizing Map* (SOM) is a special class of neural network based on competitive learning [94]. The SOM was applied to speaker recognition in [95,96]. The disadvantage of SOM is that it does not use class information while modelling speakers, resulting in a poor speaker model that leads to degradation in the performance. Linear vector quantization (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. LVQ was proposed for speaker recognition in [97]. Speaker recognition using VQ, LVQ and GVQ (group vector quantization) was demonstrated for YOHO database in [98].

Auto-associative neural network (AANN) was developed for pattern recognition task [32,33,34,103,104,106,155], and was used as an alternative to GMM. AANN is a feed-forward neural network, where the number of units in the input and output layers is 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; however there is no significant evidence that AANN is superior to GMM in computational efficiency or recognition scores.

Probabilistic Neural Network (PNN)

The probabilistic neural network (PNN) [234] is a feed forward network derived from the Bayes decision method. It estimates the probability density function for each class based on the training samples. It calculates Parzen estimates of the probability density function for each test vector.

The PNN structure consists of three layers: the input layer, the hidden layer and the output layer. The input layer represents the test vectors, and it is fully connected to the hidden layer. The hidden layer has a node for each training vector. Each hidden node calculates the dot product between the input vector and the test vector, subtracts 1 from it, and divides the result by the standard deviation squared.

The output layer has a node for each class. The sum for each hidden node is sent to the output layer and the output node with the highest value determines the class for the input test vector. The PNN has a very short training time compared with other classifiers, since the training is done in a single pass of each training vector, rather than several. However, due to its structure the execution of the PNN program requires large amount of memory, especially when the training and testing datasets are large.

The PNN shows rather high sensitivity to noisy data compared with other classifiers. It does not work well with data that is not highly representative.

Support Vector Machines (SVM)

In recent years, support vector machines (SVMs) have been widely used to solve binary classification problems. In a binary classification problem, a SVM constructs a hyperplane in a multidimensional vector space, which is then used to separate vectors that belong to two different classes. A good separation is achieved by the hyperplane that has the largest distance to the nearest training vectors of each class.

Given a two-class problem and a training set of N vector-class pairs: $\{[\mathbf{x}_1, y(\mathbf{x}_1),...,[\mathbf{x}_N,y(\mathbf{x}_N)]\}$, where $\mathbf{x}_i, \mathbf{x}_j \in R^D$ are the D-dimensional feature vectors, and $y(\mathbf{x}_i) \in \{-1; +1\}$ are the actual class labels for vectors $\mathbf{x}_i$, the classification labels for vectors $\mathbf{x} \in R^D$ are produced by the decision function s(**x**) defined as:

$$s(\mathbf{x}) = sign(\sum_{i=1}^{N} y(\mathbf{x}_i)\alpha_i K(\mathbf{x}, \mathbf{x}_i) + b) \qquad \textbf{(2.8)}$$

Where $\boldsymbol{\alpha} = \{\alpha_i\}_{i=1,..,N}$ and b are the SVM model parameters, and $K(\mathbf{x}, \mathbf{x}_i)$ is a positive definite kernel chosen to be the Gaussian type of the *Radial Basis Function* (RBF):

$$K(\mathbf{x}, \mathbf{x}_i) = -\gamma \exp\left(\left\|\mathbf{x} - y\right\|^2\right) \qquad \textbf{(2.9)}$$

The values of the model parameters $\boldsymbol{\alpha} = \{\alpha_1,...,\alpha_N\}$ and b are not unique and have to be determined during the training process using a quadratic optimization procedure. The SVM algorithm finds an optimal vector of parameters $\boldsymbol{\alpha} = \{\alpha_i\}_{i=1,..,N}$ that minimizes the following objective function:

$$f_{obj}(\boldsymbol{\alpha}) = -\sum_{i=1}^{N}\alpha_i + \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_i\,\alpha_j\,y(\mathbf{x}_i)\,y(\mathbf{x}_j)K(\mathbf{x}_i,\mathbf{x}_j) \tag{2.10}$$

Subject to:

$$\underset{i=1,\ldots,N}{\forall} 0 \le \alpha_i \le C \quad \text{and} \quad \sum_{i=1}^{N}\alpha_i\,y(\mathbf{x}_i) = 0 \tag{2.11}$$

The parameter C is a suitable positive constant value controlling how strictly the classifier fits the training data. For a given vector $\boldsymbol{\alpha}$, the model offset parameter b can be calculated using:

$$b = \frac{1}{N_{SV}}\sum_{i=1}^{N_{SV}}\left(\left(\sum_{i=1}^{N}\alpha_i\,y(\mathbf{x}_i)\mathbf{x}_i\right)\cdot\mathbf{x}_i - y(\mathbf{x}_i)\right) \tag{2.12}$$

Where $\mathbf{x}_i$ in Eq. (2.12) are feature vectors with nonzero values of the corresponding α_i and N_{SV} is the number of nonzero valued α_i coefficients. Once the optimal set of model parameters $\boldsymbol{\alpha}$ and b are determined during the training process, test vectors $\mathbf{x}$, can be classified using Eq. (2.8).

The two-class SVM method can be expanded to a multiclass problem. It is usually done by reducing the single multiclass problem into multiple binary classification problems. Each of the problems yields a binary classifier, which is assumed to produce an output function that gives relatively large values for examples from the positive class and relatively small values for examples belonging to the negative class.

In [110, 111] applications of SVM to the speaker recognition have been reported. In these studies, efficiency of the score-space kernels which are generalization of Fisher's kernel functions were examined It was demonstrated that the SVM reduces error rates by 34%

comparable to the GMM classifier. However, a relatively large number of computations required by the SVM have been identified as a major drawback.

The generalized linear discriminate sequence (GLDS) kernel for the speaker recognition and language identification tasks was introduced in [101]. It was shown that although the SVM performance was very close to the GMM performance, the combination of SVM and GMM yielded better recognition rates than the individual methods. The combination of SVM with GMM was also found to provide good results for the task of speaker recognition [112,113].

Gaussian Mixture Models (GMM)

The Gaussian mixture model (GMM) method models speech as a weighted sum of multivariate normal probability density functions (pdf) [19]. Each pdf is called a component of the GMM. A GMM with M components is said to be a GMM of order M. For an R-dimensional feature vector x, the posteriori probability for M component GMM and the probabilistic model λ is defined as,

$$P(x|\lambda) = \sum_{m=1}^{M} w_m p_m(x) \qquad \textbf{(2.13)}$$

The Eq. (2.13) corresponds to the weighted linear combination of M unimodal Gaussian densities. The probability density function $p_m(x)$, is given by,

$$p_m(x) = \frac{1}{(2\pi)^{R/2}\left|\sum_m\right|^{1/2}} e^{-\frac{1}{2}(x-\mu_m)^T \sum_m^{-1}(x-\mu_m)} \qquad \textbf{(2.14)}$$

Each of the pdf is parameterized by an R–dimensional mean vector μ_m, RxR-dimensional covariance matrix $\sum_m$ and a mixture weight w_m, also known as a priori probability. The a priori probability satisfies the following constraint,

$$\sum_{m=1}^{M} w_m = 1 \qquad \textbf{(2.15)}$$

The set of parameters, $\lambda = \{\mu_m, \sum_m, w_m, 1 \le m \le M\}$ completely define a GMM. The use of single covariance matrix for the entire set of components is adapted in some of the applications; however for speaker recognition technology this practice is not common.

Given a GMM $\lambda = \{\mu_m, \sum_m, w_m, 1 \le m \le M\}$ and a set of feature vectors $X = \{x_1, x_2, ..., x_T\}$, the log likelihood of the model is computed as,

$$\log P(X|\lambda) = \sum_{t=1}^{T} \log P(x_t|\lambda) \qquad \textbf{(2.16)}$$

The constraints applied to the GMM include: *a priori*-probability, covariance matrix, and initialization of the GMM parameters.

Priori probability

The *a priori* probabilities of the Gaussian components maintain the requirement that it should be summed to 1, as shown in Eq. (2.15). This constraint keeps the reliability of posteriori probability estimate of the GMM. The parameter w_m represents the a priori probability of each Gaussian component so it maintains the condition $0 \le w_m \le 1$. In other words a minimum value except zero may be enforced so that each Gaussian density may

have a reliable share in optimization of probabilistic model. This approach would ultimately lead avoid singularities or over-fitting to the training data.

Covariance Matrix

For speaker recognition technology a local covariance matrix for each Gaussian density is adapted, thus it can lead to a substantial computational burden so a number of careful practical restrictions are then applied to selection of covariance matrix. As it can be depicted from Eq. (2.14) that the covariance matrix is a matrix of size RxR. Typically in speaker recognition applications the covariance matrix is restricted to being a diagonal matrix. Reynolds [19,40] suggests based on the empirical evidence that diagonal covariance matrices outperform full covariance matrices. This restriction reduces the trainable covariance parameters to R parameters per Gaussian component. The covariance matrix for R=4, is shown below.

$$\sum_{m,D=4} = \begin{Bmatrix} \sigma_m(1) & 0 & 0 & 0 \\ 0 & \sigma_m(2) & 0 & 0 \\ 0 & 0 & \sigma_m(3) & 0 \\ 0 & 0 & 0 & \sigma_m(4) \end{Bmatrix} \quad \textbf{(2.17)}$$

Initialization of GMM Parameters

The initial distribution of the training data before tuning by EM procedure would have significant impact on the overall training procedure of the speaker models [29,129]. K-means clustering procedure is most commonly used technique to initialize the training speaker data. The K-means initializes the clusters for the feature vectors, each cluster would then become a single component of the GMM. The initial values of weights,

means and covariances for each of the Gaussian densities are calculated using conventional statistics. The weights are determined by reciprocating the total number of Gaussian densities. The K-means algorithm is defined in detail in Chapter 3, where LBG and information theoretic clustering are used to classify the speaker models. Several K-means variants also exist, however for this thesis it is principally used to initialize Gaussian components.

Maximum Likelihood Estimation of GMM Parameters

The maximum likelihood (ML) approach to parameter estimation is mostly adapted by GMM based speaker recognition. The ML approach estimates parameters to maximize the likelihood. In other words ML estimation would lead to maximizing the posteriori probability that the GMM produced the observed feature vectors belonging to class. The expectation maximization (EM) algorithm is widely used to obtain a ML estimate λ, given an initial estimation for λ.The Expectation Maximization (EM) algorithm was developed by Dempster [122]. It is an optimization procedure that enables the ML parameter estimation. This procedure does not take into account that which Gaussian component any particular observation belongs to. The EM procedure used for training speaker models is demonstrated in [19,40].

The expectation maximization procedure is elaborated in detail in Chapter 4, where the proposed EM-ITVQ is described. However in this chapter a general description of the procedure is given.

The EM algorithm is performed in two stages, expectation and maximization. During expectation stage the posterior probability based on the Gaussian densities is evaluated and during maximization stage the parameters are re-evaluated in a manner that

guarantees the improvement. This is equivalent to saying that $p(x|\overline{\lambda}) \geq p(x|\lambda)$. A thorough description of EM algorithm can be found in [40,122].

The ML estimates for a priori probability, means and covariance for the m[th] component update of a target speaker model are summarised below,

Weight update: $\overline{w}_m = \frac{1}{T}\sum_{t=1}^{T} P(m|x_t, \lambda)$ **(2.18)**

Mean update: $\overline{\mu}_m = \frac{\sum_{t=1}^{T} P(m|x_t, \lambda) x_t}{\sum_{t=1}^{T} P(m|x_t, \lambda)}$ **(2.19)**

Covariance update: $\overline{\sum}_m = \frac{\sum_{t=1}^{T} P(m|x_t, \lambda) x_t^2}{\sum_{t=1}^{T} P(m|x_t, \lambda)}$ **(2.20)**

The posteriori probability for m[th] component can be evaluated as,

$$P(m|x_t, \lambda) = \frac{w_m p_m(x_t)}{\sum_{m=1}^{M} w_m p_m(x_t)} \quad \textbf{(2.21)}$$

Where $p_m(x)$ is given in Eq. (2.14).

Maximum a Posteriori (MAP) Estimation

The idea of the MAP estimation is applied to derive the optimized speaker model by updating the trained parameters of the prior model. ML estimates the probabilistic model

λ which maximizes the likelihood of training vectors x, referred as $P(x|\lambda)$, however MAP estimates the probabilistic model λ and maximizes the likelihood $P(x|\lambda)P(\lambda)$, where $P(\lambda)$ is the priori probability. Thus in case of MAP estimation the prior knowledge is also used for the EM updates to form a UBM.

Assuming that a GMM based UBM is created which provides the initial estimation of the parameters $\lambda = \{w_m, \mu_m, \Sigma_m\}_{m=1}^{M}$ and a set of feature vectors $X = \{x_1, x_2, ..., x_T\}$ is trained as a large GMM , the mth component update for the target speaker GMM of m Gaussian densities can be calculated as,

$$\overline{w}_m = \alpha_m \frac{1}{T}\sum_{t=1}^{T} P(m|x_t, \lambda) + (1-\alpha_m) w_m \quad \textbf{(2.22)}$$

$$\overline{\mu}_m = \alpha_m \frac{\sum_{t=1}^{T} P(m|x_t, \lambda) x_t}{\sum_{t=1}^{T} P(m|x_t, \lambda)} + (1-\alpha_m)\mu_m \quad \textbf{(2.23)}$$

$$\overline{\Sigma}_m = \alpha_m \frac{\sum_{t=1}^{T} P(m|x_t, \lambda) x_t^2}{\sum_{t=1}^{T} P(m|x_t, \lambda)} + (1-\alpha_m)(\Sigma_m + \mu_m^2) - \overline{\mu}_m^2 \quad \textbf{(2.24)}$$

Where α_m is a weight used to define the relative importance of the prior which is calculated by,

$$\alpha_m = \frac{\sum_{t=1}^{T} P(m|x_t, \lambda)}{\tau + \sum_{t=1}^{T} P(m|x_t, \lambda)} \quad \textbf{(2.25)}$$

Where τ is called constant relevance factor, it determines to what extent new data will affect the estimate of the updated GMM parameter.

Imposter Modeling

Imposter model can minimize non-speaker related variability by normalizing the likelihood ratio scores. Generally, there are two approaches to represent the imposter models.

Likelihood Sets (Background Sets)

It is a collection of other speaker models. For each speaker, a specific model is constructed using the models of all non-claimant speakers.

Universal Background Modeling (UBM)

It is a single speaker-independent model that is used by all speakers. In addition to smaller storage space required, it usually provides better performance.

The UBM introduced by Reynolds [19,39,40] was used where no enough training data was available for GMM training. It is a single large GMM trained from a pool of speakers; the speech data used in the training of a UBM is not used for the training of the individual speaker models. In other words the speech involved in the creation of UBM does not involve the utterances taken from the target speakers. ML estimation described above can also be used to estimate UBM parameters; however in this thesis MAP

estimation is used to evaluate the UBM parameters. The UBM can be used as the initial GMM for training target speaker dependent GMM. Below it is described how to train the target speaker model by using MAP adaptation.

Gaussian mixture model (GMM) classifier was for the first time applied to the speaker recognition task by Reynolds [19], since then GMM has been widely used in speaker modeling. The GMM needs sufficient data to model the speaker, to achieve good performance. The distribution of feature vectors is modeled by the parameters mean, covariance and weight.

GMM requires sufficient data to model the speaker well [19], to avoid this issue, Reynolds *et al.* introduced GMM-universal background model (UBM) for the speaker recognition task [39]. For UBM-GMM system, a substantial amount collected from the enrolled speakers is pooled and the UBM is trained, which acts as a speaker-independent model. The speaker-dependent model is then created from the UBM by performing maximum *a posteriori* (MAP) adaptation technique using speaker-specific training speech. As a result, the GMM-UBM gives better results than the GMM. The advantage of the UBM-based modelling technique is that it provides good performance even though the speaker-dependent data is small.

Gaussian Mixture Models (GMMs) have been widely used for speech modelling. GMMs can be termed as the state of the art modelling for text independent speaker verification technology. GMMs can be regarded as a specific case of a Radial Basis Function (RBF) [93,127] neural network. GMMs are defined in this section along with summary of maximum likelihood (ML) and maximum a posteriori (MAP) estimation methods. Log-likelihood ratio (LLR) test is also described to perform the verification decisions.

More details on the GMM theory can be found in Chapter 4.

2.5 Performance Evaluation and Comparison Methods for Speaker Recognition Task

The majority of the work reported in this thesis is focused on the speaker verification task. There are two types of possible errors in speaker verification: the false acceptance error also known as the false alarm probability and the false rejection error [7,50,143], also known as the miss probability.

A false acceptance (or false alarm) error occurs when the system accepts a claim of identity from an impostor speaker.

A false rejection (or miss probability) error occurs when the system rejects a legitimate speaker as an impostor.

2.5.1 The Detection Cost Function (DCF)

The performance of speaker verification system can be characterized using the false acceptance probability and the false rejection probability. A cost based performance measure C_{Det} can be calculated based on the false acceptance and the false rejection probabilities and used to evaluate the system performance. The NIST speaker recognition evaluation plans [139,140,141] defined the performance measure parameter C_{Det} as a weighted sum of the false acceptance and the false rejection error probabilities given as:

$$C_{Det} = C_{\text{FalseRejection}} P(\text{FalseRejection} \mid \text{Target}) P(\text{Target}) + \\ + C_{\text{FalseAcceptance}} P(\text{FalseAcceptance} \mid \text{NonTarget})(1 - \text{P}(\text{Target})) \quad \textbf{(2.26)}$$

Where $P(\text{FalseRejection} \mid \text{Target})$ is the probability that an actual target speaker was rejected, $P(\text{FalseAcceptance} \mid \text{NonTarget})$ is the probability that a non-target speaker was accepted.

The parameters $C_{FalseRejection}$ and $C_{FalseAcceptance}$ are the costs (or weights) of the false rejection and false acceptance errors respectively, and *P(Target)* is the *a priori* probability of the specified target speaker. Table 2.3 shows the values of $C_{FalseRejection}$, $C_{FalseAcceptance}$ and *P(Target)* recommended by the NIST speaker recognition evaluation rules for all speaker detection tests.

Table 2.3 Speaker Detection Cost Model Parameters.

$C_{FalseRejection}$	$C_{FalseAcceptance}$	$P(\text{Target})$
10	1	0.01

The cost value C_{Det} can be further improved by the following normalization:

$$C_{Norm} = C_{Det} / C_{Default} \quad \textbf{(2.27)}$$

where,

$$C_{Default} = \min\{C_{FalseRejection} P(\text{Target}), C_{FalseAcceptance} P(\text{NonTarget})\} \quad \textbf{(2.28)}$$

Where $P(\text{NonTarget})$ is the *a priori* probability of a non target speaker. $C_{Default}$ is called the optimal decision cost function (DCF).

There are two variants of the DCF, namely the actual DCF and the optimal decision cost function (ODCF). The actual DCF is defined as, the actual decisions that the specific

system have made, and depends on the choice of value for the speaker independent speaker verification threshold. The optimal decision cost function (ODCF) is defined as the minimal decision cost attained for the given experiment. The optimal DCF is an indication of the potential performance that a system could achieve, while the actual DCF gives the true measure of the system performance.

A major drawback of using the DCF measure is that it is not as sensitive to the changes in the system performance as the Equal Error Rate (EER) measure. When computing the EER, we assume equal weights for the cost parameters, $C_{False\,\mathrm{Re}\,jection} = C_{FalseAccep\,\mathrm{tan}\,ce} = 1$.

Since the decision in a speaker verification task is binary (accept or reject), a threshold of certainty may be included in the decision rule. A claim of identity is then accepted only when the decision can be made with a pre-determined level of certainty. By varying this threshold one can vary the ratio of false acceptance to false rejection errors [143].

In speaker verification system it is typically assumed that the ratio of likelihood of the claimant speaker model and the likelihood of the imposter speaker model should be greater than some threshold ζ. The threshold ζ measures how many times it was more likely that the claimant speaker spoke the test sample than any other speaker (or imposter). Thus, the value of ζ provides the certainty of the recognition decision.

The claim made by the speaker is accepted if:

$$\frac{P(\mathrm{FalseRejection} \mid T\,\mathrm{arg}\,et)}{P(\mathrm{FalseAcceptance} \mid NonT\,\mathrm{arg}\,et)} > \xi \qquad \textbf{(2.29)}$$

Since the division in Eq. (2.29) can lead to round off problems in numerical computations, Eq. (2.29) is usually replaced by the following logarithmic version:

$$\log\left(P(\text{FalseRejection} \mid T\,\text{arg}\,et)\right) - \log\left(P(\text{FalseAcceptance} \mid NonT\,\text{arg}\,et)\right) > \log(\xi) \quad \textbf{(2.30)}$$

In most cases, speaker verification systems are judged by the equal error rate (EER) parameter.

2.5.2 The Equal Error Rates (EER) and the Detection Error Trade-off (DET) Plots

The error rates for speaker recognition system were initially measured using receiver operating characteristic (ROC) curves [7]. However in the more recent studies of the speaker recognition systems, the nonlinear ROC curves are replaced by the Detection Error Trade-off (DET) plots [142], which are believed to provide more efficient representation of the system performance because of their linear behavior in the logarithmic coordinate system. In this thesis DET plots are used to evaluate the performance of speaker verification systems.

The DET plots are related to the equal error rate (EER) parameter representing a normalized measure of the system error rates.

The detection error trade-off (DET) plot is a curve representing the percentage of the false rejection probability as a function of the percentage of the false acceptance probability. An example of a DET plot is shown in Figure 2.7. Points on the DET curve correspond to the different values of the acceptance threshold ζ or different values of the ration given in Eq. (2.29).

As illustrated in Figure 2.7, the false rejection probability is an inverse proportion to the false acceptance probability. Which means that, by decreasing the false rejection probability the false acceptance probability will be increased and vice versa.

Since the ultimate goal of all speaker verification is to simultaneously minimize both errors (false rejection and false acceptance), the best compromise can be achieved when both errors are equal. The value of the percentage of the false rejection (or false acceptance) at the point when these two errors are equal is called the equal error rate (EER).

As illustrated in Figure 2.7, the equal error rate can be determined graphically as the percentage of false rejection (or false acceptance) at the intersection point between a 45^0 line (in red) and the detection error trade-off (DET) curve (in blue).

The smaller is the EER for a given speaker verification system, the better is the system performance.

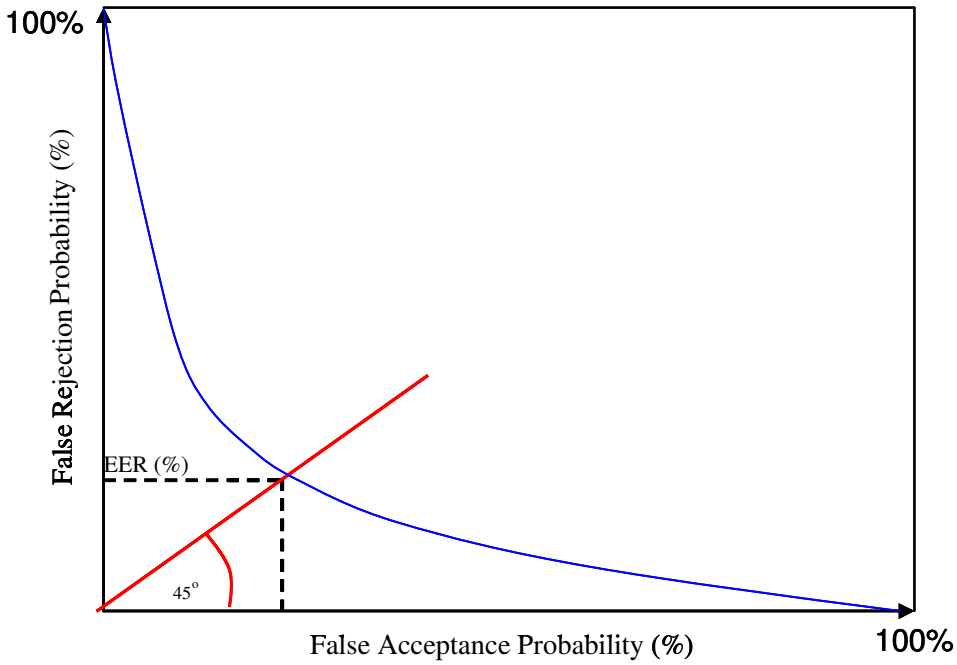


Figure 2.7 An example of the Detection Error Tradeoff (DET) curve (blue) and the process of determining the Equal Error Rates (EER).

The EER is of little practical significance since in most potential speaker verification systems a false acceptance error would be far more costly then a false rejection [143]. The EER is however, an effective technique for comparing the performance of different speaker recognition systems.

Since, different classification thresholds ζ may be applied by different applications; speaker verification systems typically use some type of score normalization techniques.

The score normalization is important in practical speaker verification systems, however since this study is primarily concerned with a closed set of speakers, and used the same classification rules across all tests, the score normalization was not used.

The work reported in this thesis belongs to speaker verification task and the EER has been adopted as the system performance measure in all cases.

2.6 Speech Corpora for Speaker Recognition Research

The speech corpora used in the speaker verification tests described in this thesis are: TIMIT, NIST 2001, NIST 2002 and NIST 2004. This section provides brief description of these corpora as well as few other speech corpora used for speaker recognition.

The selection of suitable speech corpora is of key importance in testing the performance of developed speaker recognition techniques. Ideally, the database used for performance evaluation should reflect environmental characteristics determined by possible applications.

Practical speaker recognition systems are typically used in non-ideal environments including acoustic noise and telephone line band limitations. In addition, most applications involve recognizing an individual at a later date then the date of the provided speech sample, therefore reliability over a long period of time is important.

Looking at the potential commercial applications of a speaker recognition system and in particular telephone-based speaker recognition, the following key requirements for speech corpora can be identified:

- Speech recorded over a telephone line [130] with the speaker in natural environment;
- The time duration of a single recording session should be at least 60 seconds.
- The data should be recorded for each speaker during a number of sessions spaced in time and covering a significant time interval (at least 1 year);

- The corpora should contain speech samples from a sufficiently large number of speakers;
- The corpora should contain speakers using the same language;
- The recording conditions should be well documented and the speech samples correctly labeled to avoid misuse of data.

The advantage of using publicly available corpora lies in the fact that, the collection of an appropriate data corpus takes a significant amount of time. In addition, the costs of recruiting and managing subjects to provide speech data over a long period of time can be significant. Existing speech databases have the additional advantage of enabling direct comparison of results between different studies using the same database. Publicly available data makes it also possible to reproduce reported research results.

Publicly available data comes from various commercial and academic sources and has been produced for a wide variety of applications and developed under different conditions.

Although, in the recent years the NIST database became the most frequently used corpora, other data sets are still being used as they can provide performance evaluation across different recording environments, populations of speakers and different languages.

The following list provides brief descriptions of selected corpora most often cited in the speaker recognition research.

TIMIT speech corpus and it's varaiants

The TIMIT speech corpus [132] consists of 630 speakers (438 male and 192 female). For each speaker only one recording session was used. The speech data was recorded in a

sound booth and contains fixed-text sentences read by speakers and recorded over a fixed wideband channel. The speakers used American English.

The main limitation of the TIMIT corpus is that the speech is recorded for only during one session per each speaker, therefore the data does not reflect time related variations in speech characteristics.

Moreover, the clean wideband speech environment in TIMIT has an ideal character and does not simulate the real world conditions appearing in typical speaker recognition applications.
A number of TIMIT variants also exist including:

- CTIMIT; a cellular bandwidth adjunct to the TIMIT corpus;
- HTIMIT; a re-recording of a subset of TIMIT corpus through different telephone handsets and
- NTIMIT; a telephone bandwidth adjunct to the TIMIT corpus.

SIVA speech corpus

SIVA is an Italian speech corpus [133] consisting of 840 speakers and has an even gender distribution. A small subset of only 40 speakers out of 840 had multiple recording sessions within time intervals ranging from 3 days to a few months. The speech was recorded over Public Switched Telephone Network (PSTN) channels in a home office acoustic environment. All speakers were fluent Italian speakers. The major drawback of this corpus is the lack of multiple sessions for the majority of the speakers.

POLYVAR Speech Corpus

This speech corpus [136] contains 143 speakers (85 male and 58 female). There are 3600 sessions in total recorded with speakers recorded during 1 to 229 sessions each with an intercession interval ranging from days to months. The speech samples include read digits, words and sentences, and spontaneous speech. The speech is recorded using different telephone handsets over PSTN channels at home office acoustic environment. The speakers use Swiss, French and other European languages.

POLYCOST Speech Corpus

This corpus consists of 133 speakers (74 male and 59 female) [134]. Each speaker provided more than 5 sentences with an intercession interval ranges from days to weeks. The speech samples include fixed and prompted digit strings, read sentences and free monologue. The recordings were made using variable telephone handsets over digital ISDN channels in a home office acoustic environment. The speakers used non-native English as well as various European languages.

KING Speech Corpus

The King corpus consists of 51 male speakers; each speaker was recorded over 10 sessions providing speech data with intervals ranging from weeks to a month [137]. The speech was recorded using a wideband microphone and an electret handset over clean and PSTN channels. It was recorded in a sound booth.

YOHO Speech Corpus

The YOHO corpus consists of 138 speakers (106 male and 32 female) [135]. Each speaker provided data for 4 enrollment sessions and 10 verification sessions with intercession intervals ranging from days to months. The speech samples included prompted digits and phrases and were recorded over clean 3.8 kHz channels in an office acoustic environment. All speakers used American English.

SWITCHBOARD Speech Corpus

The Switchboard corpus is an extensive data set, frequently used in speaker recognition tasks [138]. A number of subsets of the Switchboard corpus have been also used as the speaker recognition benchmark sets by the speech group at National Institute of Standards and Technology (NIST).

Switchboard I consists of 543 speakers and Switchboard II consists of 657 speakers, both corpora have approximately even gender distributions. Each speaker was recorded over 1 to 25 sessions with intercession intervals ranging from days to weeks. The corpus consists of conversational telephone speech using different telephone handsets over a PSTN channel. The speech was recorded within a home office acoustic environment. All speakers used American English from different regions of U.S.A.

NIST 2001 SRE Speech Corpus

The "*one-speaker detection*" corpus known as the NIST 2001 Speaker Recognition and Evaluation (SRE) corpus is a subset of the Switchboard-Cellular corpora, post-processed to remove any significant silence intervals and cancel transmission channel echoes contained by speech signal.

The NIST 2001 contains spontaneous speech from 174 speakers (74 male and 100 female) speakers recorded in different environmental conditions. For each speaker approximately 2 minutes of speech is available. The enrollment and test data consist of speech recorded over TDMA, CDMA, Cellular, GSM, and land transmission channels, thus different handsets and different transmission channels are available for each speaker. All speakers use American English. For each speaker approximately 2 minutes of speech is available for training for the "*one-speaker detection task*". The test trials are divided based on the length of speech segments, 0-15sec, 16sec-25sec, 26sec-35sec, 36sec-45sec, and 46sec-60sec are the available length of test segments. The complete "*one-speaker detection task*" including description of the evaluation database and evaluation rules is described in the 2001 NIST SRE Plan [139].

NIST 2002 SRE Speech Corpus

The NIST 2002 speech corpus consists of spontaneous speech from 330 speakers (139 male and 191 female) recorded in different environmental conditions. It consists of conversational speech recorded over a telephone line, from a microphone and from the news broadcast.

For each speaker approximately 3 minutes of speech are available for training for the speaker detection task. The test data for each speaker consists of speech segments of the total length of 3 minutes. A comprehensive description of the evaluation database and evaluation rules is available in the 2002 NIST SRE Plan [140].

NIST 2004 SRE Speech Corpus

The NIST 2004 speech corpus consists of 616 speakers (248 male and 368 female) recorded in different environmental conditions. It consists of conversational speech

recorded mostly over a telephone line. For each speaker approximately 5 minutes of speech is available for training as well as for testing. Most of the training data is in American English, but some training conversations involving bi-lingual speakers may be collected in Arabic, Mandarin, Russian, and Spanish. A comprehensive description of the evaluation database and evaluation rules is available in the 2004 NIST SRE Plan [141].

NIST Post-2004 Speech Corpora

The National Institute of Standards and Technology (NIST) has been coordinating Speaker Recognition Evaluations since 1996 [269]. The goal of the NIST Speaker Recognition Evaluation (SRE) series is to contribute to the direction of research efforts and the calibration of technical capabilities of text independent speaker recognition. The evaluation plans post-2004 are similar to the NIST 2004 corpora. Details of the new releases of NIST corpora can be found on: http://www.nist.gov/index.html.
The speech corpora used in the speaker verification tests described in this thesis are: TIMIT, NIST 2001, NIST 2002 and NIST 2004.

CHAPTER 3

SPEAKER VERIFICATION BASED ON THE INFORMATION THEORETIC VECTOR QUANTIZATION

This chapter investigates the Vector Quantization (VQ) approach to the speaker modeling for the speaker verification task. A relatively new vector quantization method based on the Information Theoretic principles (ITVQ) is for the first time used in the task of speaker verification and compared with two classical VQ approaches: the K-means algorithm and the Linde-Buzo-Gray (LBG) algorithm. The chapter provides a brief theoretical background of the vector quantization techniques, which is followed by experimental results illustrating their performance. The results demonstrated that the ITVQ provided the best performance in terms of classification rates, equal error rates (EER) and the mean squared error (MSE) compare to K-means and the LBG algorithms. The outstanding performance of the ITVQ algorithm can be attributed to the fact that the Information Theoretic (IT) criteria used by this algorithm provide superior matching between distribution of the original data vectors and the codewords.

3.1 Overview

3.1.1 Vector Quantization

The Vector Quantization (VQ) method is a classical signal processing technique which models the probability density functions by the distributions of prototype vectors. The VQ was originally designed to be used as a data compression technique where a large set of points (vectors) in a multidimensional space could be replaced by a smaller set of representative points with distribution matching the distribution of the original data.

A typical VQ algorithm divides a large set of vectors into clusters having number of points. Each cluster is represented by its central point. According to the Shannon's rate distortion theory [79], the central points for each cluster should be calculated as centers of gravity (or centroids); and the cluster members should be ideally selected such that, for each cluster member, the cluster centroid is the nearest centroid.

The Vector Quantization techniques have been widely adapted as a speaker modeling technique in speaker recognition /verification tasks [87,116].

A VQ technique encompasses two fundamental tasks:

1. An encoding process which involves a nearest neighbor (NN) search, assigning the closed codeword to a given vector.
2. A codebook generation process which finds an optimal, small set of vectors (codebook) representing a given large set of vectors. The elements of codebook are called the codewords.

At the simplest level, the task of nearest neighbor search can be performed using a linear search, although this approach becomes highly inefficient when a large number of highly dimensional data vectors needs to be repeatedly searched in applications like speaker verification/recognition. Many fast encoding algorithms have been proposed including various tree-search techniques [61].

The second VQ task of codebook generation is a complex multidimensional global optimization problem. For deterministic applications such as symbol identification in communication systems, the codebook is already defined by a given set of symbols being used. For non-deterministic applications such as data compression or speaker recognition, the VQ codebook has to be estimated using a data-driven procedure. The process of estimating the VQ codebook involves division of the observed data into clusters. The

centroid of each cluster becomes the codeword representing that cluster. The set of all centroids constitutes the VQ codebook. If the Cartesian distance measure is used, then the centroid simply represents the mean vector calculated from all vectors belonging to the given cluster.

The best known and very efficient VQ codebook generation algorithm used in speaker verification/recognition tasks include: the K-means algorithm [58], the Linde Buzo Gray (LBG) algorithm [59], and the Kohonen's self-organizing map (KSOM) [95]. In these algorithms the process of finding an optimal codebook is guided by minimization of the average distortion function (objective or cost function) representing an average total sum of distances between the original vectors and the codewords. It is also called the quantization error. Different types of distance measures for the quantization error have been proposed in literature [79].

The VQ codebook generation is a large scale global optimization problem, however the vast complexity of this problem means that in reality only sub-optimal solutions can be found. Codebook generation algorithms differ in the way that some algorithms are less and some more powerful in finding acceptable local minima of the objective function.

An ideal codebook should contain a set of uncorrelated (linearly independent) centroid vectors. In reality there is always remaining a certain amount of correlation between centroids.

3.1.2 Information Theoretic Learning

Some of the most recent trends in vector quantization include VQ based on the Information Theoretic Learning [80].

The idea of the Information Theoretic Learning (ITL) was conceived in the late 90's at the Computational Neuro-Engineering Lab (CNEL), University of Florida.

The ITL uses descriptors from information theory (entropy and divergences) estimated directly from the data to substitute the conventional statistical descriptors of variance and covariance.

Applications of ITL include vector quantization, adaptation of linear or nonlinear filters as well as different unsupervised and supervised machine learning approaches [56].

3.1.3 VQ in speaker recognition and verification

Vector Quantization may be used as a classification process in a number of ways [116]. The most often used approach is to generate a separate codebook for each speaker using speech recordings that belong to that speaker. During the testing phase the set X_{ID} of observed feature vectors from the unknown speaker are compared with codebooks representing the reference speakers. This process is graphically illustrated in Figure 3.1.

The quantization errors for the observed feature vectors of the unknown speaker when quantized using each of the reference codebooks are used as a measure of how close the observed feature vectors are to codewords representing each speaker. The speaker whose codebook is the closest to the observed feature vectors is then taken as the identified speaker.

In speaker verification task, an arbitrary threshold is often applied to the quantization error to determine if the observed feature vectors are close enough to the codebook for the claimant speaker to accept the claim.

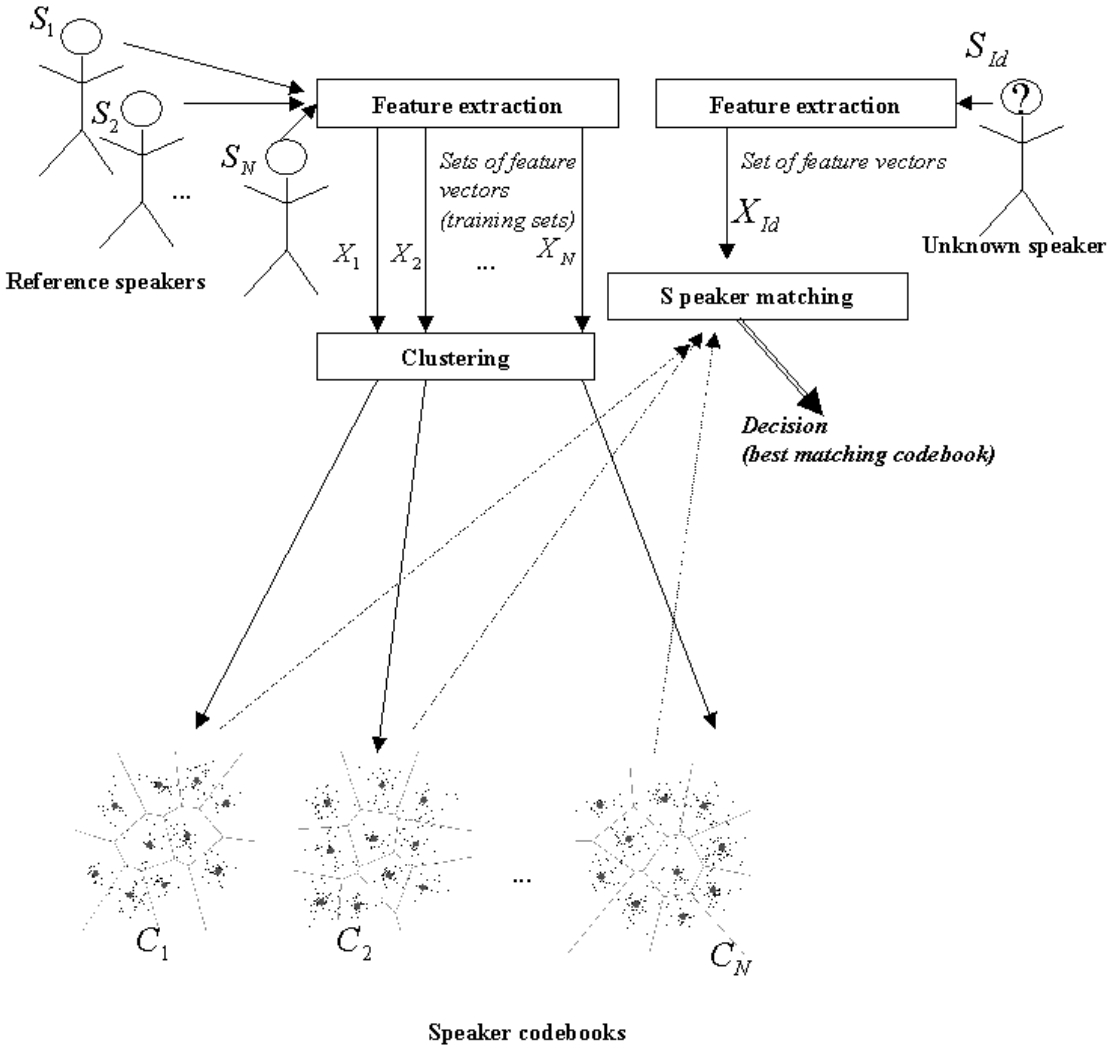


Figure 3.1 Structure of the VQ based speaker recognition system; adapted from [73].

3.1.4 Relationship between VQ and GMM

In Chapter 4, a new classification approach which combines the classical Gaussian Mixture Model with the Information Theoretic Vector Quantization (ITVQ) is introduced; therefore few comments are made here about the relationship between VQ and GMM classifiers.

The GMM and VQ techniques are closely related [57]. The GMM is often interpreted as a soft form of vector quantization [24,115]. In VQ incoming feature vectors are assigned to one of the codewords in the codebook.

Assuming that the observed vector **Y** is assigned to codeword i, then the quantization error E can be calculated as a distance measure D between the observed vector **Y** and the assigned codeword vector $\mathbf{V}_i$. It can be denoted as:

$$E = D(\mathbf{Y}, \mathbf{V}_i) \quad \textbf{(3.1)}$$

In VQ only the distance between the observed vector and the closest codeword is considered [24].

In the case of the GMM approach one can consider each Gaussian component to be a codeword in a VQ codebook. The Gaussian components are given as the *posteriori* probabilities:

$$p(\mathbf{Y} \mid \Lambda) = \sum_{i=1}^{M} p_i \, b_i \, (Y_i \, ; \Lambda) \quad \textbf{(3.2)}$$

where p_i are the *a priori* probabilities and b_i are the Gaussian component probabilities.

The *a priori* component probability p_i in Eq. (3.2) can be treated as probability of belonging to the i-th Gaussian component. Each of the multivariate Gaussian components b_i can be interpreted as a measure of distance between the observed vector **Y** and the i-th codeword $\boldsymbol{\mu}_i$.

For an appropriately chosen distance function D in Eq. (3.1) the quantization error E is in the inverse relationship to the Gaussian component probability if the mean vector for the codeword and Gaussian component are equal; that is $\boldsymbol{\mu}_i = \mathbf{V}_i$.

If all of the *a priori* probabilities in the GMM approach are equal and it is assumed that only the closest Gaussian component is significant, then the GMM and VQ approaches become identical. The GMM can therefore be interpreted as a "soft" form of VQ where membership to all the codewords is considered in a weighted form based on the *a priori* probability of a feature vector belonging to a given codeword [115].

The above interpretation can be used to reduce computational costs of the GMM based speaker verification /recognition systems. If the contributions of only the first few most significant Gaussian components are considered then there is no need to calculate the contributions of the remaining components since they can be assumed negligible [24].

The following sections describe the most frequently used VQ algorithms: K-means and the Linde-Buzo-Gray (LBG) algorithm. This is followed by sections describing the Information Theoretic Vector Quantization (ITVQ).

3.2 K-means modeling algorithm

The K-means algorithm [58,105] is a clustering algorithm used for the vector quantization codebook generation. It clusters data based on attributes or features into K groups where, K is a positive integer. The clustering is achieved by minimizing the squared Euclidean distance between vectors $\mathbf{x}_i$ and the corresponding cluster centroid vector $\boldsymbol{\theta}_j$. The centroid vector represents each cluster as a mean vector of the cluster.

Lets assume that a set of T vectors $\mathbf{X}=\{\mathbf{x}_1,\mathbf{x}_2,\mathbf{x}_3,\ldots,\mathbf{x}_T\}$ is to be divided into K clusters represented by their mean vectors $\boldsymbol{\theta}=\{\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \boldsymbol{\theta}_3,\ldots, \boldsymbol{\theta}_K\}$. The objective of the K-means algorithm is to minimize the total distortion (or quantization error) given by

$$D = \sum_{i=1}^{T}\sum_{j=1}^{K} \| \mathbf{x}_i - \boldsymbol{\theta}_j \|^2 \qquad \textbf{(3.3)}$$

K-means is an iterative approach, in each successive iteration; it redistributes the vectors in order to minimize the distortion *D* (quantization error).

The K-means algorithm consists of the following basis steps:

Step 1. Choose arbitrary initial estimates $\boldsymbol{\theta}_j(0)$ for the centroid vectors $\boldsymbol{\theta}_j$' *s, j=1,2,….,K.* Calculate the initial value of the distortion *D(0).*

Step 2.

For *i*=1 to T

For a vector $\mathbf{x}_i$, determine the nearest centroid , say $\boldsymbol{\theta}_j$,

Set *centroid(i)=j* (centroid or cluster for the jth vector)

End

For *i*=1 to K

Calculate new centroids $\boldsymbol{\theta}_j$ as the mean of the vectors $\mathbf{x}_i \, \epsilon \, \mathbf{X}$ with *centroid(i)=j.*

Calculate the distortion value *D(i).*

End

Step 3 Repeat Step 2 until either a maximum number of iterations is reached or the distortion value *D(i)* falls below a preset threshold or until no change in $\boldsymbol{\theta}_j$' *s* occurs between a few successive iterations.

The above procedure iteratively moves the cluster boundaries. When the distortion D is minimized, subsequent iterations do not result in any movement of vectors between clusters and the cluster boundaries become stabilized. This could be used as one of possible indicators to terminate the algorithm. The total distortion can also be used as an indicator of convergence of the algorithm. Upon convergence, the total distortion does not change as a result of redistribution. A great advantage of this algorithm is its computational simplicity. An example of K-means procedure is illustrated in Figure 3.2.

In the case of speaker recognition the speech files are preprocessed and a set of feature vectors is calculated. The K-means clustering can be then used to group feature vectors for each speaker into K sets (clusters) which efficiently describe the acoustic attributes of a given speaker. Thus, each speaker is modeled by a set of K clusters of feature vectors.

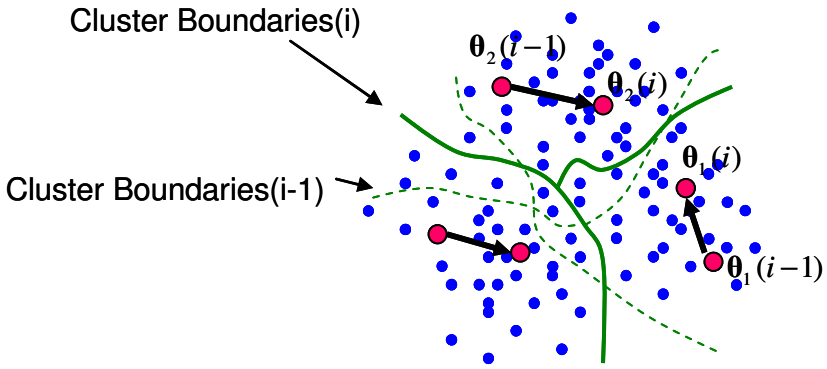


Figure 3.2 An example of the K-means clustering for 3 clusters; the blue dots represent data vectors, i is the iteration number and $\boldsymbol{\theta}_j$ denote centroid vectors (red dots). The green lines represent boundaries between clusters.

There is no general theoretical solution to find the optimal number of clusters for any given data set. A simple approach is to compare the results of multiple runs with different numbers of classes (different values of K) and choose the best one according to a given

criterion; however it needs to be measured carefully because increasing of K, results not only in smaller value of the distortion but also it increases the risk of overfitting. More details on advantages and drawbacks of K-means can be found in [9].

The drawback of overfitting can be largely eliminated by using the Information Theoretic based Vector Quantization which works on the principle of physical interpretation of the data clusters.

3.3 Linde-Buzo-Gray (LBG) clustering algorithm

The LBG algorithm [59] is an enhanced version of the K-means clustering. It consists of a sequence of iterative steps minimizing the distortion measure. The algorithm consists of two phases:

1. Codebook initialization phase;
2. Codebook optimization phase.

The codebook optimization process is guided by minimization of the average distortion of the maximum quantization error (MQE) given as:

$$MQE = D(\mathbf{X}, \mathbf{Y}, q) = \frac{1}{N} \sum_{i=1}^{N} d(\mathbf{x}_i, q(\mathbf{x}_i)) \qquad \textbf{(3.4)}$$

where Y is a given codebook, X={$\mathbf{x}_1$, $\mathbf{x}_2$, ..., $\mathbf{x}_N$} is the set of observation data vectors, N is the total number of observation data vectors, d is a vector distance measure, q is the vector quantizer function, defined such that q($\mathbf{x}_i$) is the codeword assigned to vector $\mathbf{x}_i$ based on the nearest neighbor criterion.

The LBG algorithm requires the user to provide an initial estimate of the codebook and to specify the desired number of clusters. Due to the nature of the classical LBG algorithm, which usually generates the initial codebook by randomly splitting codewords into two new codewords, the desired number of clusters needs to be a power of 2. The following sections describe the subsequent phases of the LBG algorithm.

3.3.1 Codebook initialization phase

The choice of initial codebook can be critical for the quality of the final solution. The poor choice of the initial codebook will lead to a final quantizer with a relatively large value of the quantization error.

A number of methods such as random initialization [107], initialization by splitting [59] and maximum distance initialization [108] have been proposed to perform codebook initialization. One of the most often used approaches is based on random splitting codewords until a desired codebook size is reached.

As illustrated in Figure 3.3, the process of generating an initial codebook starts with a single random initial codeword. The single codeword is then randomly split into two codewords by a small random perturbation. The procedure proceeds until a pre-set number of codewords is reached. This type of codebook initialization results in a codebook size which is a power of 2.

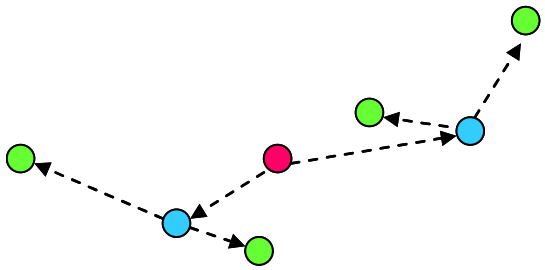


Figure 3.3 Initial codebook generation by randomly splitting the codewords. Red dot-represents the first codeword at iteration 0, blue dots-iteration 1, green dots-iteration 2, etc.

3.3.2 Codebook optimization phase

The initialization step is followed by the iterative codebook optimization procedure which gradually improves the codebook estimate by minimization of the total distortion (quantization error) D given in Eq. (3.1).

The optimization phase of the LBG algorithm proceeds as follows [10]:

Step 1: Assign the initial codebook as the current codebook Y^k and the current iteration number k=1.

Step 2: Using the current vector quantizer q^k, divide the training data into a set of nearest neighbour (NN) clusters (also called the Voronoi clusters [10]). Then calculate the average distortion $D(Y^k,q^k)$ using Eq. (3.4).
If $abs(D(Y^k,q^k)- D(Y^{k-p},q^{k-p}))$ is less than a preset threshold ζ, then terminate the algorithm.
else, go to *Step 3*. The p value is a control step denoting a mall number of iterations.

Step 3: Set k=k+1, and update the codebook Y^k by calculating the centroids of the new clusters, update the nearest neighbour quantizer q^k and go to *Step 2*.

The cycle of iterations usually continues until the decrement in average distortion value calculated over a specific small number of iterations falls below a pre-set threshold ζ. Alternatively the algorithm can be terminated when a pre-set maximum of iterations is reached.

The LBG algorithm offers a constructive solution to a very complex problem of generating an optimal VQ codebook. The great advantage of the LBG is that it does not require knowledge about the underlying statistics of the observation data.

However, the quality of the final solutions depends on the quality of the initial codebook. The procedure has a gradient descent character and has no mechanisms allowing escaping from local minima, therefore the algorithm has a tendency to end up in low quality local minima. Moreover, computationally, the LBG algorithm is highly demanding.

3.4 Information theoretic based vector quantization (ITVQ)

In K-means and LBG algorithms, data points are associated with the nearest code vector reducing the size of the original data. The challenge is to find the set of code vectors (the codebook) that reduces the data to a smaller set preserving the distribution of the original data vectors.

Unlike the K-means or LBG, the Information Theoretic Vector Quantization (ITVQ) [80] has a clear physical interpretation and relies on minimization of a well defined cost function.

The ITVQ uses descriptors from information theory (entropy and divergences) estimated directly from the data to substitute the conventional statistical descriptors of variance and covariance. The ITVQ is based on a number of core concepts of the information theory such as Parzen density estimator, Kullback Leibler divergence, Cauchy Schwartz Inequality and Renyi's Quadratic Entropy [80].

In the light of the information theory minimization, the free distance between the codeword's distribution and the original data distribution is equivalent to the minimization of the divergence measure between these two distributions. The divergence measure is calculated directly from the data using the Parzen density estimator.

The divergence minimization algorithm can be also seen as a probability density matching method, where the distance between the Parzen density estimator for the codewords and the Parzen density estimator for the original data is minimized.

The potential field created by a single vector (particle) can be described by a kernel of the form $\mathbf{K}(\cdot)$. Placing a kernel on each particle, the potential energy at a point in space $\mathbf{x}$ is given by:

$$p(\mathbf{x}) = \frac{1}{N}\sum_{i=1}^{N}\mathbf{K}(\mathbf{x}-\mathbf{x}_i) \qquad \textbf{(3.5)}$$

Where $\mathbf{x}_i$ are the data vectors. Eq. (3.5) is known as the Parzen density estimator [4].

In order to match the distribution of the codewords with the distribution of the original data, Eq. (3.5) can be used to estimate their densities and then minimize the divergence between the densities.

The distribution of the data points ($\mathbf{x}_i$) can be written as:

$$f(\mathbf{x}) = \sum_i G(\mathbf{x} - \mathbf{x}_i, \sigma_f) \quad \textbf{(3.6)}$$

Similarly, the distribution over codewords ($\mathbf{w}_i$) can be written as:

$$g(\mathbf{x}) = \sum_i G(\mathbf{x} - \mathbf{w}_i, \sigma_g) \quad \textbf{(3.7)}$$

Where, G(.) represents the Gaussian kernel given as

$$G(\mathbf{x}, \sigma) = \frac{1}{\left(\sqrt{2\pi}\sigma\right)^N} e^{-\frac{|\mathbf{x}|^2}{2\sigma^2}} \quad \textbf{(3.8)}$$

Numerous divergence measures exist, of which the Kullback-Leibler (K-L) divergence is the most commonly used [5]. The Integrated square error and the Cauchy-Schwartz (C-S) inequality, are both linear approximations to the K-L divergence

The Kullback-Leibler (K-L) divergence represents a measure of the difference between two probability distributions: from a true probability distribution X to an arbitrary probability distribution Y. Typically X represents data, observations, or a precise calculated probability distribution. The measure Y typically represents a theory, a model, a description or an approximation of X.

For probability distributions X and Y of a discrete random variable the K–L divergence of Y from X is defined as,

$$D_{KL}(X \| Y) = \sum_i X(i) \log \frac{X(i)}{Y(i)} \quad \textbf{(3.9)}$$

The Cauchy–Schwarz (C-S) inequality is a linear approximation of the K–L divergence. For vectors **x** and **y**, the inequality is written as,

$$\left|\langle \mathbf{x}, \mathbf{y} \rangle\right| \le \|\mathbf{x}\| \cdot \|\mathbf{y}\| \tag{3.10}$$

Substituting Eq. (3.8) to Eq. (3.6) and Eq. (3.7), the distribution $f(\mathbf{x})$ of the data points $\boldsymbol{x_i}$ is given as:

$$f(\mathbf{x}) = \sum_i G(\mathbf{x} - x_i, {\sigma_f}^2) = \frac{1}{\left(\sqrt{2\pi}\sigma_f\right)^N} e^{-\frac{|\mathbf{x}-x_i|^2}{2{\sigma_f}^2}} \tag{3.11}$$

and the distribution $g(\mathbf{x})$ of the codevectors $\mathbf{c_j}$. is given as:

$$g(\mathbf{x}) = \sum_j G(\mathbf{x} - \mathbf{c}_j, {\sigma_g}^2) = \frac{1}{\left(\sqrt{2\pi}\sigma_g\right)^M} e^{-\frac{|\mathbf{x}-\mathbf{c}_j|^2}{2{\sigma_g}^2}} \tag{3.12}$$

Applying the Cauchy-Schwartz (C-S) inequality of Eq. (3.10) to $f(\mathbf{x})$ and $g(\mathbf{x})$, we have,

$$\left|\langle f(\mathbf{x}), g(\mathbf{x}) \rangle\right| \le \|f(\mathbf{x})\| \cdot \|g(\mathbf{x})\| \tag{3.13}$$

Eq. (3.13) represents an equality only when $f(\mathbf{x})$ and $g(\mathbf{x})$ are collinear. Hence, maximizing the ratio between the numerator $\left|\langle f(\mathbf{x}), g(\mathbf{x}) \rangle\right|$ and the denominator $\|f(\mathbf{x})\| \cdot \|g(\mathbf{x})\|$ is equivalent to minimizing the divergence between $f(\mathbf{x})$ and $g(\mathbf{x})$.

To avoid the division, the logarithm can be maximized instead. This is valid since the logarithm is a monotonically increasing function. In order to minimize the divergence between the distributions *f(***x***)* and *g(***x***)* the following expression is minimized,

$$D_{C-S}\left(f(\mathbf{x}), g(\mathbf{x})\right) = -\log \frac{\left(\int (f(\mathbf{x})g(\mathbf{x}))dx\right)^2}{\int f^2(\mathbf{x})dx \int g^2(\mathbf{x})dx} =$$

$$= \log \int f^2(\mathbf{x})dx - 2\log \int f(\mathbf{x})g(\mathbf{x})dx + \int g^2(\mathbf{x})dx \qquad \textbf{(3.14)}$$

In Eq. (3.14) the first term contains the information about the interactions between the data points. The second term addresses the interaction between the data points x_i and code vectors $\mathbf{c_j}$. However, the third term is containing the information about the interactions between the code vectors itself. The interaction between the data points would not lead to an improvement, however the interactions between the data points and code vectors and between the code vectors would lead to improvement. This is because the position of data points is fixed and the only random position selection and change is associated with code vectors. Therefore first term in Eq.(3.14) can be ignored. The cost function with respect to code vectors can therefore be written as,

$$J(\mathbf{c}) = -2\log \int f(\mathbf{x})g(\mathbf{x})dx + \int g^2(\mathbf{x})dx \qquad \textbf{(3.15)}$$

The cost function *J(***c***)* is minimized with respect to the location of the code vectors $\mathbf{c_j}$. When the codevectors are located such that the local minima is achieved, no effective force acts on the code vectors. Moving the code vectors in the opposite direction of the gradient will bring them to such a potential minimum. This is also known as the gradient descent method. The gradient descent method states that the derivative of Eq. (3.15) with respect to the location of the codevectors must be calculated. For the sake of simplicity

the Eq. (3.15) is divided into two parts. The first part is denoted by **C** and the second part is denoted by **V**.

Considering first term of Eq. (3.15),

$$\mathbf{C} = \int f(\mathbf{x}) g(\mathbf{x}) dx$$
$$= \frac{1}{MN} \int \sum_{i}^{N} G_f(\mathbf{x} - x_i, {\sigma_f}^2) \sum_{j}^{M} G(\mathbf{x} - \mathbf{c}_j, {\sigma_g}^2) dx \quad \textbf{(3.16)}$$

Where the covariance of the Gaussian after integration is ${\sigma_a}^2 = {\sigma_f}^2 + {\sigma_g}^2$. M is the number of code vector kernels and N is the number of data point kernels.

The gradient update for the code vectors $\mathbf{c_j}$ from the above term then becomes,

$$\frac{d}{d\mathbf{c}_j} 2 \log \mathbf{C} = -2 \frac{\mathbf{\Delta C}}{\mathbf{C}} \quad \textbf{(3.17)}$$

Where $\mathbf{\Delta C}$ denotes the derivative of **C** w.r.t code vectors, it is calculated as,

$$\mathbf{\Delta C} = -\frac{1}{MN} \sum_{i}^{N} G_f(\mathbf{c}_j - \mathbf{x}_i, \sigma_f) \sigma_f^{-1} (\mathbf{c}_j - \mathbf{x}_i) \quad \textbf{(3.18)}$$

Similarly for the second term *V* we have,

$$\mathbf{V} = \int g^2(\mathbf{x}) dx$$
$$= \frac{1}{M^2} \sum_{j}^{M} \sum_{k}^{M} G(\mathbf{c}_j - \mathbf{c}_k, \sqrt{2}\sigma_g) \quad \textbf{(3.19)}$$

The gradient update for the code vectors $\mathbf{c_j}$ from the second term then becomes,

$$\frac{d}{d\mathbf{c}_j}\log \mathbf{V} = \frac{\mathbf{\Delta V}}{\mathbf{V}} \tag{3.20}$$

Where $\Delta\mathbf{V}$ denotes the derivative of $\mathbf{V}$ w.r.t code vectors, and it is calculated as,

$$\mathbf{\Delta V} = -\frac{1}{M^2}\sum_{j}^{M} G\left(\mathbf{c}_k - \mathbf{c}_j, \sqrt{2}\sigma_b\right)\sigma_b^{-1}\left(\mathbf{c}_k - \mathbf{c}_j\right) \tag{3.21}$$

Where k denotes the current centroid for which the update is obtained. By substituting the simplification of the above two terms obtained in Eq. (3.18) and Eq. (3.21) to Eq. (3.15), the update formula for the ITVQ can be established as,

$$\mathbf{c}_k(n+1) = \mathbf{c_k}(n) - \eta\left(\frac{\mathbf{\Delta V}}{\mathbf{V}} - 2\frac{\mathbf{\Delta C}}{\mathbf{C}}\right) \tag{3.22}$$

Where η is the step size, the ITVQ consists of n updates for each of the codevector $\mathbf{c_k}$.

3.5 Experiments Comparing Speaker Verification based on ITVQ, K-means and LBG Modeling Techniques

3.5.1 Overview of the Speaker Verification System

The speaker verification tests were performed using a general approach illustrated in Figure 3.4 including training and testing phases. This approach was described in detail in Chapter 2.

Mel Frequency Cepstral Coefficients (MFCC) features are used to perform the speaker verification tests. The aim was to compare the performance of two classical Vector Quantization algorithms K-means and LBG with the Information Theoretic vector Quantization (ITVQ).

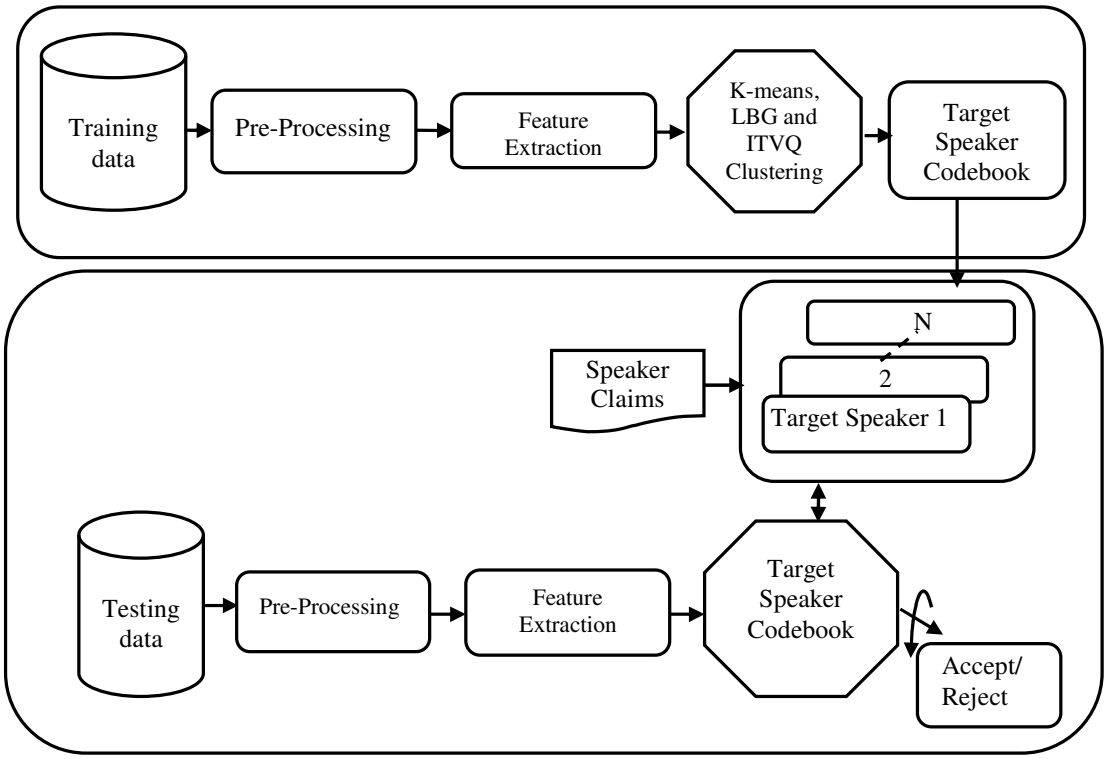


Figure 3.4 Block diagram of the speaker verification system [53].

The system in Figure 3.4 can operate in one of the two possible modes:

- The target speaker enrollment (training), and
- The testing mode.

For both of the system modes identical speech detection and feature extraction methods were used.

3.5.2 Speech Corpora

The speaker verification experiments were performed using two speech corpora: TIMIT and NIST 2004.

Details of these two speech corpora are given in Chapter 2, Section 2.6.

The TIMIT corpus was used to obtain speech samples of 630 speakers (438 male and 192 female). The recordings were made in a sound booth using fixed-text sentences read by speakers and recorded over a fixed wideband channel. The speakers used American English. The TIMIT corpora had a low environmental value since the clean wideband speech has an ideal character and does not simulate the real world conditions.

In order to provide a speech corpora that provides a better representation of the real life conditions, The NIST 2004 was used with 616 speakers (248 male and 368 female) recorded in different environmental conditions. The recordings include conversational speech recorded mostly over a telephone line. For each speaker approximately 5 minutes of speech was available for training as well as for testing. Most of the training data is in American English.

Table 3.1 shows a summary of TIMIT and NIST 2004 corpora used to perform the experiments.

Table 3.1 Properties of the speech corpora.

Description	*TIMIT*	*NIST 2004*
Language	English	English
Client speakers	630	616
Speech type	Read	conversational
Record condition	Lab	Telephone
Handset mismatch	No.	No.
Sampling rate	8KHz (down-sampled)	8KHz
Quantization	16 bit	8 bit μ-law
Train speech	45 sec	5 min
Test speech	12 sec	50 sec

3.5.3 Pre-processing and Feature Extraction

The pre-processing method followed the Speech Activity Detection procedure introduced by Reynolds in [144] described in Chapter 2. The voiced/silence interval were detected using an energy threshold.

For feature extraction 12 mel frequency cepstral Coefficients (MFCC) are used. As illustrated in Figure 3.5, the MFCC parameters were calculated by mapping the voiced speech spectrum into mel frequency scale. This mel frequency mapping was done by

multiplying the magnitude of speech spectrum for a preprocessed frame by magnitude of triangular filters in mel filterbank followed by log-compression of sub-band energies of the mel-scale filters and finally DCT. More details on the MFCC can be found in Chapter 5.

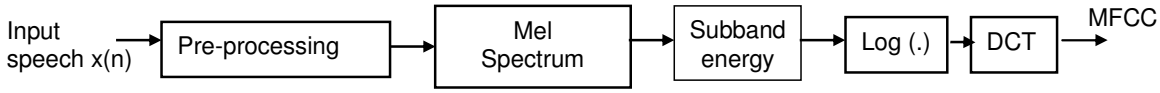


Figure 3.5 Calculation of the MFCC parameters.

As discussed in Chapter 2, the MFCC feature extracted from fixed length signal frames effectively capture the characteristics of the speakers. It was also reported that the MFCC performs well for the task of speaker verification if the frame size ranging from 20 ms to 50 ms, and the frame step ranging from 1/6 to 1/3 of the frame size is used to analyze the speech. Thus keeping in view these recommendations, the MFCC based feature extraction method was implemented on short-time signal (frame by frame basis) using frames of length 20ms with 10ms of overlap between adjacent frames.

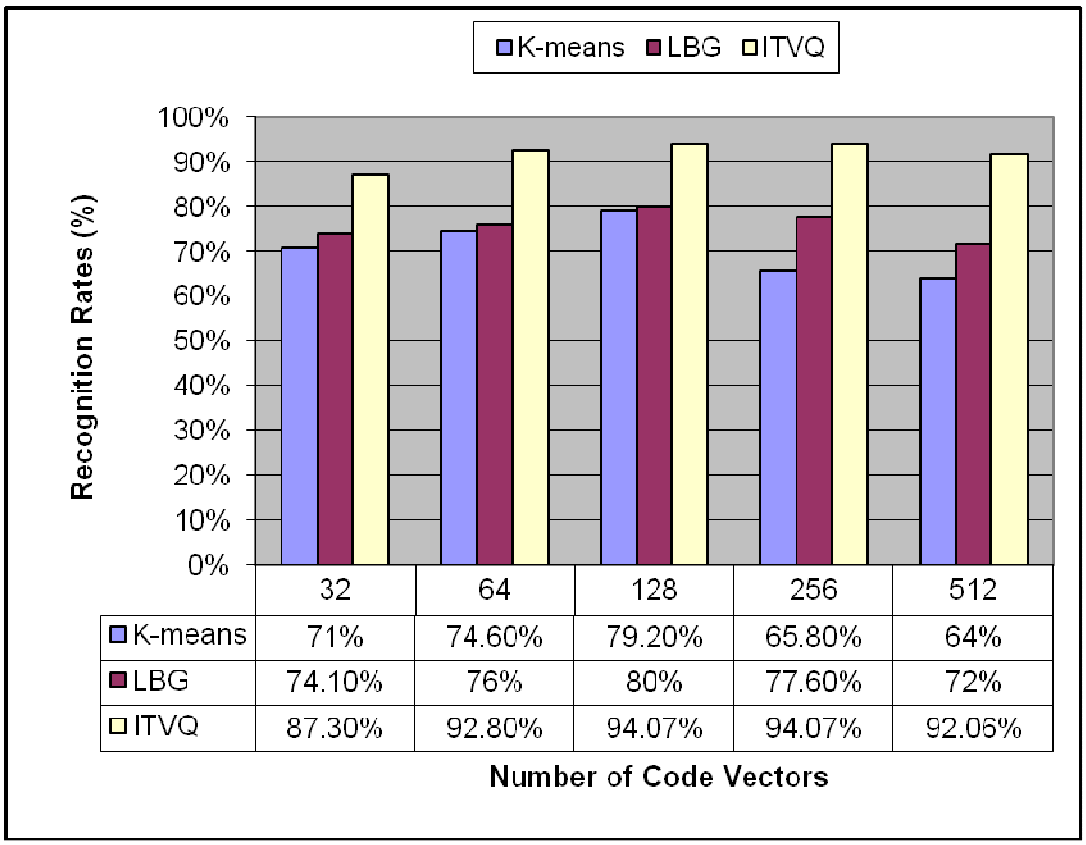
3.5.4 Speaker Verification Results

The performance of the VQ methods was evaluated using speaker recognition rates, EER values and the mean square error with respect to codebooks.

The speaker recognition rate is the most widely used measure to evaluate the performance of a speaker verification system. However the introduction of EER measure gives a more suitable tool for the evaluation of the performance of detection systems in general and speaker verification systems in particular. More details on calculation of EER can be found in Chapter 2, Section 2.5.

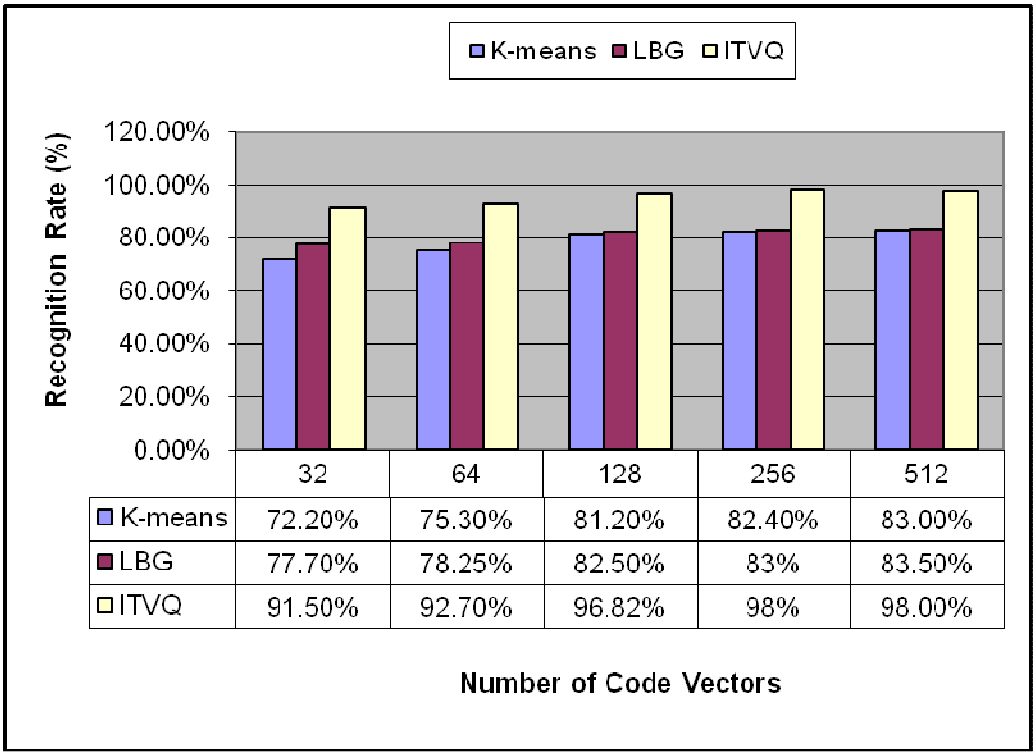
Speaker Recognition Rates

The speaker recognition rates for all three speaker modeling methods based on VQ are summarized in Figures 3.6 (a) and (b). Figure 3.6 (a) shows the results based on the TIMIT corpora and Figure 3.6 (b) shows the results based on the NIST 2004 corpora.



	32	64	128	256	512
K-means	71%	74.60%	79.20%	65.80%	64%
LBG	74.10%	76%	80%	77.60%	72%
ITVQ	87.30%	92.80%	94.07%	94.07%	92.06%

Figure 3.6(a) Recognition scores for K-means, LBG and ITVQ Classifiers for TIMIT Speech Corpora.



	32	64	128	256	512
K-means	72.20%	75.30%	81.20%	82.40%	83.00%
LBG	77.70%	78.25%	82.50%	83%	83.50%
ITVQ	91.50%	92.70%	96.82%	98%	98.00%

Figure 3.6(b) Recognition scores for K-means, LBG and ITVQ Classifiers for NIST'04 Speech Corpora.

It can be seen in Figures 3.6(a)&(b) that both corpora show the same general trend with ITVQ outperforming both the K-means and the LBG algorithm. The worse general performance in terms of the recognition rates was obtained for the K-means algorithm.

The recognition rates in Figure 3.6(a)&(b) also indicate that for all three algorithms, an increase of the number of clusters generally leads to a noticeable increase of recognition rates when the number of cluster increases from 32 to 128, further increase from 128 to 512 shows a small degradation in performance leading to slightly lower recognition rates.

The reason for the performance degradation is observed due to the increase in number of codewords can be attributed to thinner distribution of data. With the increasing number of

codewords the data is highly distributed and the codewords are therefore not capable of modeling a particular speaker accurately, which ultimately deteriorates the performance.

The relatively high recognition rates for the ITVQ indicate that Parzen density estimation provides better representation of the data distribution than the mean values used in K-means and LBG algorithms. The C-S divergence minimizes the free distance between the data points and the code vectors more efficiently than the K-means and LBG methods.

Equal Error Rates (EER)

The second measure used to compare the performance of speaker verification based on different VQ algorithms was the equal error rate (EER). The EER is the most widely used performance measure for speaker verification systems. Therefore the performance of the VQ based speaker verification system was also measured using the EER. Since the EER can only be calculated for a fixed number of codewords, a codebook containing 512 codewords was used to illustrate the performance comparison between K-means, LBG and ITVQ algorithms.

Figures 3.7(a)&(b) illustrate the percentage miss probability versus the percentage of false alarm probability and the EER values for the Kmeans, LBG and ITVQ methods using codebook size of 512. Figure 3.7(a) shows the results for the TIMIT corpora and Figure 3.7(b) shows the results for the NIST 2004 corpora.

The miss probability measures the percent of invalid matches and the false alarm probability measures the percent of valid inputs being rejected. The EER parameter represents the rate at which both the miss probability and the false alarm probability are equal. The lower the EER, the more accurate the system is considered.

As illustrated in Figure 3.7(a) and (b), both corpora show the same trend with ITVQ outperforming both K-means and LBG algorithm. The K-means algorithm provided the highest EER (34.9% for TIMIT and 21% for NIST 2004), LBG gave medium performance (27.8% for TIMIT and 19.1% for NIST 2004). Finally, the ITVQ provided the lowest EER (15.8% for TIMIT and 11.8% for NIST 2004).

The average improvement of EER value for ITVQ method is about 19.1% over K-means and 7.1% over LBG for TIMIT corpus and 9.2% over K-means and 7.3% over LBG for NIST 2004 corpus.

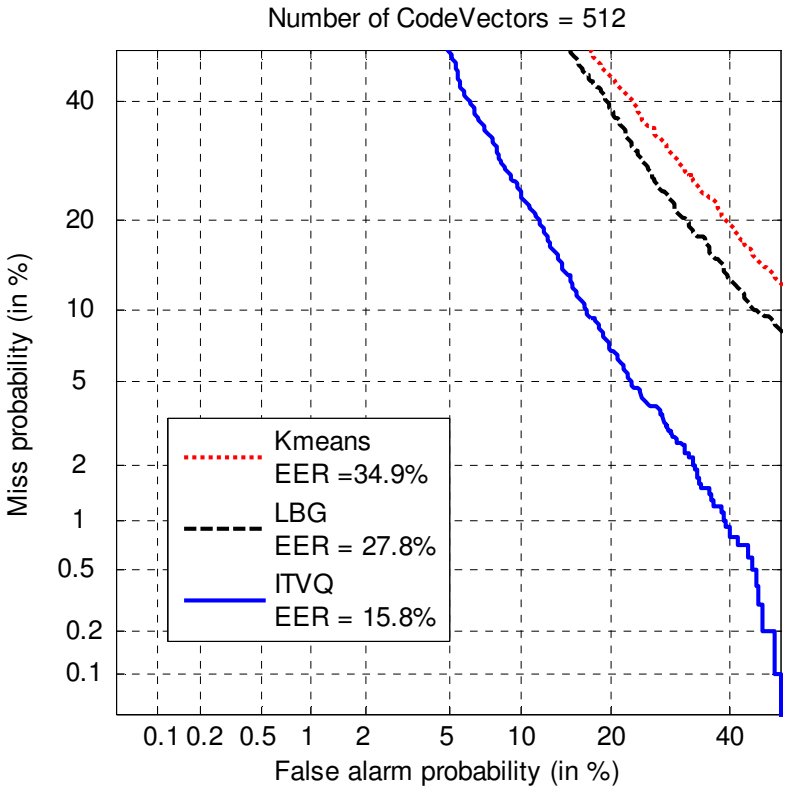


Figure 3.7(a) EER for K-means, LBG and ITVQ Classifiers for TIMIT Speech Corpora.

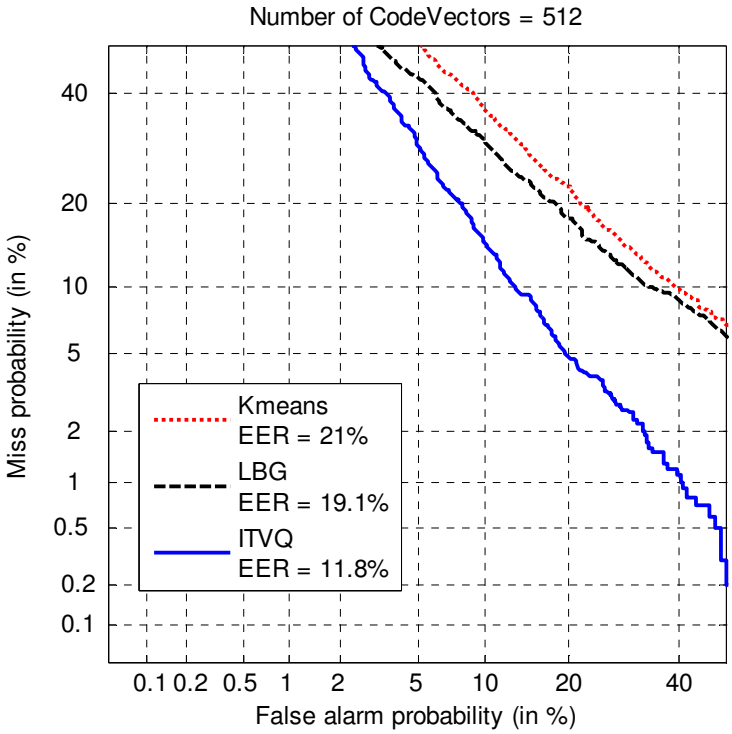


Figure 3.7(b) EER for K-means, LBG and ITVQ Classifiers for NIST'04 Speech Corpora.

Mean Square Error (MSE)

The third measure used to compare the performance of speaker verification based on different VQ algorithms was the mean squared error (MSE).

The mean square error (MSE) was calculated using the objective function for each of the evaluated procedures.

Figure 3.8(a) shows the MSE values based on the TIMIT corpora and Figure 3.8(b) shows the MSE based on the NIST 2004 corpora. Figure 3.8 (a)&(b) show the same trend as previously indicated by classification rates and EER, The ITVQ provides the lowest MSE values and the fastest convergence rates. The LBG algorithm gives the medium

performance and the K-means algorithm shows the largest MSE values and the slowest algorithm convergence rates.

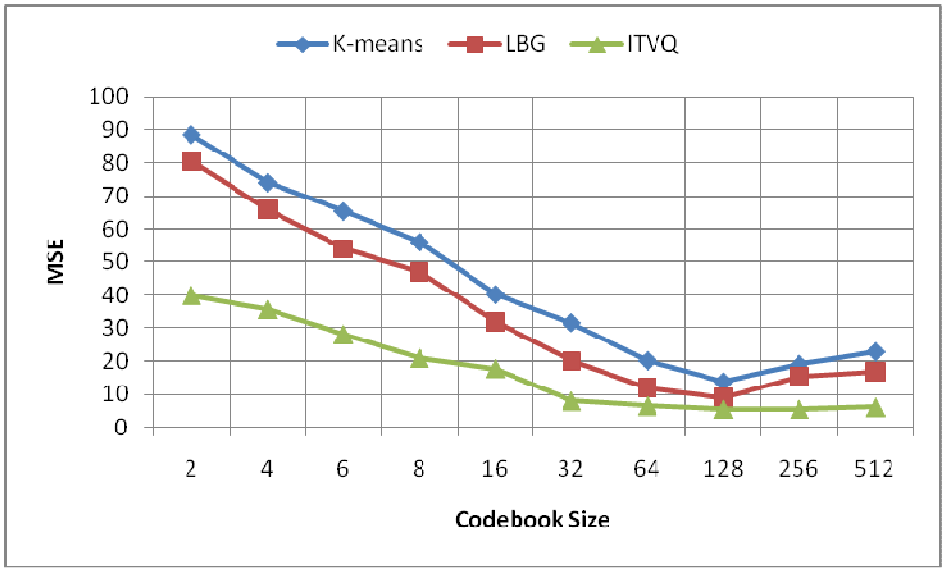


Figure 3.8(a) Mean square error for K-means, LBG and ITVQ Classifiers for TIMIT Speech Corpora.

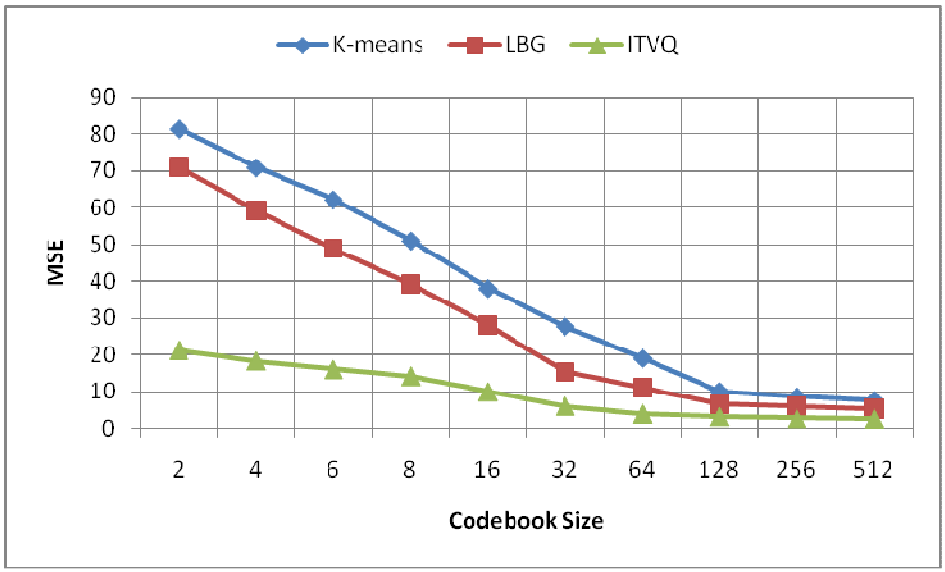


Figure 3.8(b) Mean square error for K-means, LBG and ITVQ Classifiers for NIST'04 Speech Corpora.

3.6 Summary

The chapter evaluated and compared the performance of three vector Quantization algorithms: K-means, LBG and ITVQ as modeling techniques for the speaker verification system.

The performance was compared using a feature set containing 12 MFCC coefficients. The evaluation was based on two speech corpora: TIMIT and NIST 2004. The results were evaluated it terms of three different performance measures: classification rates, equal error rates (EER) and mean squared error (MSE).

The results based on these three different measures and two speech corpora were consistent indicating that the ITVQ algorithm provides the best overall performance. The LBG algorithm was consistently showing medium performance, and the lowest results were obtained when using the K-means algorithm.

The outstanding performance of the ITVQ algorithm can be attributed to the fact that the Information Theoretic (IT) criteria used by this algorithm provide better matching between distribution of the original data vectors and the codewords.

CHAPTER 4

NEW INFORMATION THEORETIC EXPECTATION MAXIMIZATION ALGORITHM FOR THE GAUSSIAN MIXTURE MODELLING

This chapter introduces a new algorithm for the calculation of Gaussian Mixture Model parameters called Information Theoretic Expectation Maximization (ITEM). The proposed algorithm improves upon the classical Expectation Maximization (EM) approach widely used with the Gaussian mixture model (GMM) as a state-of-art statistical modeling technique. Like the classical EM method, the ITEM algorithm adapts means, covariances and weights, however this process is not conducted directly on feature vectors but on a set of centroids derived by the information theoretic vector quantization (ITVQ) procedure, which simultaneously minimizes the divergence between the Parzen estimates of the feature vector's distribution within a given class and the centroids distribution within the same class. The ITEM algorithm was applied to the speaker verification problem using NIST 2001, NIST 2002 and NIST 2004 corpora and MFCC with delta features. The results showed an improvement of the equal error rate over the classical EM approach. The EM-ITVQ also showed higher convergence rates compared to the EM.

4.1 Overview

This chapter demonstrates the performance of the speaker verification system using Gaussian mixture models (GMM) based on an information theoretic metric. The Gaussian mixture model (GMM) method is commonly regarded as the state of art modeling and classification technique. It was successfully applied in many pattern

recognition problems including speech and speaker recognition, stress and emotion classification, face recognition, and many others [109,125,128,131,177]. In its classical form the GMM applies the Expectation Maximization (EM) procedure to derive the model parameters. In Chapter 3 the speaker verification results based on the ITVQ, k-means and the LBG were compared. The results demonstrated a superior performance of the ITVQ method. In this chapter, the ITVQ algorithm was combined with the classical EM procedure and applied to estimate the GMM parameters including: weights, means and covariances. The results showed that this combination provides a significant improvement of the speaker verification results compared to the EM.

The speaker verification experiments were performed using the NIST 2001, NIST 2002 and NIST 2004 speaker recognition and evaluation (SRE) speech corpora. The evaluations based on the NIST corpora are widely used by researchers to assess and compare the performance of new speaker verification/identification methods.

The speech features used in speaker verification included: the mel-frequency cepstral coefficients (MFCC), the first derivative of MFCC (delta), the second derivative of MFCC (double delta), the energy of the each frame and the number of zero crossings for the respective speech frame.

A number of different window sizes of the delta features were examined to obtain a set of features which most efficiently represent the speaker's models. The pre-processing was used to eliminate the silence/noise speech intervals and to perform the pre-emphasis of speech.

This chapter is organized in the following way; Section 4.2 describes the theory of the GMM algorithm and the classical EM procedure. In Section 4.3 a brief review of the previous research combining the vector quantization methods with the EM method is given. The new ITEM approach is described in Section 4.4. Finally, Section 4.5 presents

the experimental results which show the performance comparison between the classical EM and the proposed ITEM algorithms.

4.2 The Gaussian Mixture Model and Expectation Maximization

The Gaussian mixture model (GMM) method [19] is commonly regarded as the state-of art modeling and classification technique successfully applied in many pattern recognition problems including speech recognition and speaker identification, image coding and many others.

In a variety of practical applications, the distribution of the parameters can be approximated by a family of finite mixture densities where, the density function is a weighted sum of component densities. The component densities are commonly modelled as Gaussians. It can be shown that any continuous probability density function can be approximated arbitrarily closely by a Gaussian mixture density [39]. In it's classical form [40], the GMM applies the expectation maximization algorithm (EM), which iteratively updates the means, covariances and weights for each class, and converges to a set of parameter vectors, providing the maximum value of the expectation function. Each set consisting of means, variances and weights constitutes a class model. The resulting models provide multivariate probability density functions for each class with the highest expectation values for given training data.

4.2.1 Gaussian Mixture Model

The GMM method iteratively develops different multivariate Gaussian probability density functions for each class. Given N classes and M components (Gaussian mixtures)

within each class, the Gaussian pdf of a feature vector $\mathbf{x}$ for the i[th] mixture within class k, is given as,

$$p_i^k(\mathbf{x}) = (1/(2\pi)^{R/2}\left|\boldsymbol{\Sigma}_i^k\right|^{1/2})\exp(-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_i^k)^T(\boldsymbol{\Sigma}_i^k)^{-1}(\mathbf{x}-\boldsymbol{\mu}_i^k)) \quad \textbf{(4.1)}$$

Where i=1,2,…M, k=1,…N, $\boldsymbol{\mu}_i$ is the component mean vector, $\boldsymbol{\Sigma}_i$ is the component covariance matrix, and R is the dimension of the feature vectors.

The set of weights, means and covariances for all components within a given class constitutes a class model $\lambda_k = \{w_i^k, \boldsymbol{\mu}_i^k, \boldsymbol{\Sigma}_i^k\}$, where k is the class index (k=1, 2,...,N). The probability that a feature vector $\mathbf{x}$ represented by a particular model λ_k belongs to any of the M components representing k[th] class is a weighted mixture of M Gaussian pdfs,

$$p(\mathbf{x}|\lambda_k) = \sum_{j=1}^{M} w_j^k p_j^k(\mathbf{x}) \quad \textbf{(4.2)}$$

Where $p_j^k(\mathbf{x})$ are the component mixture densities (pdfs) for a class k, and w_j^k are the mixture weights for a class k. The weight's values are usually constrained, such that $\sum_{j=1}^{M} w_j^k = 1$ to ensure that the maximum pdf value is equal to 1. The commonly used approach to estimate the GMM model parameters is the maximum likelihood (ML) estimation method which maximizes with respect to elements of λ_k, the conditional probability $p(\mathbf{X}_k|\lambda_k)$, where the vector $\mathbf{X}_k = \{\mathbf{x}_0^k, \mathbf{x}_1^k, ...\mathbf{x}_S^k\}$ contains all feature vectors for a particular speaker k. For simplicity it is assumed that all classes are represented by the same number of S vectors. The ML solution is derived iteratively by the expectation maximization (EM) algorithm.

4.2.2 Expectation Maximization (EM) Algorithm

The EM algorithm iteratively improves the estimates of elements of λ_k, by increasing on each iteration the probability, that the model estimate λ_k matches the observed feature vectors from a training set of data representing a given speaker k. This means that on each iteration $p(\mathbf{X}_k|\lambda_k^{iter+1}) > p(\mathbf{X}_k|\lambda_k^{iter})$, where *iter* is the iteration number and the conditional probability $p(\mathbf{X}_k|\lambda_k)$ is given as,

$$p(\mathbf{X}_k|\lambda_k) = \prod_{n=1}^{S} \sum_{i=1}^{M} w_i^k p_i^k(x_n^k) \quad \textbf{(4.3)}$$

The maximization of $p(\mathbf{X}_k|\lambda_k)$ with respect to the unknown probabilistic model λ_k can be achieved by maximizing the expectation of the log-likelihood [238] given as,

$$E\left(\log\left[p\left(\mathbf{X}_k, i \mid \lambda_k^{iter+1}\right)\right]\right) = \sum_{i=1}^{M} p\left(\mathbf{X}_k, i \mid \lambda_k^{iter}\right) \log\left[p\left(\mathbf{X}_k, i \mid \lambda_k^{iter+1}\right)\right] \quad \textbf{(4.4)}$$

It can be shown that maximizing $E\left(\log\left[p\left(\mathbf{X}_k, i \mid \lambda_k^{iter+1}\right)\right]\right)$ over λ_k^{iter+1} increases the likelihood, i.e. it makes $p(\mathbf{X}_k|\lambda_k^{iter+1}) > p(\mathbf{X}_k|\lambda_k^{iter})$.

By differentiating $E\left(\log\left[p\left(\mathbf{X}_k, i \mid \lambda_k^{iter+1}\right)\right]\right)$ given in Eq. (4.4) with respect to unknown mean, covariance and weight parameters and setting it to zero, the following parameter updating formulas can be derived:

$$\left\{ w_i^k \right\}^{iter} = \frac{1}{S} \sum_{n=1}^{S} p(i_n^k = i^k \mid \mathbf{x}_n^k, \lambda_k^{iter}) \tag{4.5}$$

$$\left\{ \boldsymbol{\mu}_i^k \right\}^{iter} = \frac{\sum_{n=1}^{S} p(i_n^k = i^k \mid \mathbf{x}_n^k, \lambda_k^{iter}) \mathbf{x}_n^k}{\sum_{n=1}^{S} p(i_n^k = i^k \mid \mathbf{x}_n^k, \lambda_k^{iter})} \tag{4.6}$$

$$\left\{ \boldsymbol{\Sigma}_i^k \right\}^{iter} = \frac{\sum_{n=1}^{S} p(i_n^k = i^k \mid \mathbf{x}_n^k, \lambda_k^{iter}) \mathbf{x}_n^k (\mathbf{x}_n^k)^T}{\sum_{n=1}^{S} p(i_n^k = i^k \mid \mathbf{x}_n^k, \lambda_k^{iter})} - \boldsymbol{\mu}_n^k (\boldsymbol{\mu}_n^k)^T \tag{4.7}$$

Where the *posterior* probabilities $p\left(i_n^k = i^k \mid \mathbf{x}_n^k \lambda_k^{iter}\right)$ are evaluated as,

$$p\left(i_n^k = i^k \mid \mathbf{x}_n^k \lambda_k^{iter}\right) = \frac{\left\{ w_i^k \right\}^{iter} p_i^{iter}(\mathbf{x}_n^k)}{\sum_{i=1}^{M} \left\{ w_i^k \right\}^{iter} p_i^{iter}(\mathbf{x}_n^k)} \tag{4.8}$$

and $p_i^{iter}(\mathbf{x}_n)$ is the ith pdf mixture component for the iteration *iter* and can be calculated using Eq. (4.1)

As illustrated in the flowchart of Figure 4.1, the EM algorithm is usually conducted in the following steps,

Step 1. Initialization.

An arbitrary initial set of models $\lambda_k^{initial}$ is generated for each class.

Step 2. Checking the stopping criteria.

Usually the algorithm proceeds until a maximum number of iterations are reached,

although other stopping criteria may be defined.

Step 3. Updating the model parameters.

The model's parameters are updated according to Eq. (4.5-4.7). Replacement of the model components is then made to obtain the next estimate of the k[th] class model λ_k^{iter}, and the procedure is repeated by returning to *Step 2*.

The flowchart of the EM algorithm is illustrated in Figure 4.1.

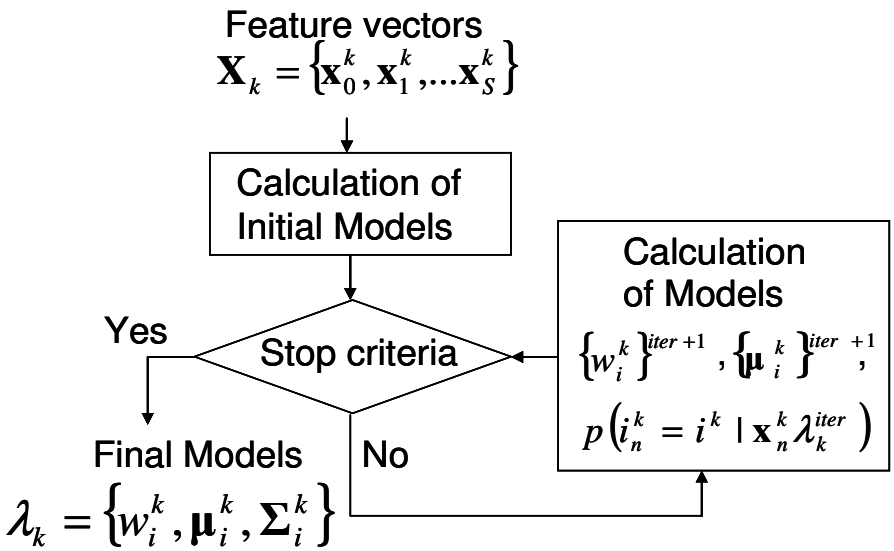


Figure 4.1 The EM algorithm flowchart.

Figure 4.2 shows that like vector quantization methods, the EM algorithm is also a clustering procedure. The multidimensional input feature vectors representing each class and denoted as black dots are grouped into M clusters (Gaussian mixtures). This grouping is denoted by the continuous-line ovals. The GMM/EM process is often regarded as a

"soft" clustering since each cluster is represented not by a single central vector but by a statistical multivariate Gaussian function given in Eq. (4.1).

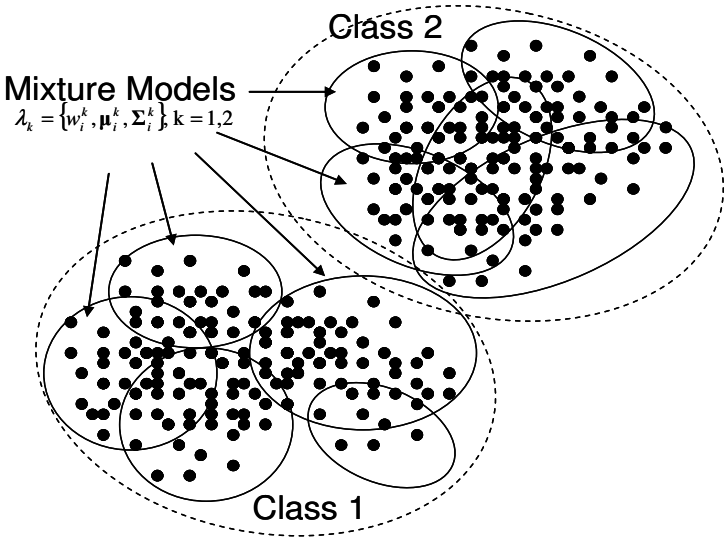


Figure 4.2 The EM viewed as a "soft" clustering process; the black dots represent feature vectors, the EM clusters or Gaussians (continuous-line black ovals) are built out of the original feature vectors.

The expectation maximization algorithm is not only used to approximate the Gaussian mixture parameters but as a general optimization procedure it can be also used to determine samples that diverge from *a priori* known distributions [122,178], evaluate the weight parameters in the re-weight least squares method [122], calculate the parameters of hidden Markov models (HMMs) [50] and make feature selection in order to achieve lowest prediction error [179].

4.2.3 Speaker Identification/Verification using the GMM models (testing process)

Assuming that N classes of the target speaker models λ_j, $j = 1,...,N$ have been estimated, the speaker identification or verification task can be performed,

Speaker Identification

In speaker identification the system identifies the target speaker to which the input utterance belongs.

Using a frame-by-frame approach feature vectors $\mathbf{x}_n$ are calculated from the speech samples of the speaker being identified. Given these features, the *a posterior* probabilities $P(\lambda_j \mid \mathbf{x}_n)$ of each speaker model are then computed and the speaker class with the highest probability is assigned to the speaker being identified. This method is called the maximum *a posteriori* probability (MAP) estimation [239]. The *a posteriori* probabilities $P(\lambda_j \mid \mathbf{x}_n)$ can be calculated from the pdf functions (Eq. (4.2)) derived by the GMM modeling using the following Bayes formula [238]:

$$P\left(\lambda_j \mid \mathbf{x}_n \right) = \frac{p(\mathbf{x}_n \mid \lambda_j) P(\lambda_j)}{P(\mathbf{x}_n)} \qquad \textbf{(4.9)}$$

Since $P(\mathbf{x}_n)$ has a constant value, maximization of Eq. (4.9) is equivalent to finding λ_j for which the numerator $p(\mathbf{x}_n \mid \lambda_j) P(\lambda_j)$ has the maximum value. The term $P(\lambda_j)$ is called the *a priori* probability of the speaker characterized by λ_j being the source of the input feature vector $\mathbf{x}_n$.

It is customary to assume that the *a priori* probabilities $P(\lambda_j)$ to be constant which further simplifies the identification problem to finding the GMM model λ_j which maximizes $p(\mathbf{x}_n \mid \lambda_j)$ given in Eq. (4.2).

In practice there is no one feature vector $\mathbf{x}_n$ for a speaker being identified but a set of N_F feature vectors $\{\mathbf{x}_1,..., \mathbf{x}_{N_F}\}$. Therefore the identification process must find the speaker model λ_j which maximizes the probability $p(\{\mathbf{x}_1,..., \mathbf{x}_{N_F}\} \mid \lambda_j)$. Typically it is assumed that the feature vectors are independent, therefore the probability $p(\{\mathbf{x}_1,..., \mathbf{x}_{N_F}\} \mid \lambda_j)$ can be calculated as the following product:

$$p(\{\mathbf{x}_1,..., \mathbf{x}_{N_F}\} \mid \lambda_j) = \prod_{n=1}^{N_F} p\left(\mathbf{x}_n \mid \lambda_j\right) \tag{4.10}$$

Since the EM algorithm calculates log of the probabilities $p(\mathbf{x}_n \mid \lambda_j)$, therefore by applying log to the both sides of Eq. (4.10), the following classification formula can be derived:

$$C = \max_{1 \le j \le N} \sum_{n=1}^{N_F} \log\left[p\left(\mathbf{x}_n \mid \lambda_j\right)\right] \tag{4.11}$$

Where C is the class index assigned to the input speaker.

Speaker Verification

The speaker verification is based on the assumption that speaker known to the system who is correctly claiming his/her identity is called a claimant and a speaker unknown to the system who is claiming to be a known speaker is called an imposter.

The speaker verification requires a binary decision stating either: that the test utterance belongs to the target speaker (hypothesis H0) or to the imposter (hypothesis H1).

Assuming that we have a GMM for the target speaker and GMM for a collection of imposters; a likelihood ratio that makes a decision between H0 and H1 is defined as a quotient between the probability $P(\lambda_C \mid X)$ that the input vectors $X = \{\mathbf{x}_1, ..., \mathbf{x}_{Ns}\}$ belong to the claimant speaker and the probability $P(\lambda_{\overline{C}} \mid X)$ that X is from the impostor speaker.

The Bayes' decision rule can be then expressed as:

$$\frac{P(\lambda_C \mid X)}{P(\lambda_{\overline{C}} \mid X)} = \frac{p(X \mid \lambda_C)P(\lambda_C)/P(X)}{p(X \mid \lambda_{\overline{C}})P(\lambda_{\overline{C}})/P(X)} \quad \textbf{(4.12)}$$

Where P(X) is the probability of the vector stream $X = \{\mathbf{x}_1, ..., \mathbf{x}_{Ns}\}$. Assuming that $P(\lambda_C)$, $P(\lambda_{\overline{C}})$ and $P(X)$ are constant and taking the log of both sides of Eq. (4.11), the following likelihood ratio can be derived:

$$\Lambda(X) = \log\left[p(X \mid \lambda_C)\right] - \log\left[p(X \mid \lambda_{\overline{C}}\right] \quad \textbf{(4.13)}$$

Choosing an arbitrary constant threshold value θ, the accepting/rejecting decision can be then made as follows:

$$\text{If } \Lambda(X) \geq \theta \text{, then accept}$$

$$\text{If } \Lambda(X) < \theta \text{, then reject.}$$

4.3 Drawbacks of the conventional EM-GMM method and previously proposed modifications

Although, in general the GMM method based on the classical EM procedure has been shown to provide very good speaker classification/verification rates, several studies pointed to drawbacks such as the sensitivity to the channel distortion, a relatively slow convergence rates (especially for large data bases) and a tendency of the EM algorithm to end up at sub-optimal solutions.

For applications of mixture modeling, one key issue is the number of parameters in the class models $\lambda_k = \{w_i^k, \boldsymbol{\mu}_i^k, \boldsymbol{\Sigma}_i^k\}$. The larger the number of parameters, the more precise description of the fine structure of the underlying data distribution can be achieved. On the other hand, a large number of parameters can lead to an overfit where the estimated model reflects random properties associated with the data. A large set of parameters can also lead to excessive complexity. Thus, the selection of the number of parameters must be a compromise. A method reducing the number of parameters was proposed in [27] where the covariance matrices $\boldsymbol{\mu}_i^k$ are assumed to be diagonal.

In [30], an improvement of the expectation–maximization (EM) algorithm for Gaussian mixture modeling was proposed using statistical tests. The first test is a multivariate normality criterion based on the Mahalanobis distance of a sample measurement vector from a certain Gaussian component center. This test was used in order to derive a

decision whether to split a component into another two or not. The second test is a central tendency criterion based on the observation that multivariate kurtosis becomes large if the component to be split is a mixture of two or more underlying Gaussian sources with common centers. If the common center hypothesis was true, the component was split into two new components and their centers are initialized by the center of the (old) component candidate for splitting. Otherwise, the splitting was accomplished by a discriminant derived by the third test. Experimental results are presented against seven other EM variants both on artificially generated data-sets and real ones demonstrate that the proposed EM variant has an increased capability to find the underlying model, while maintaining a low execution time.

Another major drawback of EM algorithm cited in the literature [29] is the tendency to converge into local minima. The reason behind this is the gradient descent character of the EM algorithm which allows the iterative solutions to proceed only towards solution giving "better" values of the objective function. A method based on the "hill climbing" search using simulated annealing which allows occasional moves towards "worse" values of the objective function was proposed in [28] to avoid the local convergence problem of EM algorithm.

In [240,241] Reynolds provided extensive evaluation of the GMM for speaker identification using clean speech from the TIMIT data base and the actual speech transmitted over the actual telephone lines from the NTIMIT database. The experiments showed significant degradation of the GMM performance due to the channel distortion.

In [242] Reynolds compared the GMM method based on the classical EM procedure with the minimum distance and the vector quantization (VQ) classifiers using different speaker identification tasks. The experiments used the KING database containing clean conversational utterances recorded with a high quality equipment as well as conversational utterances recorded over the telephone channel. The clean speech from the

KING database showed that the GMM outperformed the other two methods. The minimum distance classifier showed the worse performance. When using the telephone speech general decline in performance of all three methods was observed however their relative performance levels were the same.

Recent advancements in probabilistic models have led to increased interest in vector quantization as a possible alternative or a modifier of the EM optimization [26,27,28]. Vector quantization methods group the input feature vectors into [109] clusters. The clusters partition the input space. For any input vector the association with each cluster is calculated based on the given distance measure between the input vector and the vector representing a given class and called the cluster centre. As discussed in Chapter 3, the vector quantization methods have been successfully applied to the speaker verification problem showing good convergence rates and having the advantage of being relatively simple, computationally.

As explained in [26], the learning rules used by a number of vector quantization methods to optimize the cluster centers are equivalent to the iterative improvement of the model means provided by the EM algorithm. It was therefore suggested that, the learning rules used by a number of clustering techniques such as hard c-means (HCM), fuzzy c-means (FCM) and fuzzy learning vector quantization (FLVQ) to estimate the cluster centers can be used as approximations to the expectation maximization (EM) method as applied to Gaussian mixtures. The main benefit of using a VQ to approximate the model means was in the reduction of computational complexity and in some cases improvement of the algorithm convergence properties.
Different combination of the vector quantization (VQ) and the GMM methods have been therefore proposed as a mean of reducing the computational complexity and improving the convergence properties of the GMM modeling process.

In [243] the VQ approach was used to first sub-divide the features space into clusters for each speaker and then build a Gaussian model for each cluster. Each of the testing speech samples was only tested against the Gaussian mixture sub-model representing the closest cluster. The contribution from adjacent clusters was not taken under consideration. This approach reduced the accuracy of testing, however provided significant reduction in the testing time. A similar but simplified method was introduced in [244] where only a single Gaussian distribution was used to model each cluster. This approach not only disregarded the contribution of adjacent clusters but also the contributions of different Gaussian mixtures within a given cluster.

Another VQ-GMM combination was proposed in [23]. This method uses an extended VQ in the training phase and a Gaussian interpolation of a VQ model in the testing phase. The results using YOHO database showed improved classification rates compared to VQ and only a small deterioration compared to the full GMM. The training cost was only slightly higher than the cost required by a VQ method.

Pelicanos *et. al.* [24] proposed a method, combining vector quantization with single multi-dimensional Gaussians called the VQG algorithm for a rapid development of speaker models when using large data bases. However unlike [243,244], this method included in the testing process contributions of the adjacent clusters. A VQ was used to separate feature vectors into clusters representing different speakers and a single multi-dimensional Gaussian was calculated for each cluster. A substitute Gaussian Mixture Model was then calculated to provide pdfs for the adjacent regions by combining information from single Gaussian and fixed number of points from the adjacent regions. The tests using NIST 1996 showed comparable and in some cases improved performance to the conventional GMM method.

4.4 New Information Theoretic Expectation Maximization Algorithm

This section describes a new version of the EM algorithm which reduces the computational complexity of the classical EM algorithm and improves the convergence properties of the Gaussian mixture modeling process compare to the conventional EM-GMM method. The modifications are included in the training process; the testing part remains unchanged and proceeds as described in Section 4.2.3.

The proposed algorithm is a modified version of the EM optimization procedure and combines two objective criteria: the objective criterion used in the classical EM algorithm with the objective criterion used in the information theoretic vector quantization (ITVQ) method described in Chapter 3. The new method is referred to as the information theoretic expectation maximization (ITEM) algorithm.

In the EM-ITVQ method, the "soft" clustering process of the EM method (Figure 4.2) is enhanced by data reduction achieved through the ITVQ clustering (Figure 4.3). Figure 4.3 shows that the ITEM clusters are composed of the ITVQ centroids rather than the original feature vectors.

The ITEM algorithm convergence properties are reinforced by both maintaining the expectation maximization process of the EM algorithm, as well as an iterative improvement of centroids calculation guided by the information theoretic criteria which simultaneously minimize the divergence measure between each vector within a given cluster and the centroid of this cluster, and maximize the divergence between centroids of neighboring clusters.

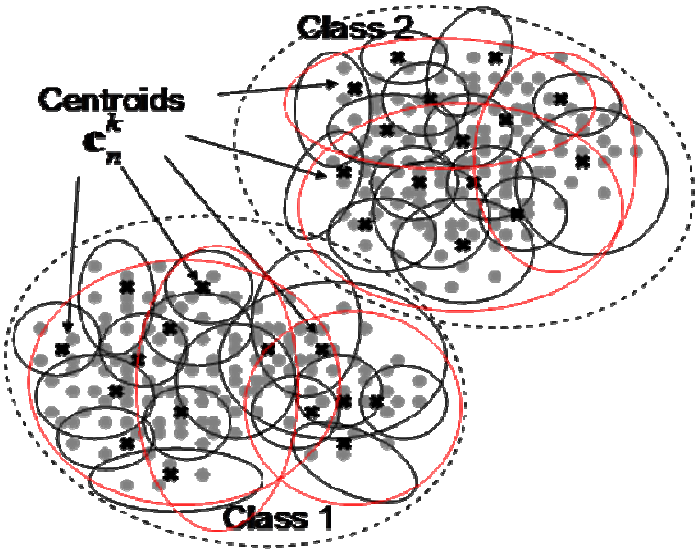


Figure 4.3 The ITEM clustering; the gray dots represent feature vectors, and the black crosses represent ITVQ centroids. The black ovals are the ITVQ clusters. The ITEM clusters (red ovals) are built out of the centroids rather than the feature vectors.

4.4.1 The ITEM Algorithm

As illustrated in Figure 4.4, the ITEM algorithm proceeds in the following steps,

Step 1. Initialization.

In this step an initial set of C centroids $\mathbf{C}_k^{init} = \{\mathbf{c}_0^k, \mathbf{c}_1^k, ...\mathbf{c}_C^k\}^{init}$ is generated for each class k using a relatively simple unsupervised clustering method such as for example the k-means algorithm. The centroids are then used to derive the initial models λ_k^{init} using Eq. (4.14-4.16).

Step 2. Checking the stopping criteria

In this step arbitrary stopping criteria are checked. Usually, the algorithm proceeds until an arbitrary number of iterations, is reached or the increase of the expectation value over a number of consecutive iterations is less than an arbitrary threshold value ζ.

Step 3. ITVQ Update

During this step, the ITVQ algorithm iteratively improves the centroids with respect to the information theoretic (IT) criteria producing a new set of centroids $\mathbf{C}_k^{iter} = \left\{\mathbf{c}_0^k, \mathbf{c}_1^k, ...\mathbf{c}_C^k\right\}^{iter}$ for each class k. The optimal number N_{ITVQ} of the ITVQ sub-iterations has to be determined experimentally.

Step 4. Updating the model parameters

In this step the model's parameters are updated using centroids $\mathbf{C}_k^{iter} = \left\{\mathbf{c}_0^k, \mathbf{c}_1^k, ...\mathbf{c}_C^k\right\}^{iter}$ calculated in *Step 3* and the following formulas:

$$\left\{w_i^k\right\}^{iter} = \frac{1}{C}\sum_{n=1}^{C} p(i_n^k = i^k \mid \mathbf{c}_n^{k,iter}, \lambda_k^{iter-1}) \tag{4.14}$$

$$\left\{\boldsymbol{\mu}_i^k\right\}^{iter} = \frac{\sum_{n=1}^{C} p(i_n^k = i^k \mid \mathbf{c}_n^{k,iter}, \lambda_k^{iter-1})\mathbf{c}_n^{k,iter}}{\sum_{n=1}^{C} p(i_n^k = i^k \mid \mathbf{c}_n^{k,iter}, \lambda_k^{iter-1})} \tag{4.15}$$

$$\left\{\boldsymbol{\Sigma}_i^k\right\}^{iter} = \frac{\sum_{n=1}^{C} p(i_n^k = i^k \mid \mathbf{c}_n^{k.iter}, \lambda_k^{iter-1})\mathbf{c}_n^{k,iter}(\mathbf{c}_n^{k,iter})^T}{\sum_{n=1}^{C} p(i_n^k = i^k \mid \mathbf{c}_n^{k,iter}, \lambda_k^{iter-1})} - \boldsymbol{\mu}_i^{k,iter}(\boldsymbol{\mu}_i^{k,iter})^T \tag{4.16}$$

A replacement of the model components is then made to obtain the next set of estimates λ_k^{iter} , and the procedure is repeated by returning to *Step 2.*

In general, the ITVQ process can be viewed as a "sharp" clustering (Figure 4.3), where each class is divided into a number of clusters and each cluster is represented by C centroid vectors. Thus, on each iteration of the ITEM, the large number of S original feature vectors in each class is updated along with a smaller number of C (C<S) representative centroid vectors. The centroids are iteratively refined using information theoretic criteria nested within the EM procedure. Thus, the EM-ITVQ has a dynamic character as it applies the updating formulas of Eq. (4.14-4.16) not to a constant set of feature vectors but to a gradually more and more refined configurations of centroids which change at each iteration. It is worth noticeable in Eq. (4.14-4.16) that the weight, mean and covariance updates are applied on centroid vectors $\boldsymbol{c_n}$ instead of feature vectors.

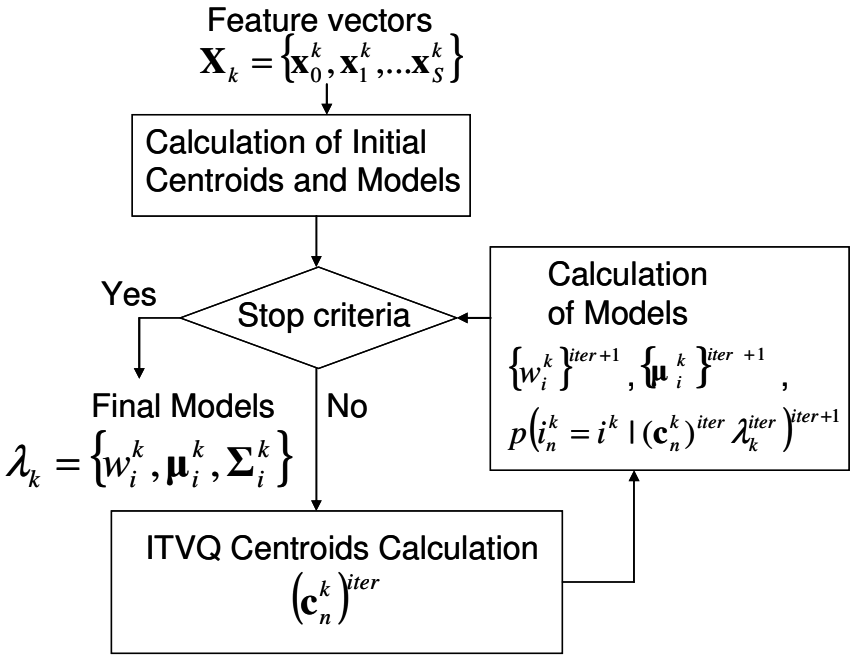


Figure 4.4 The ITEM algorithm.

4.4.2 ITVQ Centroids Calculation

For a given set of feature vectors $\mathbf{X}_k = \{\mathbf{x}_0^k, \mathbf{x}_1^k, ...\mathbf{x}_S^k\}$, finding the optimal configuration $\mathbf{C}_k = \{\mathbf{c}_0^k, \mathbf{c}_1^k, ...\mathbf{c}_C^k\}$ of C centroids is equivalent to minimizing the divergence between the Parzen estimates of the feature vector's distribution within a given class and the centroids distribution within the same class. The Parzen estimate of the feature vector's distribution has S Gaussian kernels, and it is given as,

$$\hat{f}(\mathbf{x}) = (1/S)\sum_{j=1}^{S} \exp(-(1/2)(\| \mathbf{x} - \mathbf{x}_j \|^2 / \sigma_f^2)) \quad \textbf{(4.17)}$$

The Parzen estimate of the centroids distribution has C (C<S) Gaussian kernels, and it is given as,

$$\hat{g}(\mathbf{x}) = (1/C)\sum_{j=1}^{C} \exp(-(1/2)(\| \mathbf{x} - \mathbf{c}_j \|^2 / \sigma_g^2)) \quad \textbf{(4.18)}$$

Where σ_f^2 and σ_g^2 are the kernel variances. The cost function is the divergence $J(\mathbf{c})$ between these two distributions given by the Cauchy-Schwarz formula,

$$J(\mathbf{c}) = \log\int \hat{f}^2(\mathbf{x})d\mathbf{x} - 2\log\int \hat{f}(\mathbf{x})\hat{g}(\mathbf{x})d\mathbf{x} + \log\int \hat{\mathbf{g}}^2(\mathbf{x})d\mathbf{x} \quad \textbf{(4.19)}$$

The cost function is minimized by calculating the derivatives of $J(\mathbf{c})$ with respect to the centroids $\mathbf{c}_i$, which leads to the following centroid updating formula,

$$\mathbf{c}_i(n+1) = \mathbf{c}_i(n) - \eta\left((\Delta\mathbf{V}/\mathbf{V}) - 2(\Delta\mathbf{D}/\mathbf{D})\right) \quad \textbf{(4.20)}$$

Where i=1,…, C and n is the ITVQ index, η is a constant step size, and the vectors $\mathbf{D}$ and $\mathbf{V}$ are given as,

$$\mathbf{D} = \int \hat{f}^2(\mathbf{x})\,\hat{g}(\mathbf{x})d\mathbf{x} \quad \textbf{(4.21)}$$

$$\mathbf{V} = \int \hat{g}^2(\mathbf{x})d\mathbf{x} \quad \textbf{(4.22)}$$

The terms $\Delta\mathbf{D}$ and $\Delta\mathbf{V}$ are the vectors of derivatives of $\mathbf{D}$ and $\mathbf{V}$ respectively, calculated with respect to the centroids $\mathbf{c}_i$. The η value of 0.03 provided satisfactory results when applied to the speaker verification problem.

Summarizing, the proposed ITEM algorithm can be either seen as a sequential application of the ITVQ to derive the cluster centroids and then the classical EM applied to these centroids to approximate Gaussian mixture parameters. However, since both algorithms the EM and the ITVQ operate on the same set of vectors (centroids), the ITEM can be simply described as a classical EM optimization applied to centroids rather than original feature vectors and using two rather than one objective criteria (expectation maximization and minimization of the divergence between feature vectors distribution and centroids distribution).

4.5 Speaker Verification Experiments using the Proposed ITEM Method and the Conventional EM

In this section results obtained when applying the proposed ITEM algorithm to the GMM based speaker verification are presented and compared with results obtained when using the conventional EM method to derive the Gaussian mixture model parameters for speaker verification.

4.5.1 Overview of the Speaker Verification System

The configuration of the speaker verification system used in the experiments examining the performance of the proposed ITEM method is shown in Figure 4.5.

The system can operate in one of the three possible modes:

- The Universal Background Model (UBM) training,
- The target speaker enrollment (training), and
- The testing mode.

In each case identical speech detection and feature extraction methods are used.

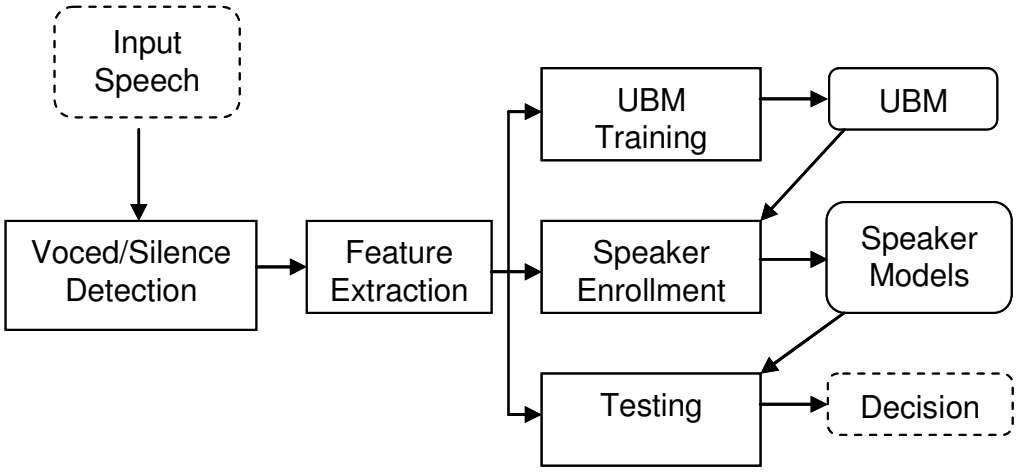


Figure 4.5 UBM-GMM based Speaker Verification System.

In the voiced/silence detection block in Figure 4.5, an energy based silence detector which identifies the low energy portions of the signal as silence regions was used; details of the applied speech activity detection method are described in Chapter 2.

As indicated in Chapter 2, the MFCC based speaker verification is relatively robust to the changes in the frame size ranging from 20 ms to 50 ms, and the frame step ranging from 1/6 to 1/3 of the frame size. Following these recommendations, the MFCC feature extraction method was implemented on the frame by frame basis using frames of length 20ms with 10ms (50% of the frame length) of overlap between adjacent frames.

The feature vector representing a given frame had 38 dimensions including: 12 MFCC parameters, 12 delta parameters ΔMFCC (first derivative of MFCC), 12 double delta parameters ΔΔMFCC (second derivative of MFCC), 1 averaged spectral energy parameter calculated for the speech signal within a given frame, and 1 zero crossing parameter calculated for the speech waveform within a given frame.

The sequences of feature vectors were subsequently modeled using the GMM based on the conventional EM algorithm and the proposed ITEM algorithm.

The number of Gaussians (Gaussian mixtures) used for every target speaker was 1024. The speech samples for training and testing were taken from the NIST 2004 and NIST 2002 corpora. In the speaker enrollment (training) stage each speaker was represented by speech utterances of the total length of 5 minutes (for NIST 2004) and 3 minutes (for NIST 2002) and in the testing stage by speech utterances of the total length of 5 minutes (for NIST 2004) and 3 minutes (for NIST 2002). The training and testing sets contained mutually exclusive sets of speakers.

After the target speaker's enrollment, the universal background model (UBM) parameter's inference was accomplished using a large corpus of speech containing only the non-target speakers (speakers not used in the enrolment and testing stages). To generate the UBM parameters speech recordings of the total length of 1 hour from the NIST 2001 were used. The details about creating a UBM are given in Chapter 2, Section 2.4.4.

The speaker verification (testing) was performed using approach described in Section 4.2.3.

The system performance was assessed using the equal error rate (EER) measure and by plotting detection error trade-off (DET) curve as described in Section 4.5.4.

4.5.2 Description of Speech Corpora

Table 4.1 contains a summary of speech corpora used in the speaker verification experiments.

The annual NIST speaker recognition evaluations (SRE) provide speech corpora widely used in evaluations of new methods introduced in the field of speaker recognition.

The speaker recognition experiments based on the EM-ITVQ and EM methods were conducted using data from NIST corpora: NIST 2001, NIST 2002 and NIST2004.

The NIST 2004 SRE data consisted of telephone conversational speech and excerpts from the Linguistic Data Consortium's Mixer project. The NIST 2004 experimental protocol used the 1side-training and 1side-testing task [141]. For each of the 616 target speakers (248 males and 368 females), 5 minutes of un-transcribed, concatenated (after silence/unvoiced removal) speech was used for target speakers training and 5 minutes of speech utterances for testing. The training and testing sets were mutually exclusive.

The NIST 2002 SRE data consisted of the telephone conversational speech and excerpts from the Switchboard corpus. The NIST 2002 experimental protocol used the one-speaker detection task [140]. For each of the 330 speakers (139 males and 191 females), 3 minutes of un-transcribed, concatenated (after silence/unvoiced removal) speech was used for target speakers training and 3 minutes of speech utterances for testing. The training and testing sets were mutually exclusive.

A subset of the NIST 2001 speech data consisting of about 1 hour of the cellular telephone conversational speech recorded from 174 speakers (74 males and 100 females)

was used to train the UBM model parameters. The NIST 2001 experimental protocol applied in experiments uses the one-speaker detection task [139].

The speech from all three corpora was sampled at 8 KHz. It was ensured that the UBM data did not share speakers with the target training and testing sets, and any duplicate speakers were removed. The summary of the properties of the training data is given in Table 4.1.

Table 4.1 Summary of Speech Corpora Used in Experiments with ITEM.

Description	***NIST 2001***	***NIST 2002***	***NIST 2004***
Language	English	English	English
Number of speakers	174 speakers (74 males and 100 females)	330 speakers (139 males and 191 females)	616 speakers (248 males and 368 females),
Speech type	conversational	conversational	Conversational
Record condition	Cellular Telephone	Broadcast news, microphone speech, telephone speech	Telephone Speech
Sampling rate	8KHz	8KHz	8KHz
UBM training	1 hour	-	-
Target training	-	3 min	5 min
Testing	-	3 min	5 min

4.5.3 Comparison of the Convergence Rates and the Computational Complexity of EM and ITEM

The effect of different number of ITVQ iterations on the convergence rates of the reciprocal of the log-likelihood function in Eq. (4.4) is illustrated in Figure 4.6. It can be noted that, the ITEM maintains the monotonic behavior of the EM algorithm providing on each iteration improved log-likelihood values. The ITEM algorithm introduced additional complexity to the modeling process by applying the ITVQ centroids updating procedure. Based on the convergence rates illustrated in the Figure 4.6, this additional complexity can be reduced to 15 ITVQ sub-iterations (updates) while still maintaining significantly higher convergence rate of the ITEM procedure compared to the conventional EM approach. Figure 4.6 shows that while increasing the number of ITVQ updates from 3 to 15, the convergence rate of EM-ITVQ were improved on an average by 32% compared to the EM algorithm. An increase from 15 to 50 updates provided further improvement of the average convergence rates by 13.5%. The convergence rates were approximated by selecting several points on the convergence curves and calculating an average gradient.

The issue of computational complexity for the task of speaker verification is of vital importance. The complexity of a speaker verification system depends upon following,

- ➔ The computations originating from the distance or likelihood between the feature vectors of the unknown speaker and the models in the database,
- ➔ It vitally depends on the number of feature vectors and their dimensionality,
- ➔ The complexity of the training method used to generate speaker models
- ➔ The number of speakers.

From the numerical computation point of view, the proposed ITEM method uses the EM and the information theoretic criteria which obviously increases the computational

complexity, However with the careful analysis of Figure 4.6 (convergence rates) we can observe,

- ➔ The likelihood of 3 is obtained by EM algorithm at around 55th EM update, see Figure 4.6.
- ➔ The likelihood of 3 is obtained by ITEM algorithm at around 8th ITEM update, see Figure 4.6.

With this analysis, it can be established that on one hand ITEM increases the computational complexity by using the additional criteria of ITVQ but on other hand it shows the significant reduction in computational complexity by reducing the number of iterations from 55 to 8.

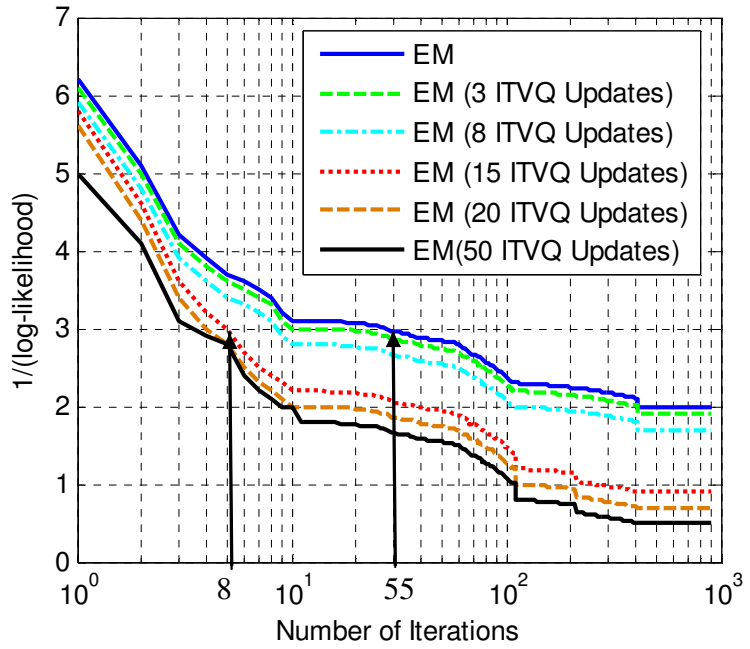


Figure 4.6 Convergence rates for the EM and ITEM algorithms.

4.5.4 Comparison of the Speaker Verification Results

The speaker verification scores discussed were based on the ITEM procedure using 15 ITVQ updates. In this approach log-likelihood ratio is evaluated, and the convergence is actually aligned with the improved likelihood estimates.

Figure 4.7 and Figure 4.8 illustrate the percentage miss probability versus the percentage of false alarm probability for both the EM and the ITEM optimization procedures. The miss probability was calculated as the probability that the system incorrectly declares a successful match between the input features and a non-matching model in the database. It measures the percent of invalid matches. The false alarm probability was calculated as the probability that the system incorrectly declares failure of match between the input features and the matching model. It measures the percent of valid inputs being rejected. The EER parameter represents the rate at which both the miss probability and the false alarm probability are equal. The lower the EER, the more accurate the system is considered. A MAP-UBM based GMM system was defined which involves training for both optimization algorithms. The algorithms were tested using 38-dimensional (R=38) feature vectors. The ITEM based modeling shows an improvement of the average equal error rate (EER) value over the classical EM algorithm. The average improvement of EER is about 1.95% (R=38) for NIST 2002 and 1.5% (R=38) for NIST 2004.

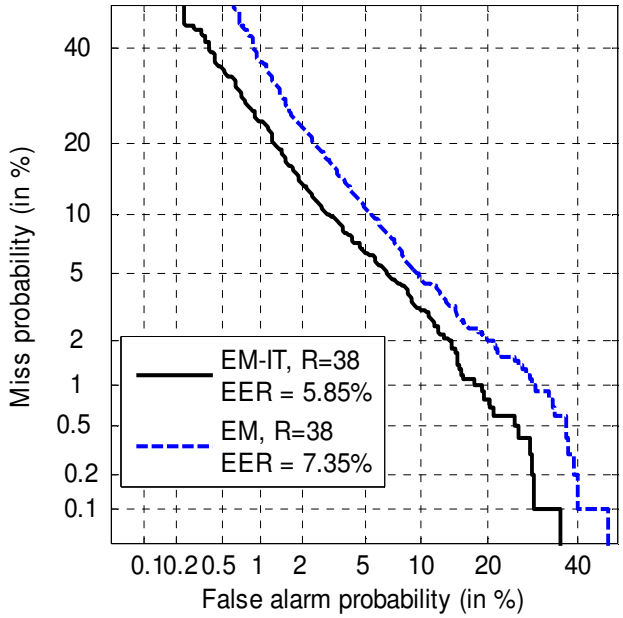


Figure 4.7 Miss Probability versus false alarm for EM and ITEM using NIST 2004 for speaker enrolment and testing. The UBM was developed using NIST 2001.

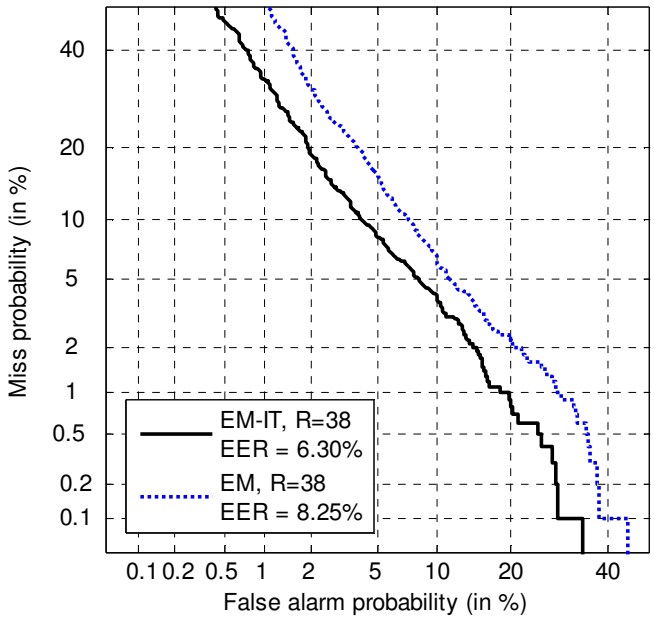


Figure 4.8 Miss Probability versus false alarm for EM and ITEM using NIST 2002 for speaker enrolment and testing. The UBM was developed using NIST 2001.

4.6 Summary

A novel approach to the GMM training of speaker models has been described.

It has been empirically demonstrated that when applied to the speaker verification task, the ITEM modeling algorithm achieves higher convergence rates and provides smaller EER values compared with the classical EM algorithm.

In contrast to EM, the ITEM method works with averaged feature vectors (centroids) rather than the original feature vectors. The averaging process removes noise, reduces the data, and captures only the essential characteristics, which results in a quality improvement of the final models. The essential characteristics refer to the refinement of the data clusters.

Unlike EM, which relies only on maximization of the expectation function, the EM-ITVQ method is guided by an additional objective given in Eq. (4.19) helping to minimize a divergence measure between the distribution of the original feature vectors and the distribution of the centroids. The resulting centroids have the Parzen density matching the density of the original features. The replacement of the original feature vectors by a set of centroids with matched distribution has a key importance in increasing the speed of the mean adaptation process.

From the structural point of view, the proposed ITEM algorithm can be seen as a sequential application of the ITVQ deriving the cluster centroids followed by the classical EM applied to these centroids to approximate the Gaussian mixture parameters. Since both algorithms, the EM and the ITVQ operate on the same set of vectors (centroids), the ITEM can be alternatively described as a classical EM optimization applied to centroids rather than original feature vectors and using two rather than one objective criteria

(expectation maximization and minimization of the divergence between feature vectors distribution and centroids distribution).
The proposed ITEM algorithm does not alter the testing (verification) process; the only changes are introduced at the training stage.

Unlike previously proposed methods, which limited to the number of mixtures to a single Gaussian [24,244], the new ITEM algorithm does not alter the fundamental structure of the classical EM algorithm, and allows modeling of multiple Gaussians.

CHAPTER 5

LINEAR VERSUS NON-LINEAR FEATURES FOR SPEAKER VERIFICATION

This chapter compares the classical features based on linear models of speech production with recently introduced features based on the nonlinear model. A number of linear and nonlinear feature extraction techniques that have not been previously tested in the task of speaker verification are tested. New fusions of features carrying complimentary speaker-dependent information are proposed. The tested features are used in conjunction with the new ITEM-GMM speaker modeling method described in Chapter 4, which provided an additional evaluation of the new method. The speaker verification experiments presented in this chapter demonstrated a significant performance improvement when the conventional MFCC features were replaced by a fusion of the MFCCs with complimentary linear features such as the inverse MFCCs (IMFCCs), or nonlinear features such as the TMFCCs and TEO-PWP-Auto-Env. Higher overall performance of the nonlinear features was observed when compared to the linear features is observed.

5.1 Overview

The speech features must provide an ample representation of the speech signal. A number of sources can lead to the redundant or inaccurate information being added to the speech signal, which actually affects widely the speaker-specific information and design of a speaker model. Such sources include interference from the environment and distortions added by the transmission channel. In the speech/speaker recognition tasks, it is required

that the speech features represent the specifics of particular voice with sufficient accuracy.

Speech carries information about the message to be conveyed, speaker characteristics, and the language. According to the source-filter model, speaker characteristics in the speech signal can be attributed to the excitation source characteristics and the vocal tract characteristics.

The speaker-specific vocal tract information is most often represented by the vocal tract features including Mel-frequency cepstral coefficients (MFCCs) and linear prediction (LP) cepstral coefficients [77].

A relatively smaller number of studies investigated the usefulness of features extracted from excitation source characteristics for speaker recognition [55,155,157,180].

Another approach to finding the most representative features is to combine different types of features which carry complimentary information about speakers.

In this chapter different linear and nonlinear features and feature combinations are examined. Some of these features have not been previously tested in the task of speaker verification; however they showed good performance in the related disciplines such as speech recognition, as well as stress and emotion recognition in speech.

The chapter starts with examination of different variants of the classical MFCC parameters based on the linear source-filter model of speech production, and then moves into applications of the inverted MFCC (IMFCC). This is followed by the study of features based on the latest nonlinear models of speech production and the Teager Energy Operator (TEO). The nonlinear features include TMFCC and TEO-PWP-Auto-Env parameters.

Fusion of features carrying complimentary speaker-dependent information are also proposed and tested. These new methods include: MFCC/IMFCC, MFCC/TMFCC and MFCC/ TEO-PWP-Auto-Env.

In all cases, the feature performance was tested using speech from the NIST corpora and the new ITEM approach to the GMM speaker modeling described in Chapter 4.

5.2 Importance of the human auditory characteristics for speech parameterization

Fletcher in [182] conducted experiments, which measure the threshold of hearing of a sinusoidal signal as a function of the bandwidth of a band-pass noise masker. Based on these experiments it was found that the human auditory system behaves as if it consisted of a bank of band-pass filters with overlapping pass-bands. These filters are now known as *auditory filters*. The work of Zwicker [184] led to the definition of the *bark scale*, which improved the definition of auditory filters and finally another improvement was made in [185] leading to the so called Equivalent Rectangular Bandwidth (ERB) scale. In [181,183,184] a mapping between objective frequency in Hz and a subjective perception of pitch leading to a development of the mel *scale* was described.

According to psychophysical studies, human perception of the frequency content of sounds follows a subjectively defined nonlinear scale called the mel scale. The mel (derived from the word melody) scale, is a heuristically determined perceptual scale and provides the relation between subjectively perceived frequency (or pitch) of a pure tone as a function of its objective acoustic frequency. The mapping curve between the acoustic frequency in Hz and the subjective pitch in mels is shown in Figure 5.1.

Studies of speaker, stress and emotion recognition in speech clearly indicate that characteristic features based on human auditory characteristics provide better performance than features that do not take these characteristics into account [226]. One reason for it is the fact that the speaker-characteristic information embedded into the speech signal is optimized for the human (not machine) perception and therefore the information is encrypted into the structure of human auditory filters. The widely used mel-frequency cepstral coefficients (MFCC) [163] described in Chapter 2 provide an example of feature parameters based on the human auditory perception. It was demonstrated in [19,186] that in noisy conditions MFCC show higher robustness than features such as LPCC, PLP, which do not incorporate human auditory characteristics.

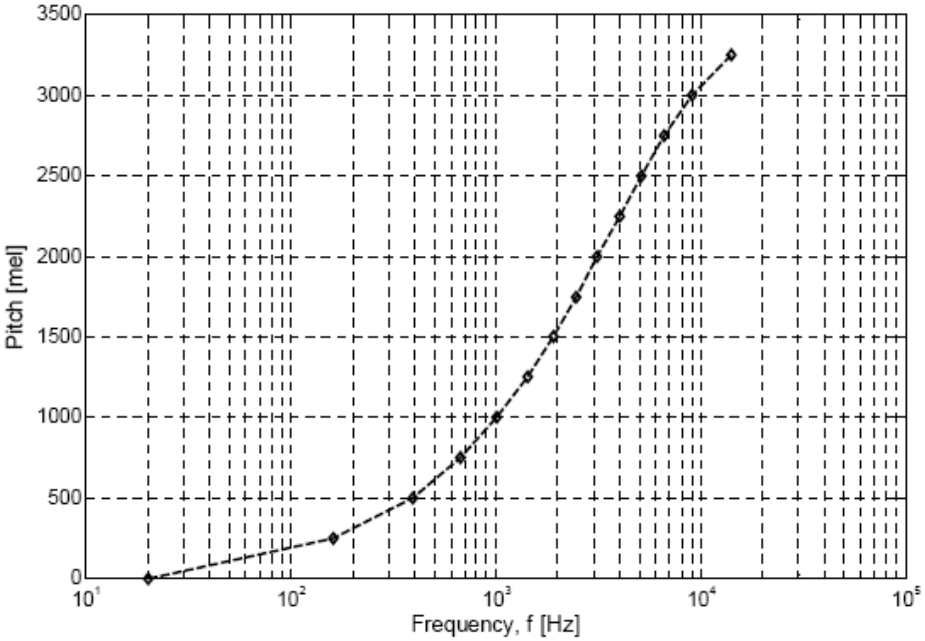


Figure 5.1 Pitch in mels versus frequency adapted from [181].

5.3 Different versions of features based on the MFCC parameters

The MFCC were first introduced and applied to speech processing in [163]. Since then a number of MFCC variants are proposed and compared with its original implementation [187,188]. The MFCC variants differ based on choice of the number of filters, the shape of the filters, the way the filters are spaced, the bandwidth of the filters, and the manner in which the power spectrum is deformed. In addition based on the requirement of a particular set of speakers or the design attributes of a corpus, MFCC variants also differ based on the frequency range of interest, number of cepstral coefficients that are chosen to design a speaker model.

The diversity in the MFCC implementations was also caused by the advancement made in psychoacoustics which gradually provided more and more refined models of the human auditory perception.

A number of approximations were established based on how the pitch perception is related to the human auditory system.

The mel scale is defined as a logarithmic scale of frequency based on human pitch perception. Equal intervals in mel units correspond to equal pitch intervals. The following mapping formula between frequency in Hz and the corresponding subjective pitch in mels is the most widely used for the MFCC implementation in speech and speaker recognition applications.

$$f_{mel} = 2595 \log_{10}\left(1 + \frac{f}{700}\right) \quad \textbf{(5.1)}$$

Where f_{mel} is the subjective pitch in mels corresponding to f which is the actual frequency in Hz.

Table 5.1 includes a list of different MFCC implementations introduced by researchers.

Table 5.1 Variants of the MFCC features.

MFCC variants
• **MFCC FB-20** – introduced by Davis [163] (1980).
• **MFCC FB-24** – from the Cambridge HMM Toolkit (HTK version 3.4.1 2009) described in [189]
• **MFCC FB-40** – from the MATLAB Auditory Toolbox of Slaney [190].

The MFCC variants differ by the number of filters and the methods of calculation of the filter's centre frequencies. Detailed descriptions of these different methods can be found in [163,189,190].

5.3.1 Calculation of the MFCC parameters

This section describes as an example the steps used in the evaluation of the mel frequency cepstral coefficients MFCC FB-20 [163]; which subsequently leads to the mathematical derivation and experimental evaluation of the inverted MFCC (IMFCC) parameters in Section 5.4.

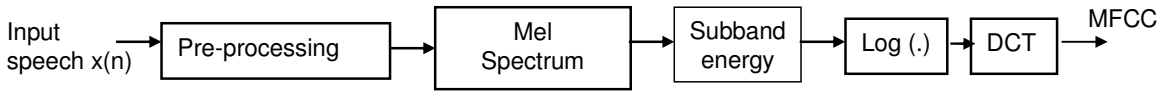


Figure 5.2 Calculation of the MFCC parameters.

As illustrated in Figure 5.2, the MFCC parameters are calculated by mapping the speech spectrum into mel frequency scale. This mel frequency mapping is done by multiplying the magnitude of speech spectrum for a preprocessed frame by magnitude of triangular filters in mel filterbank followed by log-compression of sub-band energies of the mel-scale filters and finally DCT.

Let **x**(n) represent a speech frame that is pre emphasized and hamming windowed. Firstly **x**(n) is converted to frequency domain by an N point DFT of the input signal,

$$\mathbf{X(k)} = \sum_{n=0}^{N-1} \mathbf{x(n)}.\exp\left(\frac{-j2\pi nk}{N}\right), k = 0,1,....., N-1. \qquad \textbf{(5.2)}$$

This is followed by a filter bank, with M equal height triangular filters (*i=1,2,…,M*). Each of these M equal height filters is defined as,

$$\mathbf{H_i(k)} = \begin{cases} 0 & for \quad k < f_{b_{i-1}} \\ \dfrac{k - f_{b_{i-1}}}{f_{b_i} - f_{b_{i-1}}} & for \quad f_{b_{i-1}} \le k \le f_{b_i} \\ \dfrac{f_{b_{i+1}} - k}{f_{b_{i+1}} - f_{b_i}} & for \quad f_{b_i} \le k \le f_{b_{i+1}} \\ 0 & for \quad k > f_{b_{i+1}} \end{cases} \qquad \textbf{(5.3)}$$

where *i=1,2,…,M* stands for the i^{th} filter, f_{bi} are the boundary points of the filters, and k=1,2,….,N corresponds to the k^{th} coefficient of the N-point DFT. The boundary points f_{bi} are expressed in terms of position. Their relative position depends on the sampling frequency *Fs* and the number of points *N* in the DFT,

$$f_{b_i} = \left(\frac{N}{F_s}\right) f_{mel}^{-1}\left[f_{mel}(f_{low}) + \frac{i\{f_{mel}(f_{high}) - f_{mel}(f_{low})\}}{M+1}\right] \qquad \textbf{(5.4)}$$

f_{mel} is defined in Eq. (5.1), f_{low} and f_{high} are respectively the low and high boundary frequency for the entire filter bank. M is the number of filters and f_{mel}^{-1} is the inverse of Eq. (5.1) and can be written as,

$$f_{mel}^{-1} = 700.\left(\exp(\frac{f_{mel}}{2595}) - 1 \right) \quad \textbf{(5.5)}$$

The filter bank used comprises of twenty equal height filters. The centre frequencies of the first ten filters are linearly spaced between 100 Hz and 1000 Hz, and the next ten have centre frequencies logarithmically spaced between 1000 Hz and 4000 Hz. The centre frequency for the i^{th} filter can be approximated as [191],

$$f_{c_i} = \begin{cases} 100.i & i = 1,.....,10 \\ f_{c_{10}}.2^{0.2(i-10)} & i = 11,.....,20 \end{cases} \quad \textbf{(5.6)}$$

Where the centre frequency is assumed to be in Hz.

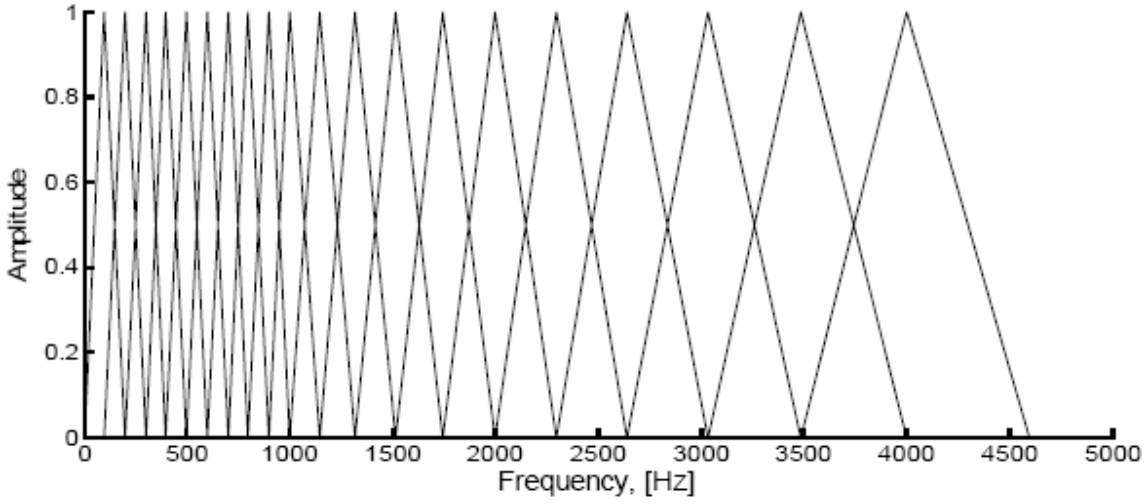


Figure 5.3 A mel spaced filter bank with 20 filters; the centre frequencies of the first ten filters are linearly spaced and the next ten are logarithmically spaced.

The endpoints of each one of the triangular filters are determined by the centre frequencies of adjacent filters as shown in Figure 5.3. The bandwidths of the filters depend upon the spacing between the centre frequencies of the adjacent filters, which is a function of the sampling rate of the signal and the number of the filters in the filter bank. Therefore, for a given sampling frequency, increase of the number of filters results in decrease of their bandwidth.

The MFCC coefficients are calculated as,

$$\mathbf{C_j} = \sum_{i=1}^{M} \mathbf{X_i}.\cos\left(j.(i-1/2).\frac{\pi}{M} \right) \text{ for } j=1, 2..., J. \quad \textbf{(5.7)}$$

Where M is the number of filters in the filter bank, J is the number of cepstral coefficients $\boldsymbol{X}_i$ is formulated as the log-energy output of the i^{th} filter is given as,

$$\mathbf{X_i} = \log_{10}\left(\sum_{k=0}^{N-1} |\mathbf{X(k)}|.\mathbf{H_i(k)} \right) \text{ for i=1,2,….,M} \quad \textbf{(5.8)}$$

5.3.2 Experimental evaluation of the MFCC variants: FB-20, FB-24 and FB-40

The MFCC variants FB-20 [163], FB-24 [189] and FB-40 [190] were evaluated on the 2004 NIST corpus using the UBM modeling based on the data from NIST 2004. The new ITEM-GMM method introduced in Chapter 4 was applied to train the target speakers and the UMB for the text-independent speaker verification system.

The experimental rules described in the 2004 NIST SRE plan [141] were followed to conduct the experiments for the MFCC variants, and only the core test as defined in [141] were performed.

In principle the MFCC FB 20 should use 20 filters, the MFCC FB 24 should use 24 filters and the MFCC FB 40 should use 40 filters. Since the NIST 2004 corpora had speech bandwidth of 4 KHz, some of the filters falling beyond this range could not be implemented. For example to conduct the experiments with the MFCC FB-20, nineteen filters are used, ten with linearly spaced centre frequencies and nine with logarithmically spaced ones. To define the filter bank for HTK MFCC FB-24, only 20 filters were used by following the way as defined in [189]. In the experiment with the MFCC FB-40, the first 32 filters are kept, which cover the frequency range {133-3954} Hz.

The NIST 2004 SRE's 1side-1side speaker verification task had 5 minutes speech available for training of the target speakers and 5 minutes speech available for the test trials.

The UBM was created by using the training speech available with the 2001 NIST SRE corpus [139]. One hour and forty minutes of voiced speech was used for that purpose. After training, the user models were tested carrying out trials as defined in the 1side-1side speaker recognition task.

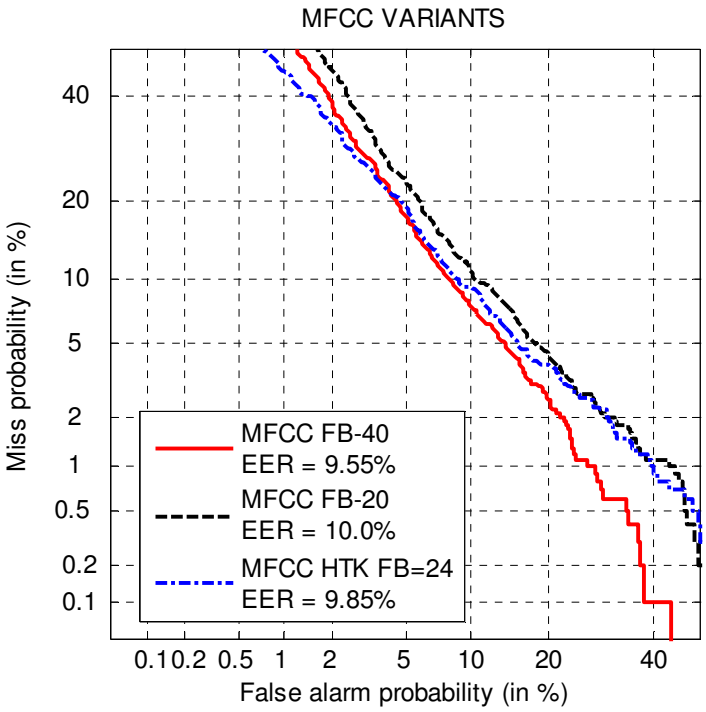


Figure 5.4 Miss probability versus false alarm probability and the equal error rates for the MFCC variants.

The miss probability versus the false alarm probability curves for the evaluated MFCC variants are presented in Figure 5.4, alongside with the corresponding equal error rates (EERs).

It can be observed that the system performance did not differ significantly when different approximations of the non-linear pitch perception of human were used. At the same time it can be noted that regardless of the specific filter bank design the larger is the number of filters, the better are the speaker recognition rates. Beside the number of filters in the filter bank, the larger amount of overlapping between the neighboring filters can also improve the results.

5.4 Inverse MFCC (IMFCC)

The MFCCs have been successfully used as a characteristic features in many speaker recognition applications. However, as the results in Section 5.3.2 showed, the selection of the number of filters in the filter bank design and the overlap between filters can enhance the performance of the speaker recognition applications.

The filter bank used in the MFCC FB-20 procedure captures vocal tract characteristics more effectively within the lower frequency regions.

In this section, a set of features which uses a complementary filter bank structure called the inversed MFCC (IMFCC) is evaluated in the speaker verification task.

The IMFCC introduced in [35] extract the speaker specific cues present within the higher frequency regions. Unlike high level features [32,33,34,192] that are often difficult to extract, the IMFCC offer computational simplicity during the extraction process.

The calculation steps for the IMFCCs are almost identical to the steps involved in the calculation of MFCCs. The only difference lies with the formulas used to calculate the filter bank structure [35].

The IMFCC method inverts the filter bank structure used in the MFCC method such that the lower frequencies are averaged by using small number of widely spaced filters and the higher frequencies are averaged by using narrower spacing of filters as shown in Figure 5.5. Thus, the IMFCC effectively capture information available at the high frequency formants which is ignored by the MFCC.

The frequency range considered for the speaker recognition is between 100-3900Hz, thus the reversed Mel scale can be obtained by flipping the filter bank at the point f=2kHz. The flipping of the filter bank can be expressed mathematically as,

$$\hat{\mathbf{H}}_{\mathbf{i}}(\mathbf{k}) = \mathbf{H}_{M+1-i}(\frac{N}{2}+1-k) \qquad \textbf{(5.9)}$$

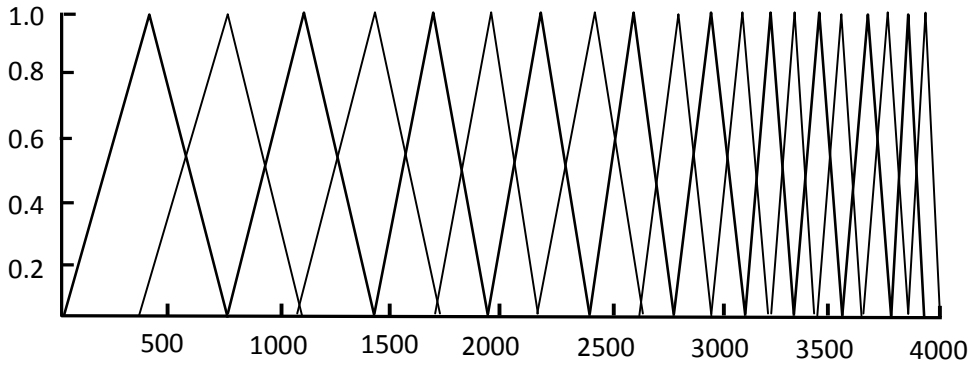


Figure 5.5 Structure of the filters for the inversed mel scale.

Where $\hat{\mathbf{H}}_{\mathbf{i}}(\mathbf{k})$ is the response of reversed filter bank. The inverse mel-frequency as evaluated in *appendix A* is given by,

$$\hat{f}_{mel}(f) = 2195.2860 - 2595\log_{10}\left(1+\frac{4031.25-f}{700}\right) \qquad \textbf{(5.10)}$$

Where $\hat{f}_{mel}$ is the inverted mel scale pitch value in mels.

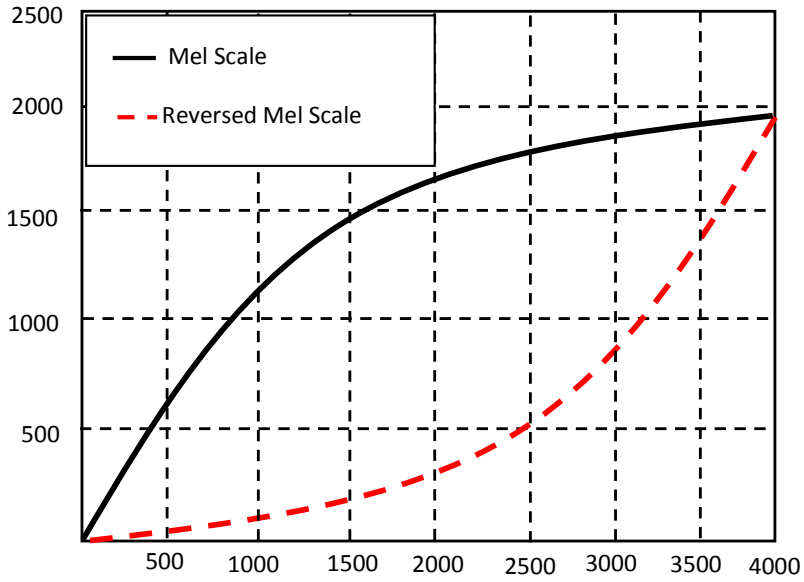


Figure 5.6 The mel scale (red line) and the inversed mel scale (black line); adapted form [35].

The mel frequency cepstral coefficients method established a way of transforming a physically measured spectrum of speech into a perceptually meaningful subjective spectrum based on the human auditory system [193] with low resolution at high frequency ranges. However the new reversed mel scale shown in Figure 5.6 provides a complimentary structure capturing high frequency formants with higher accuracy than the mel scale. Figure 5.6 shows that for the mel scale, pitch increases less rapidly as the frequency increases, on the other hand, for the inverted mel scale pitch values increase rapidly with frequency.

5.4.1 Experimental evaluation of the feature level MFCC/IMFCCs fusion

In this section a new MFCC/IMFCC fusion strategy, at the feature level is proposed and tested in speaker verification applications using NIST SRE 2004 speech corpus and the ITEM-GMM classifier. The idea behind the fusion was to capture formant characteristics at both low and high frequency ranges.

In [35] a parallel implementation of the MFCC and IMFCC was tested. However, the integration of MFCC and IMFCC was performed not at the feature's level but at the classifier level. Speaker models were generated separately for the MFCCs and for the IMFCCs were classified using the classical GMM which provided two classification scores: S_{MFCC} and S_{IMFCC}. The classification decision was then made using the following weighted sum of individual classification scores:

$$S_{com} = \lambda S_{MFCC} + (1-\lambda) S_{IMFCC} \quad \textbf{(5.11)}$$

Where the weight λ was chosen to a constant value of 0.5 as detailed in [32]. However, more suitable weights can be investigated further to enhance the performance of the combined system. The value of The speaker recognition results for MFCC/IMFCC conducted on the YOHO (microphone speech) data and POLYCOST (telephone speech) with the classical GMM as a classifier showed significant improvement of the classification rates when compared to the MFCC and IMFCC features used alone.

This section introduces a different approach to combining the MFCCs with the IMFCCs. Instead of deriving separate sets of speaker models for MFCCs and IMFCCs, and then making a decision at the classification level, these two types of feature parameters are

concatenated into combined feature vectors and these concatenated vectors are used to generate speaker models. Speakers are then verified based on the classification score obtained for the concatenated feature vectors.

Since the MFCC features provide speaker characteristics with fine resolution at the low frequency range and the IMFCC, on the other hand provide features with high resolution at the high frequency range, the proposed combined feature vectors have the advantage of a uniform resolution across the entire frequency band.

The difference between the MFCC/IMFCC features and features calculated over constant-width bands across all frequencies is that, the MFCC/IMFCC approach maintains the nonlinear human auditory characteristics.

The new MFCC/IMFCC feature vectors were tested in the context of speaker verification. The tests aimed to compare the fused MFCC/IMFCC feature vectors with the MFCC and IMFCC features used alone.

The general framework of the speaker verification system used to conduct the experiments was same as described in Section 4.5.1 and Figure 4.5. The system was designed to operate in one of the three possible modes: universal background model (UBM) training mode, target speaker enrollment mode and testing mode.

After the pre-processing of the speech signals, the feature extraction was performed based on the short-time window analysis with speech frames of length 20ms and 10 ms step size. For each frame 12 MFCC coefficients and 12 IMFCC coefficients were calculated and used to generate the following feature vectors: 12-dimensional MFCC feature vector, 12-dimensional IMFCC feature vector, 24-dimensional MFCC/IMFCC fused feature vector and 38-dimensional MFCC+ΔMFCC (first derivative of MFCC)+ΔΔMFCC (second derivative of MFCC) +E (average spectral energy)+Z (number of zero crossings

for the speech time waveform) baseline feature vector. The modeling and testing was based on the new ITEM algorithm described in Chapter 4.

The number of Gaussian mixtures used to model a target speaker was 1024. The speech corpus used for training and testing was NIST 2004. Each speaker was represented by speech utterances of the total approximate length of 10 minutes of the concatenated speech obtained after silence/unvoiced removal. In the speaker enrollment (training) stage for each speaker about 5 minutes of speech was used and in the testing stage also 5 minutes of speech was used (available utterances for 1side-1side task for NIST 2004). The training and testing sets contained mutually exclusive sets of speakers.

After the target speaker's enrolment, the universal background model (UBM) parameter's inference was accomplished using a large corpus of speech containing only the non-target speakers (speakers not used in the enrolment and testing stages). To generate the UBM parameters speech recordings of the total length of 1 hour from the NIST 2001 were used.

The results illustrated in Figure 5.6 show the performance of MFCC, IMFCC, MFCC/IMFCC and ΔMFCC (MFCC+ΔMFCC+ΔΔMFCC+Energy+Zero-crossings) features.

It can be noted that the 38-dimensional feature vectors containing the ΔMFCC parameters provided the best overall performance. The MFCC showed better performance than IMFCC, however the MFCC/IMFCC fusion outperformed both, the MFCC and the IMFCC features alone.

These results indicate that the speaker characteristic information is present in both low and high frequency ranges. Although the MFCC with their high resolution at the low frequency range provide relatively good speaker verification rates when applied on their

own, the addition of IMFCC with high resolution at the high frequencies helps to improve the verification results.

As illustrated in Figure 5.7, the equal error rate (EER) based on the IMFCC is only 1.10% lower than for the MFCC. The 24-dimensional MFCC/IMFCC fusion shows a significant improvement of the EER value by 3.9% compared to ERR for the 12-dimensional MFCC feature vectors. The 38-dimensional ΔMFCC features show further improvement of 0.25%, compared to the MFCC/IMFCC, however this is achieved at the cost of much higher dimensionality of the feature vectors and therefore much higher computational complexity.

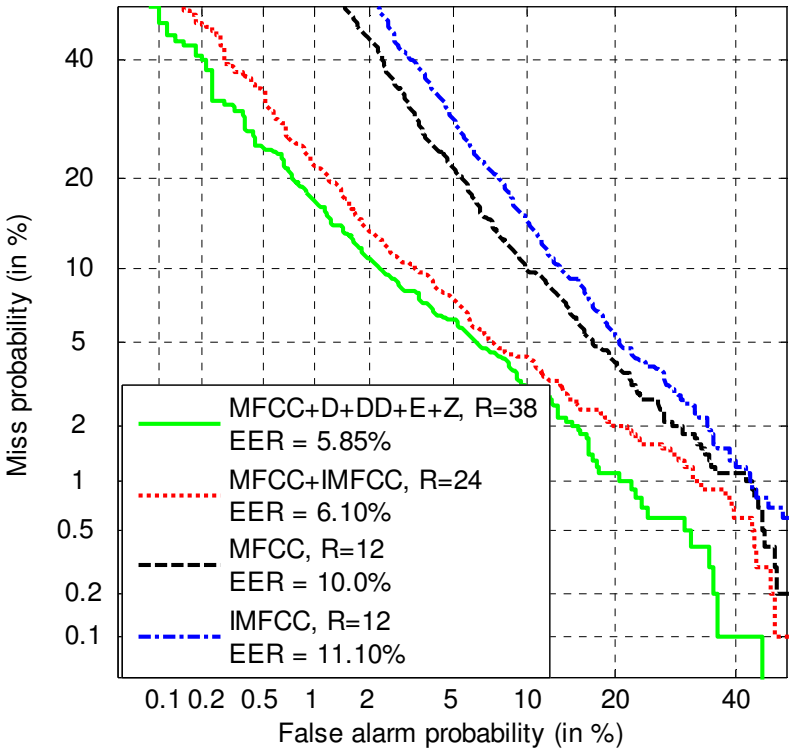


Figure 5.7 Miss probability versus false alarm probability and the equal error rates (EER) for MFCC, IMFCC, MFCC/IMFCC fusion and MFCC+Δ+ΔΔ+E+Z (ΔMFCC).

Recent laryngological experiments [194] on animals demonstrated that there is a strong correlation between the symmetry of the vocal folds vibration and the acoustic energy in the higher frequencies. In vocal folds showing periodic and symmetric motion, the energy of the high frequency spectral components was larger than in vocal folds showing an asymmetric motion. Assuming that these observations apply to humans, it can be speculated that different speakers differ in the viscosity and elasticity of their vocal folds and the symmetry of the vocal vibration. Acoustically these differences would be evident in the distribution of the spectral energy of speech. In this case the energy at high frequency harmonic components can be expected to provide vital speaker-specific information missed by the MFCC due to their coarse division of the high frequency range.

5.5 Features based on the Teager energy operator (TEO)

In this section characteristic features based on the parameter called the Teager energy operator (TEO) are described and tested in the context of speaker verification. Unlike MFCC parameters which are derived from the linear model of speech production, the theory behind the TEO parameter assumes a non-linear model of speech production. The following sections explain briefly the classical model and provide introduction to the more recent non-linear concepts of the air flow occurring during the speech phonation process. This introduction leads to the definition of the TEO based features. Finally speaker verification test and results based on the TEO features are presented.

5.5.1 Linear model of speech production

The majority of feature vectors such used the speaker verification process represent acoustic speech parameters derived from the classical source-filter model of speech

production [196]. This includes parameters such as the linear predictive coefficients (LPC), the linear predictive cepstral coefficients (LPCC) and the mel frequency cepstral coefficients (MFCC).

The classical source-filter theory of voice production assumes that the air flow through the vocal folds (source) and the vocal tract (filter) is unidirectional and has a laminar character. During phonation, the vocal folds vibrate. One vibration cycle includes the opening and closing phases in which the vocal folds are moving apart or together, respectively. The number of cycles per second determines the frequency of the vibration, which is subjectively perceived as pitch or objectively measured as the fundamental frequency F_0. The sound is then modulated by the vocal tract configuration and the resonant frequencies of the vocal tract are known as formants.

The speaker verification process assumes that certain speaker-specific characteristics have an effect on the acoustic parameters of the source-filter model. Therefore statistical modeling of these parameters can be used to derive speaker models used in the speaker verification/recognition process.

5.5.2 Nonlinear model of speech production

In his pioneering work, Teager [197] indicated the importance of the energy measures in speech analysis. His experimental studies [197,203,204], pointed to the fact that in addition to the laminar air flow during the speech phonation, certain non-linear and turbulent phenomena can be observed in the form of the supra-glottal air vortices. These vortices are formed above the vocal folds as shown in Figure 5.8. Teager's suggestions were further supported by the air flow modeling [248] and simulation experiments [249].

As indicated in [250], the supra-glottal vortices have the potential to generate sounds when heating hard surfaces of the vocal tract or interacting with each other. In the study of stress classification in speech Zhou *et. al.* [250] proposed new type of features based on the Teager energy operator (TEO) and called the TEO-Auto-Env-Area. The high performance of the TEO-Auto-Env-Area was attributed to the fact that this parameter is highly sensitive to the presence of additional harmonics and cross-harmonics in speech. It was assumed that in the state of stress, supra-glottal vortices are formed providing additional harmonic components. Different levels of stress would results in different energies and frequencies of these harmonics.

In [247], a modified version of the TEO-Auto-Env-Area called the TEO-PWP-Auto-Env was introduced and tested providing very good results in the stress and emotion recognition in speech. In [246] another TEO based parameter TMFCC was successfully applied to the anger recognition in speech.

The following sections explain the theory behind different TEO based feature parameters. This is followed by sections describing application of the various TEO based features to the speaker verification process.

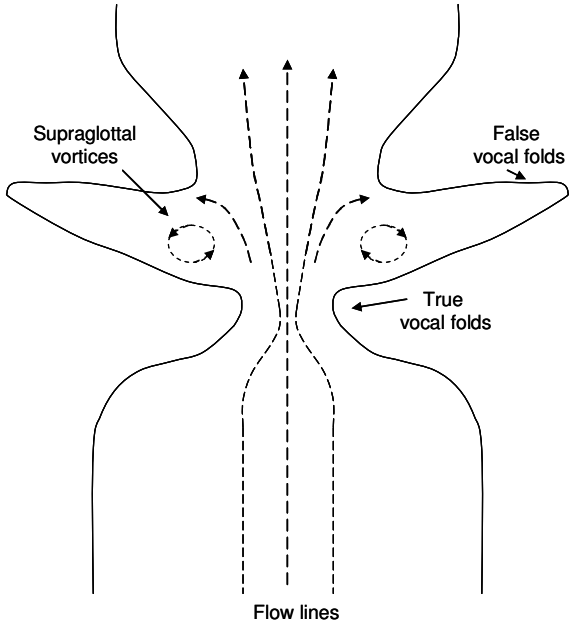


Figure 5.8 Nonlinear model of sound propagation along the vocal tract [274].

5.5.3 Teager energy operator (TEO)

In the light of the recent non-linear models of speech production, the speech signal could be regarded as an effect of amplitude and frequency modulation of separate oscillatory waves and modeled as a combination of several amplitude and frequency modulated (AM-FM) oscillatory components. Maragos [209,210] proposed a nonlinear model of speech, which represents a discrete-time speech signal $s[n]$ as a sum of M components,

$$s[n] = \sum_{i=1}^{M} x_i[n] \qquad \textbf{(5.12)}$$

Each component $x[n]$ of speech can be modeled as an AM-FM sine wave given in the discrete time domain as,

$$x[n] = a[n]\cos(\Phi[n]) = a[n]\cos\left(\omega_c n + \omega_h \int_0^n q[k]dk + \theta\right) \quad \textbf{(5.13)}$$

Where $q[k]$ is the modulating signal, ω_c is the source frequency (carrier), $\omega_h \in [0;\omega_c]$, is the maximum frequency deviation, θ is a constant phase offset, and $a[n]$ is the instantaneous amplitude.

Eq. (5.13) can be also expressed in the continuous time domain as:

$$x(t) = a(t)\cos(\Phi(t)) = a(t)\cos\left(\int_0^t \omega(\tau)d\tau + \varphi(0)\right) \quad \textbf{(5.14)}$$

Where a(t) is the time varying instantaneous amplitude, and $\omega(t) = \frac{d\Phi(t)}{dt}$ is the instantaneous frequency.

Assuming the above AM-FM modulation of speech, Kaiser and Teager [205,245] proposed the following estimate of the speech instantaneous energy known as the Teager energy operator (TEO):

$$\Psi[x(t)] = \left[\frac{dx(t)}{dt}\right] - x(t)\frac{d^2x(t)}{dt^2} \quad \textbf{(5.15)}$$

In the discrete-time domain Eq. (5.15) becomes:

$$\Psi\left(x[n]\right) = x^2[n] - x[n+1]x[n-1] \quad \textbf{(5.16)}$$

Applying Eq. (5.15) to the AM-FM speech signal given in Eq. (5.14), the following formula can be derived [211]:

$$\Psi[x(t)] \approx \left[a(t) \frac{d\varphi(t)}{dt} \right]^2 \quad \textbf{(5.17)}$$

Eq. (5.17) shows that TEO can track the modulation energy and identify the instantaneous amplitude and frequency.

5.5.4 TMFCC

In this section a feature extraction method called the TMFCC based on the Teager energy operator for the speaker verification applications is described.

Application of the TMFCC features showed promising results in anger detection [246], and stress classification [196], language recognition [211], and speech enhancement in the presence of noise [212,213].

The idea of using TEO instead of the commonly used averaged energy is to take advantage of the instantaneous amplitude and frequency tracking capability of the TEO. This leads to a better representation of formant information in the feature vector than MFCC [212].

As described in Section 5.3.1, computation of the MFCC parameters involves the mapping of the speech spectrum into the mel frequency scale. Figure 5.9 illustrates computational steps involved in the calculation of the Teager MFCC parameters, which are similar to steps involved in the calculation of the MFCC parameters.

As illustrated in Figure 5.9, the TMFCC differ from the traditional MFCC in the definition of energy measure, i.e., MFCC employs energy in frequency domain (due to Parseval's equivalence) at each sub-band whereas TMFCC employs Teager energy in time domain and determines the spectrum.

The mapping of the speech spectrum is achieved the by multiplying the magnitude of speech spectrum for a preprocessed frame by magnitude of triangular filters in Mel filter bank followed by log compression of the whole bandwidth and finally DCT [211].

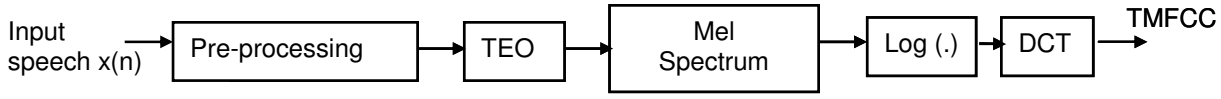


Figure 5.9 Calculation of the TMFCC parameters.

5.5.5 TEO-PWP-Auto-Env

This section describes another TEO based type of features called the TEO-PWP-Auto-Env. The TEO-PWP-Auto-Env features were introduced in [247] and successfully applied to the stress recognition in speech. In this chapter TEO-PWP-Auto-Env were for the first time applied to the speaker verification problem.

The human auditory system is assumed to perform a filtering operation which partitions the entire range of the audible frequencies into critical bands [214,215] listed in Table 5.1. The width of these bands increases logarithmically with the frequency. Observations of the changes in the numbers of harmonics within critical bands can provide cues for the recognition of stress in speech.

Based on this assumption the TEO-PWP-Auto-Env feature extraction process employs approximations of the critical bands to filter the speech signal followed by the TEO processing as shown in the Figure 5.10.

The critical bands are approximated using the perceptual wavelet packet analysis [247]. The perceptual wavelet packets bands have the advantage of having much faster roll off rates for the side lobes and less oscillations in the pass band than the band-pass filters designed using standard Matlab procedures.

The steps involved in the calculation of the TEO-PWP-Auto-Env features are illustrated in Figure 5.10. After the voiced/unvoiced detection, the voiced speech is filtered using a bank of Perceptual Wavelet Packet (PWP) filters listed in Table 5.2. The filters centre frequencies were set to the centre frequencies of the critical bands, and effective RMS bandwidth of each filter was set to the width of the corresponding critical band. For each band the Teager Energy Operator and the area under the normalized TEO autocorrelation envelope were calculated.

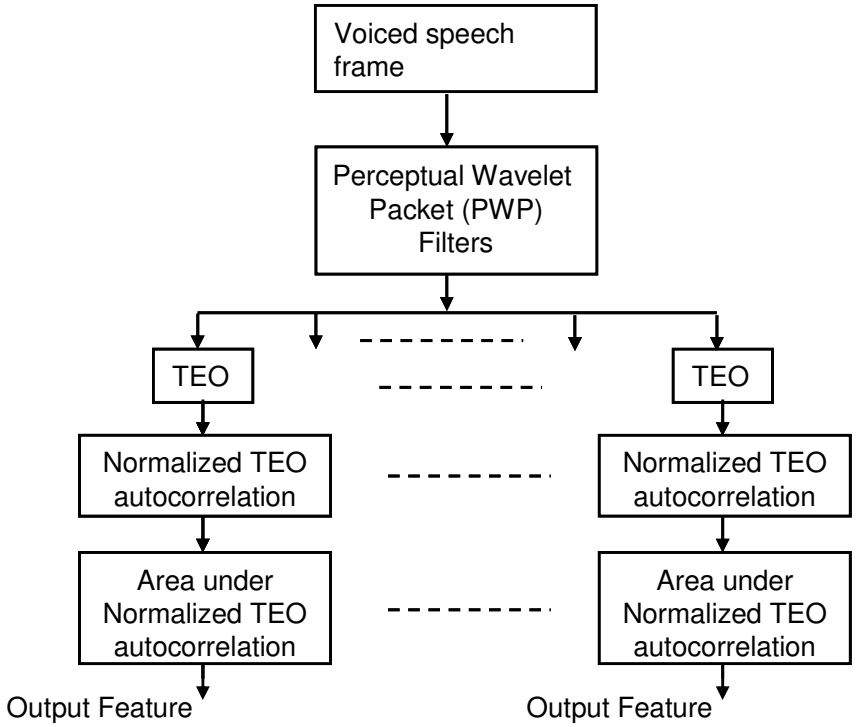


Figure 5.10 Flowchart of the TEO-based feature extraction process. Adapted from [247].

The Wavelet Packet (WP) analysis is a modified form of the Discrete Wavelet Transform where the signal is passed iteratively through a larger number of filters than in DWT. In the PWP analysis each decomposition level is calculated by passing the previous approximation coefficients though a high and low pass filters (see Figure 5.11).

The WP analysis provided high resolution at both low and high frequency ranges. The TEO-PWP-Auto-Env features were calculated for the outputs from 17 bands. The corresponding frequency ranges of these bands are listed in Table 5.1. The detailed mathematical description of the wavelet packet theory is given in [216,217].

In the speaker verification experiments, the TEO-PWP-Auto-Env feature extraction was performed with mother wavelet db2.

Table 5.2 The PWP and critical bands (CB) under 4 kHz. Adapted from [247].

Band	Perceptual Wavelet Packet		
	Lower	Upper	Bandwidth
1	0	125	125
2	125	250	125
3	250	375	125
4	375	500	125
5	500	625	125
6	625	750	125
7	750	875	125
8	875	1000	125
9	1000	1250	250
10	1250	1500	250
11	1500	1750	250
12	1750	2000	250
13	2000	2250	250
14	2250	2500	250
15	2500	3000	500
16	3000	3500	500
17	3500	4000	500

Band	Critical Band Frequency			
	Lower	Centre	Upper	Bandwidth
1	100	150	200	100
2	200	250	300	100
3	300	350	400	100
4	400	450	510	110
5	510	570	630	120
6	630	700	770	140
7	770	840	920	150
8	920	1000	1080	160
9	1080	1170	1270	190
10	1270	1370	1480	210
11	1480	1600	1720	240
12	1720	1850	2000	280
13	2000	2150	2320	320
14	2320	2500	2700	380
15	2700	2900	3150	450
16	3150	3400	3700	550

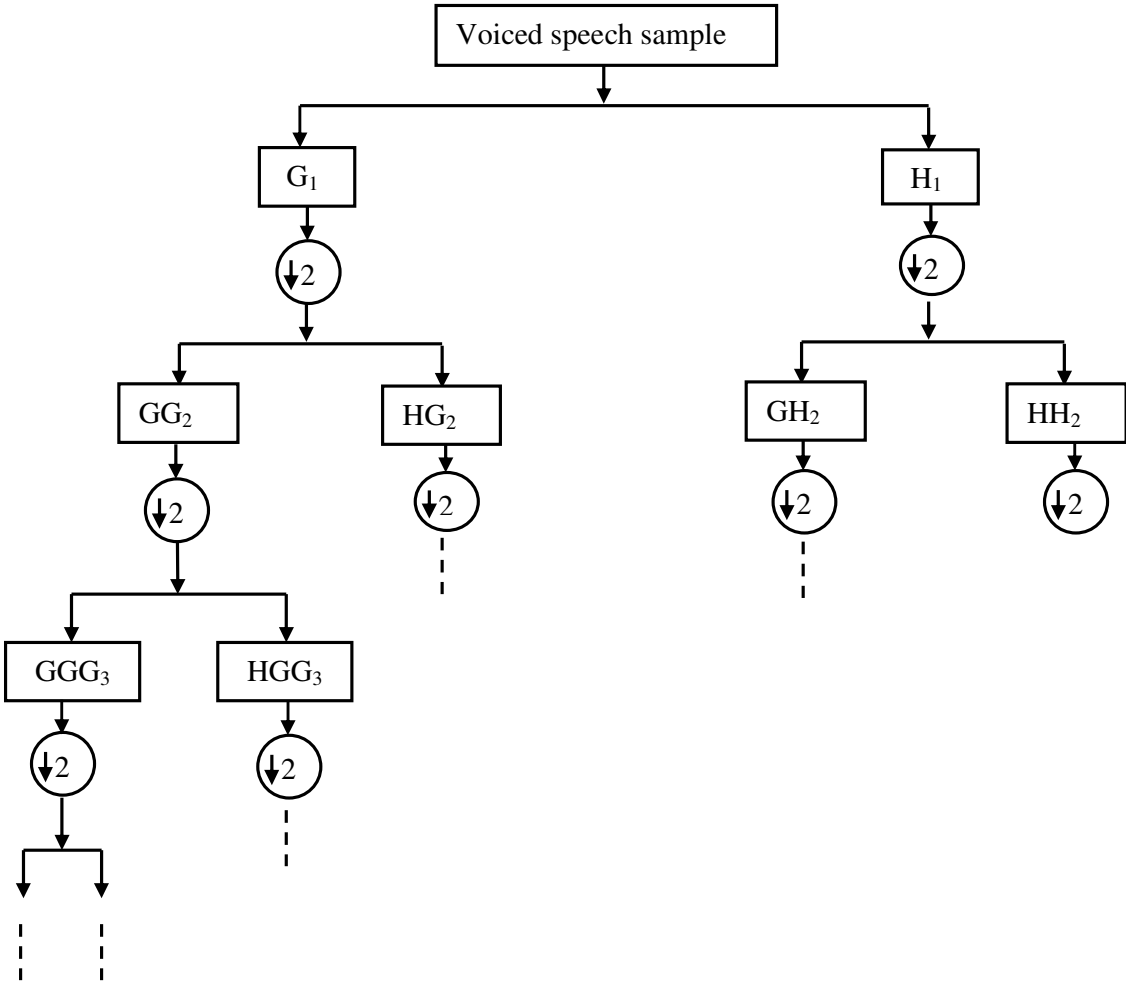


Figure 5.11 The wavelet packet (WP) decomposition tree; G-low pass filters, H-high pass filters [275].

The TEO-PWP-Auto-Env features were calculated for the outputs from the following filters (see Figure 5.11): $GGGGG_5$, $HGGGG_5$, $GHGGG_5$, $HHGGG_5$, $GGHGG_5$, $HGHGG_5$, $GHHGG_5$, $HHHGG_5$, $GHGG_4$, $HGHG_4$, $GHHG_4$, $HHHG_4$, $GGGH_4$, $HGGH_4$, HGH_3, GHH_3, HHH_3.

If a speech frame contains only a single harmonic with constant instantaneous amplitude $a[n]$ and constant instantaneous frequency $\omega_i[n]$, then Eq. (5.15) indicates that the

corresponding TEO profile is a constant number. If the signal frame contains more harmonics then the TEO profile $\Psi()$ changes as a function of n.

In reality, speech signals always contain a number of harmonic components. If there is only one excitation source with the fundamental frequency F0, then there will be a whole harmonic series of integer multiples of F0. Additional excitation sources (vortices) will generate their own harmonic series. High pitch values will generate smaller numbers of harmonic components, and low pitch values will generate larger numbers of harmonic components within a speech bandwidth.

As suggested in [196], if the speech signal is broken into small bands, and the TEO is calculated for each band, it is easier to observe the presence or absence of harmonic component within each band. Moreover, the speech analysis becomes more robust if the characteristic features are derived not directly from the TEO but from the normalized TEO autocorrelation function $R_{\Psi(x)}[k]$ given as,

$$R_{\Psi(x)}[k] = \frac{1}{2M+1}\sum_{n=-M}^{M}\Psi(x[n])\Psi(x[n+k]) \qquad \textbf{(5.18)}$$

Where M is the number of samples within the analyzed speech frame.

The normalized TEO autocorrelation function reflects the same trends as the TEO profile itself but it is less sensitive to sudden changes in TEO values. In general, the TEO profile may contain multiple harmonics and cross-harmonic terms causing very rapid changes in the TEO profile. The normalized TEO autocorrelation function can suppress some of these changes while still maintaining fluctuations which are due to changes in the vortex formation patterns. These changes could be attributed to different individual characteristics of speakers, different stress levels or emotions.

In the simplest case of a single harmonic with a constant instantaneous amplitude and constant instantaneous frequency, the normalized TEO autocorrelation function of Eq. (5.18) will produce a straight line decaying from the point (0,1) to the point (N,0), where N is the number of samples in the analyzed speech frame. The area under the autocorrelation line in this case, will be equal to N/2. If the analyzed speech frame contains more harmonic components, the normalized autocorrelation function will produce a time varying contour decaying in an oscillatory way to zero. The area under the autocorrelation contour in this case will be less than N/2. The area under the normalized TEO autocorrelation contour can be used as an indicator of changes in the harmonic components of speech due to conditions such a stress or emotion.

As indicated in Figure 5.10, the values of the area under the normalized TEO autocorrelation contour were calculated on the frame-by-frame basis for each of the analyzed frequency bands, and used as characteristic features in the speaker verification experiments.

5.5.6 Speaker verification experiments using TEO based features

In this section, the TEO based features including TMFCC and TEO-PWP-Auto-Env were for the first time applied to the speaker verification problem. The idea behind these tests was to determine if the TEO based features can efficiently capture speaker's characteristics.

The theory behind the nonlinear speech production model presented in Section 5.5.2 lead to an assumption that speakers-dependent differences in elasticity, and symmetry of the vocal folds can generate to different patterns of vortex formation during speech production. These differences would result in variations of harmonic components of speech and could be detected by changes in the TEO based parameters.

The general framework of the speaker verification system used to conduct the experiments was the same as described in Section 4.5.1 and Figure 4.5. The system was designed to operate in one of the three possible modes: universal background model (UBM) training mode, target speaker enrollment mode and speaker testing mode.

After the pre-processing of the speech signals which removed the silence and unvoiced intervals, the feature extraction was performed based on the short-time window analysis with speech frames of length 20ms and 10 ms step size.

In the case of the TMFCC features, for each frame of voiced speech, 12 TMFCC coefficients and 12 MFCC coefficients were calculated and used to generate the following feature vectors: 12-dimensional TMFCC feature vector, 12-dimensional MFCC feature vector, 24-dimensional MFCC/TMFCC fused feature vector and 38-dimensional MFCC+ΔMFCC (first derivative of MFCC)+ΔΔMFCC (second derivative of MFCC) +E (average spectral energy)+Z (number of zero crossings for the speech time waveform) baseline feature vector. The modeling and testing was based on the new ITEM algorithm described in Chapter 4.

In the case of the TEO-PWP-Auto-Env features, for each frame of voiced speech, 17 TEO-PWP-Auto-Env coefficients and 12 MFCC coefficients were calculated and used to generate the following feature vectors: 17-dimensional TEO-PWP-Auto-Env feature vector, 12-dimensional MFCC feature vector, 29-dimensional MFCC/TEO-PWP-Auto-Env fused feature vector and 38-dimensional MFCC+ΔMFCC (first derivative of MFCC)+ΔΔMFCC (second derivative of MFCC) +E (average spectral energy)+Z (number of zero crossings for the speech time waveform) baseline feature vector. The modeling and testing was based on the new ITEM algorithm described in Chapter 4.

The number of Gaussian mixtures used to model a target speaker was 1024. The speech corpus used for training and testing was NIST 2004. Each speaker was represented by

speech utterances of the total approximate length of 10 minutes of the concatenated speech obtained after silence/unvoiced removal. In the speaker enrollment (training) stage for each speaker about 5 minutes of speech was used and in the testing stage also 5 minutes of speech was used (available utterances for 1side-1side task for NIST 2004). The training and testing sets contained mutually exclusive sets of speakers.

After the target speaker's enrolment, the universal background model (UBM) parameter's inference was accomplished using a large corpus of speech containing only the non-target speakers (speakers not used in the enrolment and testing stages). To generate the UBM parameters speech recordings of the total length of 1 hour from the NIST 2001 were used.

The results in Figure 5.12 demonstrate the performance of TMFCC, MFCC, MFCC/TMFCC and ΔMFCC (MFCC+ΔMFCC+ΔΔMFCC+Energy+Zero-crossings) features.

It can be observed that the 24-dimensional feature vectors containing a fusion of MFCC/TMFCC provided the best overall performance. The ΔMFCC parameters outperformed both MFCC and TMFCC used alone, however the TMFCC futures, showed better performance than the MFCC (an EER improvement of 0.95% is observed).

These results indicate that the nonlinear TMFCC parameters based on the Teager energy operator, which are sensitive to instantaneous changes in the signal energy and frequency, are more effective in capturing speaker-characteristic changes in the distribution of the spectral energy then the linear MFCC parameters based on averaged values of the spectral energies. The addition of derivatives to the MFCC features (in ΔMFCC) is not as efficient in detecting the vital energy and frequency changes as addition of the TMFCC parameters (in MFCC/TMFCC fusion).

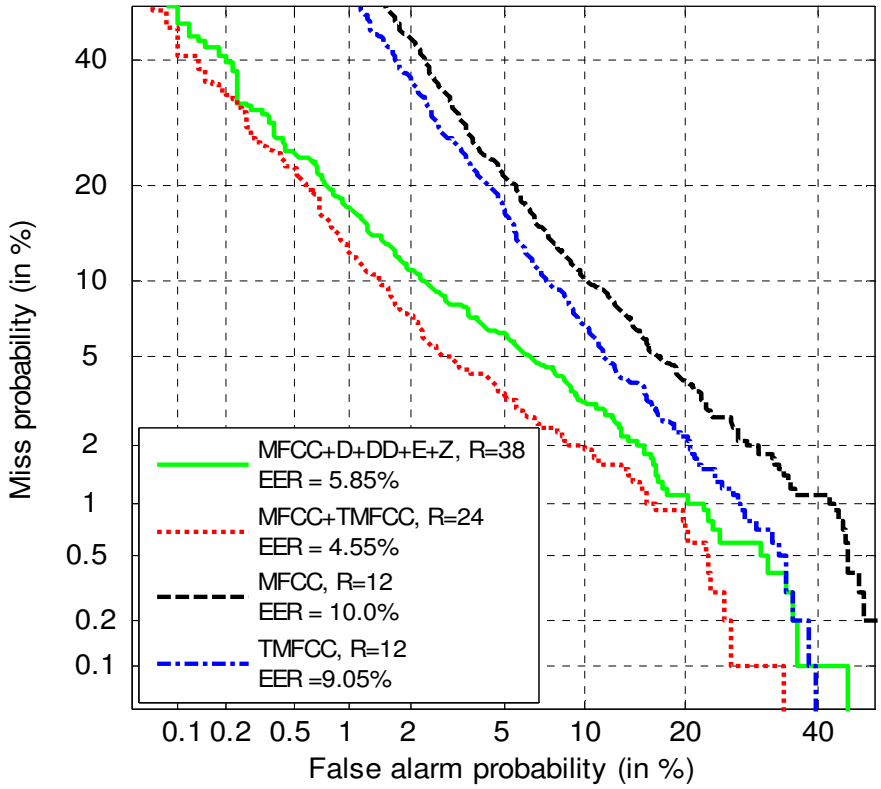


Figure 5.12 Miss probability versus false alarm probability and the equal error rates for the MFCC, TMFCC and the MFCC/TMFCC fusion. The R values indicate the dimensions of feature vectors.

Figure 5.13 shows the results obtained for the TEO-PWP-Auto-Env, MFCC, MFCC/TEO-PWP-Auto-Env fusion and ΔMFCC features.

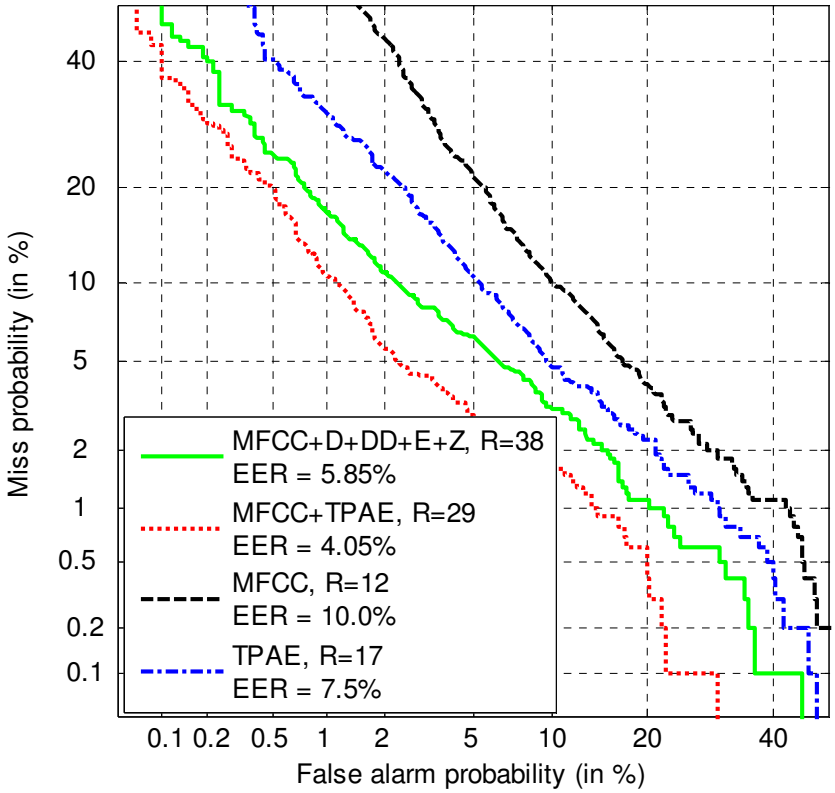


Figure 5.13 Miss probability versus false alarm probability and the equal error rates for the TEO-PWP-Auto-Env (TPAE) features. The R values indicate the dimensions of feature vectors.

It can be observed in Figure 5.13, that the 29-dimesional MFCC/TEO-PWP-Auto-Env fusion provides the best overall performance. The ΔMFCC parameters outperformed both MFCC and TEO-PWP-Auto-Env used alone, however the TEO-PWP-Auto-Env features, showed better performance than the MFCC (an EER improvement of 2.5% is observed).

These results show very consistent trend with the results presented in Figure 5.12. Again, the nonlinear TEO-PWP-Auto-Env parameters based on the Teager energy operator are more effective in capturing speaker-characteristic changes in the distribution of the spectral energy then the linear MFCC parameters. Moreover, the addition of derivatives to the MFCC features (in ΔMFCC) is not as efficient in detecting the vital energy and

frequency changes as addition of the TMFCC parameters (in MFCC/ TEO-PWP-Auto-Env fusion).

When comparing the results in Figures 5.12 and 5.13, it can be noted that the TEO-PWP-Auto-Env (ERR=7.50%) clearly outperform the TMFCC (ERR=9.05%). Similarly, the MFCC/ TEO-PWP-Auto-Env (ERR=4.05%) works slightly better than the MFCC/TMFCC fusion (ERR=4.5%). This can be attributed to the fact that the TMFCC parameters are based on the critical band sub-division of the speech bandwidth, which gives lower frequency resolution at the high frequencies (see Table 5.2). The TEO-PWP-Auto-Env parameters, on the other hand use more evenly distributed frequency resolution across both low and high frequency bands. These results again indicate the importance of including low as well as high frequency speaker-dependent information in the feature vectors.

5.6 Summary

A number of feature vectors based on the classical linear model of speech production as well as features based on the recent nonlinear models were described and applied to the speaker verification task.

Table 5.3 summarizes the performance of the tested features by showing the values of the equal error rates (EER) produced by these features.

It can be observed that when used alone the non-linear features including TMFCC and TEO-PWP-Auto-Env show stronger performance (lower EER values) than the classical nonlinear features including MFCC and IMFCC.

The best overall performance was achieved by the proposed new fusions of linear features (MFCC/IMFCC) and linear/nonlinear features: MFCC/ TEO-PWP-Auto-Env and MFCC/TMFCC. It is likely that these combinations of features contain vital complimentary information about speaker characteristics.

The strong performance of the TEO based nonlinear features is consistent with the theories describing speech as a linear combination of AM-FM signals with the speaker-specific information included in the instantaneous changes of the signal amplitude and frequency.

Since the TEO based features are sensitive to the presence of additional harmonics and cross-harmonics in the speech signal, their high performance is consistent with the recent models of speech productions assuming nonlinear air flow and generation of vortices providing additional sound sources during the phonation process.

The high performance of the TEO-PWP-Auto-Env features when compared to the TMFCC features indicated an importance of including speaker-specific information from the low and high frequency ranges in the feature vectors.

Table 5.3 Summary of the linear and nonlinear feature performance in the speaker verification task based on the % equal error rates (EER).

Features	***EER (%)***
MFCC/TEO-PWP-Auto-Env	4.05
MFCC/TMFCC	4.55
ΔMFCC	5.85
MFCC/IMFCC	6.10
TEO-PWP-Auto-Env	7.50
TMFCC	9.05
MFCC	10.0
IMFCC	11.10

The new modeling technique introduced in Chapter 4, as well as the best performing feature extraction methods introduced in Chapter 5 are tested in the Chapter 6 using a clinical speech corpus, which includes the speaker's suffering with the clinical depression. Effects of the clinical environment on the speaker verification rates are determined.

CHAPTER 6

EFFECTS OF CLINICAL DEPRESSION ON AUTOMATIC SPEAKER VERIFICATION RATES

This chapter, for the first time investigated the effects of a clinical environment on the speaker verification. Speaker verification within a homogeneous environment consisting of the clinically depressed speakers was compared with the speaker verification within a neutral (control) environment containing of non-depressed speakers. Experiments based on mixed environments containing different ratios of depressed/non-depressed speakers were also conducted in order to determine how the depressed/non-depressed ratio relates to the speaker verification rates. The experiments used a clinical speech corpus consisting of 68 clinically depressed and 71 non-depressed speakers. Speaker models were built using the new ITEM-GMM method introduced in Chapter 4. Two types of feature vectors were tested, the classical ΔMFCC coefficients and the TEO-PWP-Auto-Env features. Experiments conducted within homogeneous environments showed a significant decrease of the equal error rates (EER) by 5.1% for the clinically depressed environment when compared with the non-depressed environment. Experiments conducted within mixed environments showed that an increasing number of depressed speakers lead to a logarithmic increase of the EER values; where the increase of the percentage of depressed speakers from 0% to 30% has the most profound effect on the increase of the EER. It was also demonstrated that the TEO-PWP-Auto-Env provided more robust performance in the clinical environments compare to ΔMFCC, lowering the EER from 24.1% (for ΔMFCC) to 17.1% (for TEO-PWP-Auto-Env).

6.1 Speaker Verification in Adverse Environments

This chapter aims to demonstrate the need for the development of a speech modelling approaches which take into account the dynamics of speech under adverse conditions.

It has been reported that the performance of the speaker recognition systems which assume a noise-free tranquil environment, degrades due to both intra-speaker variability as well as the background noise and channel distortion.

The effects of noise [259,260,263] and channel distortion [260,261] on the speaker verification rates have been thoroughly investigated, and a number of compensation methods for the adverse effects have been proposed [259,262]. The adverse effects of the intra-speaker variability on the other hand, received relatively small attention from researchers.

The various sources of the intra-speaker variability include: aging, health problems, emotional state, stress level, use of alcohol and drugs.

Previous psychological research suggested that intra-speaker variations in the voice can be traced in the psychological and physiological state of a speaker [225,226].

In [226] Scherer postulated that automatic speaker verification can be improved by training the algorithms on emotional speech. In [264] Scherer reported on a project to improve on current speaker verification systems by the development of phonetically informed methods of coping with intra-speaker variation due to emotion and stress.

One of the important, yet not fully investigated health factors that can have potential impact on the speaker verification rates is clinical depression.

Clinical depression belongs to the mood disorders. It is characterized by prolonged periods of sadness and social withdrawal [255]. The psychomotor retardation often associated with clinical depression is described as general slowing of body movement, mental processing, and speech production [228]. This impacts the speaking mechanisms by creating slower and monotone speech delivery. Speech contents of depressed speaker consist of more abstractive flow of conversations, higher frequency of pauses and more non verbal sounds than speech of normal speaker. Kuny and Stassen [256] indicated that clinical depression effects voice characteristics and speaking behavior. Recent experiments [228,235,236,237,252,253,254,254] demonstrated that clinical depression changes acoustic characteristics of speech to the degree that makes it possible to detect depression signs through an automatic acoustic speech analysis.

It is estimated that up to one in eight individuals will require treatment for depressive illness in their lifetime. The occurrence of depression is the world's fourth most serious health problem and it is also expected to rise linearly with the increasing age of population [265,266]. Statistics from the World Health Organization (WHO) [257] indicate that about 121 million people are recognized to as affected by depression worldwide. Depression affects almost 10% of the population, or 19 million Americans, in a given year. In Australia, about 20% of people will be affected by depression and 6% will experience a major depressive illness [258].

These large numbers of depressed people use our telephone networks and undergo various security checks using speaker verification/recognition systems. At this point in time, it is not known to what degree the depression symptoms affect the performance of these systems.

Although, the effects of clinical depression of the speaker verification/recognition rates have not been tested yet, it is likely that they are quite profound. The prevalence of depression in our society makes these tests particularly important.

This chapter for the first time investigates the effects of clinical depression on the speaker verification rates.

In Section 6.2 the clinical speech data base used in the experiments is described. Section 6.3 presents the general framework of the speaker verification system. In Section 6.4 preliminary experiments which determine the optimal number of Gaussian mixtures and the optimal testing/training sets sizes are presented. In Section 6.5 speaker experiments based on the classical ΔMFCC features are described. In Section 6.6 speaker experiments based on the TEO-PWP-Auto-Env features are described. Finally, Section 6.7 provides the chapter's summary.

6.2 Clinical Speech Corpus

A clinical speech corpus was used to investigate the effects of clinical depression on the speaker verification rates.

The clinical speech corpus was obtained as a result of research cooperation with the Oregon Research Institute (ORI). The corpus consists of speech recordings from 139 speakers including 93 females and 46 males. The speakers were 12-19 years of age. The speech was a soundtrack of video recordings (in MPEG) format made during problem-solving, event planning and family consensus discussions between family members. Potential speakers were excluded from the recording sessions if they evidenced any substance dependence or conduct disorders or if they were taking any medications that affect the cardiac system.

For the purpose of this study, the speech was used in the mono channel format and was dawn-sampled from the original sampling rate of 44.1 kHz to 8 kHz to match the

sampling rates of the NIST corpora used in the experiments described in the previous chapters. Parts of recordings which contained more than one speaker were manually removed. For each speaker, utterances of approximate length of 7 to 8 minutes were available.

Through the self-report and interview measures of depression [229], 68 speakers (49 females and 19 males) were diagnosed by psychologists from ORI, as suffering from major depressive disorder (MDD), and the remaining 71 speakers (44 females and 27 males) were diagnosed as non-depressed controls (i.e., showing no current or lifetime history of the major depressive disorder). A detailed description of the ORI database and the way it was made can be found in [230,231,232].

6.3 Speaker Verification Framework

In order to maintain consistency with the experiments described in Chapters 4 & 5, the general framework of the speaker verification system used to conduct the environmental experiments was the same as described in Section 4.5.1 and Figure 4.5. The system was designed to operate in one of the three possible modes: universal background model (UBM) training mode, target speaker enrollment mode and testing mode.

In the pre-processing stage silence and unvoiced intervals were removed using speech activity detection (SAD) procedure as defined in Chapter 2, Section 2.8. The voiced speech was concatenated providing for each speaker speech samples of an approximate length of 6 minutes. Out of these samples, about 5 minutes of speech was used for the speaker enrolment (training) and 1 min for testing.

The concatenated speech samples were used to calculate feature vectors on the frame-by-frame basis with speech frames of length 20 ms and 10 ms step size (50% overlap). For each frame, two types of feature vectors were calculated.

As a baseline, classical 38-dimensional ΔMFCC vectors consisting of MFCC+ΔMFCC (first derivative of MFCC) +ΔΔMFCC (second derivative of MFCC) +E (average spectral energy)+Z (number of zero crossings for the speech time waveform) were tested.

The second type of feature vectors was comprised of the TEO-PWP-Auto-Env parameters described in Chapter 5. The TEO-PWP-Auto-Env futures were chosen for testing in the clinical environment because it was demonstrated in Section 5.5.6 that within neutral environment they showed better performance than TMFCC and ΔMFCC parameters. It was therefore reasonable to expect that TEO-PWP-Auto-Env may show a similar level of robustness within adverse clinical environment.

After the target speaker's enrolment, the universal background model (UBM) parameters were calculated using a large corpus of speech containing only the non-target speakers (speakers not used in the enrolment and testing stages). To generate the UBM parameters speech recordings of the total length of 1 hour from the NIST 2001 and NIST 2002 were used.

For each test, three-turn cross validations were performed. Each turn was run with different randomly chosen training and testing set. The classification rate was calculated as an average value for the three turns.

To maintain consistency with experiments described in Chapters 4 and 5, the modeling and testing was based on the new ITEM algorithm introduced in Chapter 4.

6.4 Preliminary Experiments

Prior to the main tests, preliminary research has been conducted to determine:

- An optimal values of the Gaussian mixtures for the ORI data
- Optimal sizes of the training and testing data sets

6.4.1 Optimizing the number of Gaussian mixtures

In order to determine the optimal numbers of Gaussian mixtures for the two ORI data sets, one containing depressed speakers and one containing non-depressed speakers, the performance of the speaker verification system was analyzed on each set using different numbers of Gaussian mixtures.

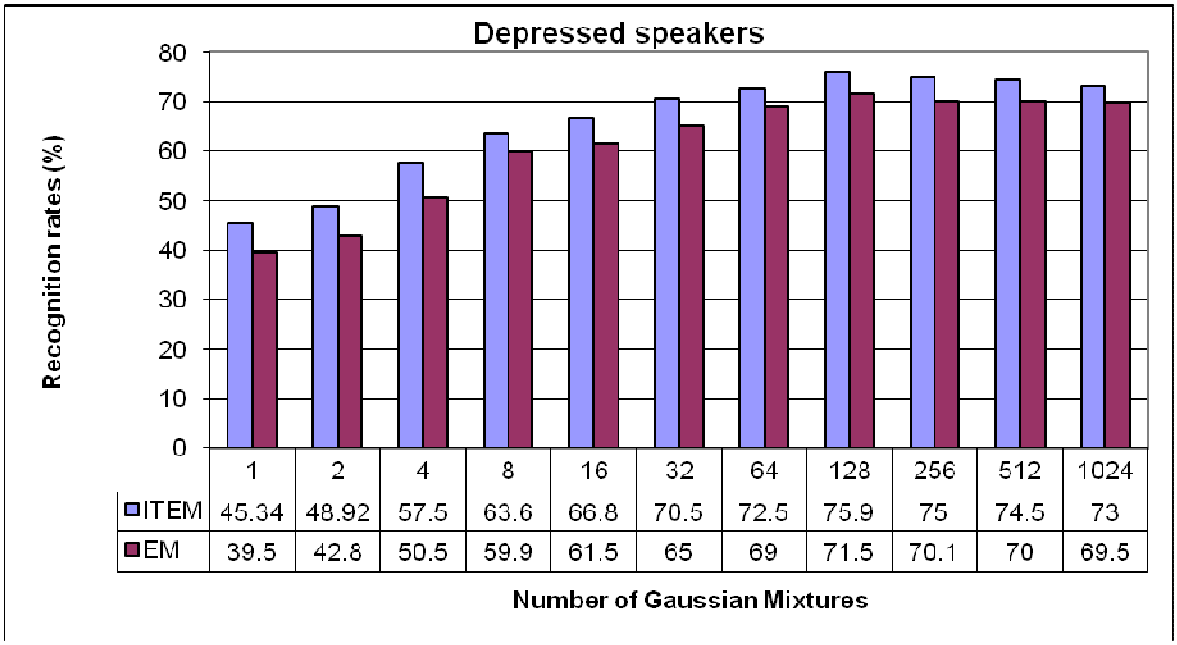


Figure 6.1 Correct recognition rates (in %) versus the number of Gaussian mixtures with GMM modeling based on the classical EM algorithm (purple bars) and the new ITEM algorithm (blue bars). Calculated for the depressed (D) speakers from the ORI data base.

While the increasing number of mixtures was generally expected to provide better classification accuracy, the associated increase of the computational complexity could make the applications impractical. Therefore, the experiments aimed to determine numbers of Gaussian mixtures which provide best compromise between the computational complexity and the classification accuracy.

To ensure consistency of observations, the tests were performed using the GMM modelling based on the new ITEM procedure for calculating the model parameters, as well as the classical EM algorithm. The ΔMFCC parameters were used as characteristic features.

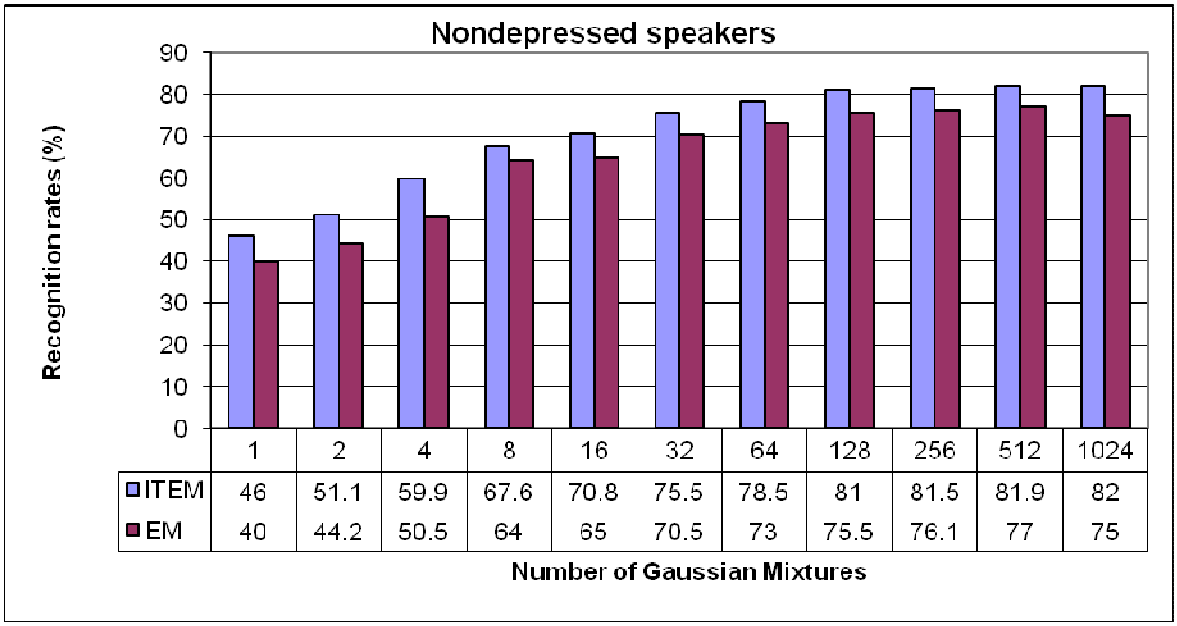


Figure 6.2 Correct recognition rates (in %) versus number of Gaussian mixtures with GMM modeling based on the classical EM algorithm (purple bars) and the new ITEM algorithm (blue bars). Calculated for the non-depressed (ND) speakers from the ORI database.

The number of Gaussian mixtures was gradually increased from 1 to 1024 with the step size of 2^n where n was equal to 1, 2, 3,…,11. For each number of Gaussian mixtures, speaker verification was performed and the correct classification rate was calculated.

Figure 6.1 shows the correct recognition rates calculated for the depressed (D) speakers from the ORI data base versus the number of Gaussian mixtures with GMM modeling based on the classical EM algorithm (purple bars) and the new ITEM algorithm (blue bars).

Similarly, Figure 6.2 shows the correct recognition rates calculated for the non-depressed (D) speakers from the ORI data base versus the number of Gaussian mixtures with GMM

modeling based on the classical EM algorithm (purple bars) and the new ITEM algorithm (blue bars).

It can be observed in Figure 6.1 that for the depressed data sets, when the number of Gaussian mixtures increases from 1 to 128, the correct classification rates also increases, however any further increase of the numbers of Gaussian mixtures from 128 to 1024 leads to a slow decrease of the correct classification rates. The decrease was an effect of rapidly decreasing numbers of vectors within Gaussian clusters (mixtures) due to thinner distribution of data.

Figure 6.2, on the other hand shows that for the non-depressed data set, when the number of Gaussian mixtures increases from 1 to 256, the correct classification rates increase logarithmically reaching a constant plateau, and any further increase of the numbers of Gaussian mixtures from 128 to 1024 had almost no effect on the correct classification rates.

The above trends were consistent for both modeling techniques: EM/GMM and ITEM/GMM.

Based on these results, it was decided that 128 Gaussian mixtures provided satisfactory classification rates for both data sets (depressed and non-depressed) without making the computational complexity prohibitively large.

6.4.2 Optimizing the training and testing sets sizes

In this section optimal sizes of the training and testing sub-sets for the two ORI data sets, one containing depressed speakers and one containing the non-depressed speakers are determined.

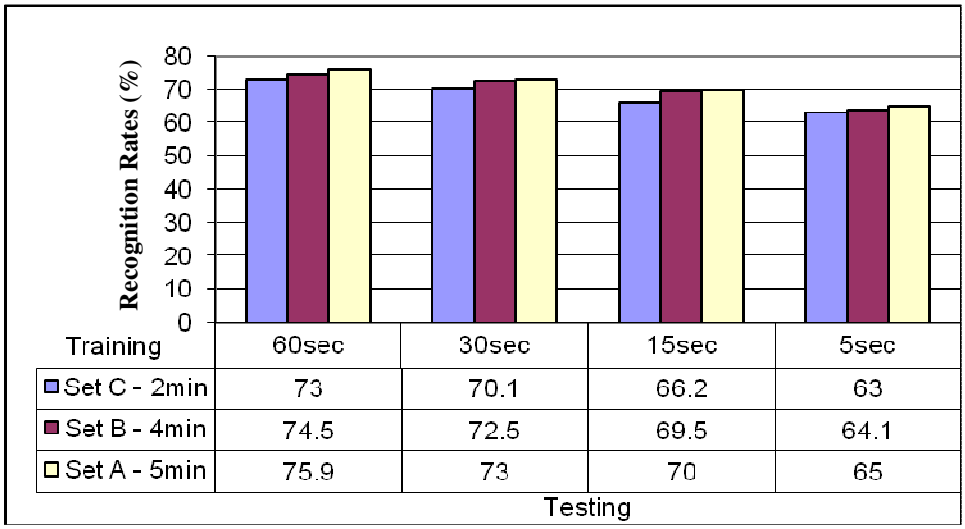
The speaker verification tests were based on the new ITEM procedure described in Chapter 4, and the ΔMFCC parameters were used as characteristic features.

Three training sets of different sizes were tested: 5 min (set A), 4 min (Set B) and 2 min (set C). These three training sets were tested in combinations with four testing sets of size: 60 sec, 30 sec, 15 sec and 5 sec. For each combination, three-turn cross validations were performed. Each turn was run with different randomly chosen training and testing set. The correct classification rates were calculated as an average value for the three turns.

The % of correct classification for all the 12 combinations of training/testing set's sizes are illustrated in Figure 6.3 (for the depressed speakers) and in Figure 6.4 (for the non-depressed speakers).

The results in Figures 6.3 and 6.4 show the same general trends for the depressed and non-depressed data. For a given size of the testing set, the correct classification rates increase with the training size increasing from 2 min to 5 min. For a given size of the training set, the correct classification rates decrease with the decreasing size of the testing set.

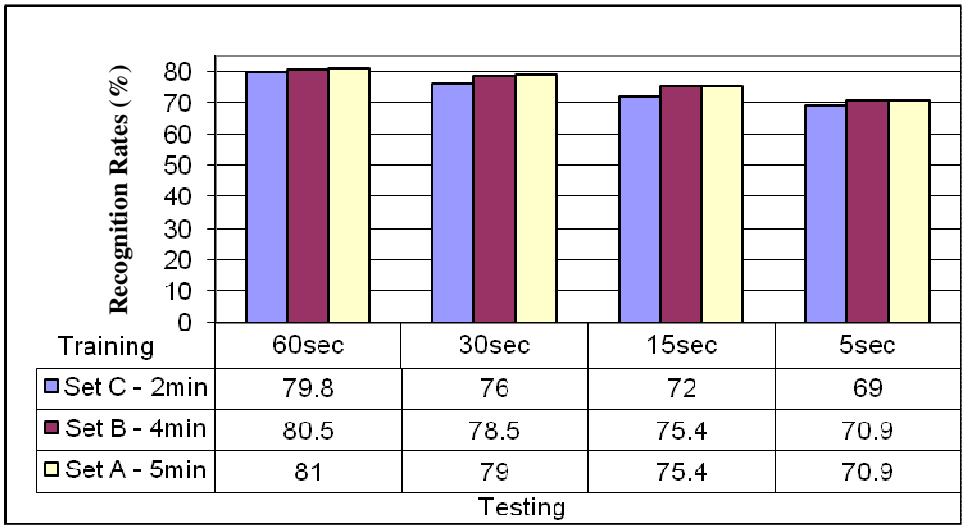
Based on these results, the testing set size of 5 min and the training set size of 60 seconds were chosen to perform experiments testing the effect of clinical depression on the speaker verification rates. These experiments are described in Sections 6.5 and 6.6.



Training	60sec	30sec	15sec	5sec
Set C - 2min	73	70.1	66.2	63
Set B - 4min	74.5	72.5	69.5	64.1
Set A - 5min	75.9	73	70	65

Testing

Figure 6.3 Correct classification rates in % for depressed speakers (from the ORI data base) using different training (set A, 5min, set B, 4 min & set C, 2 min) and testing (60 sec, 30 sec, 15 sec and 5 sec) sets sizes.



Training	60sec	30sec	15sec	5sec
Set C - 2min	79.8	76	72	69
Set B - 4min	80.5	78.5	75.4	70.9
Set A - 5min	81	79	75.4	70.9

Testing

Figure 6.4 Correct classification rates in % for non-depressed speakers (from the ORI data base) using different training (set A, 5min, set B, 4 min & set C, 2 min) and testing (60 sec, 30 sec, 15 sec and 5 sec) sets sizes.

6.5 Speaker Verification Using Classical ΔMFCC Features

In the first set of experiments the effect of clinical depression on speaker verification rates was tested using classical feature vectors ΔMFCC. The ΔMFCC vectors are most often used features in speaker verification [239,240,241,242] and the ΔMFCC feature vector consists of: MFCC+ΔMFCC (first derivative of MFCC)+ΔΔMFCC (second derivative of MFCC) +E (average spectral energy)+Z (number of zero crossings for the speech time waveform).

The ΔMFCC were used to determine the verification rates within homogeneous environments i.e. environments consisting only of depressed speakers or non-depressed speakers (Section 6.5.1) and mixed environments i.e. environments containing a mixture of both depressed and non-depressed speakers (Sections 6.5.2 & 6.5.3).

6.5.1 Speaker verification within homogeneous environments using classical ΔMFCC features

In this section two speaker verification experiments were conducted within homogeneous environments.

In the first experiment, the speaker verification rates were measured within an environment consisting of 68 clinically depressed speakers (D) from the ORI data base.

In the second experiment, the speaker verification rates were measured within an environment consisting of 70 non-depressed speakers (ND) from the ORI data base.

Since the numbers of speakers in both clinically depressed and non-depressed sets of the ORI data are approximately the same, and the recordings were made within identical

laboratory conditions including similar levels of the background noise, the speaker verification results within these two sets are directly comparable.

Figure 6.5 shows the miss probability versus false alarm probability and the equal error rates (EERs) for the homogeneous environments using ORI data (clinically depressed (D) – red line and non-depressed (ND) –green line).

It can be observed in Figure 6.5 that the speaker verification equal error rate (EER) within homogeneous environment containing only depressed speakers is 5.1% higher than for the homogeneous environment containing only non-depressed speakers.

These results indicate that the accuracy of speaker verification within clinical environment containing only depressed speakers decreases significantly when compared with the environment containing only non-depressed speakers. In other words, clinical depression makes the speaker verification task more challenging due to increase of the intra-speaker variability.

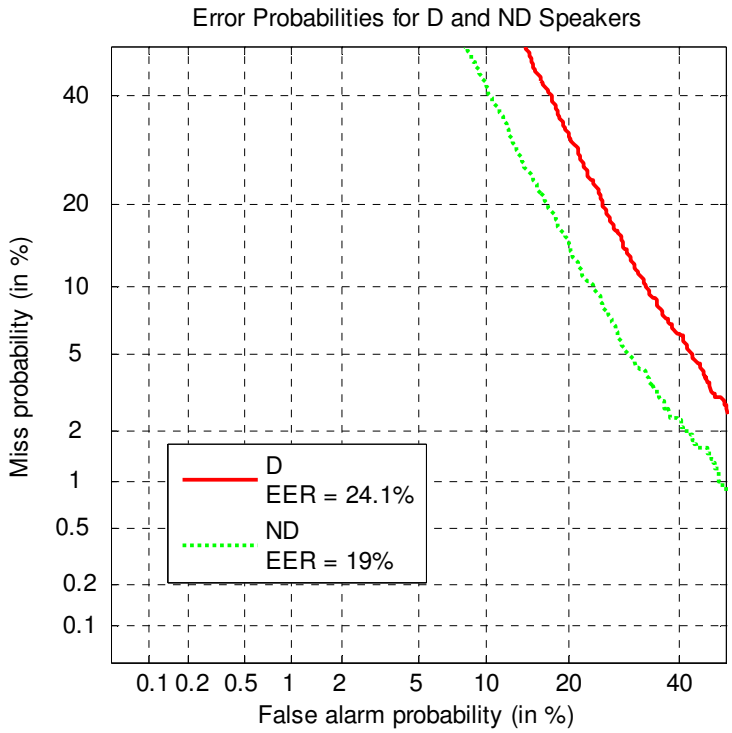


Figure 6.5 Miss probability versus false alarm probability and the equal error rates (EERs) for homogeneous environments using ORI data (clinically depressed (D) – red line and non-depressed (ND) –green line) and for the mixed environments.

6.5.2 Speaker verification within mixed environments using classical ΔMFCC features

This section describes speaker verification tests performed on data sets containing both clinically depressed and non-depressed speakers.

Described here experiments were based on mixed environments containing different ratios of depressed /non-depressed speakers. The aim was to determine how the depressed /non-depressed ratio relates to the speaker verification rates.

Four data sets were used: the first data set contained (68) 100% of depressed speakers, the second set contained a mixture of (34) 33% of depressed speakers and (68) 77% of non-depressed speakers, the, the third set contained a mixture of (17) 20% of depressed speakers and (68) 80% of non-depressed speakers, and finally, the fourth set contained (68) 100% of non-depressed speakers.

Speaker verification results for these four mixed environments are presented in Figure 6.6. The plots in Figure 6.6 show the miss probability versus false alarm probability and the equal error rates (EERs) (black line-100% ND, blue line – 25% D + 75% ND, red line -12% D + 88% ND, green line – 100% D).

It can be clearly observed in Figure 6.6 that with the increasing numbers of depressed speakers within the tested environments, the miss probability versus false alarm curves move towards areas corresponding to larger EER values.

The effect of increasing numbers of depressed speakers on the EER values is illustrated in Figure 6.7. It shows a logarithmic increase of the EER values with the increasing percentage of depressed speakers within a given environment. An increase of the numbers of depressed speakers from 0% to about 30% has the largest impact, and rapidly increases the EER from 19% to 24.8%. Further increase of the numbers of depressed speakers from 30% to 100% causes a slow increase of the EER values from 24.8% to 25.3%.

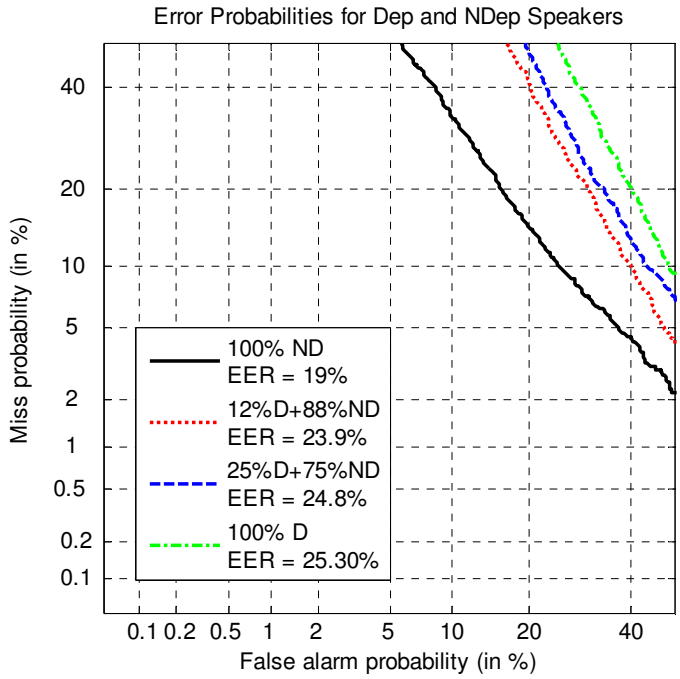


Figure 6.6 Miss probability versus false alarm probability and the equal error rates (EERs) for mixed environments using ORI data (black line-100% ND, red line -12% D + 88% ND, blue line – 25% D + 75% ND, green line – 100% D).

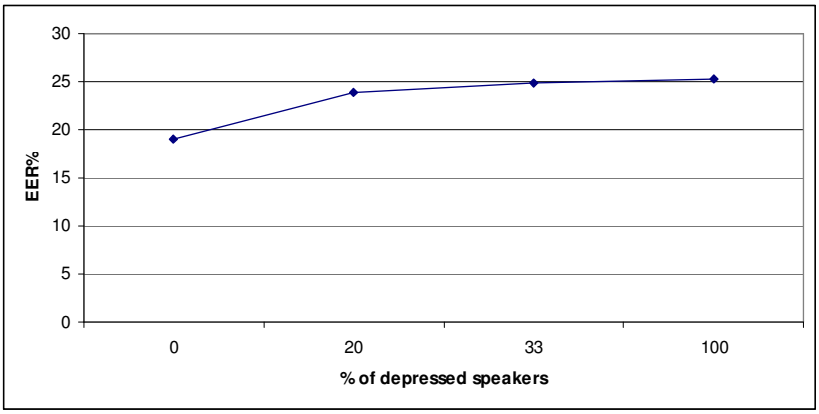


Figure 6.7 EER versus the % of depressed speakers in mixed environments using ORI data.

Additional two experiments related to the mixed environments test were conducted. In the first experiment, depressed speakers were verified within a mixed environment

consisting of equal fractions (50%) of depressed and non-depressed speakers (68 depressed and 68 non-depressed speakers). In the second experiment, non-depressed speakers were verified within the same mixed environment. These experiments aimed to determine if there is a significant difference in the verification accuracy between the depressed and the non-depressed speakers within an environment containing equal amounts of both types of speakers.

Results of these additional experiments are illustrated in Figure 6.8, which shows miss probability versus false alarm probability and the equal error rates (EERs) for mixed environments, (black line –verifying depressed speakers in the mixture of 50% depressed and 50% non-depressed speakers, blue line – verifying non-depressed speakers in the mixture of 50% depressed and 50% non-depressed speakers).

It can be observed in Figure 6.8, that the EER resulting from the verification of the depressed speakers is significantly higher (28.2%) than the ERR resulting from the verification of the non-depressed speakers (23.5%).

Looking at the results for homogeneous environments in Figure 6.5 and for mixed environments in Figure 6.8, it can be observed that the EER resulting from the verification of depressed speakers within a homogeneous environment containing only depressed speakers is lower (24.1%) than the EER resulting from the verification of depressed speakers within a mixed environment containing equal amounts of depressed and non-depressed speakers (28.2%). This possibly indicates that the depressed speakers were under-represented within the mixed environment containing 50% of depressed and 50% of non-depressed speakers.

Similarly, an inspection of the results for homogeneous environments in Figure 6.5 and for mixed environments in Figure 6.8, it can be observed that the EER resulting from the verification of non-depressed speakers within a homogeneous environment containing

only non-depressed speakers is lower (19%) than the EER resulting from the verification of depressed speakers within a mixed environment containing equal amounts of depressed and non-depressed speakers (23.5%). This could indicate that the non-depressed speakers were also under-represented within the mixed environment containing 50% of depressed and 50% non-depressed speakers and the presence of depressed speakers introduced an environmental noise.

These results are consistent with the previous results in Section 6.5.1 indicating that the verification task in general is more challenging within environments consisting of depressed speakers and the verification of depressed speakers is in general more challenging than identification of non-depressed speakers.

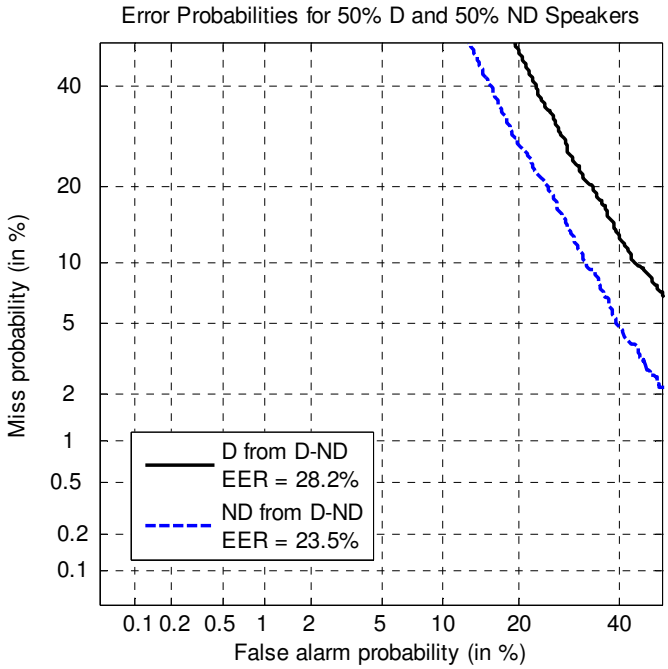


Figure 6.8 Miss probability versus false alarm probability and the equal error rates (EERs) for mixed environments; black line –verifying depressed speakers in the mixture of 50% depressed and 50% non-depressed speakers, blue line – verifying non-depressed speakers in the mixture of 50% depressed and 50% non-depressed speakers.

6.6 Speaker Verification in Homogenous Environments Using TEO-PWP-Auto-Env Features

Speaker verification tests presented in Section 6.5 demonstrated that addition of clinically depressed speakers to the testing and training data results in significantly lower equal error rates.

These results were obtained using the benchmark speaker verification system including ΔMFCC features and GMM classifier, tested previously in numerous speaker verification studies [239, 241, 242]. Majority of these studies used NIST corpora or similar data bases consisting of "normal" speaker, i.e. speakers which were not identified as suffering from any particular type of disorder that could potentially alter acoustic characteristics of their speech. In other words, the benchmark speaker verification system was optimized without taking into account the possibility of an environmental noise that could be introduced by the presence of clinically depressed speakers in the testing and training data.

Since the characteristic features are of key importance in majority of the pattern recognition systems, the possible panacea for the adverse effects of depression on the speaker verification rates could be provided by the right choice of the feature vectors.

Aiming to compensate for the adverse effects of clinical depression on the speaker verification rates, the TEO-PWP-Auto-Env features which showed the best performance in Section 5.5.6 were applied and compared with the classical ΔMFCC features.

As explained in Section 5.5, the TEO-PWP-Auto-Env were derived [196] based on recent laryngological experiments [194, 197] investigating the air flow during the glottal flow formation. The nonlinear character of the glottal flow results in a number of vortices,

which affect the acoustic properties of speech. Namely, they change the spectral energy distribution and generate additional harmonics and cross-harmonics in the speech signal.

The high performance of the TEO-PWP-Auto-Env demonstrated in Chapter 5 can be attributed to the fact that this parameter is highly sensitive to the presence of additional harmonics and cross-harmonics in speech. Assuming that in the state of depression alters the vocal system physiology and generates speaker-specific vortices providing additional harmonic components; the TEO-PWP-Auto-Env-Area was expected to provide better performance than the classical ΔMFCC features, which were derived assuming a laminar air flow during the phonation process.

Speaker verification experiments comparing the performance of the nonlinear TEO-PWP-Auto-Env features with the classical ΔMFCC features were conducted within two homogeneous environments. The first environment contained only depressed speakers (68 speakers from the ORI data base), and the second environment contained only non-depressed speakers (71 speakers from the ORI data base).

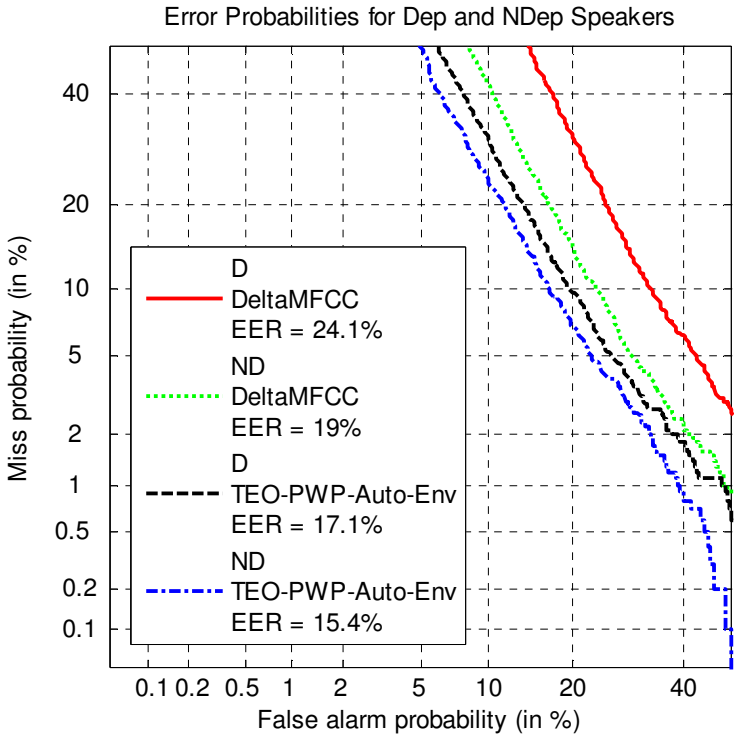


Figure 6.9 Miss probability versus false alarm probability and the equal error rates (EERs) for homogeneous environments using ΔMFCC features and TEO-PWP-Auto-Env features.

The results in Figure 6.9 indicate that within the homogeneous environment containing only depressed speakers, the TEO-PWP-Auto-Env provided better performance compare to the ΔMFCC features. The TEO-PWP-Auto-Env features significantly decreased the EER from 24.1% (for ΔMFCC) to 17.1% (for TEO-PWP-Auto-Env).

Figure 6.9 shows that the TEO-PWP-Auto-Env features provided more robust performance in the homogeneous non-depressed environment (based on ORI data) compare to ΔMFCC, lowering the EER from 19.0% (for ΔMFCC) to 15.4% (for TEO-PWP-Auto-Env).

The nonlinear TEO-PWP-Auto-Env features provided superior performance compared to the classical ΔMFCC features and showed higher robustness by maintaining high level of performance despite of a change of environment and introduction of higher intra-speaker variability. These results are consistent with the previous results based on the NIST 2004 corpora, described in Section 5.5.6

6.7 Summary

In this chapter, effects of a clinical environment on the speaker verification rates were investigated for the first time.

Speaker verification within a clinical environment consisting only of the clinically depressed speakers was compared with the speaker verification within a neutral (control) environment containing only non-depressed speakers.

Experiments based on mixed environments containing different ratios of depressed/non-depressed speakers were also conducted in order to determine how the depressed /non-depressed ratio relates to the speaker verification rates.

The experiments used a clinical speech corpus consisting of 68 clinically depressed and 71 non-depressed speakers.

Speaker models were built using the new ITEM-GMM method introduced in Chapter 4.

The feature vectors consisted of the classical ΔMFCC content including mel frequency cepstral coefficients (MFCC), their first and second derivatives, short time energy coefficient and the zero-crossing rates calculated for speech samples on the frame-by-frame basis.

The results based on this classical approach indicated that the speaker verification within the clinical environment provides a challenging task.

Experiments conducted within homogeneous environments showed a significant decrease of the equal error rates (EER) by 5.1% for the clinically depressed environment when compared with the non-depressed environment.

Experiments conducted within mixed environments containing different depressed/non-depressed ratios showed that an increasing number of depressed speakers lead to a logarithmic increase of the EER values; where the increase of the percentage of depressed speakers from 0% to 30% has the most profound effect on the increase of the EER.

In order to compensate for the adverse effects of depression on the speaker verification rates, TEO-PWP-Auto-Env features which showed the best performance in Chapter 5 were applied and compared with the classical ΔMFCC features.

The results showed that the TEO-PWP-Auto-Env provided more robust performance in the clinical environment compare to ΔMFCC, lowering the EER from 24.1% (for ΔMFCC) to 17.1% (for TEO-PWP-Auto-Env). The TEO-PWP-Auto-Env also provided better performance within the non-depressed environment compare to ΔMFCC, lowering the EER from 19.0% (for ΔMFCC) to 15.4% (for TEO-PWP-Auto-Env), which was consistent with previous results based on NIST 2004, described in Chapter 5. This demonstrates that the TEO based features are more representative of the clinical depression in speech.

CHAPTER 7

CONCLUSIONS AND FUTURE RESEARCH

This chapter summarizes the key conclusions from this thesis and highlights potential future work in exploring the data-driven and time-frequency approaches to feature extraction for speaker recognition. In particular it addresses the key challenges that have been met, and those remaining for the field, as well as discussing limitations of the work.

7.1 Summary of Research and Conclusions

The thesis aimed to investigate three major areas of improvements of the existing speaker recognition methodology.

Firstly, the aim was to propose an improved modelling and classification methodology for speaker recognition. This aim was achieved by the development of a new algorithm for the calculation of Gaussian Mixture Model parameters called Information Theoretic Expectation Maximization (ITEM). The proposed algorithm improves upon the classical Expectation Maximization (EM) approach widely used with the Gaussian mixture model (GMM) as a state-of-art statistical modeling technique. Like the classical EM method, the ITEM algorithm adapts means, covariances and weights, however this process is not conducted directly on feature vectors but on a set of centroids derived by the information theoretic vector quantization (ITVQ) procedure, which simultaneously minimizes the divergence between the Parzen estimates of the feature vector's distribution within a

given class and the centroids distribution within the same class. The ITEM algorithm was applied to the speaker verification problem using NIST 2001, NIST 2002 and NIST 2004 corpora and MFCC with delta features. The results showed an improvement of the equal error rate over the classical EM approach. The EM-ITVQ also showed higher convergence rates compared to the EM.

Secondly, the aim was to determine the usefulness of features derived from nonlinear models of speech production for speaker recognition.
This aim was achieved by comparing the classical features based on linear models of speech production with recently introduced features based on the nonlinear model. A number of linear and nonlinear feature extraction techniques that have not been previously tested in the task of speaker verification are tested. New fusions of features carrying complimentary speaker-dependent information are proposed. The tested features are used in conjunction with the new ITEM-GMM speaker modeling technique, which provided an additional evaluation of the new method. The speaker verification experiments presented demonstrated significant improvement of performance when the conventional MFCC features were replaced by a fusion of the MFCCs with complimentary linear features such as the inverse MFCCs (IMFCCs), or nonlinear features such as the TMFCCs and TEO-PWP-Auto-Env. Higher overall performance of the nonlinear features when compared to the linear features was observed.

Thirdly, the aim was to determine the effects of a clinical environment containing clinically depressed speakers on speaker recognition rates, and to investigate if the features based on nonlinear models of speech production have the potential to counteract the inverse effects of the clinically depressed environment.
For the first time, the thesis investigated the effects of a clinical environment on the speaker verification. Speaker verification within a homogeneous environment consisting of the clinically depressed speakers was compared with the speaker verification within a neutral (control) environment containing of non-depressed speakers. Experiments based

on mixed environments containing different ratios of depressed/non-depressed speakers were also conducted in order to determine how the depressed/non-depressed ratio relates to the speaker verification rates. The experiments used a clinical speech corpus consisting of 68 clinically depressed and 71 non-depressed speakers. Speaker models were built using the new ITEM-GMM method introduced in Chapter 4. Two types of feature vectors were tested, the classical ΔMFCC coefficients and the TEO-PWP-Auto-Env features. Experiments conducted within homogeneous environments showed a significant decrease of the equal error rates (EER) by 5.1% for the clinically depressed environment when compared with the non-depressed environment. Experiments conducted within mixed environments showed that an increasing number of depressed speakers lead to a logarithmic increase of the EER values; where the increase of the percentage of depressed speakers from 0% to 30% has the most profound effect on the increase of the EER. It was also demonstrated that the TEO-PWP-Auto-Env provided more robust performance in the clinical environments compare to ΔMFCC, lowering the EER from 24.1% (for ΔMFCC) to 17.1% (for TEO-PWP-Auto-Env).

7.2 Future Challenges

From the perspective of real-time implementations of the speaker recognition technology, further work is needed in the optimization of the speaker modeling and classification approaches. Issues such as improvement of convergence rates of the modeling process, distance measures and likelihood estimation between the feature vectors of the unknown speaker and the speaker models are of key importance for the future research.

More work is also needed in the area of feature extraction. This line of research can largely benefit from the developments of new nonlinear models of speech production which explicitly take into account mechanisms responsible for changes of acoustic

properties of speech due to emotions, diseases, aging, use of alcohol and drugs. Through the development of such models, new parameters can emerge providing more robust speaker characteristics.

Another area for future research includes studies of effects of different environments. This is linked to the development of speech processing, and speech enhancement methods compensating for the adverse effects of different environments.

Bibliography

[1] R. Brunelli, D. Falavigna, "Person identification using multiple cues", IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 17, Issue.10, pp. 955-966, 1995.

[2] K.A. Toh, "Fingerprint and speaker verification decisions fusion", In Proceedings of 12th International Conference on Image Analysis and Processing, pp.626-631, 2003.

[3] S. Prabhakar, S. Pankanti, A. Jain,"Biometric recognition: security and privacy concerns", IEEE Security & Privacy Magazine, Vol.1, pp.33–42, 2003.

[4] E. Parzen,'On estimation of a probability density function and mode'. The Annals of Mathematical Statistics, Vol. 27, pp.1065–1076, 1962

[5] S. Kullback, R. A. Leibler. 'On information and sufficiency'. The Annals of Mathematical Statistics, Vol. 22, pp.79–86, 1951.

[6] P. Rose, "Forensic Speaker Identification", Taylor & Francis, London, 2002.

[7] J. Campbell, "Speaker recognition: a tutorial", Proceedings of the IEEE, Vol. 85, Issue.9, pp.1437-1462, 1997.

[8] S. Furui, "Recent advances in speaker recognition", Pattern Recognition Letters, Vol. 18, Issue.9, pp.859-872, 1997.

[9] A. Gresho, "Vector quantization and signal compression", 1992: Kulwer Academic Publishers.

[10] M. Lech, "Algorithms for the Vector Quantization of Images", PhD Thesis, The University of Melbourne, 1993.

[11] J. Markowitz, "Using speech recognition in customer relationship management to be more effective", In DCI customer relationship management conference, 1999.

[12] A.M. Ju, "Could Biometrics Make Skies Safer?", In PC World, 2002.

[13] J. Kerin, "Biometric passport demand likely", 2004, www.news.com.au.

[14] G.R. Doddington, "A new method of speaker verification", Journal of acoustical society of America, Vol. 49, p.139, 1971.

[15] P.D. Bricker, “Statistical techniques for talker identification”, Bell systems technical journal, Vol.50, pp.1427-1454, 1971.

[16] R.S. Cheung, “Feature selection via dynamic programming for text independent speaker identification”, IEEE Transactions on Acoustics, Speech and Signal Processing, Vol.26, No.5, pp.397-403, 1978.

[17] Yuk, C.C.Q.L.D.-S., “An HMM approach to text independent speaker verification”, In IEEE International conference on Acoustics, Speech and Signal Processing, 1996.

[18] F.K. Soong, et al., “A vector quantization approach to speaker recognition”, AT & T Technical Journal, Vol.66, No.2, pp.14-26, 1987.

[19] D.A. Reynolds, “Speaker identification and verification using Gaussian mixture speaker models”, Speech Communication, Vol.17, pp.91-108, 1995.

[20] C. Wang, D. Xu, C.P. Jose, “Speaker verification and identification using gamma neural networks”, In international conference on neural networks, 1997.

[21] S. Bengio, J. Mariethoz, “Learning the decision function for speaker verification”, In IEEE International conference on Acoustics, Speech and Signal Processing, 2001.

[22] X. Zhu, et al., “A comparative study of mixture-Gaussian VQ, ergodic HMMs and left-to-right HMMs for speaker recognition”, In international symposium on Speech, Image processing and Neural Networks, 1994.

[23] G. Kolano, P. Regel-Brietzmann, "Combination of vector quantization and Gaussian mixture models for speaker verification with sparse training data", In *EUROSPEECH*, pp.1203-1206, 1999.

[24] J. Pelecanos, S. Myers, S. Sridharan, V. Chandran, “Vector quantization based Gaussian modeling for speaker verification”, International Conference on pattern recognition, Vol. 3, pp. 294-297, Spain, 2000.

[25] H. Jialong, L. Liu, P. Gunther, ”A new codebook training algorithm For VQ-based speaker recognition”, IEEE international conference on acoustics, speech and signal processing, Vol. 2, pp.1091-1094, 1997.

[26] A. Ethem, “Soft vector quantization and the EM algorithm”, Neural Networks, Vol.11, No.3, pp. 467-477, April 1998.

[27] P. Hedelin, J. Skoglund, “Vector quantization based on Gaussian mixture models”, IEEE Transactions on speech and audio processing, Vol. 8, No. 4, pp. 385-401, 2000.

[28] N. Ueda, R. Nakano, "Deterministic annealing EM algorithm," Neural Networks, No. 11, pp. 271–282, 1998.

[29] X. Lei, I.J. Michael, "On Convergence Properties of the EM Algorithm for Gaussian Mixtures, Neural Computation", Vol. 8, No. 1, pp. 129-151, January 1996.

[30] D. Ververidis, C. Kotropoulos, "Gaussian mixture modeling by exploiting the mahalanobis distance", IEEE transactions on signal processing, Vol. 56, No. 7, pp. 2797-2811 , July 2008.

[31] K.N. Stevens, "Sources of inter and intra-speaker variability in acoustic properties of speech sounds", In 7th international congress on phonetic sciences, 1971, Montreal, Canada.

[32] B. Yegnanarayana, S.R.M Prasanna, J.M. Zachariah, C.S. Gupta, "Combining evidence from source, suprasegmental and spectral features for a fixed-text speaker verification system", IEEE Transactions on Speech and Audio Processing, Vol. 13, No. 4, pp. 575-582, July 2005.

[33] K. Sri Rama Murty and B. Yegnanarayana, "Combining evidence from residual phase and MFCC features for speaker recognition", IEEE Signal Processing Letters, vol 13, no. 1, pp. 52-55, Jan. 2006.

[34] S.R.M. Prasanna, S.G. Cheedella, B. Yegnanarayana, "Extraction of speaker-specific excitation information from linear prediction residual of speech", Speech Communication, Vol. 48, Issue 10, pp. 1243-1261, October 2006.

[35] S. Chakroborty, A. Roy, S. Majumdar, G. Saha, **"**Capturing Complementary Information via Reversed Filter Bank and Parallel Implementation with MFCC for Improved Text-Independent Speaker Identification", International conference on Computing theory and applications, pp. 463-467, March 2007.

[36] C.K. Reddy, C. Hsiao-Dong, B. Rajaratnam, "TRUST-TECH-Based Expectation Maximization for Learning Finite Mixture Models", IEEE transactions on Pattern Analysis and Machine Intelligence, Vol. 30, Issue. 7, pp. 1146-1157, 2008.

[37] V. Hautamäki, T. Kinnunen, I. Kärkkäinen, J. Saastamoinen, P. Fränti, "Maximum a posteriori adaptation of the centroid model for speaker verification, IEEE Signal Processing Letters, Vol.15, 2008.

[38] NIMH. "The Numbers Count: Mental Illness in America," Science on our Minda Fact Sheet Series.

[39] D.A. Reynolds, T. Quatieri, R. Dunn, "Speaker verification using adapted Gaussian mixture models", Digital Signal Processing Vol.10, No.1, 2000.

[40] D.A. Reynolds, R. Rose, "Robust text-independent speaker identification using Gaussian mixture speaker models.", IEEE Transactions on Speech and Audio Processing Vol.3, 1995.

[41] A. Higgins, L. Bahler, J. Porter, "Speaker verification using randomized phrase prompting", Digital Signal Processing, Vol. 1, 1991.

[42] K.-P. Li, J. Porter, "Normalizations and selection of speech segments for speaker recognition scoring", In Proc. Int. Conf. on Acoustics, Speech, and Signal Processing, pp. 595-598, (ICASSP 1988).

[43] P. Sivakumaran, J. Fortuna, A. Ariyaeeinia, "Score normalization applied to open-set, text-independent speaker identification" In Proceedings of 8th European Conf. on Speech Communication and Technology, pp. 2669-2672, (Eurospeech 2003).

[44] H. S. Jayanna and S. R. Mahadeva Prasanna, "Analysis, Feature Extraction, Modeling and Testing Techniques for Speaker Recognition", IETE Technical Review, Vol.26, Issue.3, pp.181-190, 2009.

[45] A.K. Jain, A. Ross, S. Prabhakar, "An Introduction to Biometric Recognition" IEEE Transactions on Circuits and Systems for Video Technology, Special Issue on Image and Video Based Biometrics, Vol. 14, No. 1, January 2004.

[46] K. Delac, M. Grgic , "A Survey Of Biometric Recognition Methods", 46th International Symposium on Electronics in Marine, ELMAR-2004, 16-18 June 2004, Zadar, Croatia.

[47] T.H. Kinnunen, "Optimizing Spectral Feature Based Text-Independent Speaker Recognition", PhD Dissertation, University of Joensuu, Finland, 2005.

[48] J.M. Naik, "Speaker Verification: A Tutorial", IEEE Communications Magazine. 1990.

[49] J.P. Campbell, "Speaker recognition: A Tutorial", Proceedings of the IEEE, Vol. 85, Issue 9, pp. 1437-1462, Sep 1997.

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

[51] J. Wolf, "Efficient acoustic parameters for speaker recognition" Journal of the Acoustical Society of America Vol. 51, No. 6, pp.2044-2056, 1972.

[52] H. Nakasone, "Automated speaker recognition in real world conditions: Controlling the uncontrollable.", 8th European Conference on Speech Communication and Technology, pp. 697-700, Geneva, Switzerland, 2003.

[53] T.D. Ganchev, "Speaker Recognition", PhD dissertation, University of Patras, Greece, 2005.

[54] P. Thevenaz, H. Hugli, "Usefulness of the LPC-residue in text- independent speaker verification," Speech Communication, vol. 17, pp. 145-57, 1995.

[55] 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. 569-85, 1999.

[56] Computational Neuro-Engineering Lab, http://www.cnel.ufl.edu/research/ITL.php

[57] G. Singh, A. Panda, S. Bhattacharyya, T. Srikanthan, "Vector quantization techniques for GMM based speaker verification.", IEEE international conference on acoustics, speech and signal processing. Vol. 2, pp. II65-II68, 2003.

[58] J. MacQueen, "Some methods for classification and analysis of multivariate observations", Fifth Berkeley Symposium on Mathematical statistics and probability Vol.1, pp.281–297, 1967.

[59] Y. Linde, A. Buzo, R.M. Gray," An algorithm for vector quantizer design", IEEE Transactions on Communications, Vol.28, No. 1., pp. 84-95, 1980.

[60] N.M. Kajarekar, H. hermansky, "Analysis of source of variability in speech", In Eurospeech99. 1999.

[61] J.S. Pan, F.R. McInnes, M.A. Jack, "Fast clustering algorithms for vector quantization", Pattern Recognition, Vol. 29, Issue 3, March 1996, Pages 511-518.

[62] K. Farrell, R. Ramachandran, R. Mammone, "An analysis of data fusion methods for speaker verification.", International Conference on Acoustics, Speech, and Signal Processing (ICASSP) vol. 2, pp. 1129-1132, 1998.

[63] S. Slomka, S. Sridharan, V. Chandran, "A comparison of fusion techniques in mel-cepstral based speaker identification.", International Conference On Spoken Language Processing (ICSLP), pp. 225-228, 1998.

[64] P. Sivakumaran, A. Ariyaeeinia, M. Loomes, “Sub-band based speaker verification using dynamic recombination weights.”, International Conference on Spoken Language Processing (ICSLP), pp. 77-80, 1998.

[65] R. Ramachandran, K. Farrell, R. Ramachandran, R. Mammone, “Speaker recognition: general classifier approaches and data fusion methods.”, Pattern Recognition Vol. 35, pp. 2801-2821, 2002.

[66] R. Damper, J. Higgins, ”Improving speaker identification in noise by sub band processing and decision fusion.” Pattern Recognition Letters Vol. 24, pp. 2167-2173, 2003.

[67] M.-W.Mak, M.-C. Cheung, S.-Y. Kung, “Robust speaker verification from GSM-trascoded speech based on decision fusion and feature transformation”, International Conference on Acoustics, Speech, and Signal Processing (ICASSP), pp. 745-748, 2003.

[68] A. Hannani, D. Petrovska-Delacretaz, G. Chollet, “Linear and non-linear fusion of ALISP-based and GMM systems for text-independent speaker verification”, In Proceedings of Speaker Recognition Workshop (Odyssey 2004), pp. 111-116, Spain 2004.

[69] N. Malayath, H. Hermansky, A. kain, “Towards decomposing the sources of variability in speech”, In Eurospeech 1997, Greece.

[70] S. Kajarekar, N.M., H hermansky, “Analysis of speaker and channel variability in speech”, In proceedings of the workshop on automatic speech recognition and understanding, 1999.

[71] S. Pruzansky, "Pattern-matching procedure for automatic talker recognition," Journal of Acoustical Society of America (JASA), vol. 35(3), pp. 354-8, Mar. 1963

[72] J.W. Glenn, N. Kleiner, "Speaker identification based on nasal phonation Journal of Acoustical Society of America (JASA) , vol. 43(2), pp. 368-72, June 1967

[73] T. Kinnunen, T. Kilpeläinen, P. Fränti, “Comparison of Clustering Algorithms in Speaker Identification,.”, Proceedings of the IASTED International Conference on Signal Processing and Communications (SPC), pp. 222-227, Marbella, Spain, Sep, 2000.

[74] M.R. Sambur, "Selection of acoustic features for speaker identification," IEEE Transactions on Acoustics Speech and Signal Processing, vol. ASSP-23(2), pp. 176-82, April 1975.

[75] A.E. Rosenberg, M.R. Sambur, "New techniques for automatic speaker verification," IEEE Transactions on Acoustics Speech and Signal Processing, vol. ASSP-23(2), pp. 169-76, Apr. 1975.

[76] M.R. Sambur, "Speaker recognition using orthogonal linear prediction," IEEE Transactions on Acoustics Speech and Signal Processing, vol. ASSP-24(4), pp. 283-9, Aug. 1976.

[77] S.I Furui, "Cepstral analysis technique for automatic speaker verification," IEEE Transactions on Acoustics Speech and Signal Processing, vol. 29(2), pp. 254-72, Apr. 1981.

[78] X. Yang, J.B, Millar, I. Macleod, "On the source of inter & intra- speaker variability in the acoustic dynamics of speech", 4th international conference on spoken language, 1996.

[79] R. Gray, "Vector quantization," IEEE Magazine on Acoustics Speech and Signal Processing, vol. 1, pp. 4-29, Apr. 1984.

[80] L.S. Tue, H. Anant, E. Deniz, C.P. Jose, "Vector quantization using Information Theoretic Concepts", Natural Computing, Vol. 4, pp.39-51, Springer 2005.

[81] J.C. Bezdek, J.D. Harris, "Fuzzy portions and relations; an axiomatic basis for clustering," Fuzzy Sets and Systems, vol. 1, pp. 111-27, 1978

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

[83] L. Lin, S. Wang, "A Kernel method for speaker recognition with little data," International Conf. on signal Processing, Budapest, Hungary, May, 2006

[84] V. Chatzis, A.G. Bors, I. Pitas, "Multimodal decision-level fusion for person authentication," IEEE Transactions on Man Cybernetics Part A: Systems and Humans, vol.29, pp. 674-81, Nov. 1999

[85] A.E. Rosenberg, S. Parthasarathy, "Speaker background models for connected digit password speaker verification", International Conference on Acoustics, Speech and Signal Processing, Atlanta Georgia, May 1996.

[86] J.M. Naik, L.P. Nestch, and G.R. Doddington, "Speaker verification using long distance telephone lines," International Conference on Acoustics, Speech and Signal Processing, Glasgow, UK, May 1989.

[87] T. Matsui, and S. Furui, "Comparison of text-independent speaker recognition methods using VQ-distortion and Discrete/continuous HMMs," IEEE Transactions on Speech Audio Processing, vol. 2(3), pp. 456-9, July 1994

[88] O. Kimball, M. Schmidt, H. Gish, and J. Waterman, "Speaker verification with limited enrollment data," European Conf. on Speech Communication and Technologies. (EUROSPEECH'97), pp. 967-70, Rhodes, Greece, Sep. 1997,

[89] R.P. Lipmann, "An introduction to computing with neural nets," IEEE Transactions on Acoustics, Speech and Signal Processing, vol. 4, pp. 4-22, Apr. 1989.

[90] G. Bannani, and P Gallinari, "Neural networks for discrimination and modelization of speakers," Speech Communication. , vol. 17, pp. 159-75, 1995.

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

[92] J. Oglesby, and J.S. Mason, "Optimization of neural models for speaker identification," International Conf. on Acoustics, Speech, and Signal Processing, pp. 261-4, May 1990,

[93] J. Oglesby, "Radial basis function networks for speaker recognition," International Conf. on Acoustics, Speech and Signal Processing, pp. 393-6, Toronto, Canada, May 1991.

[94] T. Kohonen, "The self-organizing map," Proceedings of IEEE , vol. 78(9), pp. 1464-80, Sep. 1990

[95] M. Inal, Y.S. Fatihoglu, "Self organizing map and associative memory model hybrid classifier for speaker recognition," in seminar on application of Neural Netwroks in Electrical Engg. (NEUREL'02), pp. 71-4, Belgrade, Yugoslavia, Sep. 2002.

[96] A.T. Mafra, M.G. Simoes, "Text independent automatic speaker recognition using self-organizing maps," Industry Applications conference, vol. 3, pp.1503-10, Brazil, Oct. 2004.

[97] G. Bannani, F. Fogelman, P. Gallinari, "A connectionist approach for speaker identification," International Conf. on Acoustics, Speech and Signal Processing, pp. 265-8, May 1990.

[98] J. He, L. Liu, G. Palm, "A discriminative training algorithm for VQ-based speaker identification," IEEE Transactions on Speech Audio Processing, vol. 7, pp. 353-6, May 1999.

[99] M. Friedman, Introduction to pattern recognition: Statistical, structural, neural, and fuzzy logic approaches. 1999, Singapore: World Scientific.

[100] A. Webb, Statistical pattern recognition. 1999, London: Arnold

[101] 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. 210-29, 2006.

[102] K. Fukunaga, Introduction to statistical pattern recognition. 2nd edition ed. 1990: Academic Press Inc.

[103] B. Yegnanarayana, S.P. Kishore, "AANN: An alternative to GMM for pattern recognition," Neural Networks , vol. 15, pp. 459-69, 2002.

[104] M.S. Iqbal, H. Misra, B. Yegnanarayana, "Analysis of auto associative neural networks," International Joint Conf. on Neural Networks, Washington, USA, 1999.

[105] S. Furui, Digital Speech Processing, Synthesis and Recognition, Marcel Dekker Inc., New York, (1989).

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

[107] N. Pal, J. Bezdek, E. Tsao, "Generalized clustering networks on Kohonen's self organizing scheme.", IEEE Transactions on Neural Networks Vol.4, pp. 549–557, 1993.

[108] I. Katsavounidis, C.-C. Kuo, Z. Zhang, "A new initialization technique for generalized Lloyd iteration.", IEEE Signal Processing Letters Vol. 1, pp. 144–146, 1994

[109] R.O. Duda, P.E. Hart, " Pattern classification and scene analysis.", New York: Wiley, 1973.

[110] V. Wan, and S. Renals, "Speaker verification using sequence discriminant support vector machines," IEEE Transactions on Speech Audio Processing, vol. 13, pp. 203-10, 2005.

[111] V. Wan, S. Renals, "Evaluation of kernel methods for speaker verification and identification," IEEE International Conference on Acoustics, Speech, and Signal Processing, vol. 1, pp. 1-669 - 1-672, 2002.

[112] W.M. Campbell, D.E. Sturim, D.A. Reynolds, "Support vector machines using GMM supervectors for speaker verification," IEEE Signal Processing Letters, vol. 13(5), pp. 308-11, May 2006.

[113] C.H. You, K.A. Lee, H. Li, "An SVM kernel with GMM-supervector based on the Bhattacharyya distance for speaker recognition," IEEE Signal Processing Letters, vol. 16(1), pp. 49-52, Jan. 2009.

[114] S. Theodoridis, Pattern Recognition. 1999, San Diego: Academic Press

[115] H.N. James, L. Margaret, "Data driven and time frequency based feature extraction for speaker recognition" RMIT University Australia, PhD Thesis.

[116] S. Furui, "Vector quantization based speech recognition and speaker recognition techniques", 25th Asilmor conference signals, systems and computers, 1991.

[117] B.D. Ripley, Pattern recognition and neural networks, 1996, New York: Cambridge University Press.

[118] X. Wang, "Text dependent speaker verification using recurrent neural time delay neural networks for feature extraction", In IEEE signal processing workshop, 1993.

[119] S.E. Fredrickson, L. Tarassenko, "Text independent speaker recognition using neural networks techniques" In 4th international conference on neural networks, 1995.

[120] R.A. Finan, A.T. Sapeluk, R.I Damper, "Comparison of multilayer and radial basis function neural networks for text independent speaker recognition" In IEEE international conference on neural networks, 1996.

[121] J.-H. Lee, et al, "Speech feature extraction using independent component analysis", In ICASSP 2000.

[122] A.P. Dempster, N.M. Laird, D.B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm", Journal of Royal Statistical Society, B39, pp.1-38, 1977.

[123] I. margin-Chagnolleau, G. Durou, "Applications of time-frequency principal component analysis to speaker verification", Elsevier-Digital Signal Processing, Vol.10, Issues.1-3, pp.226-236, 2000.

[124] M. Kotani, et.al, "Applications of independent component analysis to feature extraction of speech", International conference on Neural networks, 1999.

[125] F. Cardinaux, C. Sanderson, S. Bengio, "User authentication via adapted statistical models of face Images", IEEE Transactions on Signal Processing, Vol. 54, No. 1, pp. 361-373, Jan 2006.

[126] H.Y. Watanabe, T. Katagiri, "Discriminative metric design for pattern recognition" ICASSP 1995.

[127] C.M. Bishop, Neural Networks for pattern recognition, 1995, Oxford University Press.

[128] H. Bredin, N. Dehak, G. Chollet, "GMM-based SVM for face recognition", International Conference on Pattern Recognition, Vol. 3, pp.1111-1114, 2006.

[129] X. Lei, "Comparative Analysis on Convergence Rates of The EM Algorithm and Its Two Modifications for Gaussian Mixtures", Neural Processing Letters, Vol. 6, No. 3, pp. 69-76, Dec 1997.

[130] New US-VISIT Biometric Entry-Exit System Begins. 2004, Embassy of the United States of America, Canberra Australia.

[131] H. Hu, X. Ming-Xing, W. Wei, " GMM Supervector Based SVM with Spectral Features for Speech Emotion Recognition", IEEE Transactions on Acoustics, Speech and Signal Processing, Vol. 4, No. 15, pp. IV-413 – IV-416, 2007.

[132] S. G. John, F.L. Lori, M.F. William, G.F. Jonathan, S.P. David, L.D. Nancy, Z. Victor, "TIMIT Acoustic-Phonetic Continuous Speech Corpus.", Linguistic Data Consortium, 1993.

[133] M. Falcone, A. Gallo, "The "SIVA" speech database for Speaker verification:description and evaluation", 4th International conference on spoken language processing (ICSLP), Vol. 3, pp. 1902-1905, Oct 1996.

[134] D. Petrovska, J. Hennebert, H. Melin, D. Genoud, "POLYCOST: A Telephone-Speech Database for Speaker Recognition", RLA2C, pp. 211-214, Avignon, France, April 20-23, 1998.

[135] J.P. Campbell, "Testing with The YOHO CD-ROM Voice Verification Corpus", ICASSP. Detroit, May 1995,pp. 341-344.

[136] J.P. Campbell, D.A. Reynolds, "Corpora for the Evaluation of Speaker Recognition Systems", International Conf. on acoustics, Speech and Signal Processing (ICASSP), 1999.

[137] H. Alan, V. Dave, "KING Speaker Verification", Linguistic Data Consortium, 1995.

[138] J.J. Godfrey, E.C. Holliman, J. McDaniel, "SWITCHBOARD: telephone speech corpus for research and development", International Conf. on acoustics, speech and Signal processing (ICASSP), 1992.

[139] 2001 NIST Speaker Recognition Evaluation.
http://www.itl.nist.gov/iad/mig/tests/spk/2001/

[140] 2002 NIST Speaker Recognition Evaluation.
http://www.itl.nist.gov/iad/mig/tests/spk/2002/

[141] 2004 NIST Speaker Recognition Evaluation.
http://www.itl.nist.gov/iad/mig/tests/spk/2004/

[142] A. Martin, G. Doddington, T. Kamm, M. Ordowski, M. Przybock, "The DET Curve in Assessment of Detection Task Performance", EUROSPEECH, pp.1895-1898, 1997.

[143] J. Oglesby, "What's in a number? Moving beyond the equal error rate", Speech Communication, Vol. 17, pp. 193-208, 1995.

[144] D.A. Reynolds, R.C. Rose, M. J. T. Smith, "PC-based TMS320C30 implementation of the Gaussian mixture model text-independent speaker recognition system", In Proceedings of the International Conference on Signal Processing Applications and Technology, pp. 967–973, November 1992.

[145] Z. Shi-xiong, M. Man-wai, M.M. Helen, "Speaker verification via high-level feature-based phonetic-class pronunciation modeling", IEEE Transactions on Computers, Vol. 56, No. 9, Sep 2007.

[146] A. Biem, S. katagiri, "Cepstrum based filter bank design using discriminative feature extraction training at various levels", In ICASSP 1997.

[147] A. Biem, et. al, "An application of discriminative feature extraction to filter bank based speech recognition", IEEE transactions on speech and audio processing, Vol.9, No.2, pp.96-110, 2001.

[148] J.E. Luck, "Automatic speaker verification using cepstral measurements," Journal of Acoustical Society of America, vol. 46(2), pp. 1026-32, Nov. 1969.

[149] G. Doddington, "Speaker recognition -identifying people by their voices," Proc. IEEE, vol. 73, pp. 1651-64, 1985.

[150] K.P. Li, J.E. Dammann, W.D. Chapman, "Experimental studies in speaker verification using an adaptive system," Journal of Acoustical Society of America, vol. 40(5), pp. 966-78, Nov. 1966.

[151] B.S. Atal, "Automatic speaker recognition based on pitch contours," Journal of Acoustical Society of America, vol. 52, no. 6(part 2), pp. 1687-97, 1972.

[152] D.A. Reynolds, "Experimental evaluation of features for robust speaker identification," IEEE Trans. Speech Audio Process., vol. 2(4), pp. 639-43, Oct. 1994.

[153] 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.

[154] 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. 1243-61, 2006.

[155] 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.

[156] 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.

[157] 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.

[158] L. Mary, K.S. Rao, S.V. Gangashetty, 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.

[159] F. Farahani, P.G. Georgiou, 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.

[160] F. Weber, L. Manganaro, B. Peskin, 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. 141-4.

[161] B.S. Atal, S.L. Hanauer, " Speech analysis and synthesis by linear prediction of the speech wave.", In journal of the acoustical society of America. Vol.50, No.2, pp. 637-655, 1971.

[162] B.S. Atal, "Effectiveness of linear prediction characteristics of the speech wave for automatic speaker identification and verification." In journal of the acoustical society of America. Vol.55, No.6, pp. 1304-1312, 1974.

[163] S.B. Davis, P. Mermelstein, "Comparison of parametric representation for monosyllabic word recognition in continuously spoken sentence", In IEEE transactions on acoustic speech and signal processing. Vol. 28, No.4,pp. 357-366, 1980.

[164] A.V. Oppenheim, "A speech analysis-synthesis system based on homomorphic filtering. In journal of the acoustical society of America.", Vol.45, pp. 458-465. 1969.

[165] de la Torre, A., et al, "A DFE-based algorithm for feature selection in speech recognition", ICASSP 1997.

[166] H. Hermansky, "Perceptual linear predictive (PLP) analysis for speech", In journal of the acoustical society of America. Vol.87, No.4, pp. 1738-1752.

[167] K.T. Assaleh, R.J. Mammone, "Robust cepstral feature for speaker identification.", In proceedings of the IEEE ICASSP, Vol 1, pp. 129-132, April 1994.

[168] K.T. Assaleh, R.J. Mammone, "New LP derived features for speaker identification.", IEEE transactions on speech and audio processing. Vol. 2 No. 4, pp. 630-638, 2002.

[169] Y. Liu, M. Russell, M. Carey, " The Role of Dynamic Features in Text-Dependent and Independent Speaker Verification", IEEE international conf on acousto=ics, speech and signal processing (ICASSP), Vol. 1, May 2006.

[170] N. Mohaddeseh, A. Eliathamby, E. Julien, "Speaker Verification Using A Novel Set of Dynamic Features," International Conference on Pattern Recognition, , vol. 4, pp. 266-269, Vol. 4, 2006.

[171] S.A. Eric, K. Akira, N.M. Mariko, M. Perez, "speaker recogmntion using gaussian mixture models" Lecture notes in computer science, Bio inspired application of connectionism, pp.287-294, Springer Verlag, Berlin 2001.

[172] C. Griffin, T. Matsui, and S. Furui., " Distance measures for text-independent speaker recognition based on MAR model.", In ICASSP94,Vol.1, pages 309—312, 1994.

[173] W. Yutai, L. Bo, J. Xiaoqing, L. Feng, W. Lihao, "Speaker recognition based on dynamic MFCC parameters"; International Conference on Image Analysis and Signal Processing, pp. 406-409, 2009.

[174] T. Kinnunen, E. Koh, L.Wang, H. Li, and E. Chng, "Temporal discrete cosine transform: Towards longer term temporal features for speaker verification," in Proc. Of 5th Int. Symposium on Chinese Spoken Language Processing, pp. 547–558, Singapore, December 2006.

[175] S. Young, The HTK Book: for HTK Version 2.1, Cambridge, England: Cambridge University Press, 1997.

[176] H. Fujisaki, K. Hiros, "Analysis of voice fundamental frequency contours for declarative sentences of Japanese", In journal of acoustical society of Japan (E), vol. 5, No.4, pp. 233,241, 1984.

[177] E. Bozkurt, E. Erzin, C.E. Erdem, A.T. Erdem, "Automatic emotion recognition for facial expression animation from speech", 17th IEEE conference on signal processing and communications, pp. 989-992, 2009.

[178] G. McLachlan, T. Krishnan, *The EM Algorithm and Extensions*. New York: Wiley, 1997.

[179] M. Law, M. Figueiredo, A. Jain, "Simultaneous feature selection and clustering using mixture models," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 9, pp. 1154–1166, Sep. 2004.

[180] G. Doddington, "Speaker recognition based on idiolectal differences between speakers" European Conference on Speech Processing Technology, pp. 2521–2524, Aalborg, Denmark, Sep. 2001.

[181] S.S. Stevens, J. Volkmann, "The relation of pitch to frequency: A revised Scale", The American Journal of Psychology, Vol. 53, No. 3, pp. 329-353, July 1940.

[182] H. Fletcher, "Auditory patterns", In reviews of modern physics, vol. 12, pp. 47-65, 1940.

[183] E. Zwicker, “subdivision of the audible frequency range into critical bands”, In journal of acoustical society of America, Vol. 33, pp. 248-249, 1961.

[184] E. Zwicker, E. Terhardt, ”Analytical expressions for critical band rate and critical bandwidth as a function of frequency” In journal of acoustical society of America, Vol. 68, No.5, pp. 1523-1525, 1980.

[185] B.C.J. Moore, An introduction to the psychology of hearing, Academic press London 5th Ed. 2003.

[186] C. Ke, W. Lan, C. Huisheng, “Methods of Combining Multiple Classifiers with Different Features and Their Applications to Text-Independent Speaker Identification” International Journal of Pattern Recognition and Artificial Intelligence, vol.11, No.3, pp. 417-445, 1997.

[187] F. Zheng, G. Zhang, Z. Song, “Comparison of different implementations of MFCC”,Journal of Computer Science & Technology, vol. 16 no. 6, pp. 582-589, Sept. 2001.

[188] T. Ganchev, N. Fakotakis, G. Kokkinakis, “Comparative Evaluation of Various MFCC Implementations on the Speaker Verification Task”, Proceedings of SPECOM 2005, Vol.1, pp.191-194, October 2005.

[189] S. Young, et. al, “HTK Book”, http://htk.eng.cam.ac.uk/, HTK Version 3.4 March 2009.

[190] S. Malcolm, “Auditory Toolbox Version 2”, 1998, Interval Research Corporation, CA.

[191] D.S. Mark, G.H. John, “Exploiting independent filter bandwidth of human factor cepstral coefficients in automatic speech recognition”, Journal of Acoustical society of America, Vol. 116, No. 3, pp. 1774-1780, Sep 2004.

[192] M. Faundez-Zanuy, E. Monte-Moreno, “State-of-the-art in speaker recognition”, IEEE Aerospace and Electronic Systems Magazine,Vol.20, No. 5, pp. 7-12, Mar. 2005.

[193] B. Gold, N. Morgan, “ *Speech and Audio Signal Processing”*, Part- IV, Chap.14, pp. 189-203, John Willy & Sons ,2002.

[194] S. Khosla, S. Murugappan, E. Gutmark, "What can vortices tell us about vocal fold vibration and voice production" Current opinion in otolaryngology and head and neck surgery, 2008, Vol 16; No. 3, pp. 183-187.

[195] A. Biem, S. katagiri, B.H Juang, "Pattern recognition using discriminative feature extraction", IEEE Transaction on signal processing, Vol. 45, No.2, pp.500-504, 1997.

[196] G. Zhau, J.H.L. Hansen, J.F. Kaiser, "Non-linear feature based classification of speech under stress," IEEE Transactions on Speech and Audio Processing, vol. 9, pp. 201-216, 2001.

[197] H.M. Teager, "Some observations on oral air flow during phonation," IEEE Transactions on Acoustics, Speech, Signal Processing, vol. 28, pp. 599-601, 1980.

[198] H. Nam, H.-S.Kim, "Speaker verification system using hybrid model with pitch detection by wavelets", Proceedings of the IEEE-SP International Symposium on Time frequency and Time scale analysis, pp.153-156, 1998.

[199] H.M. Torres, H.L. Rufiner, "Automatic speaker identification by means of Mel cepstrum, wavelets and wavelet packets", In 22nd EMBS conference 2000.

[200] C.-T. Hsieh,Y.C. Wang, "A robust speaker identification system based on wavelet transform"' IEICE Transactions, 2001. Vol. 84, No.7, pp.839-846.

[201] E. Barnard, "Performance and generalization of the classification figure of merit criterion function", IEEE transactions on neural networks, 1991, Vol.2, No.2, pp.322-325.

[202] A. Biem, S. katagiri, "Filter bank design based on discriminative feature extraction"' In ICASSP 1994.

[203] H.M. Teager, "Evidence for nonlinear sound production mechanisms in the vocal tract", Speech production and speech modelling, 1990.

[204] H.M. Teager, "A phenomenological model for vowel production in the vocal tract", Speech Science: Recent Advances, 1982.

[205] J. Kaiser, "Some observations on vocal tract operation from a fluid flow point of view", Vocal Fold Physiology: Biomechanics, Acoustics, and Phonatory Control, pp. 358-386, 1983.

[206] K.T. Assaleh, "Supplementary orthogonal cepstral features", In ICASSP 1995.

[207] R. Sethuraman, J.N.Gowdy, "A cepstral based speaker recognition system", In 21st Southeastern symposium on System Theory, 1989.

[208] RAND Corporation study, 2008.

[209] P. Maragos, J.F. Kaiser, T.F. Quatieri, "Energy separation in signal modulations with application to speech analysis", IEEE Transactions on Signal Processing, Vol.41, No.10, pp.3024-3051, 1993.

[210] P. Maragos, J.F. Kaiser, T.F. Quatieri, "On amplitude and frequency demodulation using energy operators", IEEE Transactions on Signal Processing, Vol. 41, No.4, pp.1532-1550, 1993.

[211] A.P. Hemant, T.K. Basu, "Identifying Perceptually Similar Languages Using Teager Energy Based Cepstrum", Engineering Letters, Vol. 16, No. 1, 2008.

[212] F. Jabloun, A.E. Cetin, "The Teager energy based feature parameters for robust parameters in car noise," International conference on acoustics, speech and signal processing, vol. 1, pp. 273-276, 1999.

[213] C-T. Lu, H-C. Wang, "Enhancement of single channel speech based on masking property and wavelet transform," Speech Communication, vol. 41, pp. 409-427, 2003.

[214] B. Scharf, "Critical bands," in Foundations of Modern Auditory Theory, J. V. Tobias, Ed. New York: Academic, 1970, vol. 1, pp.157–202.

[215] W.A. Yost, Fundamentals of Hearing, 3rd ed. New York: Academic, 1994, pp. 153–167.

[216] C.S. Burrus, R.A. Gopinath, H. Guo, Introduction to Wavelets and Wavelet Transforms, A Primer, Upper Saddle River, Nj: Prentice-Hall, 1998.

[217] I. Daubechies, Ten Lectures on Wavelets, CBMS, SIAM Publ., 1992.

[218] M.S. Zilovic, R.P. Ramachandran, R.J. Mammone, "The use of robust cepstral features obtained from pole-zero transfer functions for speaker identification". In Canadian conference on electrical and computer engineering. 1995.

[219] M.M. Homayounpour, G. Chollet, "A comparison of some relevant parametric representation for speaker verification", In ESCA Workshop on automatic speaker recognition, Identification and verification.

[220] S. George, et al; "Speaker recognition using dynamic synapse based neural networks with wavelet preprocessing", 2001, pp.1122-1125.

[221] F. Phan, E. Micheli-Tzanakou, and S. Sideman, "Speaker identification using gamma neural networks", In International conference on neural networks, 1997.

[222] F. Phan, E. Micheli-Tzanakou, and S. Sideman, "Speaker identification with wavelet decomposition and neural networks", 16th IEEE conference on Engineering in medicine and biology society 1994.

[223] S.C. Woo, C.P. Lim, R. Osman, "Development of a speaker recognition system using wavelets and artificial neural networks", In International symposium on Intelligent multimedia, video and speech processing 2001.

[224] R. Sarikaya, B. Pellom, and J.H.L Hansen, "Wavelet Packet Transform Features with Application to Speaker Identification", In NORSIG-98, IEEE Nordic Signal Processing Symposium, pp. 81–84, Vigso, Denmark, 1998.

[225] I. Karlsson, T. Banziger, J. Dankovicova, T. Johnstone, J. Lindberg, H. Melin, F. Nolan, K. Scherer, "Speaker verification with elicited speaking styles in the VeriVox project," Speech Communication 31(2-3): 121-129, 2000.

[226] K. R. Scherer, T. Johnstone, G. Klasmeyer, T. Bänziger "Can automatic speaker verification be improved by training the algorithms on emotional speech", 6th international conference on spoken language processing, China, Oct 2000.

[227] W.S. Mohans, "Statistical feature evaluation in speaker identification", In department of electrical engineering. 1969, North Carolina state university: North Carolina.

[228] E.I.I Moore, M. Clements, J. Peifer, L. Weisser, "Comparing objective feature statistics for classifying clinical depression", IEMBS'04, Vol.1, pp.17-20, Sep 2004.

[229] L. Sheeber, H. Hops, J. Andrews, T. Alpert, and B. Davis, "Interactional processes in families with depressed and non depressed adolescents: reinforcement of depressive behaviour," Behaviour Research and Therapy, vol. 36, pp. 417-427, 1998.

[230] H. Hops, A. Biglan, A. Tolman, L. Sherman, J. Arthur, and N. Longoria, "Living in family environments (LIFE) coding system: Reference manual for coders," Oregon Research Institute, Eugene, OR, Unpublished manuscript , 2003.

[231] H. Hops, B. Davis, and N. Longoria, "Methodological issues in direct observation-illustrations with the living in familial environments (LIFE) coding system," Journal of Clinical Child Psychology, vol. 24, pp. 193-203, 1995.

[232] N. Longoria, L. Sheeber, B. Davis, "Living in family environment coding", A reference model for coders, OREGON Research Institute, 2006.

[233] R. Price, J. Willmore, W. Roberts, "Genetically Optimized Feed forward Neural Networks for Speaker Identification", Information Technology Division, Electronics and Surveillance Research Laboratory, DSTO-TN-0203

[234] D.F. Specht, "Probabilistic Neural Networks", *Neural Networks*, vol. 3, pp.109-118,1990.

[235] D. J. France, R. G. Shiavi, S. Silverman, M. Silverman, D. M. Wilkes, "Acoustical properties of speech as indicators of depression and suicidal risk," IEEE Transactions Biomed. Eng., vol. 47, no. 7, pp. 829–837, July 2000.

[236] A. Ozdas, D. M. Wilkes, R. G. Shiavi, S. E. Silverman, and M. K. Silverman, "Analysis of fundamental frequency for near term suicidal risk assessment," in Proceedings, IEEE Int. Conf. on Systems, Man and Cybernetics, vol. 3, 2000, pp. 1853–1858.

[237] E. Moore, M. Clements, J. Peifer, and L. Weisser, "Analysis of prosodic variation in speech for clinical depression," in Proceedings, 25th Annual Conference on Engineering in Medicine and Biology, 2003, pp. 2925–2928.

[238] A.W. Drake Fundamentals of Applied Probability Theory, McGraw-Hill, New Yourk, 1967.

[239] T. F. Quatieri, Discrete-Time Speech Signal Processing Principles and Practice, Prentice-Hall Signal Processing Series, 2002.

[240] D.A. Reynolds, "Effects of Population Size and Telephone Degradation on Speaker Identification Performance", SPIE Conference on Automatic Systems for Identification and Inspection of Humans, 1994.

[241] D.A. Reynolds, "Speaker Identification and Verification Using Gaussian Mixture Speaker models", ESCA Workshop on Automatic Speaker Recognition, pp. 27-30, Martigny, Switzerland, 1994.

[242] D.A. Reynolds, "A Gaussian Mixture Modelling Approach to Text-Independent Speaker Identification", PhD Thesis, Georgia Institute of technology, Atlanta, GA 1992.

[243] Q. Lin et. al., "Selective use of the speech spectrum and a VQGMM method for speaker identification", ICSLP, pp. 2415-2418, 1996.

[244] S. Slomka, "Multiple classifier structures for automatic speaker recognition under adverse conditions", PhD thesis, QUT, Brisbane Australia, 1999.

[245] J. F. Kaiser, "On a simple algorithm to calculate the 'energy' of a signal," in Proceedings of International Conference on Acoustic, Speech and Signal Processing, vol. 1, pp. 381-384, 1990.

[246] W. Kim, and J. H. L. Hansen, "Robust angry speech detection employing TEO-based discriminative classifier combination," in *Proc. Interspeech*, 2009, pp. 2019-2022.

[247] L. He, M. Lech, S. Memon, N. Allen, "Recognition of Stress in Speech Using Wavelet Analysis and Teager Energy Operator", Interspeech 2008, 22-26 September, Brisbane, Australia.

[248] K. R. Scherer, T. Johnstone, G. Klasmeyer, T. Bänziger "Can automatic speaker verification be improved by training the algorithms on emotional speech" University of Geneva, Switzerland.

[249] A. Barney, *et al*. "Fluid flow in a dynamic mechanical model of the vocal folds and tract", JASA.1999, 105(1), pp. 444-455.

[250] D. Shinwari, R.C. Scherer, A. Afjey, K. Dewitt, Flow visualization in a model of the glottis with a symmetric and oblique angle. JASA 2003;113:487–497.

[251] T. Becker, M. Jessen, C. Grigoras, "Forensic Speaker Verification Using Formant Features and Gaussian Mixture Models", Interspeech 2008.

[252] L.A. Low, N.C. Maddage, M. Lech, L. Sheeber, N.B. Allen, "Mel frequency cepstral features and Gaussian mixtures for modeling of clinical depression in adolescents", ICASSP 2010, March 14-19 2010, Dallas, Texas.

[253] L.A. Low, N.C. Maddage, M. Lech, L. Sheeber, N.B. Allen, "Content based clinical depression detection in adolescents" , EUSIPCO'2009, August 24-28, Glasgow, Scotland.

[254] L.A. Low, N.C. Maddage, M. Lech, N.B. Allen, "Mel Frequency Cepstral Features and Gaussian Mixtures for Modelling Clinical Depression in Adolescents", ICCI 2009, Hong Kong.

[255] W. C. Drevets, "Neuroimaging and Neuropathological Studies of Depression: Implication for the Cognitive-Emotional Features of Mood Disorders," *Journal of Current Opinion in Neurobiology,* Vol.11, No. 2, 2001.

[256] S. Kuny, H.H. Stassen, "Speaking Behaviour and Voice Sound Characteristics in Depressive Patients during Recovery" Journal of Psychiatric Research, Vol. 27, No. 3, pp. 289-307, 1993.

[257] World Health Organization (WHO) Mental Health Department [Online].Available:http://www.who.int/mental_health/management/depression/definition/en/

[258] Suicide and mental illness in the media, http://www.mindframe-media.info/site/index.cfm?display=85541

[259] Y. Gong, "A Method of Joint Compensation of Additive and Convolutive Distortions for Speaker-Independent Speech Recognition", IEEE Transactions on Speech and Audio Processing, Vol. 13, Issue. 5 , Part. 2, 2005.

[260] Y. Zhao, "Maximum likelihood joint estimation of channel and noise for robust speech recognition", Proceedings of International Conference on Acoustics, Speech, and Signal Processing, Vol. 2, 2000.

[261] Y. Shan, J. Liu, "Robust Speaker Recognition in Cross-Channel Condition, Image and Signal Processing", 2nd International Congress on Digital, 2009 , Page(s): 1 – 5.

[262] M. Wolfel, "Enhanced Speech Features by Single-Channel Joint Compensation of Noise and Reverberation", IEEE Transactions on Audio, Speech, and Language Processing, Vol.17, Issue. 2, Page(s): 312 - 323, 2009.

[263] T.F. Quatieri, D.A. Reynolds, G.C. O'Leary, "Estimation of handset nonlinearity with application to speaker recognition", IEEE Transactions on Speech and Audio Processing, Vol.8, Issue.5, Page(s): 567 – 584, , 2000.

[264] K.R. Scherer, T. Johnstone, T. Bänziger, "Automatic verification of emotionally stressed speakers" The problem of affective-sciences, SPECOM, 1998 - affective-sciences.org

[265] S. J. Blumenthal and D. J. Kupfer, *Suicide Over the Life Cycle: Risk Factors, Assessment and Treatment of Suicidal Patients*. Washington, DC: American Psychiatric, 1990, ch. 6.

[266] R. W. Hudgens, "Preventing suicide," *New Eng. J. Med.*, pp. 308–877, 1983.

[267] R. Summerfield, T. Dunstone, C. Summerfield, "Speaker Verification in a Multi-Vendor Environment",www.w3.org/2008/08/siv/Papers/Centrelink/w3c-sv_multi-vendor.pdf.

[268] M. Sokolov, "Speaker verification in the World Wide Web", Eurospeech 1997.

[269] NIST Speaker Recognition Evaluation.
http://www.itl.nist.gov/iad/mig/tests/sre/

[270] J. Liu and Y. Ye, "An instantiable speech biometrics module with natural language interface: Implementation in the telephony environment," Proc. of the ICASSP 2000, Istanbul, Turkey, June 2000.

[271] J. Liu and Y. Ye "Conversational Speech Biometrics," Chapter in "E-Commerce Agents Marketplace Solutions, Security Issues, and Supply and Demand," (Eds.): Springer Verlag, 2001, Pages 166-179.

[272] S. Khosla, S. Murugappan, E. Gutmark, "What can vortices tell us about vocal vibration and voice production", Current opinion in Ot-holaryngology & Head and Neck Surgery, vol. 16, 2008.

[273] S. Khosla, S. Murugappan, R. Paniello, J. Ying, E. Gutmark, " Role of vortices in voice production: Norma versus assymetric tension", The Larygoscope, 119, pp. 216-221, January 2009.

[274] Ling He, Margaret Lech, Ian Burnett and Nicholas Allen, "Nonlinear Features for Stress and Emotion Classification in Natural Speech", IEEE Transactions on Affective Computing, under review.

[275] C. SHI-HUANG, "Speech Enhancement Using Perceptual Wavelet Packet Decomposition and Teager Energy Operator", Journal of VLSI Signal Processing, Vol. 36, pp.125–139, 2004.

Appendix A

Analogous to Eq. (5.3), the filter bank coefficients for the reversed MFCC can be defined as,

$$\hat{\mathbf{H}}_{\mathbf{i}}(\mathbf{k}) = \begin{cases} 0 & for \quad k < \hat{f}_{b_{i-1}} \\ \dfrac{k - \hat{f}_{b_{i-1}}}{\hat{f}_{b_i} - \hat{f}_{b_{i-1}}} & for \quad \hat{f}_{b_{i-1}} \le k \le \hat{f}_{b_i} \\ \dfrac{\hat{f}_{b_{i+1}} - k}{\hat{f}_{b_{i+1}} - \hat{f}_{b_i}} & for \quad \hat{f}_{b_i} \le k \le \hat{f}_{b_{i+1}} \\ 0 & for \quad k > \hat{f}_{b_{i+1}} \end{cases} \tag{1}$$

Where $1 \le k \le N$ and $\hat{f}_{b_i}$ for $i=0$ to $M+1$ can be evaluated as,

$$\hat{f}_{b_i} = \frac{N}{2} + 1 - f_{b_{M+1-i}} \tag{2}$$

Therefore an equation analogous to Eq. (5.4) can be formulated as,

$$\hat{f}_{b_i} = \left(\frac{N}{F_s}\right) \hat{f}_{mel}^{-1} \left[\hat{f}_{mel}(f_{low}) + \frac{i\left\{\hat{f}_{mel}(f_{high}) - \hat{f}_{mel}(f_{low})\right\}}{M+1} \right] \tag{3}$$

Thus $f_{b_{M+1-i}}$ can be obtained as,

$$f_{b_{M+1-i}} = \left(\frac{N}{F_s}\right) f_{mel}^{-1} \left[f_{mel}(f_{low}) + \frac{(M+1-i)\left\{f_{mel}(f_{high}) - f_{mel}(f_{low})\right\}}{M+1} \right] \tag{4}$$

By substituting $\hat{f}_{b_i}$ and $f_{b_{M+1-i}}$ in Eq. (2), the following can be obtained,

$$\left(\frac{N}{F_s}\right)\hat{f}_{mel}^{-1}\left[\hat{f}_{mel}(f_{low})+\frac{i\left\{\hat{f}_{mel}(f_{high})-\hat{f}_{mel}(f_{low})\right\}}{M+1}\right]= \\ \frac{N}{2}+1-\left(\frac{N}{F_s}\right)f_{mel}^{-1}\left[f_{mel}(f_{low})+\frac{(M+1-i)\left\{f_{mel}(f_{high})-f_{mel}(f_{low})\right\}}{M+1}\right] \quad \textbf{(5)}$$

The above can be simplified as,

$$\hat{f}_{mel}^{-1}\left[\hat{f}_{mel}(f_{low})+\frac{i\left\{\hat{f}_{mel}(f_{high})-\hat{f}_{mel}(f_{low})\right\}}{M+1}\right]= \\ \frac{F_s}{2}+\frac{F_s}{N}-f_{mel}^{-1}\left[f_{mel}(f_{high})-f_{mel}(f_{low})+f_{mel}(f_{low})-\frac{i\left\{f_{mel}(f_{high})-f_{mel}(f_{low})\right\}}{M+1}\right] \quad \textbf{(6)}$$

The reversed mel scale shares the common boundary points with the actual mel scale so,

$\hat{f}_{mel}(f_{high})=f_{mel}(f_{high})$ and $\hat{f}_{mel}(f_{low})=f_{mel}(f_{low})$

Thus Eq. (6) becomes,

$$\hat{f}_{mel}^{-1}\left[f_{mel}(f_{low})+\frac{i\left\{f_{mel}(f_{high})-f_{mel}(f_{low})\right\}}{M+1}\right]= \\ \frac{F_s}{2}+\frac{N}{F_s}-f_{mel}^{-1}\left[f_{mel}(f_{high})-f_{mel}(f_{low})+f_{mel}(f_{low})-\frac{i\left\{f_{mel}(f_{high})-f_{mel}(f_{low})\right\}}{M+1}\right] \quad \textbf{(7)}$$

Let,

$$f=\hat{f}_{mel}^{-1}\left[f_{mel}(f_{low})+\frac{i\left\{f_{mel}(f_{high})-f_{mel}(f_{low})\right\}}{M+1}\right] \quad \textbf{(8)}$$

Thus Eq. (7) becomes,

$$f = \frac{F_s}{2} + \frac{N}{F_s} - f_{mel}^{-1}\left[f_{mel}(f_{high}) - f_{mel}(f_{low}) + f_{mel}(f_{low}) - \frac{i\{f_{mel}(f_{high}) - f_{mel}(f_{low})\}}{M+1} \right] \quad \textbf{(9)}$$

Further it can be simplified as,

$$f_{mel}^{-1}\left[f_{mel}(f_{high}) - f_{mel}(f_{low}) + f_{mel}(f_{low}) - \frac{i\{f_{mel}(f_{high}) - f_{mel}(f_{low})\}}{M+1} \right] = \frac{F_s}{2} + \frac{N}{F_s} - f \quad \textbf{(10)}$$

or,

$$\left[f_{mel}(f_{high}) - f_{mel}(f_{low}) + f_{mel}(f_{low}) - \frac{i\{f_{mel}(f_{high}) - f_{mel}(f_{low})\}}{M+1} \right] = f_{mel}\left[\frac{F_s}{2} + \frac{N}{F_s} - f \right] \quad \textbf{(11)}$$

or,

$$\left[f_{mel}(f_{low}) + \frac{i\{f_{mel}(f_{high}) - f_{mel}(f_{low})\}}{M+1} \right] = f_{mel}(f_{high}) + f_{mel}(f_{low}) - f_{mel}\left[\frac{F_s}{2} + \frac{N}{F_s} - f \right] \quad \textbf{(12)}$$

Re-arranging Eq. (12), the following can be obtained,

$$\hat{f}_{mel}(f) = \left[f_{mel}(f_{low}) + \frac{i\{f_{mel}(f_{high}) - f_{mel}(f_{low})\}}{M+1} \right] \quad \textbf{(13)}$$

Thus Eq. (12) can be written as,

$$\hat{f}_{mel}(f) = f_{mel}(f_{high}) + f_{mel}(f_{low}) - f_{mel}\left[\frac{F_s}{2} + \frac{N}{F_s} - f \right] \quad \textbf{(14)}$$

Assuming, $F_s = 8KHz$, N=256,

Using, $f_{low} = \frac{F_s}{N} = 31.25Hz$, and $f_{lhigh} = \frac{F_s}{2} = 4KHz$, Eq. (14) can be simplified as,

$$\hat{f}_{mel}(f) = 2195.2860 - 2595\log_{10}\left(1 + \frac{4031.25 - f}{700}\right) \quad \textbf{(15)}$$

Where $\hat{f}_{mel}$ is the inverted mel scale pitch value in mels.