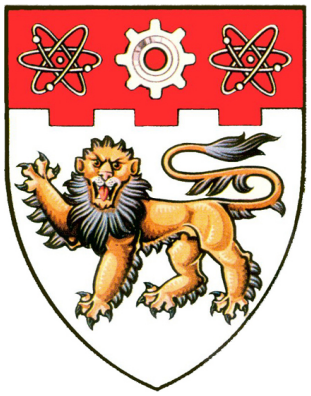
Robust Text-Independent Speaker Verification in Environmental Noise



Ashish Panda

School of Computer Engineering

Nanyang Technological University

A thesis submitted to the Nanyang Technological University in fulfilment of the requirement for the degree of

Doctor of Philosophy

2011

To
My Parents
and
Teachers.

Acknowledgements

I would like to express my gratitude to Prof. T. Srikanthan, my supervisor, for his support, guidance and extraordinary patience. Without his understanding approach, I would not have been able to complete this work.

I thank Alok Prakash for his help in some of my experiments. I also gratefully acknowledge the help provided by Dr. Jim Glass of MIT Computer Science and Artificial Intelligence Laboratory in acquiring the MITMDSVC database.

For the entire duration of my research work, I have enjoyed an excellent research environment in the Centre for High Performance Embedded Systems (CHiPES). For this, I thank the Chipeans, especially the laboratory executives and technical staffs: Ms. Nah, Merilyn and Jeremiah.

Through their acts of support, affection and understanding, my family (Baa, Maa, Nanis and Sweta) and my friends (Bharath, Abhijit, Santanu, Fauzi, Jagadeesh, Alok and Amit) acted as the barrier between me and failure. They will always have a special place in my heart.

Abstract

Automatic speaker verification has many potential applications in security, surveillance and access control. In many of these applications, it is necessary to verify the speaker based on a short and noise degraded speech utterance. This thesis addresses the problem of robust speaker verification in environmental noise conditions by introducing novel and computationally efficient techniques that are suitable for realistic conditions. It also engenders the application of psychoacoustics to realize an adaptive model compensation technique.

The probabilistic spectral subtraction (PSS) technique was investigated in detail and subsequently extended to accommodate noisy training utterance through a novel training scheme. The proposed training scheme has been shown to reduce the equal error rate, on an average, by 20% over the conventional procedure. The parallel model combination technique was investigated next due to its inherent compute efficient properties. While this provided further reduction in the equal error rate when compared to the PSS, it fell short in terms of inaccurate noise corruption function and its reliance on accurate noise estimation for better performance.

To address the issue of inaccurate noise corruption function, the max function, a non-linear function, was evaluated as an alternate noise corruption function. This led to the development of a new generalized compensation scheme in order to efficiently estimate the transformed model parameters for non-linear noise corruption functions. Experimental evaluations demonstrate that the proposed max function based compensation scheme is capable of providing better performance gain in white noise conditions. In addition, it was demonstrated that the additive function provides better performance in pink noise conditions.

In order to overcome the limitation that neither max function nor additive function can perform effectively across different types of noise, a novel psychoacoustic noise corruption function is proposed by exploiting masking

properties of noise and speech signals. The psychoacoustic noise corruption function and the generalized compensation scheme were then elegantly combined to propose a psychoacoustic model compensation technique, which is capable of effective performance across different types of noise. Experimental evaluations of the proposed psychoacoustic model compensation technique conclusively demonstrate that it provides superior performance in both white and pink noise conditions, outperforming parallel model combination by 36% and max function based model compensation by 24%.

A new multi-conditioning approach, based on the psychoacoustic model compensation, has also been proposed to deal with realistic and complex noise conditions. The proposed multi-conditioning has been shown to reduce the reliance on accurate noise estimation by simulating multiple levels of noise corruption. Experiments conducted, with the MIT mobile devices speaker verification corpus containing realistic and challenging speech utterances, recorded in highly unfavourable conditions, demonstrate that the proposed multi-conditioning technique reduces the equal error rate, on an average, by 22% over the best performing white noise based multi-conditioning technique. Moreover, the computational complexity of the proposed multi-conditioning technique is also significantly reduced due to the avoidance of compute-intensive posterior union model. Finally, the proposed techniques should pave the way for realizing speaker-aware human-computer interactions in mass volume products.

Contents

List of Figures

List of Tables

Glossary

AME Acoustic Model Enhancement; A model domain noise robustness technique

ANN Artificial Neural Network; A mathematical model inspired by the structures and functions of biological neural networks

ARMA Auto Regressive Moving Average; A mathematical model usually applied to time series data

dB Decibel; A logarithmic unit that indicates a ratio

DCT Discrete Cosine Transform; A transform that expresses finitely many data points as sum of cosine function oscillating at different frequencies

EER Equal Error Rate; A measure of the performance of speaker verification systems

EM Expectation Maximization; A parameter estimation algorithm

FFT Fast Fourier Transform; An efficient algorithm to compute discrete Fourier transform

GMM Gaussian Mixture Model; A probabilistic model for density estimation

HLDA Heteroscedastic Linear Discriminant Analysis; A method used in statistics to find a linear combination of features

HMM Hidden Markov Model; A statistical model which assumes the system being modeled as being a Markov's process

LMS Least Mean Squares; A parameter estimation method

LP Linear Prediction; A mathematical operation where future values of a discrete-time signal are estimated as a linear function of the previous samples

MFCC Mel Frequency Cepstral Coefficient; A feature for speech signals

PMC Parallel Model Combination; A model domain noise robustness technique

PSS Probabilistic Spectral Subtraction; A feature vector domain noise robustness technique, a variation of the spectral subtraction technique

PUM Posterior Union Model; A probabilistic method for finding the optimal combination of feature dimensions

RASTA Relative Spectral; A feature vector domain channel robustness technique

SNR Signal-to-Noise Ratio; A measure of how much a signal has been corrupted by noise

SS Spectral Subtraction; A feature domain noise robustness technique

TIMIT TI-MIT(Texas Instruments - Massachusetts Institute of Technology); A speech database with high quality clean speech

UBM Universal Background Model; A background model for speaker verification

1

Introduction

1.1 Motivation

Biometric identity verification has a long history. Handwriting and fingerprint verifications are some of the earliest known biometric schemes to check fraud. Currently, there are many biometric solutions for security based on iris scan, retina scan etc. Voice biometric has piqued the interest of researchers ever since Atal's effort to recognize speakers based on the pitch contours [1]. Even though recognition accuracy of voice biometric cannot, currently, match the accuracy of iris scan or retina scan, popular interest in it remains high due to its non-intrusive characteristics. Besides, an ubiquitous microphone is all that is required to collect a voice sample and therefore the cost saving potential of voice biometric is very high.

Voice biometric, also known as voice scan or speaker recognition or speaker verification, has many applications. The applications span over almost all the areas where it is desirable to secure actions, transactions or any type of interaction by identifying or authenticating the person making the transaction. Apart from forensic applications, the use of speaker recognition can be envisaged in four different areas [2]: on-site applications, remote applications, information structuring and games. On-site applications are applications where the user needs to be in front of the system to be authenticated. Access control to facilities, such as a car or a house, is an example of on-site application. Remote applications involve remote access of a system. User access to these

systems is, generally, through a telephone or a computer. Internet banking is an example of remote access. Organizing the information in audio documents falls under information structuring. Typical examples of this type of application are automatic annotation of audio archives and speaker indexing of sound tracks. Finally, speaker recognition can be applied in toys and video games etc. to improve the interactivity and user experience.

Over the last decade, speaker recognition technology has made its debut in some commercial products. Most of these products are based on scenarios with cooperative users speaking fixed digit-string passwords or uttering prompted phrases from a small vocabulary. This is known as a text-dependent or text-constrained system. Such constraints can, in general, improve the accuracy of a system. However, there are cases where such constraints can be impossible to enforce. An example of this is background verification where the speaker is verified without explicit cooperation from him/her. For cases like this, a more flexible recognition system is required where the system can perform independent of the spoken utterance. This is known as text-independent speaker recognition.

Current state-of-the-art speaker recognition systems perform extremely well in ideal conditions. Using speech collected with high quality microphone in noise-free environment, an accuracy of 99.5% has been achieved in speaker identification task with 630 speakers [3]. Using the same speech database, an equal error rate of 0.24% (A measure of error rate in speaker recognition, defined in Section 2.2.4) has been achieved in speaker recognition task [4]. However, the performance degrades to unacceptable levels with even moderate amounts of environmental noise. 109% reduction in performance due to environmental noise has been reported in [5] while a 206% drop in performance has been reported in [6].

From the intended applications of speaker recognition, described above, it can be clearly seen that the speaker recognition system is likely to operate in a variety of environments and it is not always possible to create a noise-free environment for speech sample collection. For example, it is difficult, if not impossible, to create a controlled environment for speech sample collection for access to a car in a street-side parking spot.Therefore, the usability of a speaker recognition system in real-life scenarios very much depends upon

its robustness against the environmental noise. Consequently, environmental noise has been recognized as a major impediment for real-life deployment of speaker recognition systems and noise robustness figures prominently in the future research trends of voice biometrics [2].

In this thesis, the research is aimed at real-life applications of text independent speaker recognition. First, this means that the speaker recognition system must be able to produce high recognition accuracy in real-life noisy conditions. From a practical point of view, in several applications, such as access to a car or remote access from a noisy street corner, there is no control over the type of prevailing noise. The noise can be non-stationary and difficult to predict or model. In order to maintain reliable recognition under these degraded conditions, it is important for the speaker recognition system to have some means to compensate for these type of distortions.

Second, any real-life application must take user inconvenience and cost of implementation into consideration. The speaker recognition system, in most applications, is likely to be an embedded system without much computing resources. Computationally complex algorithms pose a high demand on the processing capability of the system and hinder real-time response of the system, leading to user frustration. Such algorithms also require higher system memory and increase the cost of implementation. Therefore, it is highly desirable to design computationally efficient algorithms for noise robustness.

1.2 Major Contributions of the Thesis

The main objective of this thesis is to develop a robust and computationally efficient text-independent speaker verification system capable of high accuracy in unforeseen noisy conditions. To achieve this objective, two prominent existing approaches are studied: speech enhancement, which estimates the clean speech from the degraded speech, and model compensation, which introduces noise into the speaker model to achieve a matched condition. Speech enhancement is attractive because of its lower computational complexity and its ability to handle noisy training speech. Model compensation is attractive because of its high accuracy. After comparing the applicability and performance of both these approaches, model compensation approach is favoured.

Limitations of existing model compensation techniques are identified and novel methods are proposed and experimentally validated to overcome those limitations.

There are four major contributions of the work presented in this thesis. First is an effort to evaluate and compare the performance of speech enhancement technique and model compensation technique. The possibility of adding noise to the training speech for obtaining higher performance from the speech enhancement techniques is also investigated. Towards this goal a novel training scheme is derived. The proposed training scheme is shown to outperform the conventional training scheme. When compared to the model compensation scheme, however, the performance of speech enhancement technique is found to be less desirable. This motivates the employment of model compensation in the subsequent work. The drawbacks of existing model compensation techniques are also identified.

Second major contribution is the development of a new generalized compensation scheme, which can efficiently estimate the transformed model parameters for linear as well as non-linear noise corruption functions. Model compensation techniques employ a noise corruption function to specify the relationship between the clean speech and noisy speech and a compensation scheme to estimate the transformed or compensated model parameters. The estimation of compensated parameters, especially the compensated variance, for a non-linear corruption function is difficult. The proposed compensation scheme addresses this issue and can be used for any valid noise corruption function.

Third significant contribution is the innovative application of noise masking concepts for model compensation purpose. A novel psychoacoustic noise corruption function is developed which uses the masking thresholds, as defined by psychoacoustic model 1, to specify the relationship between the clean speech and the noisy speech. The proposed non-linear compensation scheme is employed to estimate the transformed parameters. Experiments demonstrate the excellent performance of psychoacoustic model compensation as compared to conventional model compensation schemes.

Fourth important contribution is the proposed psychoacoustic multiconditioning, which can deal with unforeseen and unpredictable noise. The model

compensation schemes require an estimation of prevalent noise and in many realistic scenarios such an estimation may not be possible. The psychoacoustic model compensation scheme provides a solution for such scenarios, which simulates a range of signal-to-noise-ratios to create multi-conditioned models. Experiments are conducted to demonstrate significant advantage of psychoacoustic multi-conditioning.

The work presented in this thesis has been communicated in international journals and conferences and has resulted in the following articles:

1. A. Panda, N. Tripathi and T. Sirkanthan,"Improved spectral subtraction technique for text-independent speaker verification", *International Conference on Digital Signal Processing*, July 2007, pp. 595-598.

2. A. Panda, A. Prakash and T. Srikanthan,"A probabilistic approach to spectral subtraction for robust text-independent speaker verification", *Journal of Signal Processing*, vol. 13, no. 5, pp. 423-430, 2009.

3. A. Panda and T. Srikanthan,"Noise adaptive models for robust speaker verification in noisy conditions", *Journal of Signal Processing*,vol. 15, no. 1, pp. 47-54, 2011.

4. A. Panda and T. Srikanthan, "Psychoacoustic model compensation for robust speaker verification in environmental noise", *IEEE Trans. on Audio, Speech and Language Processing*, [Accepted for publication]

1.3 Organization of the Thesis

The thesis is organized as follows. Chapter 2 provides a background review of the related techniques and methodologies. The basic concepts of speaker verification are explained and the methodologies to be followed for feature vector computation and speaker modeling are established. Subsequently the challenges in speaker verification are explored and the algorithms for noise robustness are analyzed and the most appropriate techniques are identified for further investigation.

Chapter 3 presents a comparative study of spectral subtraction technique (a type of speech enhancement approach) and parallel model combination

technique (a type of model compensation approach). These two techniques assume an additive noise corruption model. The difference in these two techniques lies in the domain of operation. While the spectral subtraction operates in the feature vector domain, the parallel model combination operates in the model domain. The study in this chapter suggests that model domain is more suitable for the operation of a noise corruption model.

Chapter 4 deals with the model domain compensation technique. It motivates the the search for a better noise corruption function. The max function has been suggested as an alternative noise corruption function. However, the non-linear characteristics of the max function presents significant challenges in estimating the compensated model parameters. This is addressed by a novel compensation scheme for non-linear noise corruption functions. Experiments conducted in this chapter suggest that while the max function noise corruption provides better performance in white noise, the additive noise corruption function provides better performance in pink noise.

Chapter 5 proposes the psychoacoustic model compensation technique. Based on the psychoacoustic principles, the motivation behind this technique is to determine the resultant audible signal when two competing signals are present. A novel noise corruption function is developed based on the masking thresholds defined by the psychoacoustic principles. The psychoacoustic noise corruption function can be seen as encompassing the max function and the additive function. The compensated model parameters are estimated by the compensation scheme for non-linear corruption functions developed in Chapter 4. Experimental results have been reported in this chapter, which clearly show the superiority of psychoacoustic model compensation technique.

Chapter 6 presents a computationally efficient method for creating multi-conditioned models for dealing with non-stationary and complex noise scenarios. The psychoacoustic model based compensation is used on scaled observed noise data to simulate models with different signal-to-noise-ratios. The proposed scheme is tested on a real-life noisy speech database and is shown to achieve significant performance gain.

Finally, Chapter 7 summarizes the major results and conclusions of the thesis and suggests future directions for research based on this work.

2

Literature Review

2.1 Introduction

The purpose of this chapter is to provide a review of the existing techniques and methodologies in the area of automatic speaker verification and noise robustness. Also, this chapter introduces the notations and nomenclature used throughout the thesis. Beginning with a description of the speaker verification process, various methods of feature extraction and speaker modeling are reviewed. Next, speech degradations encountered in speaker verification applications are discussed and their effects are outlined. Lastly, a comprehensive review of methods for robustness against environmental noise is undertaken and their strengths and weaknesses are assessed.

2.2 Speaker Verification

The task of speaker recognition can be divided into two classes: speaker identification and speaker verification [7]. Speaker identification involves identifying a speaker from a set of known speakers. For speaker identification, the system has access to the models of every enrolled speaker and identifies the most likely model for a given speech utterance. In this respect, speaker identification is a closed-set task. Speaker verification, on the other hand, is

an open-set task and has a claimed identity and a speech utterance to substantiate the claim. The decision of the system is, therefore, binary: either accept or reject the claim. Each of these tasks can be either text-dependent or text-independent. Figure 2.1 illustrates these divisions.

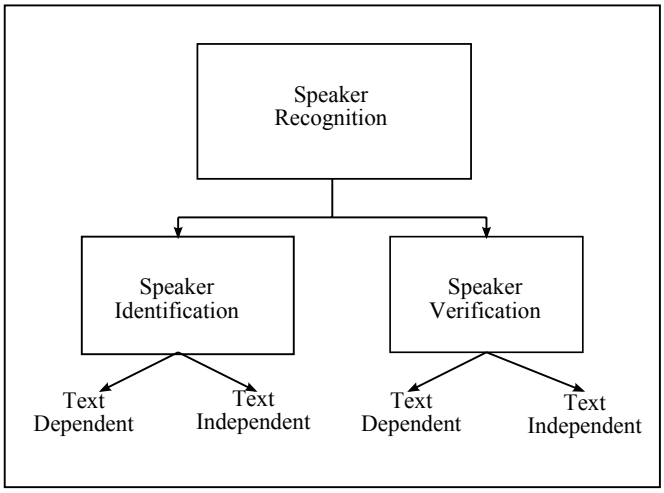


Figure 2.1: Division of Speaker Recognition Task - This figure illustrates the subclasses of the speaker recognition task

For text independent speaker recognition, the speaker is free to utter any phrase of his / her choice. For text-dependent speaker recognition, the speaker utters a pre-assigned passphrase or a prompted phrase from a limited vocabulary. The system recognizes the uttered text as well as the speaker in a text-dependent speaker recognition task. In text-dependent speaker recognition systems, the acoustic classes in the speaker models are same as the acoustic classes in the test utterance. Therefor, they provide better reliability and most of the commercial products use these techniques [2]. There are, however, scenarios involving non-cooperative users, where text-independent speaker recognition is a must. The current security concerns all over the world makes such systems even more indispensable.

Text-independent speaker verification consists of two distinct phases, a training phase and a test phase. Each of them can be seen as a succession of independent modules.The modules in training phase are feature vector extraction and speaker modeling, while the modules in test phase are feature vector extraction and scoring. Figure 2.2 illustrates the the different modules in training and test phases. The objective of the feature vector extraction

module is to identify and compute distinctive features from the speech signal. The model training module estimates the statistical distribution of the computed feature vectors. The scoring module computes a normalized match score, given a speech utterance and a model, and provides the decision. Each of these modules and the techniques involved are reviewed next.

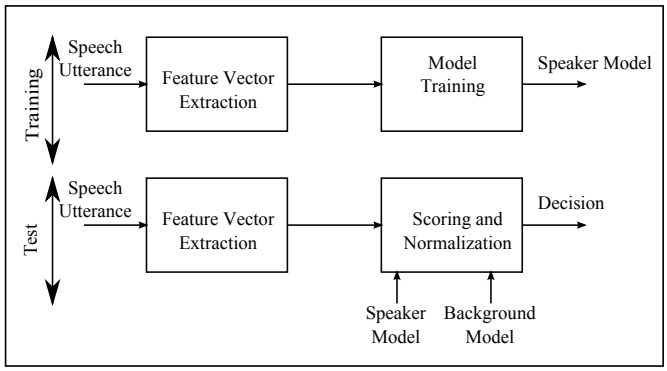


Figure 2.2: Phases of Speaker Verification Task - This figure illustrates the phases and modules of automatic speaker verification

2.2.1 Feature Vector Extraction

Feature vector extraction module transforms a speech signal into a set of feature vectors. The aim of this transformation is to obtain a new representation that is more compact, less redundant and more suitable for statistical modeling. Most current speaker verification systems rely upon a cepstral representation of the speech [2]. The cepstral representation is achieved either through Linear Prediction (LP) analysis or through filterbank approach.

The LP analysis [8] is based on a linear model of speech production and views each module of speech production, i.e., the glottal source, the vocal tract, the nasal tract and the lips, as a filter. The model usually used is an Auto Regressive Moving Average (ARMA) model and the speech production apparatus is represented by an ARMA filter. Characterizing the speech signal, therefore, translates into determining the coefficients of the filter. Several algorithms exist to compute these coefficients [9] and the cepstral coefficients can be calculated directly from these coefficients [10].

The filterbank approach for cepstral representation attempts to emulate the response of the human ear and to estimate the shape of vocal tract. A

filter bank is designed in a perceptual scale, such as Bark or mel [9], and the output of each filter, in frequency domain, is computed for a particular speech frame. The cepstral coefficient is calculated by computing logarithm of each filter-output and by applying Discrete Cosine Transform (DCT) to the log-filter-outputs [11].

Although there is no clear agreement on which of these cepstral representation is more efficient, there are some studies which indicate that filterbank approach would be more suitable for the purpose of noise robustness. One drawback of LP analysis is that it relies upon an all pole model, which may omit significant speech spectral characteristics for speech contaminated with noise [12]. The filterbank outputs, on the other hand, are direct measurements and are not subject to model constraints. Furthermore, the bandwidths of the filters can be adjusted in an advantageous manner to better capture the perceptually important characteristics of the speech signal [13]. Also, since filterbank approach attempts to estimate the shape of vocal tract, these features should be less susceptible to mimicry [14]. Therefore, the filterbank approach to cepstral representation is chosen as the method for feature vector extraction and its implementation is described next.

The silence / unvoiced portion of the speech is discarded by using a speech activity detector. The speech activity detector used for this work is database specific and will be described along with the database in the experimental set-up section of the chapters. The speech utterance is divided into 20 milliseconds frames with 10 milliseconds overlap. Hamming window is applied to each of these frame and Fast Fourier transform (FFT) is calculated. The magnitude of the FFT is extracted and only first half of these points are kept, as the spectrum is symmetric. A set of overlapping triangular filters linearly spaced in mel-scale is designed. The mel-frequency, F_{mel}, is calculated from the frequency value in Hertz (Hz), F_{Hz}, by [2]:

$$F_{mel} = 1000 \frac{\log\left(1 + F_{Hz} / 1000\right)}{\log 2} \tag{2.1}$$

The filterbank outputs are calculated and logarithm is computed. DCT is applied on the log-mel-filterbank-outputs to arrive at, what is known as, Mel Frequency Cepstral Coefficients (MFCC). This method of MFCC computation

has been followed in [15]. Figure 2.3 illustrates the steps in MFCC computation. Throughout this work, 20 MFCC are computed from 27 mel-filters on a 20 millisecond speech frame. This follows from the research work carried out in [16] on the optimal numbers of filters and MFCC.

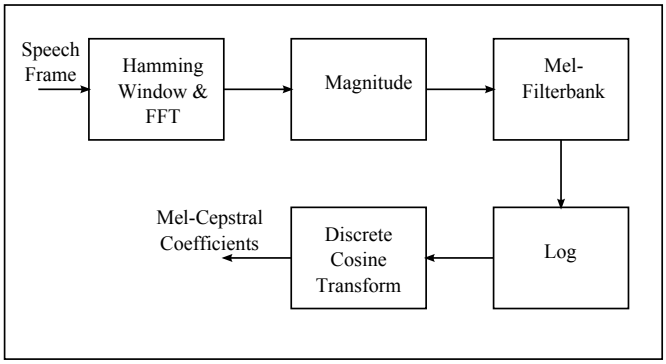


Figure 2.3: Mel Frequency Cepstral Coefficients Computation - This figure illustrates the process of cepstral computation by filterbank approach

2.2.2 Speaker Modeling

The objective of speaker modeling phase is to find the parameters of the underlying distribution of the feature vectors. These parameters are stored as a reference template for the speaker and are accessed by the scoring module during the test phase. The speaker modeling process serves two purposes. First, it discards the outliers in the training samples and second, it reduces the storage requirements by reducing the training set to manageable set of parameters. Several speaker modeling techniques have been proposed in the literature. Prominent among those are Artificial Neural Networks (ANNs), Support Vector Machines (SVMs) and Gaussian Mixture Models (GMMs).

ANNs use discrimination-based learning or prediction-based learning for speaker modeling. One of the earliest work on the application of ANN to speaker recognition [17] represented each speaker by a codebook designed by learning vector quantization. Multi-layer perceptron, radial basis functions or a hybrid of both has been used in several works [18] [19].

SVMs have received much attention in recent years. SVM classifiers are well suited to separate complex regions between two classes through optimal, nonlinear decision boundary [2]. In [20] [21], the speech feature vectors were

used as input material for the SVM. Efforts to combine GMM and SVM have been undertaken in [22] [23] [24]. Though these efforts have been moderately successful, further studies are required to achieve optimal combination of GMM and SVM.

The GMMs [25] represent the distribution of the feature vectors as a weighted combination of multivariate Gaussian distributions. The application of GMMs in speaker recognition task was proposed in [15] and it has acquired tremendous popularity since then. The reason for the success of GMMs is due to the fact that they can approximate, with sufficient accuracy, any complex distribution.

Although each of the above described methods have their advantages, GMMs appear to be the most suitable method of speaker modeling for our purpose. The main disadvantages of the ANNs are that their optimal structure has to be selected by trial-and-error procedure and the fact that the temporal structure of speech signals remains difficult to handle [2]. The performance of SVMs depends upon the appropriate kernel function. This, in addition to the inability to handle temporal structure of speech signals, makes SVMs less desirable for speaker verification [2]. Besides, as noted in [26], the computational complexity of SVM systems becomes prohibitive when noise robustness algorithms are added. Therefore, we adopt GMMs for speaker modeling and we describe them next.

For a D-dimensional feature vector, the density of a GMM with M component is a weighted linear combination of M Gaussian densities $p_i(\vec{x})$, each parameterized by a $D \times 1$ mean vector $\vec{\mu}_i$ and a $D \times D$ covariance matrix Σ_i. The density of each component is given by:

$$p_i(\vec{x}) = \frac{1}{(2\pi)^{D/2}|\Sigma_i|^{1/2}} e^{-(1/2)(\vec{x}-\vec{\mu}_i)'\Sigma_i^{-1}(\vec{x}-\vec{\mu}_i)} \tag{2.2}$$

The density of a GMM λ is given by:

$$p(\vec{x}|\lambda) = \sum_{i=1}^{M} w_i \, p_i(\vec{x}) \tag{2.3}$$

In the above equation, w_i stands for the weight of the i^{th} component and $\sum_{i=1}^{M} w_i = 1$. Collectively, the parameters of the GMM are denoted as $\lambda =$

$\{w_i, \vec{\mu}_i, \Sigma_i\},\ 1 \leq i \leq M$.

Consider a set of training feature vectors $X = \{\vec{x}_1, \vec{x}_2, \ldots, \vec{x}_T\}$. The aim of the training module is to find the parameter set λ that fits the training feature vectors best. The method of finding the parameter set is different for the decoupled GMMs and the adapted GMMs. In decoupled GMMs, the parameter set is estimated independently for each speaker and in adapted GMMs, the parameter set is adapted from a global model. The adapted GMMs provide superior performance to their decoupled counterpart [2]. This is because, in the adapted GMMs, the likelihood ratio is less likely to be affected by unseen acoustic classes in the recognition speech. The method of finding the parameters for adapted GMMs is described next.

The first step is to create a global model from a large amount of speech pooled from a large number of speakers. The initial parameter set of the global model is estimated by applying K-means algorithm [27]. Subsequently, the expectation-maximization (EM) algorithm [25] is used to refine the parameters. The iterative EM algorithm refines the parameters to monotonically increase the likelihood of the estimated model for the observed vectors. The parameters are said to have converged when the likelihood stops increasing from one iteration to the next. Generally, five to ten iterations are sufficient for parameter convergence.

The speaker model parameters are computed by adapting the global parameters using Bayesian adaptation [28]. Suppose, the global model is denoted by $\lambda = \{w_i, \vec{\mu}_i, \Sigma_i\}$, where $1 \leq i \leq M$. Let us define a probabilistic alignment of the training vector $\vec{x}_t$ into the component i of the global model as

$$p(i|\vec{x}_t) = \frac{w_i p_i(\vec{x}_t)}{\sum_{j=1}^{M} w_j p_j(\vec{x}_t)} \tag{2.4}$$

In Equation (2.4), p_i stands for the density of component i with parameters $\vec{\mu}_i$ and Σ_i and p_i is defined in Equation (2.2). The sufficient statistics for the weights, mean vectors and covariance matrices are then calculated as stated in Equations (2.5) to (2.7).

$$n_i = \sum_{t=1}^{T} p(i|\vec{x}_t) \tag{2.5}$$

$$E_i(\vec{x}) = \frac{1}{n_i} \sum_{t=1}^{T} p(i|\vec{x}_t)\vec{x}_t \tag{2.6}$$

$$E_i(\vec{x}^2) = \frac{1}{n_i} \sum_{t=1}^{T} p(i|\vec{x}_t)\vec{x}_t^2 \tag{2.7}$$

If we denote the speaker model as $\hat{\lambda} = \{\hat{w}_i, \hat{\vec{\mu}}_i, \hat{\Sigma}_i\}$, where $1 \leq i \leq M$, then the speaker model parameters can be calculated from the above sufficient statistics, with the help of Equations (2.8) to (2.10):

$$\hat{w}_i = \left[\alpha_i n_i / T + (1 - \alpha_i) w_i\right] \gamma \tag{2.8}$$

$$\hat{\vec{\mu}}_i = \alpha_i E_i(\vec{x}) + (1 - \alpha_i)\vec{\mu}_i \tag{2.9}$$

$$\hat{\Sigma}_i = \alpha_i E_i(\vec{x}^2) + (1 - \alpha_i)\left[\Sigma_i + \vec{\mu}_i^{\,2}\right] - \hat{\vec{\mu}}_i^{\,2} \tag{2.10}$$

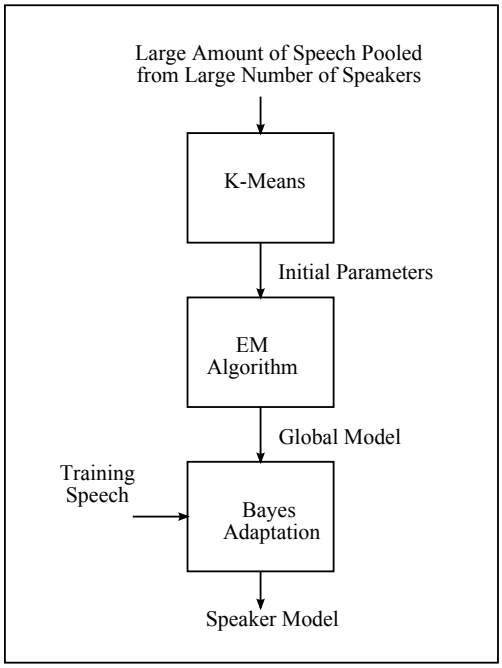


Figure 2.4: Various Steps in Computing Adapted Speaker Models - This figure illustrates the process of adaptation of a speaker model from a global model

In the above equations α_i is the adaptation coefficient which is determined as $\alpha_i = \left[n_i / \left(n_i + r\right)\right]$, where r is known as relevance factor and is taken to be 16. In Equation (2.8), γ is a constant to ensure that the weights sum to 1.

It should be noted that $\vec{x}_t^2$ is a shorthand for diag$(\vec{x}\vec{x}')$. Figure 2.4 illustrate various steps involved in computing parameters for an adapted GMM.

2.2.3 Scoring

The objective of the scoring module is to compute a match score between the claimed speaker model and the incoming speech signal and provide a decision. For the GMMs, log-likelihood score is used as the match score. Considering a set of speech feature vectors $X = \{\vec{x}_1, \vec{x}_2, \ldots, \vec{x}_N\}$ and a claimed speaker model λ, the log-likelihood score can be calculated as:

$$\log p\left(X|\lambda\right) = \frac{1}{N}\sum_{t=1}^{N} \log p\left(\vec{x}_t|\lambda\right) \tag{2.11}$$

The $p\left(\vec{x}_t|\lambda\right)$ in the above equation has been defined in Equation 2.3. The decision-making in the scoring module is a statistical hypothesis testing process. The claimed speaker model can be thought of as the "null" hypothesis. Another "alternate" model is formed and a likelihood ratio is computed. Let the claimed speaker model be denoted as λ and the alternate model be denoted as $\bar{\lambda}$. Let the log-likelihood of the feature set X with model λ be denoted as $L_\lambda(X)$. Then the log of the likelihood ratio, $\tilde{L}(X)$, can be written as:

$$\tilde{L}(X) = L_\lambda(X) - L_{\bar{\lambda}}(X) \tag{2.12}$$

Several methods have been proposed in literature for the computation of "alternate" model. In [29], a cohort of speaker models, close to the claimed speaker model, were chosen to form the "alternate" model. The claimed speaker model as well as a cohort of close speaker models formed the "alternate" model in [30]. The primary drawback of these approaches is the fact that the scoring module needs to compute match scores for the entire set of cohorts. This increases the amount of computation required for the decision-making. Later approaches, therefore, replaced the cohort of impostor models with a unique "world" model, also known as background model [31] [32] [33].

Universal Background Model (UBM) was introduced in [34] as a world model, which can also serve as the global model for speaker model adaptation described in Section 2.2.2. The UBM is trained using a large amount of speech

pooled from a large number of speakers and can therefore be viewed as a speaker-independent model representing a generic speaker. This GMMM-UBM technique has been adopted in this thesis. The scoring module computes the log-likelihood score of the incoming speech feature vectors with the claimed speaker model and the UBM. The difference of these two log-likelihood score is compared with a threshold and a decision of accept / reject is taken. Figure 2.5 illustrates the scoring and decision process.

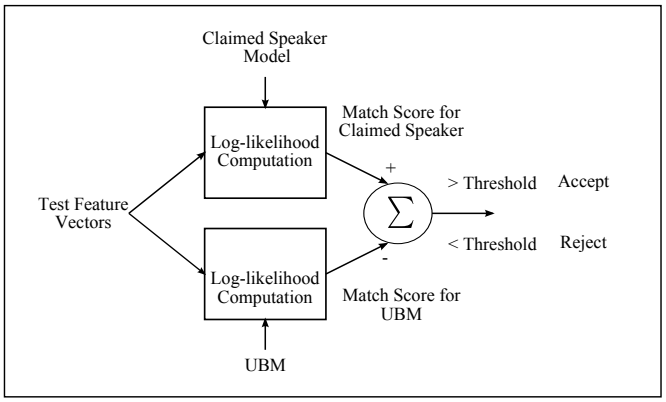


Figure 2.5: The Scoring Process - This figure illustrates the process match score computation and decision making

2.2.4 Measuring System Performance

There are two kinds of errors in case of a speaker verification system, known as type I error and type II error [7]. Rejecting a true speaker is (false rejection) is known as type I error whereas, accepting an impostor (false acceptance) is known as the type II error. False acceptance is also known as false alarm probability and false rejection is known as miss probability.

It should be noted that the false alarm probability and the miss probability depend very much on the decision threshold. By setting a higher threshold, false alarm probability can be reduced at the cost of higher miss probability. Therefore, a pair of false alarm and miss probability does not convey much information about the system performance. To have a dynamic performance measure, a plot of false alarm and miss probability at different thresholds is used. Such a plot is known as the Receiver Operating Characteristics (ROC)

plot [7]. Between two speaker verification systems, the system with the ROC curve closer to origin is the better performing system.

The point on the ROC curve at which the false alarm probability equals the miss probability is known as the Equal Error Rate (EER) point and the value is known as EER. Between two speaker verification systems, the system with lesser EER is considered a better performing system.

2.2.5 Speech Databases

Selection of a speech database is an important aspect of the experiments. Many speech databases are available to enable research in speech and speaker recognition fields. In this section, we review some of the prominent speech databases.

The YOHO voice verification corpus is one of the earliest speech databases for speaker verification [35]. It consisted of speech samples from 138 speakers (106 males and 32 females) collected over a 3-month period. The samples were collected in office environment with 4 enrollment sessions and 10 verification sessions per speaker. The sampling rate was 8 KHz with 3.8 KHz bandwidth.

The TIMIT database [3] was designed as an acoustic-phonetic speech database. It has speech samples from 630 speakers (438 males, 192 females) collected within the same session. There are 10 utterances for each speaker collected in a sound booth. The sampling rate is 16 KHz with 8 KHz bandwidth. The NTIMIT database was created by passing the TIMIT database through a telephone channel and bandwidth is restricted to 300 - 3452 Hz.

The KING database [15] contains speech samples from 51 speakers (all male), with 10 sessions of speech from each speaker. Each session consists of 30 seconds of speech on a given topic. The speech is recorded in both wideband and narrowband channel. The sampling rate is 8 Khz. The bandwidth for the wideband speech is 4 Khz, which the same for narrowband speech is 300 - 3300 Hz.

The National Institute of Standards and Technology (NIST), USA, provides a speaker recognition evaluation corpus at regular intervals [28]. The corpus contains telephonic speech provided by Switchboard corupus and Mixer corpus and the sampling rate is 8 KHz. The number of speakers in the database

vary from year to year. The NIST speaker recognition corpus has gained popularity in the recent years.

The MIT Mobile Devices Speaker Verification Corpus (MITMDSVC) [6] has been recently designed as a speech database for challenging conditions. The database contains speech samples from 48 enrolled-speakers (26 male and 22 female) and 40 imposters (23 male and 17 female). The speech utterances are extremely short (around 1 second in duration) and they were collected using mobile devices in realistic conditions such as office environment and street intersections. The sampling rate for the speech samples is 16 KHz.

In this thesis, two different speech databases have been used to validate the various techniques: the TIMIT database with artificially added noise and the MITMDSVC database. Although NIST database has been extremely popular in recent years, it is primarily designed to test algorithms for channel robustness and therefore is not very suitable for experiments concerning robustness against the additive noise.

The TIMIT provides an ideal platform to assess the true potential of an algorithm against environmental noise. With high quality microphone, 16 KHz sampling and noise-free environment for speech acquisition, the TIMIT speech database contains utterances that are close to being distortion free. This helps in isolating the problem of additive noise, when synthetic noise is added to TIMIT database. All other databases, mentioned above, contain speech samples that suffer from degradation due to multiple reasons. In such cases, it is difficult to isolate the effect of additive noise on the performance. Another advantage of using TIMIT database is that the artificial noise can be scaled to simulate a desired level of noise (different SNR conditions). In realistic speech databases, the level of noise is fixed and cannot be changed. Thus, the TIMIT database with artificially added noise is an excellent choice for developing algorithms to provide robustness against additive noise.

The MITMDSVC, on the other hand, contains speech samples collected in realistic environment. The noise in realistic environment may not be stationary and the speech might suffer from Lombard effect [36]. The algorithms developed on the TIMIT database can, thus, be tested for their suitability for deployment in real-world scenario using speech samples from this database. In other words, while the TIMIT database is suitable for development of

additive-noise-specific algorithms, the MITMDSVC is suitable for testing the maturity of such algorithms for real-life deployments.

2.3 Mismatched Condition in Speaker Verification

Speaker recognition technology has made tremendous strides in the last three decades. However, it is clear from the numerous studies and published experiments that the largest impediment to widespread deployment of speaker recognition technology is the lack of robustness in real life conditions [2]. The lack of robustness stems from a variety of issues such as speaker variability, communication channel, microphone mismatch and environmental noise [37]. Prominent among these issues are microphone mismatch and environmental noise [38].

Microphone mismatch and environmental noise (or environmental mismatch) can be grouped together and referred to as a mismatch condition. A mismatched condition is said to have occurred when there is a mismatch between the training and verification conditions [39]. In real life maintaining the conditions of the training phase in the verification phase is impractical and often impossible. Therefore, mismatched conditions present a significant research challenge and have received a lot of attention.

2.3.1 Microphone Mismatch

Most speaker verification systems rely on acoustic features based on spectra or a derivative of spectra. Since the spectrum of a signal is highly affected by channel information, the acoustic features are also affected by the same. Microphone mismatch occurs when one type of microphone is used to collect the training speech, while another type of microphone is used to collect the verification speech. The microphone acts as a filter on the speech signal. Different microphones have different filter characteristics. This can lead to significant degradation in speaker verification performance.

Considerable amount of research work has been devoted to the problem of speaker verification in microphone mismatch condition. Solutions have been proposed in feature vector domain, model domain and match score domain.

Techniques such as cepstral mean subtraction and Relative Spectral (RASTA) filtering [15] [28] [40] function in the feature vector domain. RASTA filtering, Heteroscedastic Linear Discriminant Analysis (HLDA), feature mapping and eigenchannel adaptation have been incrementally applied and compared in [41]. It was found in the aforementioned study that adding more techniques on top of other techniques does not necessarily improve performance. Other examples of feature vector domain solutions are feature warping [42], short-time Gaussianization [43] and harmonic structure features [44]. Model domain techniques for robustness against microphone mismatch use new data to learn channel characteristics. Eigenchannel compensation proposed in [45] can be applied in model domain as well as feature vector domain. Other examples of model domain techniques are latent factor analysis [46] and model adaptation [47]. Score domain techniques attempt to remove the handset dependent bias from the match score. Examples of such techniques are H-norm, Z-norm and T-norm [28] [48] [49].

2.3.2 Environmental Mismatch

Just like the type of microphone, environmental noise can significantly affect the speech spectra and consequently, the acoustic feature. For example, it has been observed, in literature, that white noise tends to reduce the variance of the cepstral coefficients within the frame [50] [51]. Another cause of mismatch stems from the involuntary physical changes in a user in response to the noisy conditions. The auditory feedback of a user is obstructed by the noise, which induces the user to raise his/her voice in response. This causes statistically significant articulation variability known as the Lombard effect [36] [52] [53]. These phenomena may produce serious mismatches between the training and verification conditions that result in degradation in accuracy.

The acquired speech can be thought of as a function of the clean speech and the condition of acquisition. If the clean speech is denoted by X, then the same speech under condition α can be represented by $f_\alpha(X)$. A mismatch condition occurs if the enrollment speech were collected under the noise condition α and test utterance under β. The degraded performance is a result of two different functions of the speech being compared with each other.

Efforts in the field of speaker verification in noisy conditions have been directed towards reducing the mismatch between training and verification conditions. Because of the complexity of modern noisy speech processing techniques, it is difficult to classify them as techniques classified in different categories may have many similarities. Nonetheless, we describe here a broad classification of the efforts in the field of speaker verification in noisy conditions. These efforts can be classified into three categories [39].

First, having noise independent systems. In other words, $f_\alpha \approx f_\beta$. Search for noise resistant features and robust distance measures fall in this category. Noise resistant features are implemented in feature vector domain, while the robust distance measures can be implemented in the match-score domain. The temporally weighted features [54] and score weighting described in [55] are examples of this class of techniques.

Second, transforming utterances into a reference environment. Speech enhancement techniques fall in this category and their goal is to find ${f_\beta}^{-1}$ and ${f_\alpha}^{-1}$. These techniques operate in the feature vector domain. Using spectral subtraction [56] and Wiener filter [57] are examples of this class of techniques.

Third, transforming models created in training condition α into verification condition β. Model compensation techniques fall in this category and their goal is to find the transformation g, such that $g(f_\alpha) = f_\beta$. These techniques operate in the model domain. Parallel model combination [58] and Jacobian model adaptation [59] are examples of this class of techniques.

The work presented in this thesis deals with the problem of speaker verification in environmental noise. Therefore, a review of the prominent and relevant existing techniques is presented in the next section.

2.4 Speaker Verification in Environmental Noise

The aim of this section is to evaluate existing noise robustness techniques for speaker verification systems. Towards this goal, the techniques under the three categories described above are explained and their reported performance are taken into consideration. This is followed by a comparative analysis, which attempts to identify the most promising techniques for further investigation.

2.4.1 Noise-Independent Systems

With noise-independent systems, the same system configuration can be used for clean and noisy speech. In other words, transformation of models or features can be avoided altogether. This has motivated researchers to look into the possibility of obtaining noise resistant features.

In [60], a new feature called "Mel-Frequency Discrete Wavelet Coefficient" (MFDWC) was employed for noise robustness. The computation of MFDWC follows closely the computation of MFCC described in Section 2.2.1, except that Discrete Wavelet Transform (DWT) [61] is used in stead of DCT on the log-mel-filter-output. The performance gain provided by MFDWC in noise is not very clear from [60] as MFDWC is used in conjunction with parallel model combination and the gain reported is not entirely due to MFDWC. However, MFDWC was used in speech recognition task in [62] and a 10% improvement in performance at 10dB Signal-to-Noise-Ratio (SNR) and 2% improvement at 5 dB SNR was reported.

A temporally weighted linear prediction feature has been proposed in [54]. Rather than removing noise as speech enhancement methods do, the weighted linear prediction method aims to increase the contribution of such samples in the filter optimization that have been less corrupted by noise. The linear prediction methods for spectral estimation used in the aforementioned work are Weighted Linear Prediction (WLP) [63] and stabilized WLP [64]. At a SNR of 10 dB with white noise, the new features provided a performance gain of 3% and at a SNR of 10 dB with factory noise, a peformance gain of around 1% was reported.

A boosted binary feature has been proposed in [65]. The proposed feature is calculated by applying a binary transform on the spectral magnitude of the speech signal or the mel-filter output of the speech signal. From the set of these binary features, a smaller set is selected by the Discrete Adaboost algorithm [66]. The feature provides moderate improvements at 10 dB SNR, although the exact percentage of improvement cannot be gauged from the plots provided in [65].

Pitch-synchronous feature extraction has been proposed in [67]. Instead of computing the feature vectors from fixed duration speech frames, pitch-synchronous feature extraction computes feature from frames which have a

length that is integral multiple of the pitch period of the speech. This has been claimed to reduce the spectral distortion of the speech signal. An average performance gain of 0.7% at 10 dB SNR and 1.5% at 5 dB SNR has been reported.

2.4.2 Feature Vectors Transformation

This class of noise robustness techniques has received a lot of attention. The most popular in this class are the speech enhancement techniques that employ spectral subtraction, Wiener filtering or Kalman filtering for estimation of clean speech from a noisy speech observation. These techniques improve the SNR of the speech signal. However, the gain in the SNR does not always translate into gain in the speaker verification performance, probably because of the spectral distortion induced by these techniques [39]. Nevertheless, these techniques have been successfully adapted by many researchers to improve the speaker verification performance in noisy conditions.

Spectral Subtraction (SS) [68] assumes noise to be additive in the spectral magnitude domain. Consider that a windowed noise signal $n(k)$ has been added to a windowed clean speech signal $s(k)$, with their sum denoted by $x(k)$. Then,

$$x(k) = s(k) + n(k) \tag{2.13}$$

Taking their Fourier transforms:

$$X(f) = S(f) + N(f) \tag{2.14}$$

The SS filter $H(f)$ is calculated by replacing the noise spectrum $N(f)$ with a spectra which can be readily measured. The average spectral magnitude measured during the nonspeech activity $|\mu(f)|$ replaces $|N(f)|$. If actual spectrum of the speech is to be estimated, then the phase information of the noise spectrum is replaced by the phase information of the noisy speech. However, this is not required in speaker verification applications, as only the spectral magnitude, rather than the spectrum itself, is of interest. The clean speech spectral magnitude estimator $|\hat{S}(f)|$ can, therefore, be written as:

$$|\hat{S}(f)| = |X(f)| - |\mu(f)| \tag{2.15}$$

The above equation holds as long as $|\mu(f)| < |X(f)|$. However, $|\mu(f)| > |X(f)|$ leads to a singularity as the spectral magnitude cannot be negative. This singularity is addressed by setting a floor value ξ for the spectral magnitude estimator. The revised estimator can be written as:

$$|\hat{S}(f)| = \max\left[\left(|X(f)| - |\mu(f)|\right), \xi\right] \tag{2.16}$$

Several variations of SS have been proposed and implemented in literature. Equation (2.16) refers to SS in spectral magnitude. SS can also be performed in the power spectral domain, which is known as power SS. When MFCC feature vectors are used, the SS can be implemented in the spectral domain or the mel-filter output domain [56]. Figure 2.6 illustrates these two types of SS implementation for MFCC.

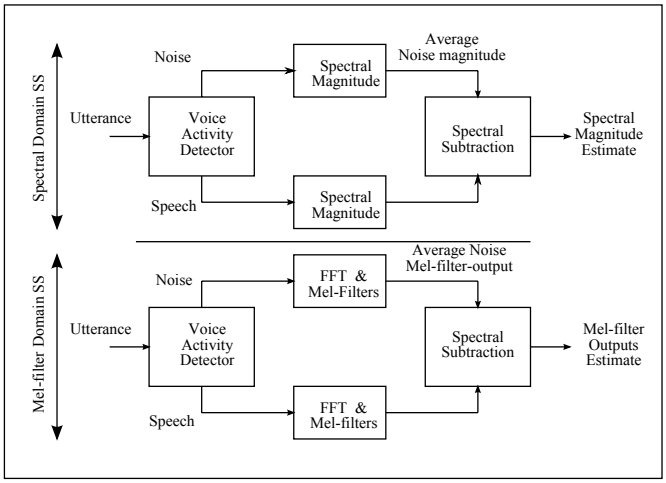


Figure 2.6: Types of Implementations of SS for MFCC - This figure illustrates the implementation of SS in spectral and mel-filter-output domain for MFCC computation

In [69], SS was employed as a missing feature detector. Bark filter outputs were used as feature vectors and if any of the spectral estimates in the bark filters equaled the floor value ξ, the corresponding bark filter output was considered unreliable and missing. During the scoring phase, the missing dimensions of the feature vectors were ignored. This was accomplished by

representing the feature vectors $\vec{x}$ as $\{\vec{x}^p \vec{x}^m\}$, where $\vec{x}^m$ stands for the missing dimensions and $\vec{x}^p$ stands for "present" or reliable dimensions. The Gaussian densities described in Equation 2.3 was modified as follows:

$$p(\vec{x}|\lambda) = \sum_{i=1}^{M} w_i \, p_i(\vec{x}^p) \tag{2.17}$$

At an SNR of 9 dB, 37% improvement in EER was reported as a result of the above technique. The significant improvement in performance shows the advantage of ignoring the highly corrupted dimensions of feature vectors. However, the above mentioned study does not reveal the performance of SS without missing feature technique.

In [70], the performances of different types of SS were compared. The baseline system consisted of clean training speech and noisy test speech. SS techniques were applied to the noisy test speech. The SS techniques used were power SS [70], Ephraim-Malah filter [71] and Virag generalized SS [72]. The performance of all three SS techniques were found to be unsatisfactory. In fact, at 10 dB and 5 dB SNRs, power SS and Ephraim-Malah filter performed worse than the system without SS, while Virag generalized SS showed only slight improvement in performance. This is probably because the residual noise left out by SS in the test utterance created a heavily mismatched condition. Marked improvements in performance for all the three techniques were reported, however, when noisy training speech was used. In this case, the residual noise in the training and test utterances created a matched condition. At 10 dB and 5 dB SNRs, the average improvements in EER reported was around 75%. This experiment demonstrates the importance of using noisy training speech for SS based algorithms.

Another improvement of the SS technique has been proposed in the form of Acoustic Model Enhancement (AME) [73]. AME attempts to create the matched condition by adapting the speaker models to the speech enhancement technique. In other words, the residual noise left behind by the SS technique is accounted for in the speaker models. This is accomplished by finding a transformation T for the model means μ_i such that the transformed means, $\hat{\mu}_i$ resembles the models means trained under SS conditions. Therefore, AME is similar to the noisy training speech strategy employed in [70].

The difference between AME and the technique in [70] is that instead of corrupting the training speech with noise, AME corrupts the model means. At 5 dB SNR, the performance gain reported for AME was about 89%.

Besides corrupting the training speech or speaker model for enhanced performance of SS, many researchers have proposed variations of basic SS model, represented by Equation (2.16), in order to improve the accuracy of the SS estimator. The primary weakness of the SS estimator is the occurrence of singularity when the average noise spectral magnitude becomes larger than the noisy speech spectral magnitude. The floor value, ξ, employed in singularity conditions adversely affects the performance of the SS estimator.

In order to address the problem of floor value smoothing of clean speech estimate and noise estimate is advocated in [74]. The smoothing is accomplished by computing weighted sum of consecutive estimates as follows:

$$|\hat{S}_i(f)|^2 = \lambda_s |\hat{S}_{i-1}(f)|^2 + (1-\lambda_s)|\hat{S}_i(f)|^2 \tag{2.18}$$

$$|\hat{N}_i(f)|^2 = \lambda_n |\hat{N}_{i-1}(f)|^2 + (1-\lambda_n)|\hat{N}_i(f)|^2 \tag{2.19}$$

In the above two equations, i is the frame index and λ_s and λ_n are memory factors such that $0.1 \leq \lambda_s \leq 0.5$ and $0.5 \leq \lambda_n \leq 0.9$. As a result of smoothing, along with improved modeling technique, an average performance gain of 71% at 5 dB SNR and 35% at 10 dB SNR was reported.

A probabilistic approach to SS was proposed in [75]. The clean speech estimate, $|\hat{S}(f)|$ was considered to be a random variable with its mean corresponding to the SS estimate. The variance of the random variable was theoretically derived from an additive noise corruption model. The variance was considered to be a measure of the uncertainty in the SS estimate. This approach was used for text-dependent speaker verification using Hidden Markov Models (HMM) [76]. A modified form of the Viterbi algorithm, called the weighted Viterbi algorithm [77] [78], was employed for the match score computation. The weighting of the Viterbi algorithm translated into computing the expected likelihood score instead of the conventional likelihood score. Performance improvements ranging from 30% to 40% were reported over a range of SNRs. It is worth noting that these improvements in performance

were achieved without any modifications to the modeling technique. This indicates that the performance gain could be further improved if noisy training speech technique proposed in [70] is employed.

Based on the sheer amount of literature available, it can be safely concluded that SS has been one of the most popular speech enhancement techniques for noise robustness in speaker verification. Nevertheless, other speech enhancement techniques have also been investigated [79]. Prominent among these are Wiener filtering and adaptive noise cancellation.

The Wiener filter can be described by the transfer function $W(f)$ [80]:

$$W(f) = \frac{g\,S(f)}{g\,S(f) + N(f)} \tag{2.20}$$

In the above equation, g is a gain matching term, $S(f)$ is the speech spectrum and $N(f)$ is the noise spectrum. Therefore, a filter can be designed to estimate the speech spectrum, given the noisy speech observations. Using Wiener filter, an average performance gain of 75% was reported at 10 dB SNR in [81]. Integrating pitch information with Wiener filter improved the performance only slightly in the same study. In [57], a two-stage Wiener filter was designed for robust performance. An average performance gain of 55% at 10 dB SNR and 66% at 5 dB SNR was reported.

Adaptive noise cancellation techniques dynamically estimate the noise waveform and subtract it from the noisy speech. The Least Mean Squared (LMS) algorithm [82] is a practical algorithm for adaptive noise cancellation. The LMS algorithm has the following form:

$$w_{n+1} = w_n + 2\,\mu\,e(n)\,y_n \tag{2.21}$$

In the above equation, w stands for the output of the algorithm, n stands for iteration number, μ is a adaptation constant which controls the rate of convergence, e is the error signal and y is the reference signal. In [83], normalized LMS algorithm was used for adaptive noise cancellation and performance gain of 21% was reported at 10 dB SNR for text-dependent speaker verification.

2.4.3 Model Transformation

The third class of noise robustness techniques, also known as model compensation techniques, involve transformation of model parameters to a reference environment. These techniques, typically, employ a theoretical model of noise corruption. Using the noise information during the verification phase and the noise corruption model, a transformation of the model parameters is undertaken to reduce the mismatch between training and verification phase. The AME technique described in Section 2.4.2 can also be listed under this category. However, it was placed under feature vector transformation in this work, as the main objective of AME is to enhance the speech and the model transformation is used to create a matched condition for the enhanced speech.

One of the most popular model transformation technique has been the Parallel Model Combination (PMC). It was first proposed for robust speech recognition in noisy conditions [84] [85]. PMC employs an additive model of noise corruption, i.e., it assumes that the noise shows up as an additive component in the mel-filter output domain. The estimation of the corrupted parameters works as follows. First, the noise characteristics are ascertained by training a noise model from the non-speech portion of the speech. Then the noise parameters are added to the model parameters in the mel-filter-output domain with a certain gain g. Mathematically, let the parameters in log-mel-filter-output domain be μ_s^l, μ_n^l, σ_s^l and σ_n^l. These parameters are then transformed to the mel-filter-output domain μ_s, μ_n, σ_s and σ_n. The corrupted speech model parameters in mel-filter-output domain are, then, given by

$$\mu = [g\,\mu_s + \mu_n] \tag{2.22}$$

$$\sigma = \left[g^2\,\sigma_s + \sigma_n\right] \tag{2.23}$$

In the above two equations, g is a gain factor used to account for the level difference between the training utterance and the test utterance and μ and σ are the transformed parameters in the mel-filter-output domain. These parameters are mapped back to the cepstral domain before scoring.

PMC has been a very popular model compensation technique and it has been applied in speaker recognition tasks in [86], [58] and [26]. Significant

improvements in speaker recognition performance have been reported in all three works. In [86], a multi-SNR speaker model is created using the noise model. A performance improvement of 77% is reported as a result of PMC induced multi-SNR model. In [58], the standard PMC is used and an average improvement of 80% is reported in speaker verification performance at 6 dB SNR. In the same work, the effect of g on performance was investigated and it was found that the performance is relatively insensitive to the choice of g and the g value calculated at an SNR of 6 dB holds for speaker verification in a wide range of SNRs. Similarly, PMC is reported to have resulted in over 70% gain in speaker verification performance in [26].

The Jacobian model adaptation technique [59] is another notable model compensation technique. It adapts the mean vectors of a model in order to reduce the mismatch. The adaptation is based on the partial derivative representation of the differential of an analytic function. Consider a model mean $\vec{\mu}$ and let the noise parameter during training be denoted as $\vec{C}$. Let the noise parameter during the verification be $\hat{\vec{C}}$. Then the adapted model mean $\hat{\vec{\mu}}$ can be expressed as:

$$\hat{\vec{\mu}} = \vec{\mu} + \frac{\partial \vec{\mu}}{\partial \vec{C}} \left(\hat{\vec{C}} - \vec{C} \right) \tag{2.24}$$

In the above equation, $\frac{\partial \vec{\mu}}{\partial \vec{C}}$ is the Jacobian matrix. During the training phase, the prevalent noise is estimated and the Jacobian matrix is computed. During the verification phase, the difference in prevalent noise is estimated and the model is updated according to Equation (2.24).

In [87], a variation of Jacobian adaptation, called α-Jacobian adaptation, has been proposed for speech recognition task. In it, the Jacobian matrix is estimated by multiplying the noise estimate and the noisy speech with a suitable factor α. This technique is claimed to be more robust in handling wider variation between training and verification condition. In [88], Jacobian adaptation is derived for frequency filtered spectral energies instead of the conventional MFCC. Frequency filtered spectral energies and Jacobian adaptation were used for speaker verification in [89] and an average performance gain of 35% was reported. In [90], a continuous noise estimation was adopted for Jacobian adaption and an average performance gain of 40% was reported.

The PMC and the Jacobian adaption employ a noise corruption model to estimate the transformed model parameters. Recently, researchers have proposed obtaining transformed parameters by adding noise directly to the training speech and creating multi-SNR models. In [91], the MFCC feature vectors are split into several sub-band. For each sub-band, GMMs are trained at various SNRs. Let $\lambda(i,j,k)$ be the model of i^{th} speaker for j^{th} sub-band at k^{th} SNR level. Then the likelihood of i^{th} speaker for j^{th} sub-band $L(i,j)$ is given by:

$$L(i,j) = \max_{k} \frac{1}{T} \sum_{t=1}^{T} p\left(\vec{x}_t(j) | \lambda(i,j,k)\right) \tag{2.25}$$

In the above equation, $\vec{x}_t(j)$ stands for the j^{th} sub-band of the feature vector $\vec{x}_t$. The match score L for the model is given by:

$$L = \sum_{j} w(j)\, L(i,j) \tag{2.26}$$

The $w(j)$ in the above equation stands for the weight of the j^{th} sub-band. Using $1/f$-noise to create multi-SNR models, 24% to 28% error reduction was reported in the above work. In [92], $1/f^{\alpha}$-noise was used to create multi-SNR models and slightly better performance was obtained.

The multi-SNR model technique was further improved by using missing feature theory in [93]. The missing feature theory is based on a posterior union model presented in [94] [95]. Consider a set of multi-SNR models λ_k and a feature vector $\vec{x}_t$. Let the subset of $\vec{x}_t$ be denoted as X_{sub}. Then

$$p(\vec{x}_t|\lambda_k) = \sum_{X_{sub} s.t. X_{sub} \subset \vec{x}_t} p(X_{sub}|\lambda_k) \tag{2.27}$$

And,

$$L = \sum_{k} p(\vec{x}_t|\lambda_k)\, p(\lambda_k) \tag{2.28}$$

In the above equation, $p(\lambda_k)$ is the prior probability of model λ_k. Consider a D-dimensional vector $\vec{x} = (x_1, x_2, \ldots, x_D)$. Then the possible subsets of this vector can be any combination of the component dimensions $x_1, x_2, \ldots, x_D$. This is likely to improve the performance as the heavily mismatched dimen-

sions score less and are, therefore, dominated by the matched dimensions. Under challenging street noise conditions, this technique provided an improvement of around 50% in speaker verification performance. Obviously, this scoring technique is computationally complex. Finding out every possible subset of a vector and computing the score is a time-consuming process. A faster method for the match score calculation has been proposed in [96]. However, the match score computation is still significantly more computationally complex than in other model compensation techniques.

2.4.4 Analysis of Environmental Noise Robustness Techniques

So far, in this section, prominent and relevant existing techniques for environmental noise robustness have been described. The goal of this sub-section is to critically examine the existing techniques and identify avenues of research that will be explored later in the thesis. The criteria for evaluation of existing techniques are performance gain, computational complexity and suitability for real-world application.

The noise independent systems are the most attractive from the real-world applicability point of view. One of the attributes for an ideal feature for speaker recognition is to remain unaffected by reasonable environmental noise [97]. Noise robust features obviate the need for additional noise robustness modules in the feature vector domain or the model domain which leads to less computational complexity. However, it can be inferred from the approaches described in Section 2.4.1 that these techniques are still in their infancy from the performance point of view. The traditional feature vectors, such as MFCC, offer superior performance when used alongside feature vector transformation or model transformation techniques. Although the feature vector transformation and model transformation add to the computational complexity of the overall system, the high performance gain justifies the added computational complexity. Besides, feature transformation and model transformation techniques are well researched and have been tested in a wide variety of environments. Therefore, these techniques are closer to real-world deployment than noise independent systems.

Among the feature vector transformation techniques, SS has been the most popular and the most well researched technique with several variations. Considering the scenario of clean training speech and SS-enhanced test speech, the best performing SS-based technique appears to be the probabilistic SS (PSS) described in [75]. It provides noticeable improvements [75] while other techniques do not [70]. A general characteristic of the SS-based techniques has been that when using clean training speech and SS-enhanced noisy test speech, they provide little improvement. In order to achieve significant performance gains, both training and test speech should be enhanced by SS. Going by the general characteristic of the SS-based techniques, therefore, the PSS performance should improve significantly, if it is used to enhance the training as well as the test utterances. However, this has not been accomplished so far and this will entail derivation of the parameter re-estimation formulae for the training algorithms such that *expected likelihood* is maximized instead of the conventional *likelihood*.

Among the model transformation techniques, the PMC boasts the best reported performance with reasonable computational complexity. As noted in [87], Jacobian model adaptation performs well only when noise prevalent during the verification phase is close to the noise prevalent in the training phase. α-Jacobian adaptation addresses this problem. However, the performance of α-Jacobian adaptation still lags behind that of the PMC [87]. Besides, the Jacobian matrix depends on the prevalent noise during training and is speaker specific. Therefore, the verification module needs to have access to the Jacobian matrix as well as the speaker model for verification. This increases the storage requirements for the speaker verification system.

Due to the reasons described above, it can be safely concluded that the PSS and the PMC are two of the most practical and best performing techniques. Between the PSS and the PMC, it is difficult to conclude which technique provides better performance based on the reported results, as the reported results stem from different speech databases and different noise types. Therefore, these two techniques should be tested under uniform conditions to ascertain the better technique.

The PSS and the PMC have two common drawbacks. First is their assumption about noise being additive in the spectral magnitude and mel-filter-

output domain. The interaction between speech and noise is more complex and the performance of the respective techniques can be improved by having less restrictive assumptions. Second drawback is that their performance very much depends on the quality of the estimated noise [38]. Reliable noise estimation may not always be possible in realistic conditions. The posterior union model described in Section 2.4.3 addresses this problem. However, the computational complexity of the posterior union model is prohibitive. A computationally efficient algorithm for handling realistic noise without dependence on accurate noise estimation will significantly improve the real-life usability of the PSS and the PMC.

Based on the literature review and analysis, the following three research objectives are arrived upon. First, the performance of the PSS and the PMC should be compared under uniform conditions to determine the better performing technique. Since the PSS could potentially gain from enhanced training speech, the possibility of improving the performance of the PSS technique with noisy training speech should also be investigated while performing comparison of the PSS and the PMC. It should be noted that the PSS and the PMC rely on the same underlying additive noise corruption model. The PSS employs the model in the feature vector domain while the PMC employs it in the model parameters domain. The comparison, therefore, is essentially to ascertain the domain that is more suitable for employing a noise corruption model. The next chapter provides this comparative study.

Second, the underlying noise corruption model should be improved in the domain determined by the comparison described above. Although environmental noise is additive in the spectrum, it is additive in the spectral magnitude only if the phase difference between the speech signal and the noise signal is zero. Under realistic conditions, the phase difference cannot be assumed to be zero. Therefore, better noise corruption models should be investigated that will improve the accuracy of the transformation of either the feature vectors or the model parameters. The improved accuracy of the transformation is likely to provide better performance gain in noisy conditions. Chapter 4 and Chapter 5 deal with the search for more robust noise corruption model.

Third, accurate noise estimation is very difficult, if not impossible, in realistic noise scenarios. Therefore, the noise-robustness technique must not rely on accurate estimation of the prevailing noise. The multi-SNR models provide a way to deal with unforeseen noise types. However, creating multi-SNR models with a certain type of noise, such as white or $1/f^{\alpha}$-noise, biases the models toward the particular type of noise. If the prevalent noise is different from the one used to create the multi-SNR models, it could result in heavy mismatch and degraded performance. The posterior union model provides a way to deal with heavy mismatch, but with a computationally complex algorithm. Hence, a computationally efficient way to reduce heavy mismatch should be investigated. Chapter 6 deals with the realistic noise scenario.

2.5 Summary

In this chapter, a review of the existing techniques in the field of speaker recognition in environmental noise has been presented. Starting with an overview of the basic speaker verification framework, the MFCC feature vectors and the GMM-UBM system have been explained. GMM-UBM system and MFCC feature vectors will be used in this thesis. Thereafter, the mismatched condition is explained and the two major type of mismatches in speaker verification, microphone mismatch and environmental mismatch, are described. This was followed by an overview of the existing environmental noise robustness techniques. Noise independent systems, feature vector transformation and model compensation techniques were explained. A critical analysis of these techniques was undertaken which identifies the limitations of the existing techniques.

3

Comparison of Probabilistic Spectral Subtraction and Parallel Model Combination

3.1 Introduction

This chapter compares the performances of the Probabilistic Spectral Subtraction (PSS) [75] and the Parallel Model Combination (PMC) [84] in environmental noise. It begins by an explanation of the PSS and the PMC techniques, followed by motivations for a new training scheme for the PSS technique. A novel training scheme is proposed by deriving the model parameters re-estimation formulae, which maximizes the expected likelihood. The proposed training scheme is suitable for employing PSS on the noisy training speech. Experiments show that the proposed training scheme does improve the performance of the PSS technique significantly. However, the performance of the PMC technique is noticeably better. Also, the spectral subtraction based techniques pose implementation difficulties, which make the model compensation technique a very attractive proposition.

3.2 PSS and PMC

In this section, the PSS and PMC techniques are explained. Since these two techniques are compared later on in the chapter, it is important to explore, in detail, the concepts involved in these techniques. Also, this section serves to introduce many terminologies and notations that will be used to develop the enhanced training scheme for the PSS technique.

3.2.1 Probabilistic Spectral Subtraction

Spectral subtraction refers to a class of subtractive algorithms which estimate the clean speech component based on the observation of noisy speech and noise. Mathematically, spectral subtraction can be expressed as:

$$\hat{x}(\omega)^p = \max[f(y(\omega)^p, \overline{n}(\omega)^p), \xi] \tag{3.1}$$

In the above equation, $\hat{x}(\omega)$ denotes the clean speech estimate, $y(\omega)$ denotes the noisy speech, $\overline{n}(\omega)$ denotes the estimate of the average noise and ξ denotes a constant. Thus, the clean speech estimate in SS is a function of the noisy speech and the noise estimate. In traditional SS,

$$f(y(\omega)^p, \overline{n}(\omega)^p) = y(\omega)^p - \overline{n}(\omega)^p \tag{3.2}$$

In the modified SS, proposed in [98],

$$f(y(\omega)^p, \overline{n}(\omega)^p) = y(\omega)^p + \ln|1 - e^{\overline{n}(\omega)^p - y(\omega)^p}| \tag{3.3}$$

If $p = 1$, then the above equations represent spectral subtraction in the spectral magnitude domain and if $p = 2$, they represent the spectral subtraction in the power spectral domain. When mel-cepstrums are used as the feature vectors, SS can be used either in the signal spectrum or in the mel filter output. In this work, SS in the mel-filter output in the magnitude domain has been considered following the method employed in [75].

In the PSS, the clean speech estimate $\hat{x}(\omega)$ is considered a random variable. Each operation of PSS yields a random variable with a certain mean and a

variance. If the noisy speech is denoted by y, then the operation of PSS results in $\hat{x} \sim N(\tilde{x}, {\sigma_x}^2)$, where $\tilde{x}$ is the mean of the estimate and ${\sigma_x}^2$ is the variance of the estimate. We have dropped the frequency notation ω for the sake of notational brevity. The mean of the estimate $\tilde{x}$ coincides with the traditional SS estimate as described in Equation (3.1). Hence, the traditional SS can be seen as using only the mean of the estimate, while the probabilistic approach can be seen as using both the mean and the variance of the estimate.

As pointed out in [99], the relationship between noisy speech and clean speech can be expressed at the output of mel-filter f as follows:

$$\overline{y_f}^2 = \overline{x_f}^2 + \overline{n_f}^2 + 2.\sqrt{c_f}\sqrt{{x_f}^2}\sqrt{{n_f}^2}.\cos(\phi) \tag{3.4}$$

In the above equation, $\overline{y_f}^2$, $\overline{x_f}^2$ and $\overline{n_f}^2$ represent energy of noisy speech, clean speech and noise respectively. c_f is a constant and ϕ is the phase difference between the clean speech and the noise signal. According to [75], even though the phase difference may be assumed to be constant within a given mel-filter, it is not constant across different frames, which is what the traditional SS supposes. The non-stationary phase difference between the clean speech signal and the noise, therefore, is the cause of the uncertainty in the traditional SS. Since the uncertainty in the estimate is a result of the non-stationary phase difference between the clean speech and noise, the variance is calculated in [75] by integrating the approximated uncertainty function over the phase difference. The approximated uncertainty function depends on the SS function $f\left((y)^p, (n)^p\right)$ and hence re-derivation of uncertainty function and its integration over the phase difference is required if the SS function changes. The derivation of the expression for ${\sigma_x}^2$ in the case of traditional SS, given by Equation (3.2), is presented in [75] and has been used in this work. Consider the noisy speech s_f, average noise estimate n_f and clean speech estimate $\tilde{s}_f$ in the mel-filter-output domain for filter f. Then the variance for the clean speech estimate is given by the following equation:

$$\text{Var}\left[\log\left(\tilde{s}_f\right)\right] = \begin{cases} \frac{2.c_f.n_f}{s_f - n_f}, & \left[s_f - n_f\right] \geq 10.c_f.n_f \\ -\frac{s_f - n_f}{50.c_f.n_f} + 0.4, & \left[s_f - n_f\right] < 10.c_f.n_f \end{cases} \tag{3.5}$$

In the above equation, c_f is a constant and is taken to be 0.25. The variance in the cepstral domain can be computed using the variance in the log-mel-filter-output domain by multiplying with squared DCT coefficients.

Consider a noisy test utterance $Y = \{\vec{y}_1, \vec{y}_2, \ldots, \vec{y}_T\}$. After the PSS operation, let the resultant mean and variance of the estimate be $X = \{\vec{\tilde{x}}_1, \vec{\tilde{x}}_2, \ldots, \vec{\tilde{x}}_T\}$ and $\Sigma_x = \{\vec{\sigma}^2_{x1}, \vec{\sigma}^2_{x2}, \ldots, \vec{\sigma}^2_{xT}\}$. Note that the conventional SS would yield only X.

Let the model parameters be $\lambda = \{w_i, \vec{\mu}_i, \Sigma_i\}, \quad 1 \leq i \leq M$, where M is the number of components in the model and Σ_i is a diagonal covariance matrix with non-zero elements $\Sigma_{i_{dd}} = \sigma^2_{id}, \quad 1 \leq d \leq D$. The likelihood score of the vector $\vec{\tilde{x}}_t$ with respect to the model λ can, then, be calculated as:

$$\begin{aligned} p(\vec{\tilde{x}}_t|\lambda) &= \sum_{i=1}^{M} w_i \prod_{d=1}^{D} \frac{1}{\sqrt{2\pi\sigma^2_{id}}} \\ &\quad \cdot \exp\left[-\frac{1}{2}\frac{(\tilde{x}_{td} - \mu_{id})^2}{\sigma^2_{id}}\right] \end{aligned} \tag{3.6}$$

In the above equation, we have assumed D dimensional vectors and $\tilde{x}_{td}$ represents the d^{th} component of $\vec{\tilde{x}}_t$. The log-likelihood score is then calculated as

$$\log\left[p(X|\lambda)\right] = \frac{1}{T}\sum_{t=1}^{T} \log\left[p(\vec{\tilde{x}}_t|\lambda)\right] \tag{3.7}$$

If conventional SS were used to estimate the clean speech, then Equation (3.7) is used to calculate the final score in the verification phase. If, on the other hand, PSS is used, an expected likelihood is calculated as follows:

$$\begin{aligned} E\left[p(\vec{\tilde{x}}_t|\lambda)\right] &= \sum_{i=1}^{M} w_i \prod_{d=1}^{D} \frac{1}{\sqrt{2\pi(\sigma^2_{id} + \sigma^2_{xtd})}} \\ &\quad \cdot \exp\left[-\frac{1}{2}\frac{(\tilde{x}_{td} - \mu_{id})^2}{(\sigma^2_{id} + \sigma^2_{xtd})}\right] \end{aligned} \tag{3.8}$$

The log of expected likelihood is then calculated as follows to obtain the final

score.

$$\log\left(E\left[p(X|\lambda)\right]\right) = \frac{1}{T}\sum_{t=1}^{T}\log\left(E\left[p(\vec{\tilde{x}}_t|\lambda)\right]\right) \tag{3.9}$$

3.2.2 Parallel Model Combination

The PMC operates on a different philosophy. Unlike the PSS, which attempts to estimate the clean speech from a noisy observation, the PMC attempts to estimate the noisy model parameters from the clean model parameters [85]. Consider a clean model $\lambda = \{w_i, \vec{\mu}_i^c, \Sigma_i^c\}, \quad 1 \leq i \leq M$. Consider a set of noise observations N and let the noise parameters be denoted by $\{\vec{\mu}_N^c, \Sigma_N^c\}$. The superscript c is to emphasize the fact that these parameters are in the MFCC domain. Also, note that the noise observations have been represented by a single multivariate Gaussian distribution. For more complex noise, a mixture of Gaussians can be considered. As described in Section 2.2.1, the MFCCs are derived from log-mel-filter-output by DCT, which can be represented by a matrix C. Therefore, the parameters in the log-mel-filter-output domain can be computed by:

$$\vec{\mu}_i^l = C^{-1}\vec{\mu}_i^c \tag{3.10}$$

$$\Sigma_i^l = C^{-1}\Sigma_i{}^c(C^{-1})' \tag{3.11}$$

$$\vec{\mu}_N^l = C^{-1}\vec{\mu}_N^c \tag{3.12}$$

$$\Sigma_N^l = C^{-1}\Sigma_N{}^c(C^{-1})' \tag{3.13}$$

The $'$ in the above equations denote transpose of a matrix. If the distributions in the log-mel-filter-output domain are considered to be Gaussian, then the distributions in the mel-filter-output domain are log-normal and the parameters $\vec{\mu}_i^l, \Sigma_i^l, \vec{\mu}_N^l, \Sigma_N^l$ can be converted to log-normal parameters $\vec{\mu}_i, \Sigma_i, \vec{\mu_N}, \Sigma_N$ [85]. The compensated model parameters $\hat{\vec{\mu}}_i, \hat{\Sigma}_i$ are given by:

$$\hat{\vec{\mu}}_i = g.\vec{\mu}_i + \vec{\mu}_N \tag{3.14}$$

$$\hat{\Sigma}_i = g^2.\Sigma_i + \Sigma_N \tag{3.15}$$

The g in the above equations is the gain factor and is calculated from the

average energy of clean speech E_s, noisy speech E_{ns} and noise E_n:

$$g = \frac{E_{ns} - E_n}{E_s} \tag{3.16}$$

The compensated parameters $\hat{\vec{\mu}}_i, \hat{\Sigma}_i$ are converted back to Gaussian parameters $\hat{\vec{\mu}}_i^{\,l}, \hat{\Sigma}_i^{\,l}$. The compensated model parameters in the MFCC domain $\hat{\vec{\mu}}_i^{\,c}, \hat{\Sigma}_i^{\,c}$ are computed by:

$$\hat{\vec{\mu}}_i^{c} = C\,\hat{\vec{\mu}}_i^{l} \tag{3.17}$$

$$\hat{\Sigma}_i^{c} = C\,\hat{\Sigma}_i^{l}\,(C)^{'} \tag{3.18}$$

The compensated parameters are used to compute the match score between the test utterance and the speaker model.

3.3 Novel Training Scheme for PSS

As has been discussed in Chapter 2, Section 2.4.4, SS-based techniques work best when speech enhancement is employed in training as well as verification utterances. In other words, the models are trained with SS-enhanced noisy training speech for obtaining better performance. This was done by adding modelized noise to the training speech in [70]. However, modelizing noise involves various approximations, which may adversely affect the performance. Therefore, in this work, noise signal, without modelization, is added to training speech in the time domain. This section describes a novel training scheme which employs PSS in the training phase.

3.3.1 Motivation for a Novel Training Scheme

Figure 3.1 illustrates the speaker verification modules when SS is employed in the training and the verification utterances. During training, the SS estimates of the training utterance is used to train models. Similarly, during the verification phase, the match score is computed by scoring the SS estimates of the noisy test utterance with the claimed model.

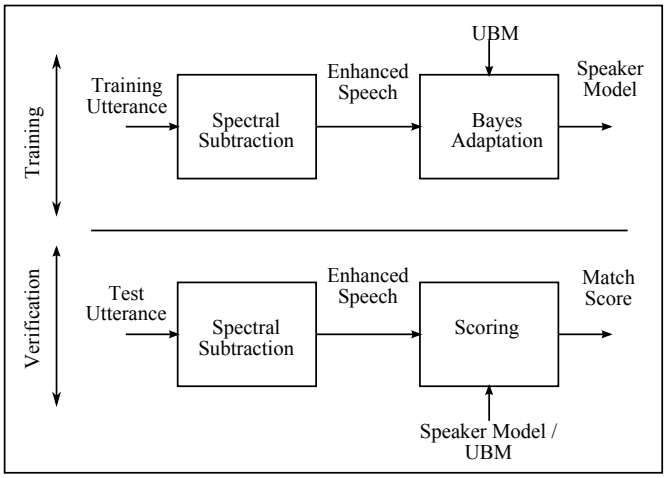


Figure 3.1: Spectral Subtraction in Speaker Verification Task - This figure illustrates the phases and modules of automatic speaker verification when spectral subtraction is employed

Now consider that PSS is employed instead of SS in the speaker verification task. The scoring module computes the log of the expected likelihood score given by Equation (3.9). No other changes are require in the verification phase. Let us now consider the training phase. As explained in Section 3.2.1, the PSS operation on training speech would result in a mean estimate X and a variance estimate Σ_x. The conventional training procedure for GMM-UBM scheme [28] consists of a weighted sum of the maximum likelihood parameters and the UBM parameters. The estimation of the maximum likelihood parameters is based on X. Therefore, the conventional training scheme is unable to exploit the variance parameter Σ_x. In order to exploit the variance parameter, the training scheme should maximize expected likelihood instead of likelihood. Also, since expected likelihood is used as match score, maximizing expected likelihood for model parameters is likely to impart better discrimination capability to the models.

3.3.2 Proposed Training Scheme for the PSS

In the Expectation Maximization (EM) algorithm [14] derivation, given an initial parameter set λ, a new set of parameters $\bar{\lambda}$ is estimated such that $p\left(X|\bar{\lambda}\right) \geq p\left(X|\lambda\right)$, where X is the training set. This is accomplished by using an auxiliary function [100], referred to as the Q-function in this work.

The Q-function allows a means to iteratively increase the likelihood function of the observed data by maximizing a function of the complete data. The term complete data is used to refer to observable and hidden data. The observable data is X, while the hidden data is the state variable. A similar approach is applied here for derivation of the maximum expected likelihood parameters. The Q-function is defined to maximize the expected likelihood of the complete data.

Considering a training set $X = \{\vec{x}_1, \vec{x}_2, \ldots, \vec{x}_T\}$, the Q-function for maximizing the likelihood is given by $Q = \sum_I p\left(X, I|\lambda\right) \log\left[p\left(X, I|\bar{\lambda}\right)\right]$ [14]. Similarly, the Q-function for maximizing the expected likelihood can be expressed as:

$$Q = \sum_I p\left(X, I|\lambda\right) \log E\left[p\left(X, I|\bar{\lambda}\right)\right] \tag{3.19}$$

In the above equation, I is the hidden state variable and λ is the current estimate of the model parameters and $\bar{\lambda}$ is the model parameters to be estimated such that $E\left[p\left(X|\bar{\lambda}\right)\right] \geq E\left[p\left(X|\lambda\right)\right]$. Consider GMMs with M components, $\lambda = \{w_i, \vec{\mu}_i, \Sigma_i\}$ and $\bar{\lambda} = \{\bar{w}_i, \vec{\mu}_i, \bar{\Sigma}_i\}$, $1 \leq i \leq M$. $E\left[p\left(X, I|\lambda\right)\right]$ in Equation (3.19) can be expressed as:

$$E\left[p\left(X, I|\bar{\lambda}\right)\right] = \prod_{t=1}^{T} E\left[\bar{w}_{i_t}\, \bar{b}_{i_t}\left(\vec{x}_t\right)\right] \tag{3.20}$$

Note that in the above equation, i_t stands for the component index that produced the observation $\vec{x}_t$ and it is unknown. From Equation (3.19) and Equation (3.20),

$$Q = \sum_I p\left(X, I|\lambda\right) \sum_{t=1}^{T} \log E\left[\bar{w}_{i_t}\, \bar{b}_{i_t}\left(\vec{x}_t\right)\right] \tag{3.21}$$

There are M possible components and i_t can be any of the M components and $\vec{x}_t$ can be any of the T observation vectors. The Q-function for an arbitrary component i and an arbitrary vector $\vec{x}_t$, denoted by Q_{i_t}, can be expressed as:

$$Q_{i_t} = p(\vec{x}_t, i|\lambda) \log E\left[\bar{w}_i\, \bar{b}_i\left(\vec{x}_t\right)\right] \tag{3.22}$$

In the above equation, $p(\vec{x}_t, i|\lambda)$ can be seen as the posterior probability of observation $\vec{x}_t$ being produced by component i of model λ. Let this probabil-

ity be denoted by $\gamma_t(i)$.

$$\gamma_t(i) = p(i_t = i|\vec{x}_t, \lambda) \tag{3.23}$$

Noting that the Q-function is the sum of Q_{i_t} over all possible components and all possible observations, the Q-function can be written as

$$Q = \sum_{t=1}^{T}\sum_{i=1}^{M} \log E\left[\bar{w}_i \bar{b}_i\left(\vec{x}_t\right)\right] \gamma_t\left(i\right) \tag{3.24}$$

In the above equation, $\bar{b}_i(\vec{x}_t)$ is the probability density function of the individual Gaussian component of the GMM and is given by Equation (3.25). In the equation below, D stands for the dimension of the vector $\vec{x}_t$.

$$\begin{aligned} \bar{b}_i\left(\vec{x}_t\right) &= \frac{1}{(2\pi)^{\frac{D}{2}}\left|\bar{\Sigma}_i\right|^{\frac{1}{2}}} \\ &\quad \cdot e^{-\frac{1}{2}\left[\left(\vec{x}_t-\vec{\mu}_i\right)'\bar{\Sigma}_i^{-1}\left(\vec{x}_t-\vec{\mu}_i\right)\right]} \end{aligned} \tag{3.25}$$

Since expected likelihood is to be maximized, the posterior probability is determined by the expected likelihood of an observation with respect to a component.

$$\gamma_t(i) = \frac{w_i E\left[b_i\left(\vec{x}_t\right)\right]}{\sum_{k=1}^{M} w_k E\left[b_k\left(\vec{x}_t\right)\right]} \tag{3.26}$$

where, considering diagonal covariance matrix for each component of the GMM and a variance for the estimated clean speech $\Sigma_x = \{\vec{\sigma}_{x1}^2, \vec{\sigma}_{x2}^2, \ldots, \vec{\sigma}_{xT}^2\}$,

$$\begin{aligned} E\left[b_i\left(\vec{x}_t\right)\right] &= \prod_{d=1}^{D} \frac{1}{\sqrt{2\pi(\sigma_{id}^2 + \sigma_{xtd}^2)}} \\ &\quad \cdot \exp\left[-\frac{1}{2}\frac{\left(x_{td} - \mu_{id}\right)^2}{(\sigma_{id}^2 + \sigma_{xtd}^2)}\right] \end{aligned} \tag{3.27}$$

Consider the Q-function for an arbitrary dimension d:

$$\begin{aligned} Q &= \sum_{t=1}^{T}\sum_{i=1}^{M}\gamma_t(i)\log\bar{w}_i \\ &\quad -\sum_{t=1}^{T}\sum_{i=1}^{M}\gamma_t(i)\log(2\pi)^{\frac{1}{2}}\left(\bar{\sigma}_{id}^2+\sigma_{xtd}^2\right)^{\frac{1}{2}} \\ &\quad -\sum_{t=1}^{T}\sum_{i=1}^{M}\frac{1}{2}\gamma_t(i)\frac{(x_{td}-\bar{\mu}_{id})^2}{(\bar{\sigma}_{id}^2+\sigma_{xtd}^2)} \end{aligned} \tag{3.28}$$

Taking partial derivative of Q with respect to $\bar{w}_i$, using Lagrange's multiplier η for the constraint $\sum_{i=1}^{M}\bar{w}_i = 1$ and equating to zero yields

$$\sum_{t=1}^{T}\frac{\gamma_t(i)}{\bar{w}_i}-\eta=0 \tag{3.29}$$

Solving the above equation of $\bar{w}_i$ and using the condition $\sum_{i=1}^{M}\bar{w}_i = 1$,

$$\eta=\sum_{t=1}^{T}\sum_{i=1}^{M}\gamma_t(i) \tag{3.30}$$

From Equations (3.29) and (3.30), the estimation formula of $\bar{w}_i$ can be found as

$$\bar{w}_i=\frac{\sum_{t=1}^{T}\gamma_t(i)}{\sum_{t=1}^{T}\sum_{i=1}^{M}\gamma_t(i)} \tag{3.31}$$

In order to find the formula for $\vec{\mu}_i$, the partial derivative of Q with respect to $\bar{\mu}_{id}$ is equated to zero, which yields

$$\sum_{t=1}^{T}\gamma_t(i)\frac{(x_{td}-\bar{\mu}_{id})}{\bar{\sigma}_{id}^2+\sigma_{xtd}^2}=0 \tag{3.32}$$

Solving for $\bar{\mu}_{id}$,

$$\bar{\mu}_{id}=\frac{\sum_t\frac{\gamma_t(i)x_{td}}{\bar{\sigma}_{id}^2+\sigma_{xtd}^2}}{\sum_t\frac{\gamma_t(i)}{\bar{\sigma}_{id}^2+\sigma_{xtd}^2}} \tag{3.33}$$

Similarly, taking partial derivative of Q with respect to $\bar{\sigma}_{id}{}^2$ and equating

it to zero,

$$\sum_t \frac{\gamma_t(i)\left(x_{td}-\bar{\mu}_{id}\right)^2}{\left(\bar{\sigma}_{id}^2+\sigma_{xtd}^2\right)^2}=\sum_t \frac{\gamma_t(i)}{\bar{\sigma}_{id}^2+\sigma_{xtd}^2} \tag{3.34}$$

It can be seen from Equations (3.33) and (3.34) that there is no closed-form solution to them. However, if we assume σ_{xtd}^2 to be independent of t, then the above two equations can be solved easily. In practice, σ_{xtd}^2 is not independent of t. Nevertheless, it is a necessary assumption to solve the equations. Taking into account the aforementioned assumption, we can derive the estimation formulae for $\bar{\mu}_{id}$ and $\bar{\sigma}_{id}$ as:

$$\bar{\mu}_{id}=\frac{\sum_t \gamma_t(i)\, x_{td}}{\sum_t \gamma_t(i)} \tag{3.35}$$

$$\bar{\sigma}_{id}^2=\frac{\sum_t \gamma_t(i)\left(x_{td}^2-\sigma_{xtd}^2\right)}{\sum_t \gamma_t(i)}-\bar{\mu}_{id}^2 \tag{3.36}$$

Equations (3.31), (3.35) and (3.36) together constitute the entire set of maximum expected likelihood estimation formulae for a GMM. It can be seen from the aforementioned equation that the estimation formulae for $\bar{w}_i$ and $\bar{\mu}_{id}$ are similar to those of the Expectation Maximization (EM) algorithm, except for the definition of $\gamma_t(i)$. Table 3.1 summarizes the similarities and differences between the conventional maximum likelihood and the proposed maximum expected likelihood parameter re-estimation.

The proposed maximum expected likelihood parameters replace the maximum likelihood parameters in the GMM-UBM parameter estimation formulae. The new GMM-UBM formulae can be written as:

$$\hat{w}_i=\left[\alpha_i \bar{w}_i+(1-\alpha_i)\, w_i\right]\nu \tag{3.37}$$

$$\hat{\vec{\mu}}_i=\alpha_i \vec{\bar{\mu}}_i+(1-\alpha_i)\,\vec{\mu}_i \tag{3.38}$$

$$\hat{\Sigma}_i=\alpha_i E_i\left(\vec{x}^2\right)+(1-\alpha_i)\left[\Sigma_i+\vec{\mu}_i^2\right]-\hat{\vec{\mu}}_i^2 \tag{3.39}$$

In the above equations, $\bar{w}_i$ and $\vec{\bar{\mu}}_i$ are calculated as in Equations (3.31) and

Table 3.1: Conventional and Proposed Re-estimation Formulae

	Conventional	Proposed
Q-function	$\sum_I p\left(X, I\|\lambda\right) \log\left[p\left(X, I\|\bar{\lambda}\right)\right]$	$\sum_I E\left[p\left(X, I\|\lambda\right)\right] \log E\left[p\left(X, I\|\bar{\lambda}\right)\right]$
$\gamma_t(i)$	$\frac{w_i b_i(\vec{x}_t)}{\sum_{k=1}^{M} w_k b_k(\vec{x}_t)}$	$\frac{w_i E[b_i(\vec{x}_t)]}{\sum_{k=1}^{M} w_k E[b_k(\vec{x}_t)]}$
$\bar{w}_i$	$\frac{\sum_t \gamma_t(i)}{\sum_t \sum_i \gamma_t(i)}$	$\frac{\sum_t \gamma_t(i)}{\sum_t \sum_i \gamma_t(i)}$
$\bar{\mu}_{id}$	$\frac{\sum_t \gamma_t(i) x_{td}}{\sum_t \gamma_t(i)}$	$\frac{\sum_t \frac{\gamma_t(i) x_{td}}{\bar{\sigma}_{id}^2 + \sigma_{xtd}^2}}{\sum_t \frac{\gamma_t(i)}{\bar{\sigma}_{id}^2 + \sigma_{xtd}^2}} \approx \frac{\sum_t \gamma_t(i) x_{td}}{\sum_t \gamma_t(i)}$
$\bar{\sigma}_{id}^2$	$\frac{\sum_t \gamma_t(i) x_{td}^2}{\sum_t \gamma_t(i)} - \bar{\mu}_{id}^2$	$\frac{\sum_t \gamma_t(i)\left(x_{td}^2 - \sigma_{xtd}^2\right)}{\sum_t \gamma_t(i)} - \bar{\mu}_{id}^2$

(3.35), ν is a constant to ensure that the $\hat{w}_i$ sum to 1, α_i are the adaptation coefficients computed as $\alpha_i = \left[n_i / \left(n_i + r\right)\right]$ (r is a constant and is taken as 32) and n_i and $E_i\left(\vec{x}^2\right)$ are calculated as follows

$$n_i = \sum_{t=1}^{T} \gamma_t\left(i\right) \tag{3.40}$$

$$E_i\left(\vec{x}^2\right) = \frac{1}{n_i} \sum_{t=1}^{T} \gamma_t\left(i\right)\left(\vec{x}_t^2 - \vec{\sigma}_{xt}^2\right) \tag{3.41}$$

3.4 Experiments

Experiments were conducted to gauge the performances of the PSS and the PMC techniques in noisy environments. This section deals with these experiments. First, the experimental set-up is described, which sheds light on the various control parameters of the experiments. Experimental results are

presented next, followed by an analysis of the experimental results.

3.4.1 Experimental Set-up

The experiments were conducted using the test portion of the TIMIT database, which has 10 utterances of speech each from 168 speakers. The first eight utterances (approximately 27 seconds of speech) were used for training and the 9th and 10th utterances were used as two test utterances. It is the same set-up as was followed in [4]. The mismatch condition was created by adding white and pink noise from NOISEX-92 database [70] to the utterances. The noise waveform being added to the speech was scaled appropriately to result in the global SNR of 5dB.

In the front-end, the silence frame were identified by comparing the frame-energy with the mean frame-energy from the first three frames. This is because the first few frames in the utterances are silence / unvoiced frames. In case of TIMIT database, this simple speech activity detector works well. The speech frames were then divided into 20ms windows progressing at 10ms. 20 mel-cepstral coefficients were calculated using 27 mel-filters from each window. The mel-filter coefficients were normalized so that they sum to 1.

The UBM was trained using speech from the train portion of the TIMIT database to ensure that the speech used for UBM is different from the speech used for speaker training and verification. The speech for the UBM was collected from 128 speakers and it was gender and region balanced. The UBM, and consequently the speaker models, had 64 components. The choice of 64 components follows from the studies in [27].

3.4.2 Experimental Results

The experiments were conducted to compare the performance of the PSS and the PMC techniques for test utterances corrupted by white and pink noise at an SNR of 5db. The first set of experiments employed the PSS technique. As has been mentioned before, to achieve the best performance for the PSS technique, noise should be added to the training utterances as well. Therefore, white and pink noise were added to the training utterances. Figure 3.2 presents the ROC curves from this set of experiments.

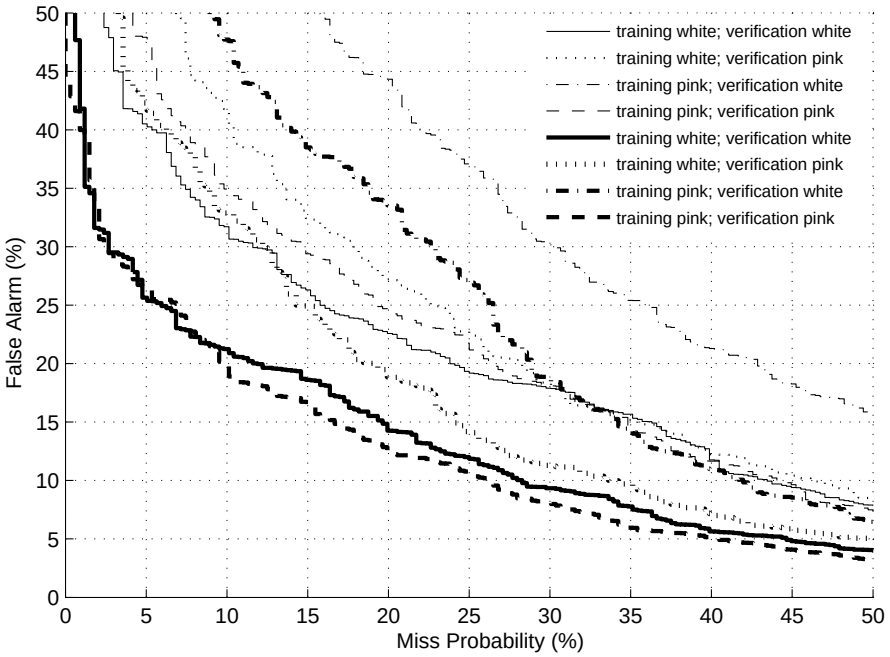


Figure 3.2: ROC Curves for the PSS Scheme - This figure shows the ROC curves for the PSS scheme. Thick curves are for the proposed training scheme while the thin curves are for the conventional training scheme

Firstly, it can be observed that the proposed training scheme provides significantly better results. This is because the conventional training scheme optimizes the models for the mean estimates X only. The proposed training scheme optimizes the models for the mean estimates X as well as the variance estimates Σ_x. It can also be observed that the best performance is obtained when the same type of noise is used for training and test utterances. This is no surprise as same type of noise in training and test speech represent the matched conditions. Table 3.2 provides the EERs from the these experiments. The best EER for test utterances corrupted with white noise is 17.1% while the same for test utterances corrupted with pink noise is 15.7%.

Table 3.2: EERs for the PSS Scheme with Noisy Training Speech (%)

Training Noise	Test Noise	Conventional Training	Proposed Training
White	White	21.4	17.1
White	Pink	23.8	19.6
Pink	White	30.0	25.9
Pink	Pink	23.0	15.7

While it has been mentioned before that the PSS scheme performs better when noisy training speech is used, it has not been explicitly demonstrated. The next set of the conducted experiments demonstrates this. In this set of experiments clean training speech and noisy test utterances were used. Table 3.3 presents the EERs from this set of experiments. It can be seen that the PSS scheme does provide some improvement compared to no speech enhancements. However, the EERs are worse than their noisy training counterparts.

Table 3.3: EERs for the PSS Scheme with Clean Training Speech (%)

Training Noise	Test Noise	Without Speech Enhancement	With PSS
None	White	32.9	29.1
None	Pink	36.6	21.3

EERs from Tables 3.2 and 3.3 lead to the following conclusions. The PSS scheme provides some performance gain when clean training speech is used. However, for the best performance, noise should be added to the training speech as well. These findings support the observations in [70]. For test

utterances corrupted with white and pink noise, the best EER for the PSS scheme is 17.1% and 15.7% respectively. These performances can be compared with the performances obtained with the PMC.

For the PMC, clean training speech was used. The models trained were compensated during the verification phase with the help of a single Gaussian distribution trained on the non-speech portion of the test utterances. The gain factor g was calculated according to Equation (3.16). It was observed that the value of g stayed close to 1 in all the cases. Figure 3.3 presents the ROC curves from these experiments. It can be seen that the PMC performs extremely well and it is better than the performance obtained through the PSS.

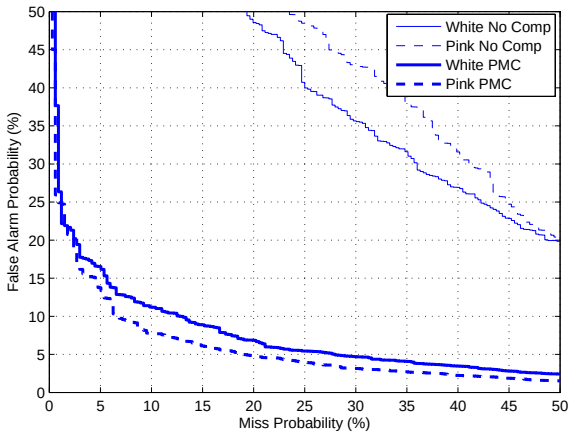


Figure 3.3: ROC Curves for the PMC Scheme - This figure shows the ROC curves for the PMC scheme. Thick curves are for the PMC while the thin curves are for speaker verification without any compensation or speech enhancement

Table 3.4 provides the EERs from these experiments. It can be observed that the PMC scheme provides on an average 71% reduction in the EER compared to speaker verification without any noise compensation or speech enhancement. Also, the PMC EERs of 11% and 8.6% for the white and and pink noise respectively are superior to their PSS counterparts of 17.1% and 15.7%.

Table 3.4: EERs for the PMC Scheme (%)

Training Noise	Test Noise	Without Compensation	With PMC
None	White	32.9	11.0
None	Pink	36.6	8.6

3.4.3 Analysis

The EER value for the TIMIT database with the set-up described in Section 3.4.1, using clean speech only is 0.6%. This level of performance is unmatched by either the PMC or the PSS. It is also unrealistic to expect this level of performance from the noise robustness techniques, as this requires recreating the clean speech waveform from a noisy speech sample, which is currently not achievable. However, the clean speech performance does indicate that there is a lot of potential for improvement in the noise robustness techniques.

The environmental noise robustness techniques strive to reduce the mismatch between training and verification conditions. The PSS and the PMC take different approaches to achieve this. The experiments conducted in this chapter have compared these approaches and it can be concluded from the reported experiments that the PMC approach results in better performance.

Speaker verification involves three types of speech utterances: training, test and utterances for the UBM. The best performance can be obtained by achieving matched condition in all three types of speech. The PSS approach is to achieve the matched condition by transforming individual feature vectors through noise removal. This would be ideal, if the process of noise removal were accurate. However, in the absence of an accurate noise estimation and an accurate noise corruption model, the PSS process leaves behind a residual noise. The residual noise create a mismatched condition, albeit this mismatched condition is less severe than the one created by the additive noise. Adding noise to the training speech and subjecting the training speech to PSS serves to reduce the mismatch between the training and the test utterances. However, this does not take into account the speech used for UBM and mismatch between UBM and training and test speech remains. This is one of the reasons for the relative under-performance of the PSS scheme.

The above explanation indicates that using enhanced noisy speech for training the UBM might improve the performance of the PSS scheme further. The proposed novel training scheme can be employed for this purpose. However, this raises some issues for the implementation phase. The speaker models and the UBM can be trained using noisy speech. However, the PSS may not provide good performance if the noise used to corrupt the training and UBM speech does not match the noise prevalent during the verification.

As can be inferred from Table 3.2, the performance of the PSS scheme reduces if the noise types differ. Since, in real life, the verification noise is unknown during the training phase, the implementation of this scheme is difficult.

The PMC, on the other hand, strives the reduce the mismatch by transforming the model parameters and model parameters are transformed based on the noise observed during the verification phase. Therefore, mismatch due to differing noise types is unlikely with the PMC. Also, since the speaker model training and the UBM training are one-off affairs, this can be done in a controlled environment, without much inconvenience to the user, ensuring acquisition of noise-free speech for creating the models.

Besides, the PSS operates on the observed samples while the PMC operates on distribution parameters. Observed samples are prone to contain outlier data. Transforming these outliers is likely to be inaccurate. Therefore, from the implementation point of view and from the performance points of view the PMC is preferable.

The PMC, however, has two drawbacks. First is the additive noise model, which uses a restrictive assumption. Second is its reliance on accurate estimation of the prevalent noise for good performance. While the additive assumption adversely effects the performance, the reliance on accurate estimation of noise limits its application in realistic conditions, where accurate estimation of the prevalent noise may not be possible.

3.5 Summary

A comparative study of the PSS and the PMC techniques was presented in this chapter. Based on the studies in Chapter 2, a novel training scheme was devised for the PSS scheme that uses the enhanced noisy training speech. This was achieved by defining an auxiliary function which iteratively increases the expected likelihood of the observed data. Experiments were conducted on the TIMIT database with artificial noise added to the speech utterances. Experimental results demonstrate the efficacy of the proposed training scheme. However, the PMC performs significantly better than the PSS and presents

less implementation difficulties. However, the PMC suffers from an inaccurate and restrictive noise corruption function. The next chapter introduces the max function as an alternative noise corruption function.

4

Max Function Based Model Compensation

4.1 Introduction

This chapter further investigates the model compensation process. It begins with a description and analysis of the model compensation as performed by the PMC [84]. Noise corruption function and compensation scheme are defined and identified as the two major components of the PMC. Then the max function, a non-linear function, is introduced as a corruption function. Next, a novel generic compensation scheme is derived to accommodate the non-linear corruption function. Experiments are conducted on the TIMIT database with artificial noise and the performances of the max function based compensation and the PMC are compared. Experiments conducted in this chapter indicate that the ideal noise corruption function should encompass the max function and the additive function.

4.2 Model Compensation

Model compensation is the process of transforming the model parameters from the training environment to the verification environment. The simplest

way to achieve this would be to add the observed noise to the training utterances and retrain the speaker and the background models. However, this gives rise to implementation difficulties. Retraining the speaker and background models for every single verification is time consuming and computationally inefficient. Also, the storage requirements become prohibitive if the training utterances need to be stored. An efficient way of transforming the model parameters might be to find the relationship between the model parameters in the training environment and the verification environment, which eliminates the reliance on the training utterances. Finding such a relationship is the underlying objective of the model compensation process.

The PMC assumes an additive relationship between the noisy speech and the clean speech in the mel-filter-output domain. In other words, the mel-filter-outputs of the noisy speech is assumed to be sum of the mel-filter-outputs of clean speech and observed noise. This relationship between the noisy speech and the clean speech is defined, in this work, as the noise corruption function. The relationship between the clean model parameters and the compensated parameters is derived from the noise corruption function. The method of computing the compensated model parameters from the clean model parameters is defined, in this work, as the compensation scheme.

4.2.1 Noise Corruption Function

The noise corruption function specifies the relationship between the clean speech and the noisy speech. The arguments of the noise corruption function are clean speech vectors and observed noise vectors, while the image of the two arguments is the noisy speech vectors. Let ξ_s be the clean speech vector component and ξ_n be the observed noise vector component. If the noisy speech vector component is denoted by ξ'', then the noise corruption function, $f(.)$, can be expressed by

$$\xi'' = f(\xi_s, \xi_n) \tag{4.1}$$

For the PMC, the noise corruption function is additive:

$$\begin{aligned} \xi'' &= f(\xi_s, \xi_n) \\ &= \xi_s + \xi_n \end{aligned} \tag{4.2}$$

The noise corruption function is an approximated relationship between the clean feature vectors and the noisy feature vectors. Consider a clean speech signal $x[t]$ and an additive noise signal $n[t]$. Let the degraded speech signal be denoted by $z[t]$. Then,

$$z[t] = x[t] + n[t] \tag{4.3}$$

The Fourier transforms of the signals, $X[f]$, $N[f]$ and $Z[f]$, maintain the additive property. Consider the output of the l^{th} mel-filter, $F_z(l)$, for the noisy signal. It can be expressed as:

$$F_z(l) = \frac{1}{A_l} \sum_{k=L_l}^{U_l} W_l[k] \, |Z[k]| \tag{4.4}$$

In the above equation, $W_l[k]$ is the filter coefficient for frequency k, A_l is the sum of all filter coefficients within the filter band, L_l is the lower end of the filter band and U_l is the upper end of the filter band.

Consider the following approximation of the mel-filter output for the noisy signal:

$$\begin{aligned} F_z(l) &= \frac{1}{A_l} \sum_{k=L_l}^{U_l} W_l[k] \, |Z[k]| \\ &= \frac{1}{A_l} \sum_{k=L_l}^{U_l} W_l[k] \, |X[k] + N[k]| \\ &\approx \frac{1}{A_l} \sum_{k=L_l}^{U_l} W_l[k] \; (|X[k]| + |N[k]|) \\ &= F_x(l) + F_n(l) \end{aligned} \tag{4.5}$$

The above equation represents the noise corruption function for the PMC. Figure 4.1 illustrates the additive approximation of the PMC.

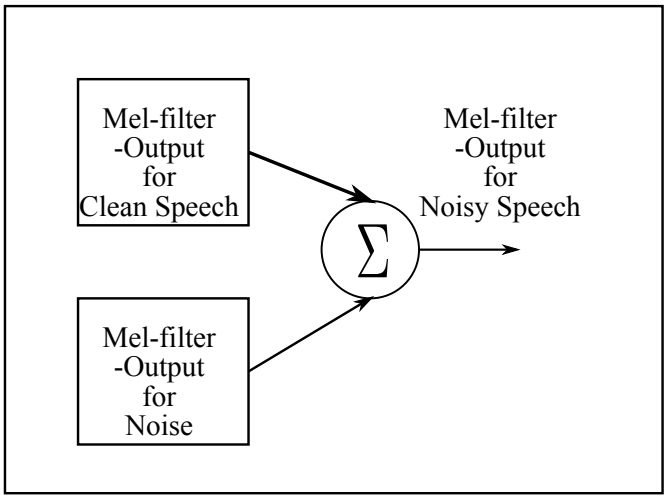


Figure 4.1: Noise Corruption Function for the PMC - This figure illustrates the additive approximation of the PMC

4.2.2 Compensation Scheme

The following relations can be derived from the approximation described in Equation (4.5):

$$E\left[F_z(l)\right] = E\left[F_x(l)\right] + E\left[F_n(l)\right] \tag{4.6}$$

$$\mathrm{Var}\left[F_z(l)\right] = \mathrm{Var}\left[F_x(l)\right] + \mathrm{Var}\left[F_n(l)\right] \tag{4.7}$$

In Equations (4.6) and (4.7), E stands for the mathematical expectation and Var stands for the variance. The assumption of independence of the clean speech and the noise is tacit in the above equations. This assumption, along with the additive assumption, makes the computation of the compensated parameters relatively straight forward. The expectation (or mean) and the variance of the clean speech signal is obtained from the clean models while the expectation and the variance of the noise signal is obtained from the cepstral representations of the observed noise.

Note that the expectation and variance are in the mel-filter-output domain. Therefore, the mel-cepstral representation of the models are transformed to the mel-filter-output domain. The parameter compensation, defined by Equations (4.6) and (4.7), is performed in the mel-filter-output domain. The compensated parameters are then transformed back to the mel-cepstral domain. Figure 4.2 illustrates the compensation scheme.

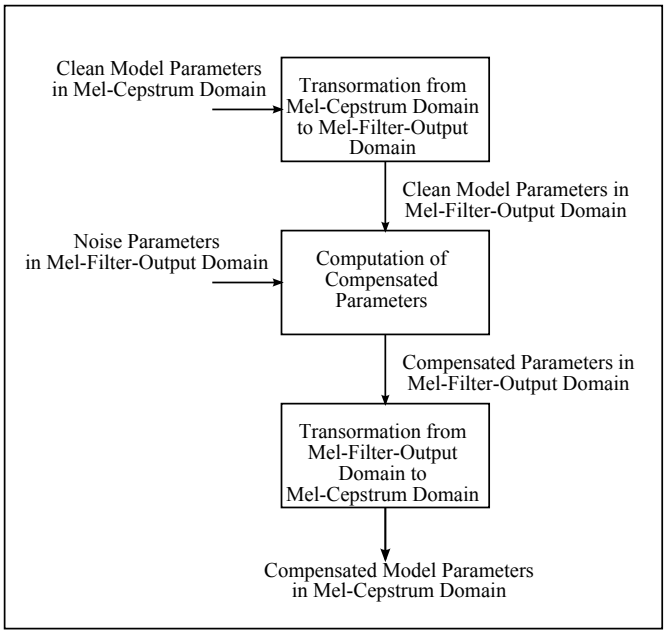


Figure 4.2: Compensation Scheme for the PMC - This figure illustrates the various steps in the compensation scheme of the PMC

4.3 Max-function Noise Corruption

According to the max function noise corruption, the mel-filter output of the corrupted speech equals the maximum of the clean speech mel-filter output and the noise mel-filter output. In other words, a given mel-filter output of a degraded speech frame is dominated by either the clean speech or the noise. The following expressions illustrate the mathematical approximation involved [14]:

$$\begin{aligned} F_z(l) &= \frac{1}{A_l}\sum_{k=L_l}^{U_l} W_l[k]\,|Z[k]| \\ &= \frac{1}{A_l}\sum_{k=L_l}^{U_l} W_l[k]\,|X[k]+N[k]| \\ &\approx \frac{1}{A_l}\sum_{k=L_l}^{U_l} W_l[k]\,\max\left(|X[k]|,|N[k]|\right) \\ &= \max\left(F_x(l),F_n(l)\right) \end{aligned} \tag{4.8}$$

The above relationship between the degraded speech signal and the clean

speech signal was observed in [101]. This relationship was also used in [102]. In both of these works, the max function corruption was used to alter the training algorithm, with the objective of training optimal models with noisy training speech. This is different from the model compensation scheme which assumes clean speech models and transforms the clean speech model parameters using the noise corruption function.

In this work, we will employ the max function as the noise corruption function for model compensation. This has potential for performing better than the additive function. It has been reported in [14] that visual inspection of the mel-filterbank log-energies for corrupted and non-corrupted speech confirms that a given filter band is dominated by either the clean speech or the noise. This has been further supported by the observations and plots of mel-filterbank log-energies in [101]. Therefore, the max-function appears to be more accurate as a noise corruption function than the additive function used by the PMC.

As can be observed from Equation (4.8), the max function is non-linear. A model compensation scheme for HMM involving non-linear corruption function was developed in [103]. However, the model compensation scheme in [103] only compensated for the model means and the model variances were left unaltered. This is because for a non-linear corruption function, deriving closed form expressions for $\mathrm{Var}\left[F_z(l)\right]$ is difficult. The next section deals with this issue and presents a novel compensation scheme for non-linear corruption functions.

4.4 Novel Compensation Scheme for Non-linear Corruption Functions

Compensation scheme describes the details of transforming the clean model parameters in order to create a matched condition for the noisy test environment. Consider a Gaussian mixture speaker model [15] $\lambda = \{w_i, \vec{\mu}_i, \Sigma_i\}, 1 \leq i \leq M$, where M is number of Gaussian components in the model, w_i is the weight, $\vec{\mu}_i$ is the mean vector and Σ_i is the diagonal covariance matrix

of the i^{th} component. Each mean vector $\vec{\mu}_i$ is a D-dimensional vector, i.e., $\vec{\mu}_i = (\mu_{i1}\, \mu_{i2}\, \ldots \mu_{iD})$.

The max function, as described in Equation (4.8), operates in the mel-filter-output domain. Therefore, the mel-cepstral model parameters must be converted to mel-filter-output domain. The corruption scheme described here follows the same three major steps as illustrated in Figure 4.2:

1. The model parameters are in mel-cepstral domain. Since the max function corruption takes place in the mel-filter-output domain, the first step in the compensation scheme is to transform the model parameters into mel-filter-output domain. This step is similar to the corresponding step in the PMC.

2. The max function is applied and the compensated parameters in the mel-filter-output domain are computed from the clean model parameters and the observed noise. This step requires an innovative estimation method. The proposed estimation technique is generic in the sense that it can be applied to any non-linear corruption function $f(\xi_s, \xi_n)$ as long as $f(\xi_s, \xi_n) > 0$.

3. The compensated parameters are transformed back to mel-cepstral domain. This step, again, mirrors the corresponding step of the PMC.

Each of the above mentioned steps are described in detail below:

4.4.1 Transforming Model Parameters into Mel-filter-output Domain

First, the clean model parameters are transformed into log-mel-filter-output domain by multiplying with Inverse Discrete Cosine Transform (IDCT) matrix.

$$\vec{\mu}_i^{\,l} = C^{-1}\, \vec{\mu}_i \tag{4.9}$$

$$\Sigma_i^l = C^{-1}\, \Sigma_i\, {C^{-1}}^T \tag{4.10}$$

In the above two equations, the mean vectors and variances in the log-mel-filter-output domain are denoted by $\vec{\mu}_i^{\,l}$ and Σ_i^l, respectively, the IDCT matrix

is denoted by C^{-1} and ${C^{-1}}^T$ denotes the transpose of the IDCT matrix. Since the feature vectors in log-mel-filter-output domain are normally distributed, the feature vectors converted into mel-filter-output domain are log-normally distributed and the parameters of normal distribution can be converted to log-normal distribution. Let the parameters in the mel-filter-output domain be denoted by $\vec{\mu}_i'$ and Σ_i' respectively. Note that, the covariance matrix Σ_i' may not be a diagonal matrix even if Σ_i^l is diagonal. For notational simplicity let us drop the component index "i", and let the elements of the mean vector be μ_p, whereas elements of the covariance matrix be σ_{pq}^2 with suitable superscript indicating their domain. Then the log-normal parameters can be obtained by the following equations [85]:

$$\mu_p' = \exp\left[\mu_p^l + \frac{{\sigma_{pp}^l}^2}{2}\right] \tag{4.11}$$

$${\sigma_{pq}'}^2 = \mu_p'\mu_q'\left[\exp\left({\sigma_{pq}^l}^2\right) - 1\right] \tag{4.12}$$

The log-normal parameters are in mel-filter-output domain, which can be used for the computation of compensated parameters.

4.4.2 Computing Compensated Parameters

Let the set of noise observations be $N = \{\vec{n}_1', \vec{n}_2', \ldots, \vec{n}_V'\}$. Note that each vector in the set N represents a frame of noise estimated from the test utterance and they are represented in the mel-filter-output domain. The noise output for mel-filter p for frame v will, henceforth, be denoted by n_{pv}'. Also, we will denote the max-function in Equation (4.8) by $f(.)$, i.e., $f\left(s_p', n_{pv}'\right)$ will stand for the max-function operation between s_p' and n_{pv}'.

In order to efficiently compute the compensated parameters, we express the compensated parameters μ_p'' and ${\sigma_{pq}''}^2$ as follows:

$$\mu_p'' = \mu_p' + \mu_{\delta p}' \tag{4.13}$$

$${\sigma_{pq}''}^2 = {\sigma_{pq}'}^2 + {\sigma_{\delta pq}'}^2 \tag{4.14}$$

The above two equations are the result of a tacit assumption that the compensation factor δ shows up as an additive component in the mel-filter-output domain. The value of δ is determined by the max-function. μ'_{δ} and ${\sigma'_{\delta pq}}^2$ are the parameters of the distribution of δ. Note that the features in the mel-filter-output domain are log-normally distributed. Therefore, we assume that δ is also log-normally distributed.

Given an element of the mean vector in the mel-filter-output domain μ'_p and a noise sample in the mel-filter-output domain n'_{pv}, the compensated mean can be calculated by $f\left(\mu'_p, n'_{pv}\right)$. We can express this as:

$$f\left(\mu'_p, n'_{pv}\right) = \mu'_p + \delta_{pv} \tag{4.15}$$

In other words, each sample of estimated noise n'_p gives rise to one sample point of δ_p. The value of δ_{pv} can be zero, according to the definition of the max-function. The value of the random variables of a log-normal distribution is always positive. Therefore, the log-normal parameters cannot be calculated directly from the sample values of δ_p. One way to solve this problem is to estimate the distribution of $\left[\mu'_p + \delta_p\right]$, which is always positive. Let us denote the mean and the covariance of $\left[\mu'_p + \delta_p\right]$ as $m_{\delta p}$ and $c_{\delta pq}$ respectively. Since μ'_p is a constant,

$$m_{\delta p} = \mu'_p + \mu'_{\delta p} \tag{4.16}$$

and,

$$c_{\delta pq} = {\sigma'_{\delta pq}}^2 \tag{4.17}$$

Consequently, we can write the compensated parameters as follows:

$$\mu''_p = m_{\delta p} \tag{4.18}$$

$${\sigma''_{pq}}^2 = {\sigma'_{pq}}^2 + c_{\delta pq} \tag{4.19}$$

The random variable $\left[\mu'_p + \delta_p\right]$ is log-normally distributed. Therefore, the

maximum-likelihood estimates of $m_{\delta p}$ and $c_{\delta pq}$ can be computed as follows:

$$m_{\delta p}{}^{l} = \frac{1}{V}\sum_{v=1}^{V}\log\left[f\left(\mu_{p}^{'}, n_{pv}^{'}\right)\right] \tag{4.20}$$

$$\begin{aligned} c_{\delta pq}{}^{l} &= \frac{1}{V}\sum_{v=1}^{V}\log\left[f\left(\mu_{p}^{'}, n_{pv}^{'}\right)\right] * \log\left[f\left(\mu_{q}^{'}, n_{qv}^{'}\right)\right] \\ &\quad - m_{\delta p}{}^{l} m_{\delta q}{}^{l} \end{aligned} \tag{4.21}$$

$$m_{\delta p} = \exp\left[m_{\delta p}{}^{l} + \frac{c_{\delta pp}^{l}}{2}\right] \tag{4.22}$$

$$c_{\delta pq} = m_{\delta p} m_{\delta q}\left[\exp\left(c_{\delta pq}{}^{l}\right) - 1\right] \tag{4.23}$$

Given the estimation in Equations (4.22) and (4.23), the compensated parameters can be calculated in mel-filter-output domain using Equations (4.18) and (4.19). The last stage in the compensation process is to transform the compensated parameters in the mel-filter-output domain to the mel-cepstral domain, which we describe next.

4.4.3 Transforming Compensated Parameters into Mel-cepstral Domain

Let us denote the compensated parameters in the log-mel-filter-output domain as $\hat{\vec{\mu}}^{l}$ and $\hat{\Sigma}^{l}$. These can be calculated from $\vec{\mu}''$ and Σ'' using the following equations [85].

$$\hat{\mu}_{p}^{l} = \log\left(\mu_{p}''\right) - \frac{1}{2}\log\left[\frac{\sigma_{pp}''^{2}}{\mu_{p}''^{2}} + 1\right] \tag{4.24}$$

$$\hat{\sigma}_{pq}^{l2} = \log\left[\frac{\sigma_{pq}''^{2}}{\mu_{p}''\mu_{q}''} + 1\right] \tag{4.25}$$

Once $\hat{\vec{\mu}}^{l}$ and $\hat{\Sigma}^{l}$ are computed, they can be transformed to mel-cepstral

domain by multiplication with the Discrete Cosine Transform (DCT) matrix.

$$\hat{\vec{\mu}}_i = C\,\hat{\vec{\mu}}_i^{\,l} \tag{4.26}$$

$$\hat{\Sigma}_i = C\,\hat{\Sigma}_i^{l}\,C^T \tag{4.27}$$

In the above two equations, $\hat{\vec{\mu}}_i$ and $\hat{\Sigma}_i$ are the mean vector and the covariance matrix of the compensated model in mel-cepstral domain, C is the DCT matrix and C^T is the transpose of the DCT matrix.

Notice that only the mean vectors and the covariance matrices of the Gaussian components have been changed. The implicit assumption is that the weights of the components remain unchanged throughout the compensation process. Also, the above described compensation scheme does not depend on the max function. Therefore, it can be applied to any non-linear noise corruption function as long as the image of the corruption function is positive. This constraint of positive image comes from the fact that the distribution of the images of the corruption function is assumed to be log-normal. This constraint is not limiting, however, since the image of the valid corruption function consists of mel-filter-outputs and mel-filter-outputs are always positive.

4.4.4 Relationship Between the PMC Compensation Scheme and the Proposed Scheme

It should be noted that the steps described in Sections 4.4.1 and 4.4.3 are identical to their counter parts in the PMC compensation scheme. The real novelty of the proposed scheme lies in the steps described in Section 4.4.2. It should also be noted that the proposed compensation scheme reduces to the PMC compensation scheme if the corruption function is additive instead of being non-linear. Consider the PMC corruption function $f(\xi_s, \xi_n) = \xi_s + \xi_n$. From Equation (4.15), it can be observed that for the PMC corruption function:

$$\delta_{pv} = n'_{pv} \tag{4.28}$$

Therefore, the compensation process described in Equations (4.13) and (4.14) reduce to the PMC compensation described in Equations (4.6) and (4.7).

Both compensation schemes assume that the compensation factor δ shows up as an additive component in the mel-filter-output domain and the compensation scheme essentially estimates the log-normal parameters of the random variable δ. For the additive corruption function of the PMC, δ equals the noise observation and therefore the parameters of δ can be estimated explicitly from the observed noise samples. For a non-linear function like the max function, δ can be zero or negative and explicit estimation of its parameters is not possible. The proposed scheme provides a work-around for the estimation of the log-normal parameters of δ without relying on the individual sample values of δ. Table 4.1 summarizes this.

Table 4.1: PMC and Proposed Compensation Scheme

	PMC	Proposed
Compensated Model Mean	$\mu'_p + E\left[n'_{pv}\right]$	$E\left[f\left(\mu'_p, n'_{pv}\right)\right]$
Compensated Model Variance	${\sigma'_{pq}}^2 + \text{Covar}\left[n'_{pv}\right]$	${\sigma'_{pq}}^2 + \text{Covar}\left[f\left(\mu'_p, n'_{pv}\right)\right]$

4.5 Experiments

Experiments were conducted to gauge the performance of the proposed max function based model compensation and to compare it with the performance of the PMC. This section presents the experimental set-up, results and analysis of these experiments.

4.5.1 Experimental Set-up

Experiments were conducted on the TIMIT database under a set-up identical to what has been described in Chapter 3, Section 3.4.1. For a detailed descrip-

tion of the experimental set-up, the reader is referred to the above mentioned section. The speaker models and the UBM were trained using clean speech. The mismatch condition was created by adding white and pink noise at SNRs of 5 dB and 10 dB.

Two systems were compared through the experiments. The first system is the PMC system, where the models were trained using the clean speech and these models were compensated by the PMC technique during the verification phase. The second system is the max function system, where the models were, again, trained using the clean speech, but were compensated by the max-function compensation scheme during the verification.

The prevalent noise was estimated by the simple voice activity detector described in Chapter 3, Section 3.4.1. For the max function compensation scheme, these observed noise frames were processed in the same manner as the speech frames up until the mel-filter-output computation stage. Therefore, the noise frames were kept in the mel-filter-output domain for the model compensation purpose. The overall verification module is depicted in Figure 4.3. The verification module is essentially same for both the PMC and the max-function based compensation. Only the compensation scheme differs as explained above in Section 4.4.4.

4.5.2 Experimental Results

First the two systems were compared for test utterances corrupted with white noise at SNRs of 10 dB and 5 dB. Figure 4.4 presents the ROC curves from these experiments. The ROC curves for the verification system without any compensation is presented in Chapter 3, Section 3.4.2. As can be seen from the DET curves, both PMC and max function based model compensation are much superior to the system without any compensation, which is expected. Comparing the curves of PMC and max function systems, it can be observed that the max function system holds a clear advantage over the PMC system over a broad range of false alarm probabilities and miss probabilities.

Table 4.2 summarizes the EERs obtained from these experiments. The EERs also reflect the better performance of the max function for the white noise. It can be observed that the PMC, on an average, reduced the EERs

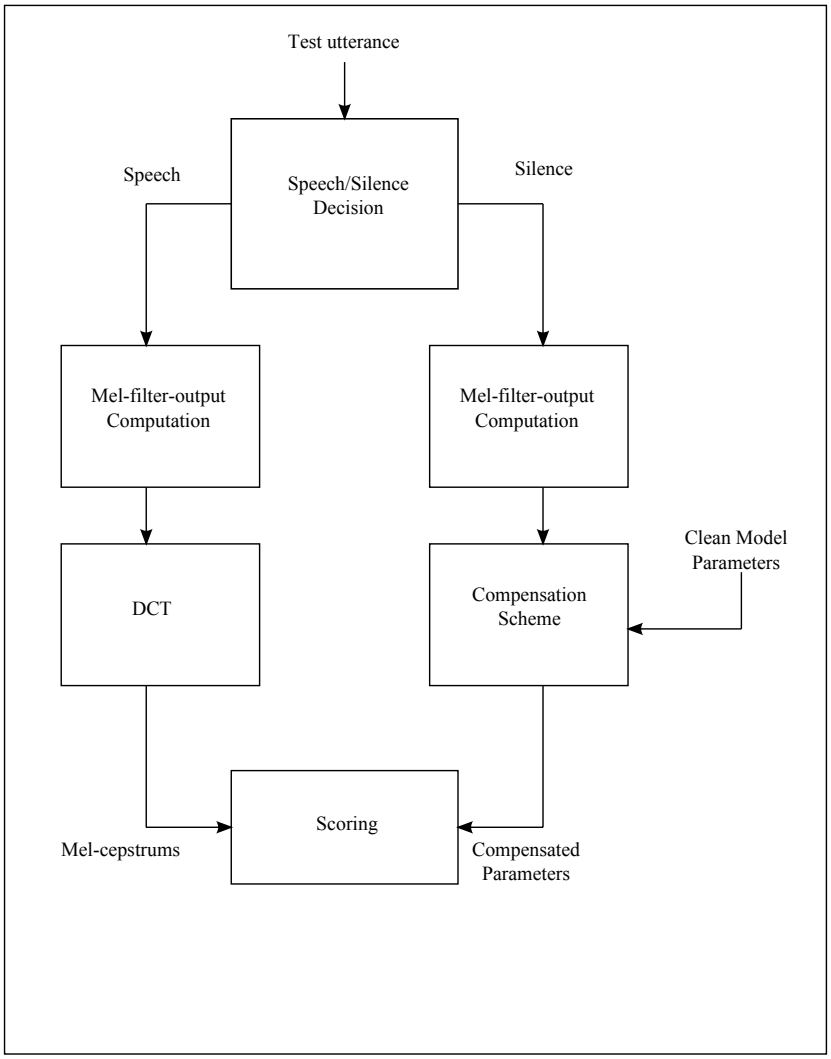


Figure 4.3: Verification Module for Non-linear Corruption Function - This figure illustrates the various steps in the verification module for a non-linear corruption function

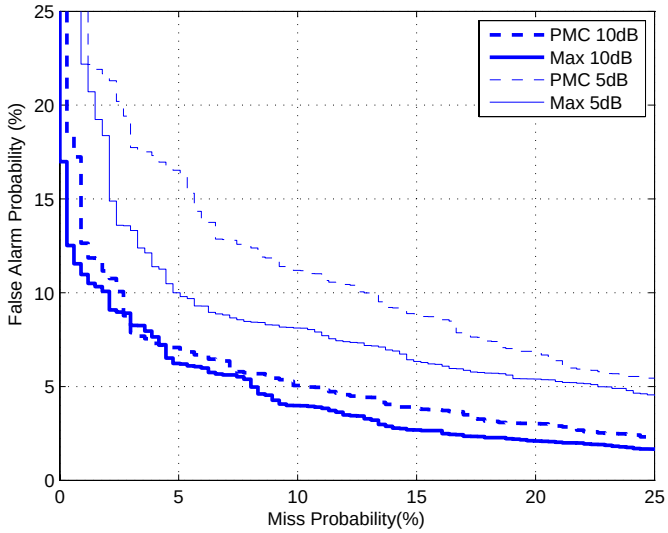


Figure 4.4: ROC Curves Comparing Max function Compensation and PMC for White Noise - This figure presents the ROC curves from the experiments comparing max function and PMC for white noise

by 69.5% while the max function based compensation reduced the same by 74.5%.

Table 4.2: EERs for Max-function Compensation and the PMC for White Noise(%)

Test Noise	White 10 dB	White 5 dB
No Compensation	23.6	32.9
PMC	6.5	11.0
Max-function	6.0	8.4

The next set of experiments compared the PMC and the max function based compensation for test utterances corrupted with pink noise at SNRs of 5 dB and 10 dB. Figure 4.5 depicts the ROC curves from these experiments. It can be observed that for the pink noise, the comparison is a bit difficult as no technique has a clear advantage. At the SNR of 5 dB, for false alarm rates above 12%, max-function based compensation provides better performance and for false alarm rates below 12%, PMC has an advantage. At the SNR of 10 dB, the ROC curves for the two techniques are virtually indistinguishable from one another.

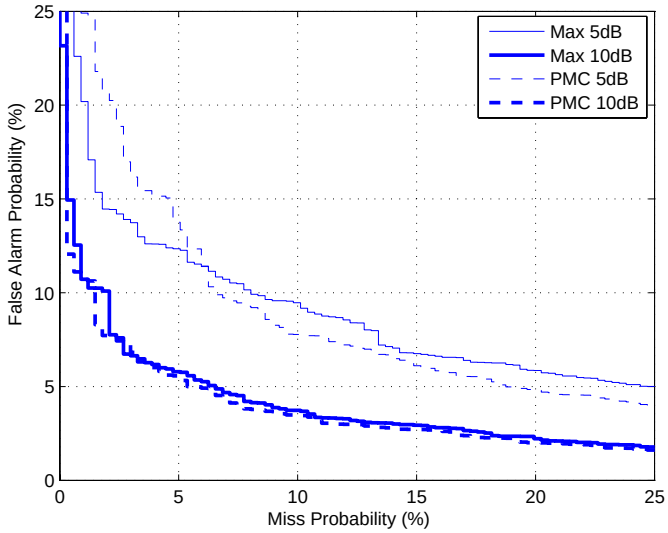


Figure 4.5: ROC Curves Comparing Max-function Compensation and PMC for Pink Noise - This figure presents the ROC curves from the experiments comparing max-function and PMC for pink noise

Table 4.3 summarizes the EERs from the experiments involving pink noise. Again, both the techniques perform significantly better than the system without any compensation. From the EER point of view, the PMC provides slightly better gains. The PMC, on an average, reduced the EER by 76.2% while the max-function based compensation reduced the same by 74.2%.

Table 4.3: EERs for Max-function Compensation and the PMC for Pink Noise(%)

Test Noise	Pink 10 dB	Pink 5 dB
No Compensation	22.1	36.6
PMC	5.3	8.6
Max-function	5.6	9.6

4.5.3 Analysis

From the experimental results presented above, it can be concluded that while the max-function based compensation provides better gains for test utterances corrupted with white noise, the PMC holds a slight advantage for test utterances corrupted with pink noise. As has been explained before, the compen-

sation procedure has two important components, namely the noise corruption function and the compensation scheme. It has been shown in Section 4.4.4 that the compensation scheme is, essentially, the same in both approaches of model compensation. Therefore, the difference in the performances can be attributed to the noise corruption function. While the max function maintained a gain of 74% in both types of noise, the additive function provides less gain for white noise and more gain for pink noise.

These observations point to the possibility that while the max function is a better approximation of corruption by white noise, the additive function is a better approximation of corruption by pink noise. An ideal noise corruption function should provide robust approximation for both types of noise and neither of these functions achieve that. This serves as a motivation for the novel noise corruption function introduced in the next chapter.

4.6 Summary

This chapter provided a detailed look into the model compensation technique and identified the noise corruption function and the compensation scheme as two important components of the model compensation process. A novel compensation scheme for the non-linear noise corruption functions was developed. The proposed compensation scheme with max function was employed as an alternative model compensation technique. Experiments conducted on TIMIT database with artificial noise show that while the max-function provides better performance in white noise, the additive function of PMC provides better performance in pink noise, indicating that neither of these functions provide robust performance under both types of noise. The next chapter introduces the psychoacoustic noise corruption function, which provides better performance irrespective of the type of the prevalent noise.

5

Psychoacoustic Model Compensation

5.1 Introduction

This chapter presents a novel model compensation technique based on the psychoacoustic principles. It begins with an overview of the psychoacoustic principles such as the concept of auditory masking and the critical bands. Next, psychoacoustic model is introduced as a set of rules for the masking procedure and the masking thresholds are explained. A new noise corruption function which employs the masking thresholds defined by the psychoacoustic model is derived. Experiments conducted on the TIMIT database with artificial noise demonstrate the superior performance of the proposed psychoacoustic model compensation.

5.2 Psychoacoustic Principles

Most current audio coders achieve compression by exploiting the fact that "irrelevant" signal information is not detectable by even a well trained or sensitive listener [104]. Irrelevant information is identified during signal analysis by incorporating into the coder several psychoacoustic principles such as critical bands and auditory masking. While the concept of critical bands tries to

emulate the spectral analysis as done by the human ear, the concept of auditory masking tries to estimate the audible signal in the presence of competing signals.

5.2.1 Critical Bands

The modern notion of critical bands stems from the empirical work by several researchers [105][106]. It turns out that a frequency-to-place transformation takes place in the inner ear and distinct regions in the cochlea are tuned to different frequency bands. The critical bandwidth can be loosely defined as the bandwidth at which subjective responses change abruptly.

Generally, the ear is modeled as a set of 25 discrete bandpass filters. The bandwidths of these filters are known as critical bandwidths. Critical bandwidth tends to remain constant upto 500 Hz and increases to about 20% of the center frequency above 500 Hz. Table 5.1 presents an idealized filterbank described in [106].

The Bark scale of perceived frequency is based on the concept of critical bands. The width of one critical band is commonly referred to as one Bark. A frequency of h Hz can be converted to z Bark using the following equation [107]:

$$\begin{aligned} z &= 13 * \arctan\left(0.00076\, h\right) \\ &\quad +3.5 * \arctan\left[\left(\frac{h}{7500}\right)^2\right] \end{aligned} \tag{5.1}$$

The non-uniform Hz spacing of the critical bands is rendered uniform on a bark scale. The concept of Bark scale is important as the masking properties associated with the inner ear often employ the bark scale. The auditory masking properties are explained next.

5.2.2 Auditory Masking

Masking refers to a process where a “weaker” sound is rendered inaudible by the presence of a “stronger” sound. The “stronger” sound is referred to as

Table 5.1: Critical Band Filterbank

Band Number	Center Frequency (Hz)	Bandwidth (Hz)
1	50	0-100
2	150	100-200
3	250	200-300
4	350	300-400
5	450	400-510
6	570	510-630
7	700	630-770
8	840	770-920
9	1000	920-1080
10	1170	1080-1270
11	1370	1270-1480
12	1600	1480-1720
13	1850	1720-2000
14	2150	2000-2320
15	2500	2320-2700
16	2900	2700-3150
17	3400	3150-3700
18	4000	3700-4400
19	4800	4400-5300
20	5800	5300-6400
21	7000	6400-7700
22	8500	7700-9500
23	10500	9500-12000
24	13500	12000-15500
25	19500	15500 -

the masker and the "weaker" sound is referred to as the maskee. There are two types of masking, namely simultaneous masking and temporal masking.

Simultaneous masking is a frequency domain phenomenon which has been observed within the critical band. Simultaneous masking can be of two types: *tone-masking-noise* and *noise-masking-tone*. In the first case, a tone like signal in a critical band renders the noise signal within the critical band inaudible provided the noise spectrum is below a predictable threshold directly related to the strength of the masking tone. The second case follows the same pattern with the masker and maskee roles reversed. A simplified explanation for the masking phenomenon is that the masker signal creates an excitation, in the basilar membrane, of sufficient strength to block the transmission of any weaker signal [104].

Inter-band masking has also been observed in simultaneous masking. A masker centered within one critical band can have some predictable effect on the masking thresholds in other critical bands. This effect is known as the spread of masking. The spread of masking is often modeled by an approximately triangular spreading function which has slopes of +25 dB and -10 dB per Bark.

Therefore, the concept of simultaneous masking can be completely described by the definition of masking thresholds for tone and noise and a spreading function. With the help of masking thresholds and spreading function, the audibility of a signal can be determined when competing signals occur simultaneously.

Temporal masking takes place in the time domain. The absolute audibility thresholds for a masked sound are artificially increased prior to, during and following the occurrence of a masking signal. Premasking tends to last only about 5 ms, whereas postmasking lasts anywhere between 50 ms to 300 ms, depending on the strength and duration of the masker [104].

5.3 Proposed Application of Psychoacoustics to Model Compensation

In an interesting study of human capability for speaker recognition, it was found that while automatic speaker recognition systems outperformed human beings in clean speech conditions, human beings outperformed the automatic systems in noisy conditions [108]. In [109] too, human beings are reported to have outperformed the automatic systems in mismatch conditions. Therefore, it is conceivable that emulating the human response in automatic speaker recognition systems might improve the performance of the automatic systems. Psychoacoustics has a long and successful history of application in the field of speech processing. The feature vectors used in this work, MFCCs, rely on the psychoacoustic concept of perceived frequency [9]. The concept of noise masking has also been found to have improved the performance of speech enhancement in [72] and [110]. However, the concept of psychoacoustics has never been applied for model compensation.

As has been mentioned before, the goal of model compensation is to transform the model parameters in order to create a matched condition. We now propose a way to employ the psychoacoustic principles, described above, to transform the model parameters.

Consider the noisy test utterances. The clean test speech and the prevalent noise are two competing signals. The resultant audible signal is, therefore, a masked version of the clean test utterance. The clean training speech, on the other hand, remains completely audible in the absence of any noise. Hence the mismatch condition is the result of the lack of the masking operation during the training phase. This can be remedied by subjecting the training speech to the auditory masking process.

However, the verification phase of the speaker verification system does not have access to the training speech. It has access to only the model parameters. Consider the mean vectors of the clean model in the mel-filter-output domain. They can be thought of as the average mel-filter outputs for the clean training speech. Using an estimate of the noise prevalent during the verification phase, the auditory masking process can be applied on the clean model means, with the model means serving as tone maskers.

Out of the two types of masking, the temporal masking cannot be employed for masking the model means. This is because, in order to employ the temporal masking, the sequence of occurrence of the speech and the noise frames must be determined. For example, in order to compute the temporal masking effect of a noise signal on the speech signal, the time gap between the two signals must be computed. This is not possible in this case since the noise estimate is from an entirely different utterance.

Therefore, the concept of simultaneous masking can be used for masking the model means. The underlying assumption is that any of the observed noise frames is equally likely to have corrupted the speech signal by occurring simultaneously. Employing simultaneous masking for the model compensation involves three tasks. Firstly, the masking thresholds and the spreading function should be specified. The computationally efficient psychoacoustic model 1 is used for this purpose.Secondly a psychoacoustic noise corruption function should be defined. An important contribution of this work lies in the innovative application of the masking thresholds in determining the psychoacoustic noise corruption function. Thirdly, a compensation scheme should be implemented that employs the psychoacoustic noise corruption function for model compensation.

5.3.1 Psychoacoustic Model - 1

Psychoacoustic model analyzes the audio signals and computes the amount of noise masking available as a function of frequency [111]. The MPEG audio standards provide two implementations of psychoacoustic model: psychoacoustic model 1 and psychoacoustic model 2. Psychoacoustic model 1 is computationally more efficient and is suitable for MPEG layer I and II compression [111]. This model is described next.

According to the psychoacoustic model 1 [112], the masking threshold of the signal depends on the type of signal (tone or noise), location of signal in the bark scale and the level of the signal. Consider a tone masker located at frequency bin j. Its masking effect on frequency bin i is given by its masking

threshold T_s:

$$\begin{aligned} T_s(i,j) &= P_s(j) - 0.275z(j) \\ &\quad + SF(i,j) - 6.025 \quad (dB) \end{aligned} \tag{5.2}$$

In the above equation, $P_s(j)$ is the level of the tone at frequency bin j in dB, $z(j)$ denotes the bark value of frequency bin j and $SF(i,j)$ is a spreading function. Similarly, the masking threshold T_n of a noise masker located at frequency bin j is given by

$$\begin{aligned} T_n(i,j) &= P_n(j) - 0.175z(j) \\ &\quad + SF(i,j) - 2.025 \quad (dB) \end{aligned} \tag{5.3}$$

The spreading function in Equations (5.2) and (5.3) is given by:

$$\begin{aligned} &SF(i,j) = \\ &\begin{cases} 17\delta_z - 0.4P(j) + 11, & -3 \leq \delta_z < -1 \\ (0.4P(j) + 6)\,\delta_z, & -1 \leq \delta_z < 0 \\ -17\delta_z, & 0 \leq \delta_z < 1 \\ (0.15P(j) - 17)\,\delta_z - 0.15P(j), & 1 \leq \delta_z < 8 \end{cases} \\ &(dB) \end{aligned} \tag{5.4}$$

In the above equation, δ_z stands for the maskee-masker separation in Bark, i.e.,$\delta_z = z(i) - z(j)$ and $P(j)$ stand for level of the masker at frequency bin j, whether noise or speech.Also, according to the spreading function, the masker's effect is limited to maskee-masker separation of -3 through 8.

From the above description of the psychoacoustic model 1, it can be observed that noise is a stronger masker than tone in the sense that for the same signal loudness, it has a higher masking threshold. Consider a tone and a noise with loudness 15 dB located at 10 bark. From Equations (5.2) and (5.3), it can be computed that while the tone has a masking threshold of 6.22 dB at 10 bark, the noise has a masking threshold of 11.22 dB at 10 bark. In other words, a tone of 15 dB loudness at 10 bark can mask a signal with loudness of up to 6.22 dB, while a noise of the same loudness in the same bark location can mask a signal with loudness of up to 11.22 dB. The masking effect of these

signals at various bark locations is plotted in Figure 5.1.

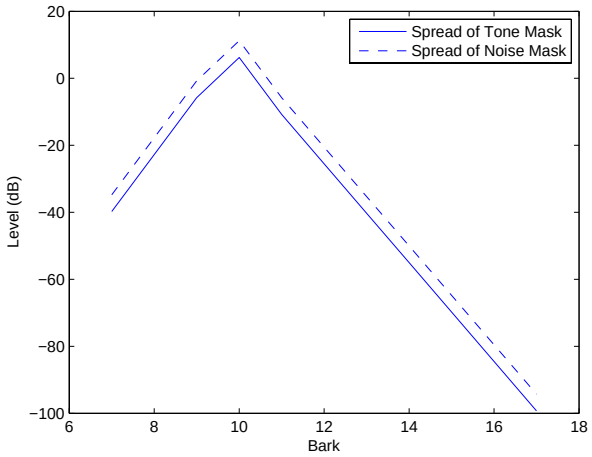


Figure 5.1: Spreading of the Masking Effect - This figure compares the spread of the masking effect of a tone and a noise signal of 15 dB loudness located at 10 bark

5.3.2 Psychoacoustic Noise Corruption Function

Now we discuss the application of the above principles to model compensation. Let us consider a mel-cepstral feature vector of a clean speech signal $\vec{s} = (s_1 s_2 \ldots s_D)$. By applying inverse discrete cosine transform and taking exponentiation, the mel-filter output of the speech signal can be determined. Let us denote the mel-filter output of the speech signal for filter f as s'_f. Similarly, we denote the mel-filter output of the noise signal for filter f as n'_f. According to psychoacoustic principles, if the speech signal and the noise signal occur simultaneously, then the psychoacoustic masking procedure would give rise to the resultant noisy signal s''_f. We estimate the resultant s''_f by treating the speech signal as the tone masker and the noise signal as the noise masker.

Applying the psychoacoustic model 1 to mel-cepstral coefficients has some limitations. The mel-filter output is a weighted combination of the spectral magnitudes within the mel-filter bandwidth. Therefore, signal level at individual frequency bins cannot be determined from the mel-filter output. This can be circumvented by considering the mel-filter output as one signal located at the central frequency of the mel-filter. To further simplify the computation

of the masking threshold, let us assume that the masking effect of a signal in one mel-filter does not extend to other mel-filters, i.e., a signal located in mel-filter f can only mask other signals located in the same mel-filter f. This assumption essentially makes the masker-maskee separation zero, i.e., $\delta_z = 0$. Therefore, according to Equation (5.4), $SF(i,j) = 0$.

Let the central frequency of the mel-filter f be c_f Hertz. Using Equation (5.1), this can be converted into c_z Bark. According to Equations (5.2) and (5.3), the masking threshold for the speech signal, T_s and the masking threshold for the noise signal, T_n can be computed as follows:

$$T_s = 20\log_{10}(s'_f) - 0.275c_z - 6.025 \quad (dB) \tag{5.5}$$

$$T_n = 20\log_{10}(n'_f) - 0.175c_z - 2.025 \quad (dB) \tag{5.6}$$

Equation (5.5) implies that the speech signal s'_f will mask out any noise signal with a level below T_s dB. Similarly, Equation (5.6) implies that any speech signal with a level below T_n dB will be masked out by the noise signal n'_f. Therefore, if $T_s > 20\log_{10}(n'_f)$, then the speech signal will completely mask out the noise signal and as a result the resultant signal s''_f would consist of clean speech only. Similarly, if $T_n > 20\log_{10}(s'_f)$, then the resultant signal s''_f would consist entirely of noise.

Consider the third possibility, where neither signal is able to completely mask the other. Due to the presence of the noise signal, T_n dB of the speech level is masked. Similarly, T_s dB of the noise level is masked due to the speech signal. The masked portion of the signal is inaudible. Let us define the potential contribution of a signal as the audible (or non-masked) portion of the signal. Therefore, the potential contribution of the speech signal towards s''_f is $s'_f - 10^{\frac{T_n}{20}}$, while the potential contribution of the noise signal towards s''_f is $n'_f - 10^{\frac{T_s}{20}}$. Note that the masking thresholds T_n and T_s are computed based on the signal level and hence it is difficult to compute them if both signals are masked simultaneously. Therefore, we assume that the signal with larger potential contribution to the resultant signal becomes the dominant signal and remains unchanged while the non-dominant signal is partially masked by it.

If the noise signal is dominant, the level of the resultant signal $s_f^{''}$ can be given by:

$$20\log_{10}(s_f^{''}) = 20\log_{10}\left(n_f^{'} + \left[s_f^{'} - 10^{\frac{T_n}{20}}\right]\right) \quad (dB) \tag{5.7}$$

If, however, the speech signal is dominant, the resultant signal can be given by:

$$20\log_{10}(s_f^{''}) = 20\log_{10}\left(s_f^{'} + \left[n_f^{'} - 10^{\frac{T_s}{20}}\right]\right) \quad (dB) \tag{5.8}$$

In Equations (5.7) and (5.8) the expressions within the square brackets represent the contribution of the non-dominant signal to the resultant signal.

The psychoacoustic masking procedure explained above can be summarized by the following equation.

$$s_f^{''} = \begin{cases} s_f^{'}, & 10^{\frac{T_s}{20}} > n_f^{'} \\ n_f^{'}, & 10^{\frac{T_n}{20}} > s_f^{'} \\ s_f^{'} + n_f^{'} - 10^{\frac{T_s}{20}}, & \text{State-I} \\ n_f^{'} + s_f^{'} - 10^{\frac{T_n}{20}}, & \text{State-II} \end{cases} \tag{5.9}$$

In the above equation, “State-I" refers to the condition: $(s_f^{'} - 10^{\frac{T_n}{20}}) > (n_f^{'} - 10^{\frac{T_s}{20}})$, while “State-II" refers to the condition: $(n_f^{'} - 10^{\frac{T_s}{20}}) > (s_f^{'} - 10^{\frac{T_n}{20}})$. Figure 5.2 illustrates the flowchart for the above equation.

Consider a frame of clean speech signal and a frame of noise signal whose mel-filter-outputs are given in Figure 5.3. Noise masking threshold is shown against the mel-filter-outputs of the clean speech signal in Figure 5.4. The speech masking thresholds are plotted against the noise mel-filter-outputs in Figure 5.5.

It can be observed that, in this particular example, while the noise masks the speech signal in mel-filters number 1 through 3 and 11 through 27, the speech signal masks the noise in mel-filters number 4 through 10. Using Equation (5.9), therefore, one can obtain an estimate of the noisy speech signal. The estimated noisy speech signal is plotted against the actual noisy signal in Figure 5.6. In this instance, the estimated noisy signal approximates the actual noisy signal pretty accurately.

Consider a Gaussian mixture model trained on clean melcepstral feature vectors. The model means are essentially melcepstral coefficients, which can

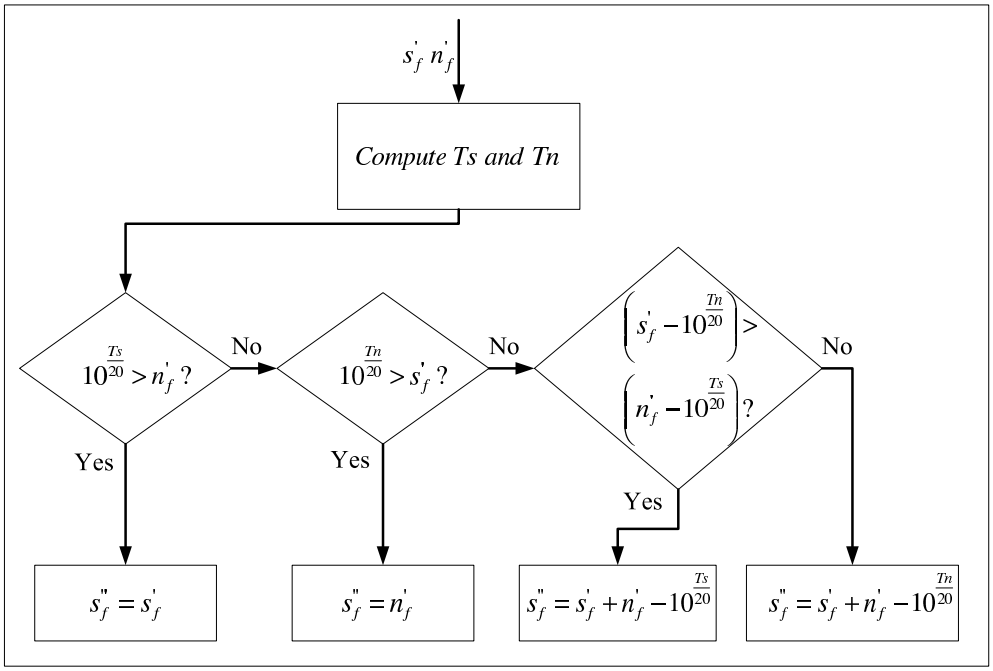


Figure 5.2: Psychoacoustic Noise Corruption Function - This figure describes the psychoacoustic noise corruption function in Equation (5.9) in the form of a flow-chart.

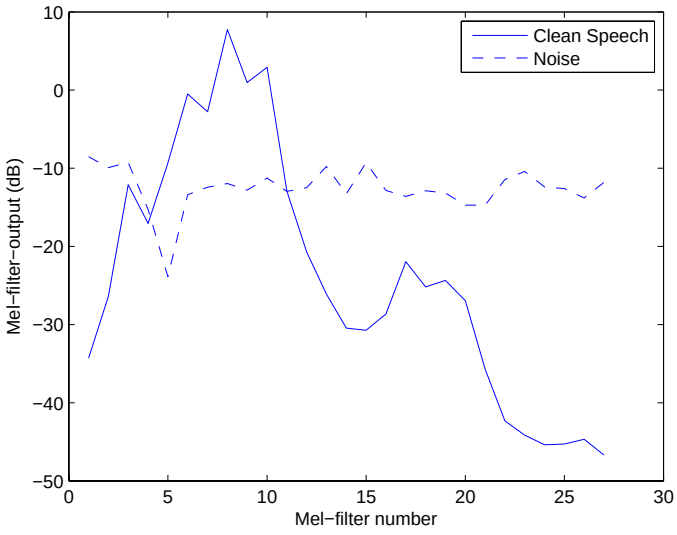


Figure 5.3: Mel-filter-outputs of Clean Speech and Noise - This figure provides an example of a frame of clean speech and noise signal

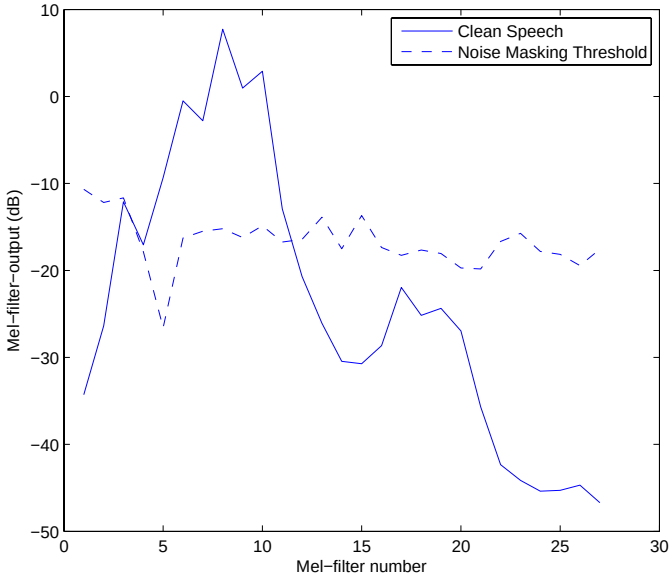


Figure 5.4: Mel-filter-outputs of Clean Speech and Noise Masking Threshold - This figure compares the clean speech with the noise masking threshold

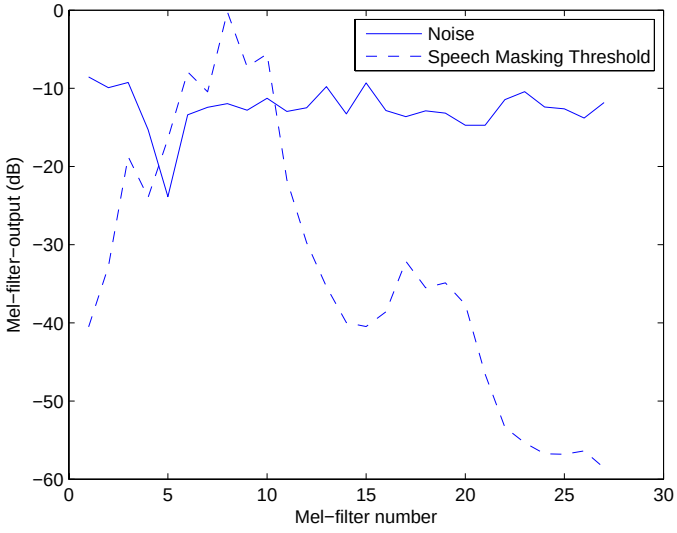


Figure 5.5: Mel-filter-outputs of Noise and Speech Masking Threshold - This figure compares the noise with the speech masking threshold

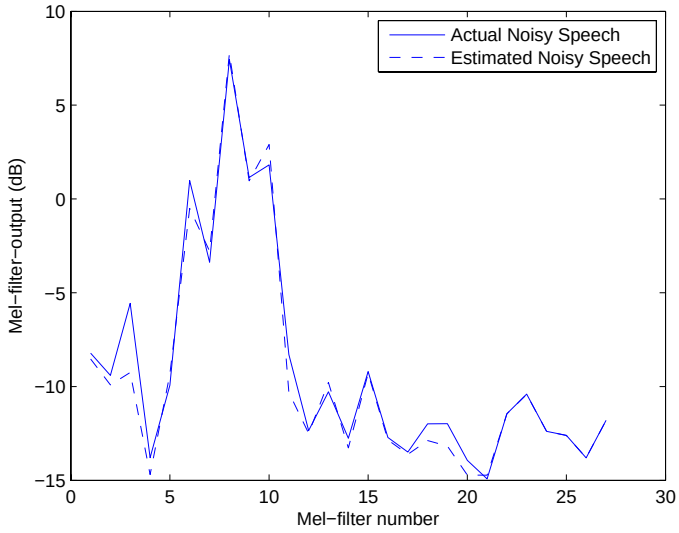


Figure 5.6: Mel-filter-outputs of Actual and Estimated Noisy Speech Signals - This figure compares the actual noisy signal with noisy signal estimated by psychoacoustic noise corruption function

be transformed into mel-filter output domain by applying inverse discrete cosine transform (IDCT). Given an estimated sample of the noise during the verification process, the model means can be compensated using the psychoacoustic masking described in Equation (5.9). The model means can, again, be transformed back to the melcepstral domain, once the compensation is achieved in the mel-filter output domain, by applying discrete cosine transform (DCT).

5.3.3 Psychoacoustic Noise Corruption Function in Relation to Max-function and Additive Function

The additive function of the PMC views the mel-filter-outputs of the noisy speech as the sum of the mel-filter-outputs of clean speech and noise signal. The max-function, on the other hand, implements a simple masking operation and the criterion for the masking operation in max-function is the signal level. The psychoacoustic noise corruption function implements both of these eventualities in a more refined manner.

When the level of one signal falls below the masking threshold of the other

signal, then the psychoacoustic function behaves like the max-function with the dominant signal completely masking the non-dominant signal. Unlike the max-function, however, the criteria for masking are level of the signals and the masking thresholds of each signal. If neither signal is able to mask the other, then the psychoacoustic function behaves like the additive function. Unlike the additive function, however, only the non-masked portions of the signals are added. The underlying assumption is that the dominant signal is not masked at all, while the non-dominant signal is partially masked.

In the last chapter, it was observed that while the max-function performs better in white noise, the additive function performs better in pink noise. It was also remarked that a corruption function that encompasses both these functions might be a better corruption function. Since the psychoacoustic noise corruption function encompasses both these functions and it incorporates advanced psychoacoustic principles, it is expected to perform better in both white and pink noise scenarios.

5.3.4 Compensation Scheme

As can be seen from Equation 5.9, the psychoacoustic noise corruption function is non-linear. If the compensated vectors are expressed in the form $\vec{x}+\vec{\delta}$, then, as explained in Section 4.4.4 of Chapter 4, the components of $\vec{\delta}$ can be zero or negative. However, the components of $\vec{x}+\vec{\delta}$ are always positive. Therefore, the compensation scheme developed in Section 4.4 of Chapter 4 can be applied for the psychoacoustic noise corruption function.

5.4 Experiments

Experiments were conducted to test the proposed algorithms. This section presents the experimental set-up, the results and the analysis of these experiments.

5.4.1 Experimental Set-up

Experiments were conducted on the TIMIT database under a set-up identical to what has been described in Chapter 3, Section 3.4.1. For a detailed description of the experimental set-up, the reader is referred to the above mentioned section. The speaker models and the UBM were trained using clean speech. The mismatch condition was created by adding white and pink noise at SNRs of 5 dB and 10 dB.

The central frequencies of the mel-filters were determined as follows. The sampling rate for the TIMIT database is 16 KHz. Therefore the speech band width is 0 - 8 KHz. When converted to mel scale, the speech bandwidth translates into 0 - 2840 mel. In this bandwidth, the 27 central frequencies are equidistant from each other. So the location of the central frequency of the first mel-filter can be computed to be 101.4 mel. The subsequent central frequencies are located at multiples of 101.4 mel. These values are then converted to Hz and then to bark. Table 5.2 lists the bark values of some of the central frequencies.

Table 5.2: Bark Values of Central Frequencies of Mel-Filters

Filter Number	Center Frequency (Bark)
1	0.65
3	2.12
5	3.82
7	5.70
9	7.66
11	9.61
13	11.44
15	13.10
17	14.58
19	15.92
21	17.16
23	18.35
25	19.54
27	20.71

Three systems were compared through the experiments: the PMC model compensation system, the max-function model compensation system and the

proposed psychoacoustic model compensation system. In all these three systems clean speech was used to train the models and the models were compensated during the test phase based on the noise estimate provided by the voice activity detector.

5.4.2 Experimental Results

The three systems were first compared for test utterances corrupted with white noise at SNRs of 5 dB and 10 dB. Figures 5.7 and 5.8 present the ROC-curves from experiments involving white noise. The curves from the psychoacoustic model compensation scheme are identified by the legend "Psychoacoustic". It can be seen from the ROC curves that the psychoacoustic model compensation scheme outperforms the PMC and the max-function based model compensation.

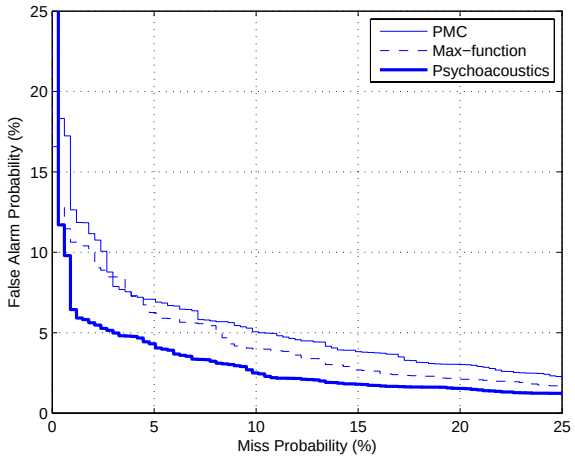


Figure 5.7: ROC Curves for White Noise at 10 dB SNR - This figure compares the compares the ROC-curves of PMC, Max-function and Psychoacoustic model compensation for white noise at 10 dB SNR

Table 5.3 summarizes the EERs from the above experiments. The "No Compensation" system refers to the system without any noise compensation. It can be observed that the psychoacoustic model compensation scheme provides the lowest EER of all schemes reducing the no compensation EER, on an average, by over 80%. As compared to the PMC and the max-function based

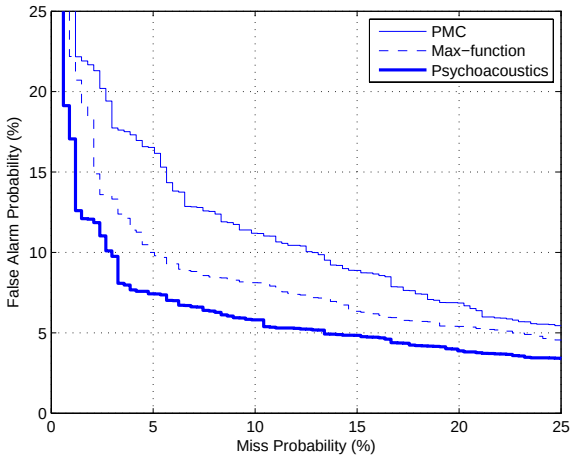


Figure 5.8: ROC Curves for White Noise at 5 dB SNR - This figure compares the ROC-curves of PMC, Max-function and Psychoacoustic model compensation for white noise at 5 dB SNR

compensations, the psychoacoustic scheme reduces the EER by 36% and 24% respectively.

Table 5.3: EERs for Various Compensation Schemes under White Noise(%)

Test Noise	White 10 dB	White 5 dB
No Compensation	23.6	32.9
PMC	6.5	11.0
Max-function	6.0	8.4
Psychoacoustic	4.4	6.6

Next the three systems were compared for test utterances corrupted with pink noise at SNRs of 10 dB and 5 dB. Figures 5.9 and 5.10 present the ROC-curves from experiments involving pink noise. Again it can be seen that the psychoacoustic scheme outperforms the other two. The ROC curves for the psychoacoustic scheme can be identified under the legend “psychoacoustic”.

Table 5.4 summarizes the EERs from the experiments involving pink noise. Again, it can be observed that the psychoacoustic model compensation scheme provides the lowest EER of all schemes reducing the no compensation EER, on an average, by over 80%. This translates to a reduction in EER of 29% and 26% as compared to the PMC and the Max-function schemes respectively.

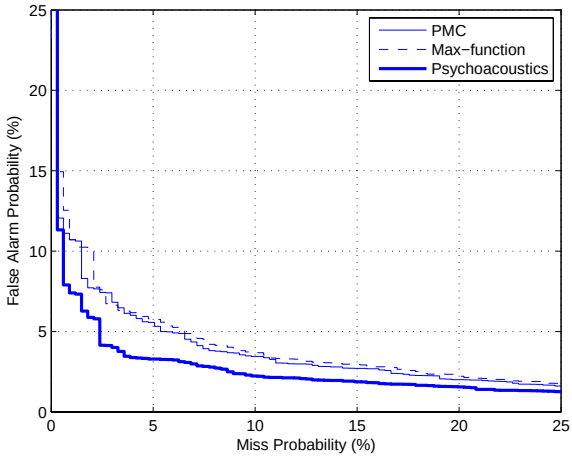


Figure 5.9: ROC Curves for Pink Noise at 10 dB SNR - This figure compares the compares the ROC-curves of PMC, Max function and Psychoacoustic model compensations for pink noise at 10 dB SNR

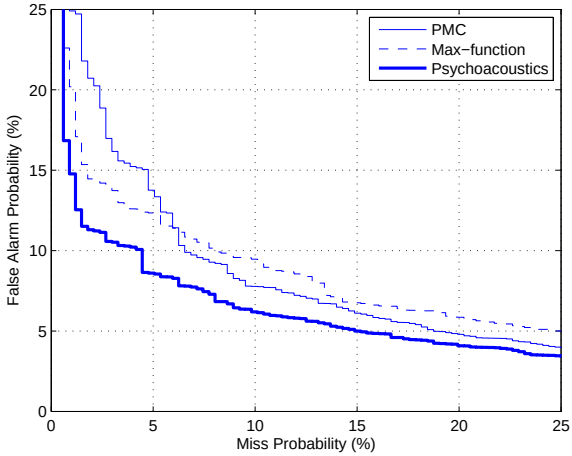


Figure 5.10: ROC Curves for Pink Noise at 5 dB SNR - This figure compares the compares the ROC-curves of PMC, Max function and Psychoacoustic model compensations for pink noise at 5 dB SNR

Table 5.4: EERs for Various Compensation Schemes under Pink Noise(%)

Test Noise	White 10 dB	White 5 dB
No Compensation	22.1	36.6
PMC	5.3	8.6
Max-function	5.6	9.6
Psychoacoustic	3.6	7.5

It should be noted that irrespective of the type of noise the psychoacoustic model compensation scheme performs consistently better than the other two schemes. It seems to support our earlier assertion that psychoacoustic scheme, encompassing both additive and max noise corruption functions, should perform better under both, white and pink, noise types.

5.4.3 Analysis

The psychoacoustics emulate the response of human ears. The experiments reported in this chapter demonstrate the effectiveness of employing psychoacoustics in model compensation. The psychoacoustic noise corruption function provides a more robust approximation of the noise corruption process than the max function or the additive function.

The experiments conducted in this chapter rely on the noise estimates provided by the simple frame-energy based voiced activity detector. The speech data (from TIMIT) used for the experiments with artificial noise provide an ideal setting for such a voice activity detector. In real-life scenarios, the noise is likely to be more challenging and the voice activity detector might not provide a good estimation of the noise, which can lead to reduction in performance.

Without a robust methodology to address the problem of inaccurate noise estimation, the applicability of the psychoacoustic model compensation has only limited potential. This problem is addressed in the next chapter. The proposed technique is tested in a realistic speech database which contains Lombard effect and session variability. A novel multi-conditioning technique is developed which addresses the problem of inaccuracies in noise estimation.

5.5 Summary

In this chapter, a novel psychoacoustic model compensation technique was proposed. Taking advantage of the progress made in the field of psychoacoustics, a robust noise corruption function is developed. A significant contribution of the proposed technique lies in the innovative use of the masking thresholds to define the psychoacoustic noise corruption function. Ex-

periments conducted on the TIMIT database corrupted with artificial noise demonstrate the superiority of the proposed technique. However, the proposed technique relies on estimated noise for the compensation procedure and inaccuracies in the noise estimation may translate into performance penalty. The next chapter addresses this issue through psychoacoustic multi-conditioning.

6

Psychoacoustic Model Compensation in Realistic Noise

6.1 Introduction

In this chapter the psychoacoustic model compensation technique is extended further to accommodate realistic noise. A common feature of the various model compensation techniques is their dependence upon the estimated noise for good performance. The noise estimation is particularly difficult in realistic scenarios as the noise could be time-varying and unpredictable. A novel psychoacoustic multi-conditioning technique is developed in this chapter which does not need an accurate estimate of the prevalent noise. A voice activity detector is used to provide a rough estimate of the prevalent noise and scaled values of the noise estimates are used to create a multi-SNR GMM system. Experiments conducted on TIMIT and MIT Mobile Devices Speaker Verification Corpus (MITMDSVC) demonstrate the efficacy of the proposed technique.

6.2 Synthetic Noise and Realistic Noise

Synthetic noise such as the white and pink noise, as has been used in this work to corrupt the TIMIT speech database, are stationary. Figure 6.1 and 6.2 show the spectrograms of a speech utterance corrupted with white and

pink noise respectively. It can be observed from the spectrograms that the noise characteristics does not change significantly with time. Such noise types are easier to estimate and the voice activity detectors, generally, provide an accurate estimation of the noise.

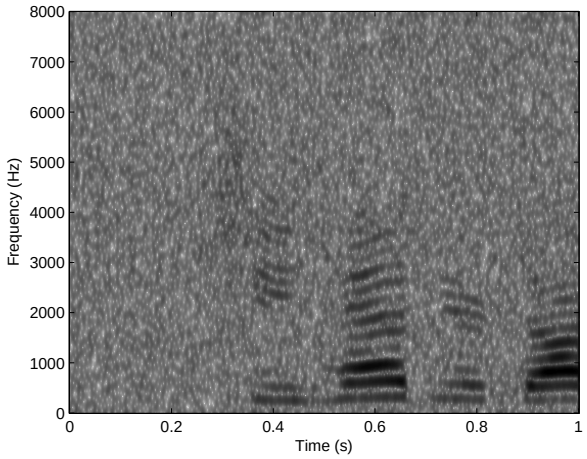


Figure 6.1: Spectrogram of a Speech Utterance Corrupted with White Noise - This figure shows the spectrogram of a speech utterance corrupted with white noise at an SNR of 10 dB

Contrast the two spectrograms with Figure 6.3 which is the spectrogram of a realistic noisy speech utterance recorded in a busy street intersection. It can be observed that the towards the beginning of the utterance, there are two significant clicks spanning across the frequency range. Also the low frequency noises are stronger in the beginning and weaker towards the end. Even though the noise intensity is less in the realistic noisy speech, it is difficult the estimate the noise.This is because most voice activity detectors depend on a noise template to identify the noise frames. Since the noise frames differ from each other considerably, it is difficult to find a template that can result in an accurate estimation of the prevalent noise. Thus, the time varying nature of the realistic noise poses significant challenge in estimation.

Besides time-varying noise, the real-world noisy speech could be degraded by other factors as well such as session variability and Lombard effect. Session variability is caused by the variation in the physical and emotional state of an individual user over different sessions. Since physical and emotional state of the user affects the voice quality, the difference in the voice between two

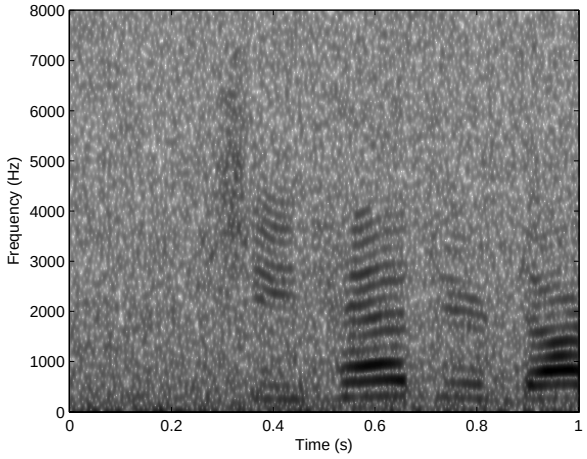


Figure 6.2: Spectrogram of a Speech Utterance Corrupted with Pink Noise - This figure shows the spectrogram of a speech utterance corrupted with pink noise at an SNR of 10 dB

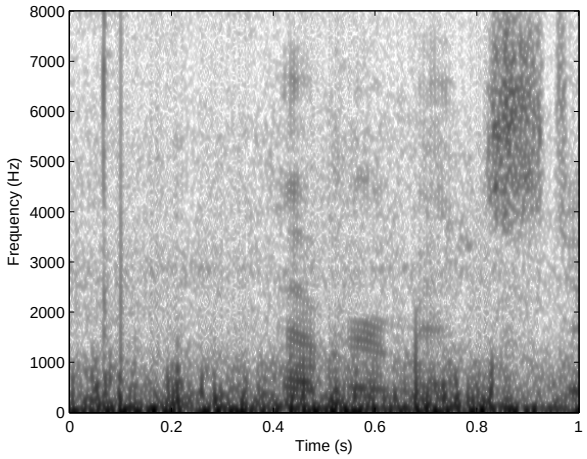


Figure 6.3: Spectrogram of a Realistic Noisy Speech Utterance - This figure shows the spectrogram of a noisy speech utterance recorded in a busy street intersection

sessions, such as training and verification session, could be significant. The session variability is not directly related to the environmental noise and it can appear even in quiet conditions. The Lombard effect is directly related to the prevalent noise. The normal voice of a user can be drowned in the environmental noise. In order to be heard, the speakers in noisy environments tend to raise their voice. This causes a physical change in the vocal tract of the user, which gives rise to the change in the voice.

It should be noted that the session variability and the Lombard effect represent a change in the voice of the user. Even though such conditions do degrade the performance of the speaker verification system, they do not fall under the scope of the research carried out in this work. Besides, the voice variability is unlikely to interfere with the function of the psychoacoustic model compensation technique as the goal of the psychoacoustic model compensation technique is to compensate the clean model parameters for changes in the environmental noise.

The unpredictable and complex noise types, such as the one illustrated in Figure 6.3, on the other hand, can interfere with the function of the compensation technique. Without a reliable estimation of the noise, the transformation of the model parameters is not likely to be reliable, thereby degrading the performance of the compensation technique. It is impossible to have control over the noise types in real-life scenarios. For example, in a street intersection, there can be a short burst of musical noise stemming from the sounding of the horn of a vehicle. Therefore, the difficulty in the estimation of the noise in real-life scenarios must be adequately addressed in order for the psychoacoustic model compensation technique to be useful in real-life deployments.

6.3 Multi-conditioning

The noise estimation for the complex noise types can be addressed, to some extent, by adopting sophisticated voice activity detectors. However, the computational complexity of such voice activity detectors are considerable. Also, voice activity detectors may not provide an accurate estimation of the prevalent noise every time. To circumvent this problem, multi-conditioned mod-

els, also known as multi-SNR models, have been employed in many existing works.

Multi-conditioning refers to the process of training multiple models each with different levels of noise. Consider a clean training set X_0. Multiple copies of the training set can be generated by corrupting it with noise at different SNRs. Consider $L+1$ different copies of X_0, i.e., $X_0, X_1, \ldots, X_L$. Let the models trained on these training data be denoted as $\lambda_0, \lambda_1, \ldots, \lambda_L$. Figure 6.4 illustrates this process. During verification a final score can be computed based on the score of the test utterance with $L+1$ different models.

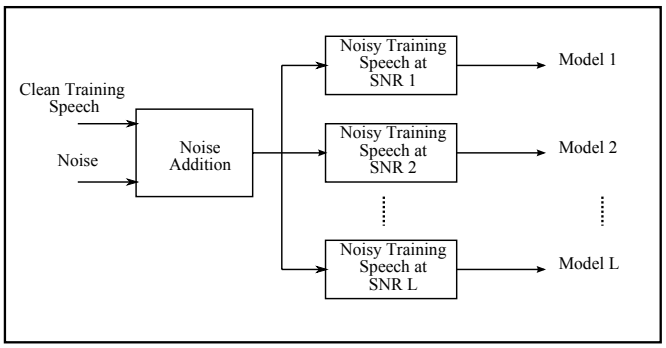


Figure 6.4: Multi-conditioning of Models - This figure shows the process of multi-conditioning as implemented in speaker verification

The idea of multi-conditioning was first introduced in [86], where the noise models were added to the speaker models at SNRs of 0 dB to 18 dB in steps of 3 dB increment. The noise used in the experiments were artificial noise and the noise models were created using the artificial noise instead of estimating the noise from the noisy utterances. In real world scenarios, the noise has to be estimated from noisy utterances.

In order to avoid the problem of noise estimation, white noise can be used to create multi-conditioned models [91][92]. White noise can be added to the training utterances at various SNRs and multiple models can be trained. This method avoids estimating the noise altogether during the verification phase. However, the real prevalent noise is likely to be very different from the white noise and therefore, there will be a mismatch between the model and the test utterances.

The mismatch due to the white noise can be addressed through Posterior Union Model (PUM) [38]. The PUM relies on the missing feature theory to

find an optimal combination of matched feature subbands. Consider a feature vector $\vec{x} = (x_1, x_2, \ldots, x_D)$. Then let us consider the set $X = x_1, x_2, \ldots, x_D$ comprising of all the elements of the vector $\vec{x}$. The subset X_{sub} such that $X_{sub} \subset X$ is considered optimal if $p(\lambda_l|\vec{x}_{sub})$ is maximum, where $\vec{x}_{sub}$ is a vector consisting of elements of the set X_{sub}. The maximum is determined by considering all possible combinations of all the elements of the feature vector $\vec{x}$.

Obviously the PUM is computationally prohibitive, rendering multi-conditioned models with white noise less attractive. The psychoacoustic model compensation technique developed in this work provides an innovative and computationally efficient way of creating multi-conditioned models. We describe it in the next section.

6.4 Psychoacoustic Multi-conditioning

The psychoacoustic model compensation transforms the model parameters to suit a particular prevalent noise. The transformation can essentially be viewed as a simulation of the effect of noise addition to the clean training speech. In other words, the psychoacoustic model compensation is an approximation of adding the prevalent noise to the clean training speech and training a model. Therefore, it is conceivable that a range of models with different SNRs can be created by appropriately scaling the observed noise.

Consider a clean speaker model λ_{α_0} resulting from the clean training speech X. Let us denote the set of noise observation as $N_{\alpha_1} = \{\vec{n}_1, \vec{n}_2, \ldots, \vec{n}_V\}$. By applying psychoacoustic model compensation on λ_{α_0} using N_{α_1}, a compensated model λ_{α_1} can be obtained. In essence, this psychoacoustic compensation process approximates the effect of X being corrupted by N_{α_1}. Then the SNR, R, for the corrupted speech, which is represented by the model λ_{α_1}, can be written as:

$$R = 10 \log_{10} \frac{E(X)}{E(N_{\alpha_1})} \tag{6.1}$$

In the above equation, $E(X)$ and $E(N_{\alpha_1})$ stand for energy of the clean speech and energy of the noise respectively. Let N_{α_1} be multiplied by a scale factor g in the mel-filter-output domain to result in a set of noise observations N_{α_g}.

Let the psychoacoustic model compensation process be applied on λ_{α_0} using N_{α_g} resulting in the compensated model λ_{α_g}. Then the SNR for the corrupted speech, represented by the model λ_{α_g} can be written as:

$$10 \log_{10} \frac{E(X)}{g^2 E(N_{\alpha_1})} = R - 20 \log_{10}(g) \tag{6.2}$$

It can be observed from the above explanation, that a number of different models can be created, by varying the value of g, which can simulate the effect of clean training speech being corrupted at various SNRs. Figure 6.5 illustrates the psychoacoustic multi-conditioning process.

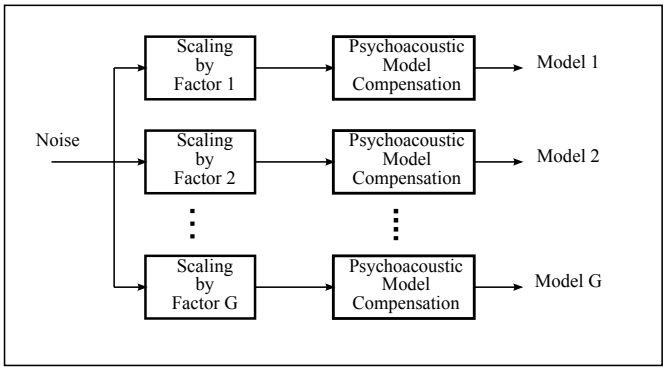


Figure 6.5: Psychoacoustic Multi-conditioning of Models - This figure shows the process of psychoacoustic multi-conditioning

It should be noted that the psychoacoustic multi-conditioning does not require creation of models at exact SNR levels, unlike other multi-conditioning schemes. Therefore, the explicit determination of the base SNR R is not necessary. Only the desired difference in SNR levels are required to determine the scale factor g. For example, if the clean speech model is compensated using the observed noise scaled by a factor of 0.5, the resulting compensated model represents a 6 dB increase in SNR as compared to the model compensated by the unscaled observed noise.

Let the set of multiconditioned models for a speaker S be $\{\lambda_{\alpha_0}, \lambda_{\alpha_1}, \ldots, \lambda_{\alpha_G}\}$. Considering equal prior for all models, the likelihood score of a test utterance

Y for the speaker S is given by [38]:

$$p(Y|S) = \frac{1}{G+1} \sum_{g=0}^{G} p\left(Y|\lambda_{\alpha_g}\right) \tag{6.3}$$

In the above equation, $G+1$ stands for the total number of elements in the set of multiconditioned models. The total number can be varied depending on the complexity of the noise encountered. Notice that if $G = 0$ and $\alpha_0 = 1$, then the above multiconditioning is just the psychoacoustic model compensation.

The multi-conditioning scheme described here addresses one of the significant drawbacks of model compensation techniques. The model compensation techniques rely heavily on the voice activity detectors to provide an accurate estimation of the prevalent noise, which is essential for good performance of the techniques. By creating multi-conditioned models, the above described scheme reduces the reliance of the psychoacoustic model compensation technique on the accuracy of the voice activity detector.

Using the scaled noise observations and psychoacoustic compensation for multiconditioning has certain advantages over the multiconditioning approaches using white noise. It reduces the storage requirements of the speaker recognition system. The models with different SNRs can be generated on the fly during verification. Therefore, only the clean speech model needs to be stored. Using training speech corrupted with white noise at different SNRs for multiconditioning, on the other hand, would require all the models at different SNRs to be stored. Secondly, since the multiconditioning described above is achieved with the observed noise, it should induce lesser number of heavily mismatched subbands, as the observed noise should be closer to the actual noise than the white noise. This should reduce the reliance on the computationally complex posterior union model for acceptable performance.

6.5 Experiments

Experiments were conducted on the TIMIT database and the MIT Mobile Devices Speaker Verification Corpus (MITMDSVC) to determine the effec-

tiveness of the psychoacoustic multi-conditioning. This section presents the experimental set-up, experimental results and analysis of the results.

6.5.1 Experimental Set-up

Two databases have been used for experimentation in this chapter. The first is TIMIT, which has been described before, and the second is MITMDSVC [6]. The experimental set-up for the experiments on TIMIT database are same as described in in Chapter 3, Section 3.4.1. The set-up for MITMDSVC is described below.

MITMDSVC is a realistic noisy speech database created by using hand-held devices for speech acquisition. Emulating the scenarios encountered by real-world speaker verification systems, the collected speech data contains two unique sets: as set of enrolled users and a set of dedicated impostors. For the enrolled set, the speech data was collected in two separate sessions, one of which can be used for training and the other can be used for verification. For each session, the data was collected in three different locations: a quiet office, a mildly noisy lobby and a busy street intersection. Two different microphones were used. The MITMDSVC does contain Lombard effect, microphone clicks and session variability. The user recited a list of names and icecream flavours. The enrolled set has 48 speakers, out of which 22 are female and 26 are male. The impostor set has 40 speakers, out of which 17 are female and 23 are male.

For the MITMDSVC, we tested our algorithms on three scenarios. First, the training was performed on the speech collected in quiet office environment using headset microphone and verification was performed using speech collected in quiet office environment using headset microphone (index: OH-OH). Second, the training was done on the speech collected in quiet office environment using internal microphone and verification was done by using speech collected in the busy street intersection using internal microphone (index: OI-SI). Third, the training was done using speech collected in quiet office environment using headset microphone and the verification was done using speech collected in the busy street intersection using headset microphone (index: OH-SH). The first scenario represents the matched condition while the

second and third scenarios represent the mismatched conditions.

The part of the database containing ice cream flavour phrases were used for experiments. The impostors saying the same phrase as the enrolled speakers were grouped to form the set of impostors. This is the same set-up as was used in [38]. The speaker models consisted of 32 Gaussian components. The prevalent noise for MITMDSVC was estimated using the *ad-hoc* procedure based on “Murphy’s Algorithm" described in [14]. The duration of each utterance in MITMDSVC is around 1 second. Therefore, the estimated noise, often, consists of only few frames. With this sparse noise information, the variance of the noise cannot be estimated accurately and therefore only the model means were compensated for the MITMDSVC.

The multiconditioning was achieved with scale factors of 0.5, 0.6, 0.8, 1, 1.25, 1.6 and 2. Notice that the scale factor of 1 corresponds to the base SNR R as explained in Section 6.4. The scale factors translate to SNRs of $R-6$ dB to $R+6$ dB with a 2 dB increment every step.

6.5.2 Experimental Results

This section presents the results from the experiments. The “baseline” results indicate the results from experiments without any noise compensation involved. The “psy-comp” results are results from experiments using the psychoacoustic model compensation and the “psy-multi” results are from experiments using the psychoacoustic multi-conditioning.

Figure 6.6 presents the ROC curves for experiments on the TIMIT database involving white noise, while Figure 6.7 presents the same for pink noise. The use of multiconditioning provides similar performance to the psychoacoustic compensation using only the observed noise. This is because the simulated noise added to the speech utterances could be estimated easily. Without heavy mismatch between the estimated noise and the actual noise, the advantages of multiconditioning is not likely to be apparent. Table 6.1 summarizes the equal error rates associated with the experiments.

Figure 6.8 presents the DET curves from the experiments in OH-OH scenario. It can be observed that there is a moderate improvement in EER with psychoacoustic compensation as well as multiconditioning. This is because,

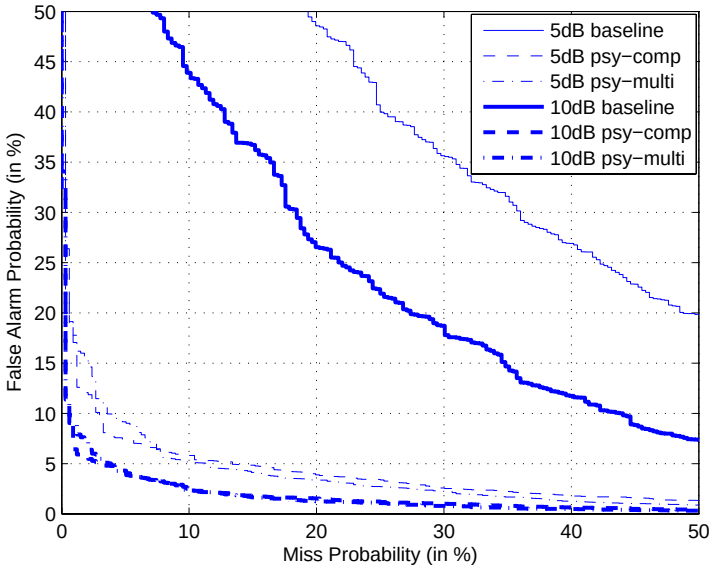


Figure 6.6: ROC Curves for TIMIT Database Corrupted by White Noise - This figure compares ROC curves for various methods on TIMIT database using white noise

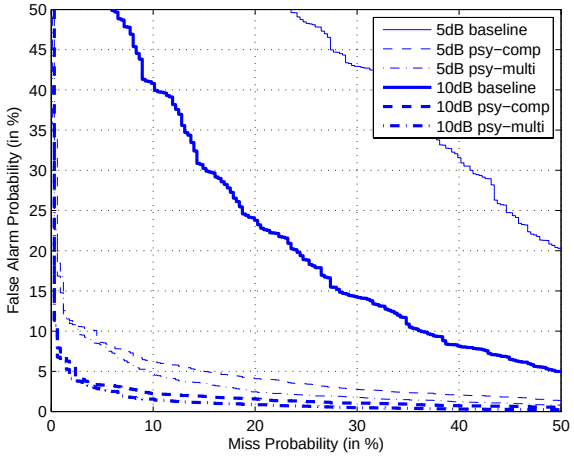


Figure 6.7: ROC Curves for TIMIT Database Corrupted by Pink Noise - This figure compares ROC curves for various methods on TIMIT database using pink noise

Table 6.1: EER (%) from Experiments with TIMIT Database Corrupted with White and Pink Noise

	White 5dB	White 10db	Pink 5dB	Pink 10dB
Baseline	32.9	23.6	36.6	22.1
Max-function	8.4	6.0	9.6	5.6
Psy-comp	6.6	4.4	7.5	3.3
Psy-multi	7.1	4.3	6.6	3.6

as mentioned in [38], the office data is not completely noise-free. However, the noise conditions for training and testing are not significantly different to produce a pronounced effect. Figure 6.9 presents the DET curves from the experiments in OI-SI scenario. In this case, psychoacoustic compensation scheme performs significantly better than the baseline. The performance is further improved with multi-conditioning. The estimated noise, in this case, is not a very good representation of the actual noise. Therefore, advantage of multiconditioning is very noticeable. The DET curves from OH-SH scenario, shown in Figure 6.10, shows similar noticeable improvements for the proposed algorithms.

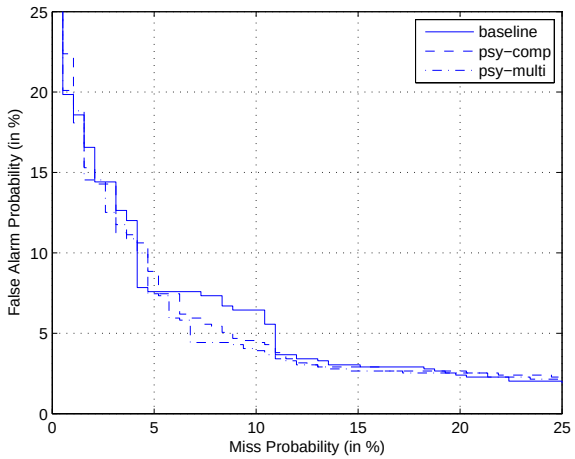


Figure 6.8: ROC Curves for OH-OH Scenario in MITMDSVC Database - This figure compares ROC curves for various methods in the OH-OH scenario of the MITMDSVC database

Table 6.2 summarizes the EERs from experiments with MITMDSVC database. It can be observed that under the realistic conditions, the psychoacoustic model compensation as well as the psychoacoustic significantly reduce the

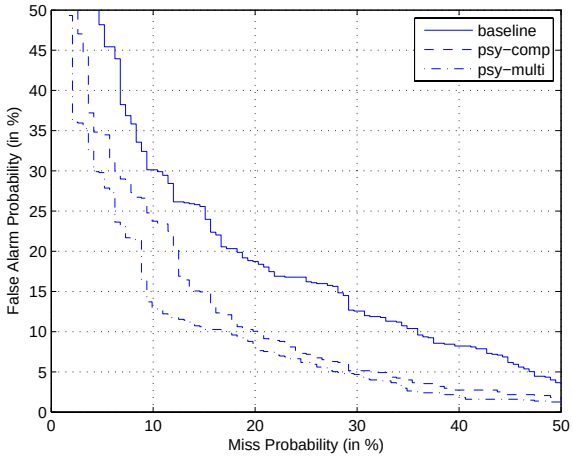


Figure 6.9: ROC Curves for OI-SI Scenario in MITMDSVC Database - This figure compares ROC curves for various methods in the OI-SI scenario of the MITMDSVC database

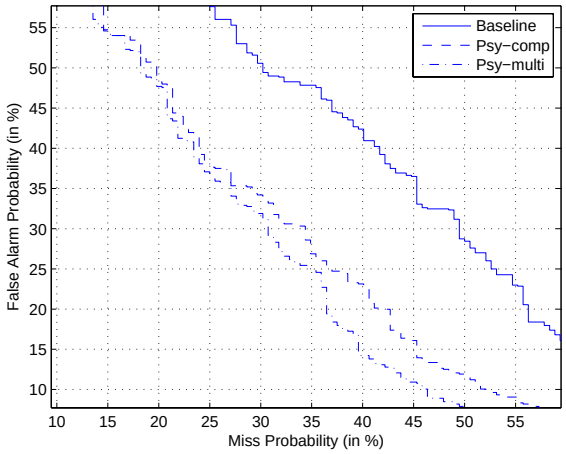


Figure 6.10: ROC Curves for OH-SH Scenario in MITMDSVC Database - This figure compares ROC curves for various methods in the OH-SH scenario of the MITMDSVC database

EER. The psychoacoustic model compensation scheme reduced the EER by 15%, 22% and 22% under OH-OH, OI-SI and OH-SH conditions respectively. The psychoacoustic multi-conditioning reduced the EER by 19%, 38% and 25% under OH-OH, OI-SI and OH-SH conditions respectivly.

Table 6.2: EER (%)from Experiments with MITMDSVC Database

	OH-OH	OI-SI	OH-SH
Baseline	7.3	19.2	40.9
Max-function	6.2	16.2	32.8
Psy-comp	6.2	15.0	31.8
Psy-multi	5.9	11.8	30.7

It can be observed that all three systems implemented perform worse in the OH-SH scenario than in OI-SI scenario. This is because of inadequate training speech. As noted in [113], the performance of GMM-UBM depends strongly on the amount of speech available for training. In OI-SI case, the phrases uttered for training and verification are the same. Therefore, the performance of GMM-UBM system is better even with very short training data. In OH-SH scenario, however, the phrases uttered for training and verification are different. Therefore, the GMMs do not adequately model the acoustic classes of test utterances and this leads to the lower performances.

6.5.3 Analysis

We have presented results from TIMIT database corrupted with artificial noise and MITMDSVC database containing real-world noise. The experimental results presented reveal that the psychoacoustic model compensation technique provides significant performance gain in both databases. Therefore, it can be concluded that the psychoacoustic model compensation technique is an effective way of dealing with environmental noise.

The psychoacoustic multi-conditioning provides remarkable improvement in the performance under realistic conditions. Table 6.3 compares the EER reported in this chapter with those reported in [38] and [93]. In the table, “Multi-white WB” refers to the multi-conditioning with wide-band white noise and “Multi-white NB” refers to the multi-conditioning with narrow-band white noise as in [38]. “Psy-multi” refers to the psychoacoustic multi-conditioning

introduced in this chapter. The OH-SH scenario has not been experimented in [38].

Table 6.3: Comparison of EERs (%) on MITMDSVC Reported in This Work with EERs Reported in [38] and [93]

Scenario	Multi-white WB noise in [38]	Multi-white NB noise in [38]	Weiner Filter in [93]	Psy-multi
OH-OH	12.65	7.29	5.66	5.9
OI-SI	23.96	15.63	19.39	11.8

It can be observed from Table 6.3 that the psychoacoustic multi-conditioning provides better performance than multi-conditioning reported in [38]. This is especially remarkable as the psychoacoustic multi-conditioning provides performance that is better than or similar to the one reported with the computationally complex posterior union model in [38]. This demonstrates the computational efficiency of the proposed algorithm.

The computational efficiency is the direct result of multi-conditioning using the observed noise. White noise, being very dissimilar to the actual noise, induces heavily mismatched subbands, which degrades the performance of the multi-conditioning. This the reason the probabilistic union model is indispensable for improved performance, if white noise is used for multi-conditioning. The observed noise, on the other hand, is much closer to the actual noise, even though it might not be the complete representation of the actual noise. This is less likely to induce heavy mismatch between the compensated model and the test utterance.

The state-of-the-art performance of the psychoacoustic multi-conditioning on the realistic and challenging MITMDSVC database shows that the proposed technique can be deployed in real-life scenarios with challenging noise conditions. However, before deciding to implement the multi-conditioning, the intended environment of the speaker verification system must be taken into consideration. If the intended environment mostly contains stationary noise, then the psychoacoustic multi-conditioning might not be need, as shown by the experiments conducted on the TIMIT database. The psychoacoustic model compensation might be sufficient for such an environment,

thereby saving the hardware costs and reducing the latency without experiencing performance penalty. If however, a random noise environment is anticipated, then the psychoacoustic multi-conditioning must be implemented.

6.6 Summary

This chapter tackled the issue of the complex and unpredictable prevalent noise. It highlighted the problems related to realistic scenarios and introduced the concept of multi-conditioning methods. The existing multi-condition methods use artificial noise during the training phase to create multi-SNR models. Such approaches, though moderately beneficial, result in heavily mismatched subbands due to the considerable difference between the artificial noise and the actual noise, which leads to lower performance. The psychoacoustic multi-conditioning, developed in this chapter, avoids the problem of heavily mismatched subbands by using the observed noise to create multi-SNR models. Experiments conducted on TIMIT and MITMDSVC databases demonstrate the effectiveness of the proposed method.

7

Conclusions

7.1 Summary of Results and Thesis Contributions

This thesis has proposed a number of novel techniques to address the problem of text-independent speaker verification in environmental noise conditions. A greater emphasis was given to the roadblocks and challenges in deploying speaker verification in real world conditions. The challenges faced were overcome through proposing and experimentally validating novel techniques after detailed evaluations of existing solutions.

A comparative study of the PSS and the PMC techniques was undertaken to understand the limitations of the prominent existing approaches. The PSS was found to suffer from performance issues due to the mismatch condition induced by the random tonal noise left behind by the spectral subtraction operation. This issue was alleviated, to some extent, by adding noise to the training utterance. To achieve this efficiently, a new training scheme was developed for the PSS, which maximizes the expected likelihood of the training data. Even though the noise added training speech improved the performance of the PSS scheme by 20%, the best performance was obtained when the noise added to the training data was similar to the noise added to the test utterance. This cannot be realized in real-life as information about the test noise is rarely available during the training phase. The PMC, on the other hand, was found to be a robust technique achieving EERs that are 40% lower than what can be

achieved through the PSS. However, subsequent analysis identified two major limitations for the PMC technique, namely, its inaccurate additive noise corruption function and its reliance on accurate noise estimation for better performance.

To address the issue of inaccurate noise corruption function, the max function based model compensation introduced the max function as an alternative to the additive noise corruption function. This study resulted in two important contributions. First, it led to the development of a new generalized compensation scheme suitable for non-linear noise corruption functions. It relies on the amount of change induced by the individual noise vectors and then estimates the distribution of the transformed parameters. The elegant estimation of the compensated model parameters makes it highly versatile, in the sense that it is not specific to the max function and can be employed for any valid linear and non-linear noise corruption function. Second, it demonstrated that neither noise corruption function is suited for all types of noise. For example, for utterances corrupted with white noise, the max function performed 18% better than the additive function and for utterances corrupted with pink noise, the additive function performed 8% better than the max function.

A novel psychoacoustic model compensation technique was introduced to perform reliably in diverse noise conditions. A psychoacoustic noise corruption function was derived through innovative application of the concept of masking thresholds. By emulating the response of the human ear, the proposed noise corruption function has been shown to better characterize the effect of noise on the clean model parameters. The generalized compensation scheme was employed to estimate the transformed model parameters. Experimental evaluations show the proposed psychoacoustic model compensation technique to be an effective noise robustness technique achieving high recognition accuracy in diverse conditions. On an average, it reduced the EER by over 80%, which is significantly better than the performance obtained by the PMC and the max function based compensation. Even more remarkable is the fact that the proposed technique maintains the 80% reduction in EER in both pink and white noise scenarios.

A new multi-conditioning approach has been proposed to reduce the reliance of the psychoacoustic model compensation on accurate noise estimation. This has made it possible to overcome the otherwise significant problem of accurate noise estimation in real-world scenarios. Poor estimation of the prevalent noise typically leads to poor performance of the model compensation techniques. The proposed framework extends the psychoacoustic model compensation for complex noise scenarios of real-world conditions and thereby enhances its applicability to practical implementations. A voice activity detector is used to obtain an estimate of the observed noise so as to create the multi-conditioned models. To reduce the dependence on the accuracy of the voice activity detector, scaled values of the observed noise is used to compensate the model parameters and multiple instances of compensated parameters are generated by varying the scale factor. The proposed method has been shown to exhibit significant advantages over the conventional multi-conditioning with white noise. Since the observed noise is closer in characteristics to the actual noise than the white noise, it appears to induce lesser number of heavily mismatched sub-bands. The experiments conducted confirm that the reduction in mismatch leads to higher performance. The results conclusively demonstrate that the proposed techniques provide superior performance, which is, on average, 22% better than what can be achieved through multi-conditioning with white noise. More importantly, the reported results have been achieved without recourse to the computationally complex probabilistic union model, which is essential for multi-conditioning with white noise for obtaining better performance.

In conclusion, this thesis introduces the novel psychoacoustic model compensation for robust speaker verification in environmental noise conditions. It is noteworthy that the proposed technique does not pose any of the implementation difficulties that are posed by the spectral subtraction based techniques. In addition, it provides superior performance when compared with the PSS and the PMC across different types of noise. Most importantly, the proposed technique is computationally efficient and hence well suited for deployment in mass volume products.

7.2 Future Research Direction

Although this thesis makes significant contributions in the field of speaker verification in environmental noise, it is by no means an exhaustive study of the general area of mismatch conditions in speaker verification. This section briefly discusses some future research directions which can extend or augment the work in this thesis.

- **Psychoacoustic Model Compensation with Limited Noise Data:** With short utterances and complex noise scenario, a voice activity detector may not result in sufficient noise data. This can lead to inaccurate estimation of the model parameters, especially the model variances. This problem was addressed in this thesis by keeping the model variances unchanged when insufficient noise data condition was encountered. Other estimation techniques such as shrinkage estimation [114] may improve the estimation accuracy and should be investigated.

- **Psychoacoustic Model Compensation for Dynamic Features:** Many researchers have suggested dynamic features such as delta-cepstrums and delta-delta cepstrums [2]. Such features are appended to the MFCC. In other words, each feature vector has two distinct halves, one with MFCCs and another with the dynamic features. With such feature vectors, the psychoacoustic model compensation can be performed on the MFCC part of the model parameters.

- **Integration of Speech Enhancement and Psychoacoustic Model Compensation :** In this work, speech enhancement and psychoacoustic model compensation have been treated as competing methods. However, there are indications that these techniques can complement each other. When speech enhancement technique is employed in the feature vector domain, there remains a residual noise. The residual noise cause mismatch and is the main drawback of the speech enhancement techniques. The psychoacoustic model compensation technique can complement the speech enhancement techniques by compensating for the residual noise

in the model domain. The challenge in integrating the speech enhancement techniques with psychoacoustic model compensation lies in the estimation of the distribution of the residual noise.

- **Compensation for Environmental Mismatch and Channel Mismatch:** This thesis considered the problem of environmental mismatch exclusively. Environmental mismatch can also occur along with channel mismatch. To address this problem, psychoacoustic model compensation should be studied along with techniques to reduce the channel mismatch such as the CMS. One way to achieve this will be to subtract the cepstral mean from the feature vectors and to alter the noise observations to accommodate the subtraction of the cepstral mean. The altered noise observations can be used for the psychoacoustic model compensation.

References

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

[2] F. Bimbot *et al.*, "A tutorial on text-independent speaker verification," *Eurasip Journal on Applied Signal Processing*, vol. 4, pp. 430–451, 2004.

[3] D. A. Reynolds, "Large population speaker identification using clean and telephone speech," *IEEE Signal Processing Letters*, vol. 2, no. 3, pp. 46–48, 1995.

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

[5] B. Ma, H. M. Meng, and M.-W. Mak, "Effect of device mismatch, language mismatch and environmental mismatch on speaker verification," in *Proc. International Conference on Acoustics, Speech, and Signal Processing (ICASSP'07)*, vol. IV, 2007, pp. 301–304.

[6] R. H. Woo, A. Park, and T. J. Hazen, "The MIT mobile device speaker verification corpus: Data collection and preliminary experiments," in *Proc. 2006 IEEE Odyssey: Speaker and Launguage Recognition Workshop*, 2006, pp. 1–6.

[7] J. P. Campbell, Jr., "Speaker recognition: A tutorial," *Proceedings of the IEEE*, vol. 85, no. 9, pp. 1437–1462, 1997.

[8] G. Fant, *Acoustic Theory of Speech Production*. The Hague, The Natherlands: Mouton, 1970.

[9] J. R. Deller, J. H. L. Hansen, and J. G. Proakis, *Discrete-time Processing of Speech Signals*. New York, USA: Wiley-Interscience, 2000.

[10] D. Petrovska-Delacretaz, J. Cernocky, J. Hennebert, and G. Chollet, "Segmental approaches for automatic speaker verification," *Digital Signal Processing*, vol. 10, no. 1-3, pp. 198–212, 2000.

[11] A. V. Oppenheim and R. W. Schafer, "Homomorphic analysis of speech," *IEEE Transactions on Audio and Electroacoustics*, vol. 16, no. 2, pp. 221–226, 1968.

[12] J. Tierney, "A study of LPC analysis of speech in additive noise," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 28, pp. 389–397, 1980.

[13] D. O'Shaughnessy, *Speech Communication: Human and Machine*. Reading: Addition-Wesley, 1987.

[14] D. A. Reynolds, "A Gaussian mixture modeling approach to text-independent speaker identification," Ph.D. dissertation, Georgia Institute of Technology, Atlanta, Ga, USA, 1992.

[15] D. A. Reynolds and R. C. Rose, "Robust text-independent speaker identification using Gaussian mixture speaker models," *IEEE Transactions on Speech and Audio Processing*, vol. 2, pp. 245–257, 1994.

[16] P. Murthy, "VLSI based system for voice authentication," Master's thesis, Nanyang Technological University, Singapore, 2001.

[17] Y. Bennani, F. Fogelman, and P. Gallinary, "A connectionist approach for speaker identification," in *Proc. International Conference on Acoustics, Speech, and Signal Processing (ICASSP'90)*, 1990, pp. 256–265.

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

[19] J. M. Naik and D. Lubenskt, "A hybrid HMM-MLP speaker verification algorithm for telephone speech," in *Proc. International Conference on Acoustics, Speech, and Signal Processing (ICASSP'94)*, vol. 1, 1994, pp. 153–156.

[20] M. Schmidt and H. Gish, "Speaker identification via support vector classifiers," in *Proc. International Conference on Acoustics, Speech, and Signal Processing (ICASSP'96)*, vol. 1, 1996, pp. 105–108.

[21] Y. Gu and T. Thomas, "A text-independent speaker verification system using support vector machines classifier," in *Proc. European Conference on Speech Communication and Technology (EUROSPEECH'01)*, 2001, pp. 1765–1769.

[22] J. Kharroubi, D. Petrovska-Delacretaz, and G. Chollet, "Combining GMMs with support vector machines for text-independent speaker verification," in *Proc. European Conference on Speech Communication and Technology (EUROSPEECH'01)*, 2001, pp. 1757–1760.

[23] X. Dong, W. Zhaohui, and Y. Yingchun, "Exploiting support vector machines in hidden Markov models for speaker verification," in *Proc. International Conference on Spoken Language Processing (ICSLP)*, 2002, pp. 1329–1332.

[24] S. Fine, J. Navratil, and R. A. Gopinath, "Enhancing GMM scores with using SVM "hints"," in *Proc. European Conference on Speech Communication and Technology (EUROSPEECH'01)*, 2001.

[25] B. S. Everitt and D. J. Hand, *Finite Mixture Distributions.* New York: Chapman and Hall, 1981.

[26] S. G. Pillay, A. Ariyaeeinia, M. Pawlewski, and P. Sivakumaran, "Speaker verification under mismatched data conditions," *IET Signal Processing*, vol. 3, no. 4, pp. 236–246, 2009.

[27] A. Panda, "High performance voice authentication system," Master's thesis, Nanyang Technological University, Singapore, 2003.

[28] D. A. Reynolds, T. F. Quatieri, and R. B. Dunn, "Speaker verification using adapted Gaussian mixture models," *Digital Signal Processing*, vol. 10, no. 1-3, pp. 19–41, 2000.

[29] A. Higgins, L. Bahler, and J. Porter, "Speaker verification using randomized phrase prompting," *Digital Signal Processing*, vol. 1, no. 2, pp. 89–106, 1991.

[30] T. Matsui and S. Furui, "Feature warping for robust speaker verification," in *Proc. 2001:A Speaker Odyssey*, 2001, pp. 213–218.

[31] M. Carey, E. Parris, and J. Bridle, "A speaker verification system using alpha-nets," in *Proc. International Conference on Acoustics, Speech, and Signal Processing (ICASSP'91)*, vol. 1, 1991, pp. 397–400.

[32] L. P. Heck and M. Weintraub, "Handset-dependent background models for robust text-independent speaker recognition," in *Proc. International Conference on Acoustics, Speech, and Signal Processing (ICASSP'97)*, vol. 2, 1997, pp. 1071–1074.

[33] G. Gravier and G. Chollet, "Comparison of normalization techniques for speaker recognition," in *Proc. Workshop on Speaker Recognition and its Commercial and Forensic Applications*, 1998, pp. 97–100.

[34] D. A. Reynolds, "Comparison of background normalization methods for speaker verification based on a posteriori probability," in *Proc. European Conference on Speech Communication and Technology (EUROSPEECH'97)*, vol. 2, 1997, pp. 963–966.

[35] J. P. Campbell, Jr., "Testing with the YOHO CD-ROM voice verification corpus," in *Proc. International Conference on Acoustics, Speech, and Signal Processing (ICASSP'95)*, vol. I, 1995, pp. 341–344.

[36] D. Pisoni, R. Bernacki, H. Nusbaum, and M. Yuchtman, "Some acoustic-phonetic correlates of speech produced in noise," in *Proc. International Conference on Acoustics, Speech, and Signal Processing (ICASSP'85)*, 1985, pp. 1581–1584.

[37] R. Sarikaya and J. H. L. Hansen, "Analysis of the root-cepstrum for acoustic modeling and fast decoding in speech recognition," in *Proc. European Conference on Speech Communication and Technology (EUROSPEECH'01)*, 2001, pp. 687–690.

[38] J. Ming, T. J. Hazen, J. R. Glass, and D. A. Reynolds, "Robust speaker recognition in noisy conditions," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 15, no. 5, pp. 1711–1723, 2007.

[39] Y. Gong, "Speech recognition in noisy environments: A survey," *Speech Communication*, vol. 16, no. 3, pp. 261–291, 1995.

[40] H. Hermansky and N. Morgan, "RASTA processing of speech," *IEEE Transactions on Speech and Audio Processing*, vol. 2, no. 4, pp. 578–589, 1994.

[41] L. Burget, P. Matejka, P. Schwarz, O. Glembek, and J. Cernocky, "Analysis of feature extraction and channel compensation in a GMM speaker recognition system," *IEEE Transactions on Audio, Speech and Launguage Processing*, vol. 15, no. 7, pp. 1979–1986, 2007.

[42] J. Pelecanos and S. Sridharan, "Feature warping for robust speaker verification," in *Proc. 2001:A Speaker Odyssey*, 2001, pp. 213–218.

[43] B. Xiang, U. V. Chaudhari, J. Navratil, G. N. Ramaswamy, and R. A. Gopinath, "Short-time Gaussianization for robust speaker verification," in *Proc. International Conference on Acoustics, Speech, and Signal Processing (ICASSP'02)*, 2002, pp. 681–684.

[44] C. Cao, X. Xiao, M. Li, J. Liu, and Y. Yan, "Harmonic structure features for robust speaker recognition against channel effect," in *Proc. International Symposium on Information Science and Engineering*, 2009, pp. 451–454.

[45] Y. Dong, J. Zhao, L. Lu, J. Lui, X. Zhao, and H. Wang, "Eigenchannel compensation and symmetric score for robust text-independent speaker verification," in *Proc. International Symposium on Chinese Spoken Language Processing (ISCSLP'08)*, 2008, pp. 1–4.

[46] D. Sturim, W. Campbell, D. Reynolds, R. Dunn, and T. Quatieri, "Robust speaker recognition with cross-channel data: MIT-LL results on the 2006 NIST SRE auxiliary microphone task," in *Proc. International Conference on Acoustics, Speech, and Signal Processing (ICASSP'07)*, vol. 4, 2007, pp. 49–52.

[47] L. F. Lamel and J. L. Gauvain, "Speaker verification over the telephone," *Speech Communication*, vol. 31, pp. 141–154, 2000.

[48] C. Barras and J. L. Gauvain, "Feature and score normalization for speaker verification of cellular data," in *Proc. International Conference on Acoustics, Speech, and Signal Processing (ICASSP'03)*, 2003, pp. 49–52.

[49] R. Auckenthaler, M. Carey, and H. Lloyd-Thomas, "Score normalization for text-independent speaker verification systems," *Digital Signal Processing*, vol. 10, pp. 42–54, 2000.

[50] D. C. Bateman, D. K. Bye, and M. J. Hunt, "Spectral contrast normalization and other techniques for speech recognition in noise," in *Proc. International Conference on Acoustics, Speech, and Signal Processing (ICASSP'92)*, vol. I, 1992, pp. 241–244.

[51] S. V. Vaseghi and B. P. Milner, "Noise adaptive hidden Markov models based on Wiener filters," in *Proc. European Conference on Speech Communication and Technology (EUROSPEECH'93)*, vol. II, 1993, pp. 1023–1026.

[52] J. H. L. Hansen and O. N. Bria, "Lombard effect compensation for robust automatic speech recognition in noise," in *Proc. International Conference on Acoustics, Speech, and Signal Processing (ICASSP'90)*, 1990, pp. 1125–1128.

[53] J.-C. Junqua, "The Lombard reflex and its role on human listeners and automatic speech recognizers," *Journal of Acoustical Society of America*, vol. 93, no. 1, pp. 510–524, 1993.

[54] R. Saeidi, J. Pohjalainen, T. Kinnunen, and P. Alku, "Temporally weighted linear prediction features for tackling additive noise in speaker verification," *IEEE Signal Processing Letters*, vol. 17, no. 6, pp. 599–602, 2010.

[55] M. Nosratighods, E. Ambikairajah, J. Epps, and M. Carey, "Score weighting in speaker verification systems," in *Proc. International Conference on Information, Communications and Signal Processing*, 2007, pp. 1–4.

[56] M. Padilla and T. Quatieri, "A comparison of soft and hard spectral subtraction for speaker verification," in *Proc. International Conference on Spoken Language Processing (INTERSPEECH'04)*, 2004, pp. 1773–1776.

[57] Y. Xie, M. Liu, Z. Yao, and B. Dai, "Improved two-stage wiener filter for robust speaker identification," in *Proc. International Conference on Pattern Recognition*, 2006, pp. 310–313.

[58] L. P. Wong and M. Russell, "Text-dependent speaker verification under noisy conditions using parallel model combination," in *Proc. International Conference on Acoustics, Speech, and Signal Processing (ICASSP'01)*, vol. 1, 2001, pp. 457–460.

[59] S. Sagayama, Y. Yamaguchi, S. Takahashi, and J. Takahashi, "Jacobian approach to fast acoustic model adaptation," in *Proc. International Conference on Acoustics, Speech, and Signal Processing (ICASSP'97)*, vol. 2, 1997, pp. 835–838.

[60] Z. Tufekci and S. Gurbuz, "Noise robust speaker verification using mel-frequency discrete wavelet coefficients and parallel model compensation," in *Proc. International Conference on Acoustics, Speech, and Signal Processing (ICASSP'05)*, vol. I, 2005, pp. 657–660.

[61] S. Mallat, *A Wavelet Tour of Signal Processing*. San Diego: Academic Press, 1998.

[62] J. Gowdy and Z. Tufekci, "Mel-scaled discrete wavelet coefficients for speech recognition," in *Proc. International Conference on Acoustics, Speech, and Signal Processing (ICASSP'00)*, vol. 3, 2000, pp. 1351–1354.

[63] C. Ma, Y. Kamp, and L. F. Willems, "Robust signal selection for linear prediction analysis of voiced speech," *Speech Communication*, vol. 12, no. 2, pp. 69–81, 1993.

[64] C. Magi, J. Pohjalainen, T. Backstrom, and P. Alku, "Stabilised weighted linear prediction," *Speech Communication*, vol. 51, no. 5, pp. 401–411, 2009.

[65] A. Roy, M. Magimai.-Doss, and S. Marcel, "Boosted binary features for noise-robust speaker verification," in *Proc. International Conference on Acoustics, Speech, and Signal Processing (ICASSP'10)*, 2010, pp. 4442–4445.

[66] J. Friedman, T. Hastie, and R. Tibshirani, "Additive logistic regression: A statistical view of boosting," *Annals of Statistics*, vol. 28, no. 2, pp. 337–407, 2000.

[67] G. Wei-Guo, Y. Li-Ping, and C. Di, "Pitch synchronous based feature extraction for noise-robust speaker verification," in *Proc. Congress on Image and Signal Processing*, vol. 5, 2008, pp. 295–298.

[68] S. F. Boll, "Suppression of acoustic noise in speech using spectral subtraction," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 27, pp. 113–120, 1979.

[69] A. Drygajlo and M. El-Maliki, "Speaker verification in noisy environments with combined spectral subtraction and missing feature theory," in *Proc. International Conference on Acoustics, Speech, and Signal Processing (ICASSP'98)*, vol. 1, 1998, pp. 121–124.

[70] C. A. Medina, J. A. Apolinario Jr., A. Alcaim, and R. G. Alves, "Robust speaker verification in colored noise environment," in *Proc. Asilomar Conference on Signals, Systems and Computers*, vol. 2, 2003, pp. 1890–1893.

[71] Y. Ephraim and D. Malah, "Speech enhancement using a minimum mean-square error short-time spectral amplitude estimator," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 32, no. 6, pp. 1109–1121, 1984.

[72] N. Virag, "Single channel speech enhancement based on masking properties of the human auditory system," *IEEE Transactions on Speech and Audio Processing*, vol. 7, no. 2, pp. 126–137, 1992.

[73] A. Moreno-Daniel, J. A. Nolazco-Flores, T. Wada, and B.-H. Juang, "Acoustic model emhancement: An adaptation technique for speaker verification under noisy environments," in *Proc. International Conference on Acoustics, Speech, and Signal Processing (ICASSP'07)*, vol. 4, 2007, pp. 289–292.

[74] T. Ganchev, I. Potamitis, N. Fakotakis, and G. Kokkinakis, "Text-independent speaker verification for real fast-varying noisy environments," *International Journal of Speech Technology*, vol. 7, pp. 281–292, 2004.

[75] N. B. Yoma and M. Villar, "Speaker verification in noise using a stochastic version of the weighted Viterbi algorithm," *IEEE Transactions on Speech and Audio Processing*, vol. 10, no. 3, pp. 158–166, 2002.

[76] L. R. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition," *Proceedings of the IEEE*, vol. 77, no. 2, pp. 257–287, 1989.

[77] N. B. Yoma, F. McInnes, and M. Jack, "Weighted Viterbi algorithm and state duration modeling for speech recognition in noise," in *Proc. International Conference on Acoustics, Speech, and Signal Processing (ICASSP'98)*, 1998, pp. 709–712.

[78] ——, "Improving performance of spectral subtraction in speech recognition using a model for additive noise," *IEEE Transactions on Speech and Audio Processing*, vol. 6, pp. 579–582, 1998.

[79] J. Ortega-garcia and J. Gonzalez-rodriguez, "Overview of speech enhancement techniques for automatic speaker recognition," in *Proc. International Conference on Spoken Language Processing (ICSLP)*, 1996, pp. 929–932.

[80] V. Beattie and S. J. Young, "Hidden Markov model state-based cepstral noise compensation," in *Proc. International Conference on Spoken Language Processing (ICSLP)*, 1992, pp. 519–522.

[81] J. Bai, R. Zheng, B. Xu, and S. Zhang, "Robust speaker recognition integrating pitch and Wiener filter," in *Proc. International Symposium on Chinese Spoken Language Processing (ISCSLP)*, 2004, pp. 69–72.

[82] B. Widrow and S. D. Stearns, *Adaptive Signal Processing*. New Jersey: Prentice Hall, 1985.

[83] M. Z. Ilyas, A. O. Abid Noor, K. A. Ishak, A. Hussain, and S. A. Samad, "Normalized least mean square adaptive noise cancellation filtering for speaker verification in noisy environments," in *Proc. International Conference on Electronic Design*, 2008, pp. 1–4.

[84] M. J. F. Gales and S. J. Young, "HMM recognition in noise using parallel model combination," in *Proc. European Conference on Speech Communication and Technology (EUROSPEECH'93)*, 1993, pp. 837–840.

[85] ——, "Cepstral parameter compensation for HMM recognition in noise," *Speech Communication*, vol. 12, no. 3, pp. 231–239, 1993.

[86] T. Matsui, T. Kanno, and S. Furui, "Speaker recognition using HMM composition in noisy environments," *Computer Speech and Language*, vol. 10, no. 2, pp. 107–116, 1996.

[87] C. Cerisara, L. Rigazio, and J.-C. Junqua, "α-Jacobian environmental adaptation," *Speech Communication*, vol. 42, no. 1, pp. 25–41, 2004.

[88] A. Abad, C. Nadeu, J. Hernando, and J. Padrell, "Jacobian adaptation based on the frequency-filtered spectral energies," in *Proc. European Conference on Speech Communication and Technology (EUROSPEECH'03)*, 2003, pp. 1621–1624.

[89] J. Anguita, J. Hernando, and A. Abad, "Improved Jacobian adaptation for robust speaker verification," *IEICE Transactions on Information and Systems*, vol. E88-D, no. 7, pp. 1767–1770, 2005.

[90] J. Anguita and J. Hernando, "Jacobian adaptation with continuous noise estimation for real speaker verification applications," in *Proc. 2006 IEEE Odyssey: Speaker and Language Recognition Workshop*, 2006, pp. 1–5.

[91] K. Yoshida, K. Takagi, and K. Ozeki, "Improved model training and automatic weight adjustment for multi-SNR multi-band speaker identification system," in *Proc. International Conference on Spoken Language Processing (ICSLP'04)*, 2004, pp. 1749–1752.

[92] L. Yang and W. Gong, "Multi-SNR GMMs-based noise-robust speaker verification using $1/f^{\alpha}$ noises," in *Proc. International Conference on Pattern Recognition*, vol. 4, 2006, pp. 241–244.

[93] J. Ming, T. J. Hazen, and J. R. Glass, "A comparative study of methods for handheld speaker verification in realistic noisy conditions," in *Proc. 2006 IEEE Odyssey: Speaker and Language Recognition Workshop*, 2006, pp. 1–8.

[94] J. Ming, P. Jancovic, and F. J. Smith, "Robust speech recognition using probabilistic union models," *IEEE Transactions on Speech and Audio Processing*, vol. 10, no. 6, pp. 403–414, 2002.

[95] J. Ming, J. Lin, and F. J. Smith, "A posterior union model with applications to robust speech and speaker recognition," *EURASIP Journal on Applied Signal Processing*, vol. 2006, pp. 1–12, 2006.

[96] J. Jung, K. Kim, and M. Kim, "Advanced missing feature theory with fast score calculation for noise robust speaker identification," *Electronics Letters*, vol. 46, no. 14, pp. 1027–1029, 2010.

[97] J. Wolf, "Efficient acoustic parameters for speaker recognition," *Journal of the Acoustical Society of America*, vol. 51, no. 6, pp. 2044–2056, 1972.

[98] J. Droppo, A. Acero, and L. Deng, "A nonlinear observation model for removing noise from corrupted speech log mel-spectral energies," in *Proc. International Conference on Spoken Language Processing (ICSLP)*, 2002, pp. 1569–1572.

[99] N. B. Yoma, F. McInnes, and M. Jack, "Improving performance of spectral subtraction in speech recognition using a model for additive noise," *IEEE Transactions on Speech and Audio Processing*, vol. 6, pp. 579–582, 1998.

[100] L. E. Baum, T. Petrie, G. Soules, and N. Weiss, "A maximization technique occuring in the statistical analysis of probabilistic functions of Markov chains," *Annals of Mathematical Statistics*, vol. 41, no. 1, pp. 164–171, 1970.

[101] A. Nadas, D. Nahamoo, and M. A. Picheny, "Speech recognition using noise-adaptive prototypes," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 37, no. 10, pp. 1495–1503, 1989.

[102] R. C. Rose, E. M. Hofstetter, and D. A. Reynolds, "Integrated models of speech and background with application to speaker identification in noise," *IEEE Transactions on Speech and Audio Processing*, vol. 2, no. 2, pp. 245–257, 1994.

[103] B. A. Mellor and A. Varga, "Noise masking in a transform domain," in *Proc. International Conference on Acoustics, Speech, and Signal Processing (ICASSP'93)*, vol. 2, 1993, pp. 87–90.

[104] T. Painter and A. Spanias, "Review of algorithms for perceptual coding of digital audio signals," in *Proc. International Conference on Digital Signal Processing (DSP'97)*, vol. 1, 1997, pp. 179–208.

[105] D. D. Greenwood, "Critical bandwidth and the frequency coordinates of the basilar membrane," *Journal of the Acoustical Society of America*, vol. 33, no. 10, pp. 1344–1356, 1961.

[106] B. Scharf, "Critical bands," in *Foundations of Modern Auditory Theory*, J. V. Tobias, Ed. New York: Academic Press, 1970.

[107] E. Zwicker and H. Fastl, *Psychoacoustics Facts and Models*. New York: Springer-Verlag, 1990.

[108] A. Alexander, F. Botti, D. Dessimoz, and A. Drygajlo, "The effect of mismatched recording conditions on human and automatic speaker recognition in forensic applications," *Forensic Science International*, vol. 146S, pp. S95–S99, 2004.

[109] A. Schmidt-Nielsen and T. Crystal., "Speaker verification by human listeners: Experiments comparing human and machine performance using the NIST 1998 speaker evaluation data," *Digital Signal Processing*, vol. 10, pp. 249–266, 2000.

[110] Y. Hu and P. C. Loizou, "Incorporating a psychoacoustical model in frequency domain speech enhancement," *IEEE Signal Processing Letters*, vol. 11, no. 2, pp. 270–273, 2004.

[111] D. Pan, "A tutorial on MPEG/audio compression," *IEEE Multimedia*, vol. 2, no. 2, pp. 60–74, 1995.

[112] A. Spanias, T. Painter, and V. Atti, *Audio Signal Processing and Coding*. New Jersey: Wiley-Interscience, 2007.

[113] A. Larcher, J.-F. Bonastre, and J. S. Mason, "From GMM to HMM for embedded password-based speaker recognition," in *European Signal and Image Processing Conference (EUSIPCO)*, 2008.

[114] J. Schäfer and K. Strimmer, "A shrinkage approach to large-scale covariance matrix estimation and implications for functional genomics," *Statistical Applications in Genetics and Molecular Biology*, vol. 4, no. 1, 2005.