SPEAKER IDENTIFICATION USING DISTRIBUTED VECTOR QUANTIZATION AND GAUSSIAN MIXTURE MODELS

LOH MUN YEE

UNIVERSITI TEKNOLOGI MALAYSIA

SPEAKER IDENTIFICATION USING DISTRIBUTED VECTOR QUANTIZATION AND GAUSSIAN MIXTURE MODELS

LOH MUN YEE

A thesis submitted in fulfilment of the
requirements for the award of the degree of
Master of Science (Computer Science)

Faculty of Computer Science and Information Systems
Universiti Teknologi Malaysia

MARCH 2010

ABSTRACT

Speaker identification is the computing task of recognizing people's identity based on their voices. There are two main difficulties in this field. First is how to maintain the accuracy rate under large amount of training data. Second is how to reduce the processing time. Previous studies reported that Gaussian Mixture Model (GMM) for speaker identification appears to have many advantages. However, due to long processing time, this process does not always produce satisfying result in practice. Meanwhile, current mechanisms for hybrid production of speaker identification are directed more towards accuracy problems, not processing time optimization. This research focuses on constructing distributed data training on Vector Quantization (VQ) modeling to achieve an initial result. The decision tree approach is applied to obtain distributed training for VQ model. GMM classification process is then employed on the initial result to achieve a final result. The efficiency of the model is evaluated by computational time and accuracy rate compared to GMM baseline models. Experimental result shows that the hybrid distributed VQ/GMM model yields better accuracy. Besides, it gives 80% reduction in processing time and is 5 times faster compared to GMM baseline models. In conclusion, this research successfully improves the computational time and accuracy of the text-independent speaker identification system.

ABSTRAK

Sistem pengecaman suara adalah tugas mengecam identiti manusia berasaskan suara. Terdapat dua masalah dalam bidang ini. Yang pertama adalah bagaimana menjamin ketepatan dalam pemprosesan data yang besar. Manakala masalah kedua adalah bagaimana untuk mengurangkan masa pemprosesan. Kajian-kajian terdahulu menunjukkan bahawa Model Bercampur Gaussian (GMM) bagi pengecaman suara mempunyai banyak kelebihan dalam bidang ini. Akan tetapi, ia tidak memuaskan kerana masa yang diambil untuk memproses data adalah lama. Sementara itu, penyelidikan semasa terhadap teknik hibrid pengecaman suara hanya memfokus kepada masalah ketepatan, bukannya kepada pengoptimuman masa pemprosesan. Kajian ini menumpukan kepada pembinaan latihan data teragih ke atas model Kuantisasi Vektor (VQ) untuk mendapat keputusan awal. Pepohon Keputusan adalah pendekatan yang digunakan dalam latihan model VQ secara teragih. Kemudian, keputusan awal tersebut digunakan oleh proses klasifikasi GMM untuk mencapai keputusan muktamad. Kebekesanan model tersebut telah diuji dari segi masa pemprosesan dan ketepatan, dibandingkan dengan model-model asas GMM. Keputusan ujikaji menunjukkan bahawa model hibrid teragih VQ/GMM adalah lebih baik dari segi ketepatan. Selain itu, ia mengurangkan masa pemprosesan sebanyak 80% dan 5 kali lebih cepat daripada menggunakan model-model asas GMM. Kesimpulannya, penyelidikan ini telah berjaya membantu menjimatkan masa pemprosesan dan ketepatan untuk sistem pengecaman suara secara bebas-teks.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

VQ	-	Vector Quantization
GMM	-	Gaussian Mixture Model
DTW	-	Dynamic Time Warping
HMM	-	Hidden Markov Models
FFT	-	Fast Fourier Transform
LPC	-	Linear Prediction Coding
SVM	-	Support Vector Machine
MFCC	-	Mel-Frequency Cepstrum Coefficients
PLP	-	Perceptual Linear Predictive
MAP	-	Maximum A Posterior
VQPGS	-	VQ Pre-classifier for Gaussian Selection
LBG-GMM	-	LBG Training for GMM Model
ZCR	-	Zero Crossing Rate
EM	-	Expectation Maximization
ML	-	Maximum likelihood

LIST OF APPENDICES

CHAPTER 1

INTRODUCTION

Speaker recognition is a process where a person is recognized on the basis of his/her voice signals (Doddington, 1985). Evolution of speaker recognition is quantum jump in artificial intelligence and forensic science technologies because it endows machines with the human-like abilities to distinguish people's identity from one another (Judith, 2000). Speaker recognition technologies are currently applying in many daily applications. For example, access control system, security control for confidential information, transaction authentication and telephone banking.

Speaker recognition can be text dependent or text independent. For text dependent system, predefined utterance is used for training and testing the system (Reynolds, 2002). This is different in text independent system, where user can simply use whatever utterance for recognition task (Bimbot, 2005). Speaker verification and speaker identification are the subset of speaker recognition, where speaker verification accepts or rejects the identity claim of a speaker whereas speaker identification determines which registered speaker provides a given utterance from a set of known speaker (Campbell, 1997).

The success of speaker recognition system depends largely on how to classify a set of feature used to characterize speaker specific information (Jiuqing and Qixiu, 2003; Sorensen and Savic, 1994). However, pattern classification from speech signal remains as a challenging problem encountered in general speaker recognition system, including speaker verification and speaker identification.

Recent development in classifying speaker data from a group of speakers is still insufficient to provide a satisfying result in achieving high performance pattern classification. There are two main difficulties in pattern classification field: how to maintain accuracy under incremental amounts of training data and how to reduce the processing time as real time systems regarding efficiency and simplicity of calculation (He and Zhao, 2003; Campbell, 2002). This research aims to solve the aforementioned problems.

1.1 Background of the Problem

Building robust speaker recognition systems are often difficult because speech signal is dynamic and influenced by many sources of variation. There have seen significant progress being made to cope with this problem using different techniques in the past two decades (Sadaoki, 1997). The problem of speaker recognition belongs to a much broader topic in scientific and engineering so called pattern classification. The goal of pattern classification is to classify objects of interest into a number of categories or classes (Richard et al., 2000). The classes here refer to individual speakers.

Pattern classification plays as a crucial part in speaker modeling component chain. The results of pattern classification strongly affect the speaker recognition engine to decide whether to accept or reject a speaker. Early pattern classification is produced by Dynamic Time Warping (DTW) (Sakoe and Chiba, 1978; Jingwei et al., 2002) and Hidden Markov Models (HMM) (Lawrence, 1989). These techniques are not really efficient for real time application due to characteristic of text dependent recognition. As an alternative to solve the problem, Vector Quantization (VQ) (Vlasta and Zdenek, 1999), Gaussian mixture model (GMM) and Support Vector Machine (SVM) (Solera et al., 2007) were introducing for speaker recognition. GMM is focus of research after Reynolds and Rose (1995) prove its effective performances in text independent speaker identification.

Previous studies have reported GMM technique of pattern classification appears to have several advantages. However, in practice the process does not always produce satisfied result due to the long computational time (Hong et al., 2004; Reynolds and Campbell, 2007). Consequently, alternative methods must be sought in order to reduce processing time problem for GMM technique.

Other than these traditional methods, there are some hybrid methods as an alternative for speaker pattern classification. These hybrid methods draw the attention of the researcher because it is proved with significant improvement for speaker recognition accuracy rates. For example, hybrid GMM/ ANN (Xiang and Berger, 2003), hybrid GMM/VQ (Pelecanos et al., 2000) and hybrid GMM/SVM (Fine et al., 2001; Minghui et al., 2006). Fenglei and Bingxi (2003) claim that most of these hybrid system use GMM because it was able be performed in a completely text independent situation.

Performance of speaker recognition systems in term of accuracy rates has been significantly improved over hybrid conditions. However, Moon et al., (2003) declare that when speaker recognition is adopted in real-world application, processing time issue is often observed. Meanwhile, current works for the hybrid production of speaker recognition are directed more towards accuracy problems, not processing time problems. Therefore, it is encouraging if a speaker recognition task can be conducted in a "good" pattern classification machine.

1.2 Motivation of Research

This research is intended to ascertain and enhance GMM pattern classification approach via hybrid modeling for speaker identification. This pattern classification approach should be able to handle large speaker database in short time limit, whereas the accuracy rate is still maintained or even higher than the conventional GMM pattern classification technique.

Shen and Reynolds (2008) analyzed the NIST speech corpus evaluation set by using GMM and concluded that more than 5 minute are used for identifying 100 sets of speaker data. Similar reviews have been done by Bruneaua et al., (2009). According to the author, GMM computational time will dramatically increase when dealing with large set of data. Therefore, banking authentication systems often verify user identity instead of identify user voice with full set of data (Shen and Reynolds, 2008). However, there are still a need on GMM speaker identification system, for example: access control system.

A survey is done to investigate suitable solution for reducing GMM technique processing time. Tae et al. (2002) claims by reducing learning data can improve training speed for speaker identification. Similarly, Tomi et al. (2003) reveals a speaker pruning algorithm for real time speaker identification which is based on reducing training data. Meanwhile, Xiaodan et al. (2006) declares decision tree approach is effective for solving large data problem which can divide the whole set of data into separable classes. However, in order to obtain better accuracy rates for speaker identification system, several hybrid pattern classification attempts need to be conducted. (Qiguang et al., 1996; Fine et al., 2001; Minghui et al., 2006).

Due to above factors, a novel hybrid method which takes the advantage of 2 typical pattern classification approaches for text independence speaker identification is constructed. These two techniques are Vector Quantization (VQ) and Gaussian Mixture Models (GMM). The primary reason of deriving hybrid idea is VQ is able to provide shorter processing time (Vlasta and Zdeněk, 1999). The subsequently reason is GMM shows it is effective and gains a stable accuracy while handling large speaker data (Auckenthaler et al., 1999; Daniel, 2004). In the first focus of proposed method, VQ are used as pre-training pattern classification approach to initialize a small set of speaker model which have closer distance measure in identification training phase. These initial speaker models are used for GMM testing in identification testing phase to calculate the log likelihood score. This is mainly for leading GMM engine testing on possible speaker models instead of comparing all speaker models in database.

However, the first stage of the proposed method is able to decrease the processing time efficiently, but the accuracy rate cannot be guarantee. This is due to VQ can only perform well with small range of data (Jialong et al., 1999). Consequently, alternative methods must be sought in order to improve accuracy rates

for VQ approach. Guangyu and Michael (2005) suggested an Adaptive Discriminative VQ technique to divide feature vector space into subspace before training. Acero et al. (1996) suggested a Sub-partitioned vector quantization technique to indicate a significant reduction in memory for speech processing. From the result of these reviews, VQ is not able to process a large set of data without any pre-processing technique directly. Therefore, it brings another focus for this research.

The second focus of the research introduced a process of distributing data by classifying data into smaller subgroup depending on speaker's pitch, which given name "distributed data training". The research hypothesis claims that training data in small subgroup will lead towards the time efficiency compared to run searching under whole set of data. Thus, a decision tree function which separates the speech signal according to their gender information via pitch analysis is added as pre-processing stage for hybrid VQ/GMM approach.

This work focuses on construction of VQ/GMM model based on distribute leading to increase processing time and result in higher accuracy for text independent speaker identification. The proposed method has been successfully applied and tested from the experiments conducted.

1.3 Problem Statement

Based on the analysis on the usage of VQ and GMM pattern classification approach in speaker identification, VQ and GMM moderately applied in text independent environments successful, but it is still suffer from several problems:

(i) Large training set of GMM result in long computational time. It compare the entire speaker models in speaker database via maximum log likelihood score.

(ii) VQ is only suitable for classifying small range of data because of its template matching characteristic.

(iii) Accuracy rates will decrease along with the increasing of training data for VQ pattern classification; it generates an unstable solution when dealing with large date set.

Conventional GMM approach in recognizing people identity from speech signal is still insufficient to produce data. It is time consuming and requires heavy computations using powerful workstation. The emergence of speaker recognition technologies require pattern classification engine for speaker recognition manage to process huge speaker data sets in limited time. Hence, a hybrid distributed VQ/GMM algorithm with less computational time and capable to work on huge dataset should be developed.

1.4 Goal

This research intended to improve time consuming issue in conventional GMM approach. It constructs a pre-processing stage; distributed data training on VQ modeling and pass the pre-processing result to GMM model for identification task. The efficiency is evaluated in term of computational time and accuracy for the whole speaker identification process in achieving accurate result compared to VQ and GMM baseline model. Besides, a comparison is conducted against other hybrid VQ/GMM models which focus on speedup speaker identification system.

1.5 Objectives

(i) To develop a novel hybrid distributed VQ/GMM model for text-independent speaker identification.

(ii) To reduce the computational time for conventional GMM speaker identification process and maintain high accuracy rate.

(iii) To optimize large data training problem by distributed training using decision tree approach.

1.6 Scope

This research is bound to the following scopes:

(i) The work reported here focus on the closed-set text-independent speaker identification task. A hybrid model, distributed Vector Quantization and Gaussian Mixture Model is applied on speaker identification pattern classification.

(ii) TIMIT speech corpus is used as speech database for evaluation purpose. Besides, experiments are conducted in clean speech environment for standard evaluation. The selective of 10 sets, 50 sets and 100 sets data from this corpus are used to go thru all phase of experiments in order to show the impact of data increasing. Finally, the experiments is conducted on full set TIMIT data to show how proposed model handle large datasets.

(iii) Comparison is made between proposed technique, baseline VQ and GMM techniques and other hybrid VQ/GMM models on enhancing speed for speaker identification system.

(iv) Performance measurement is evaluated by computational time and percentage of accuracy rate, which is a number of correctly identified speakers against the total number of tested speaker.

1.7 Significance of the research

As discussed in problem statement, this research is intended to construct a distributed VQ modeling as pre-processing stage in pattern classification. The results of distributed VQ modeling is used by GMM classification. Hence, an

algorithm with less computational time and capable to works on large dataset are developed.

The principal contribution of this study is to propose a method to reduce time consuming issue for conventional GMM modeling successfully. Besides, it presents a transformed criterion for conventional hybrid VQ/GMM classifier. Conventional hybrid VQ/GMM classifier is employed using VQ as pre-classifier directly without any pre-processing. However, the proposed method concern about distributing data into smaller range in order to reduce computational time while the accuracy rate is still maintained.

Experimental result shows that the development of hybrid distributed VQ/GMM method always yielded better improvements in accuracy and bring more than 80% reduce in processing time compared to GMM baseline system. Besides, problem of VQ can be solved by distributed VQ because VQ can only performed well on small range of data. Therefore, proposed model also leads better result compared to conventional hybrid VQ/GMM. Thus, it brings a significance improvement for speaker identification pattern classification technique as the emergence of biometrics security transforming itself into worldwide needs.

1.8 Research Contributions

The work addressed in this study has contributed to the following aspects:

(i) Distributed VQ model distribute speaker model in smaller subgroup and train data in following subgroup only, hence minimize the accuracy error for pre-processing phase.

(ii) By initializing hybrid Distributed VQ/GMM model in speaker identification task, smaller computational times were needed for testing speaker data. This study discloses a competent and less computational hybrid model in handling pattern classification task, whilst result is reliable for speaker identification.

This research can be considered as an applied research, which can benefit the following agencies:-

(i) Access Control System - To help provide faster identification process

(ii) Forensic Science Agencies - Proposed model is capable of handling large dataset and provide better accuracy rates which can help to convict a criminal or discharge an innocent in court.

(iii) Research bodies – Provide alternative way to do pattern classification task

1.9 Thesis Organization

This thesis consists of six chapters as following:

Chapter 1 briefly introduces speaker recognition pattern classification approach and some research background. Motivations of research and problem statement are also defined. Goal, objectives and scopes of research are stated clearly. Finally, research contributions are discussed.

Chapter 2 conducts a review to previous work; consist of the general speaker recognition framework, some pattern classification approach in speaker recognition, the need of a hybrid modeling and problem over hybrid modeling. Besides, this chapter also analyze some techniques focus on minimize processing time and distributed VQ classification, as well as discuss several existing process for hybrid VQ/GMM model.

Chapter 3 presents the methodology and theoretical framework of this study. It consists of following procedures: analyze the ability of the distributed VQ, design methodology, data collection and finally the output which is the hybrid distributed VQ/GMM design framework.

Chapter 4 reports on the implementation of the proposed model. The model is designed to be implemented modularly by four phase, which consist of 2 training phase, and 2 classification phase, namely distributed data pre-processing phase, distributed VQ model training phase, distributed VQ model classification phase,

GMM model identification phase. This chapter illustrates how workflow and progress has been carried out in details for this study.

Chapter 5 presents and discusses the results of conducted experiments based on the proposed approach in this study. Several experiments are carried out on increasing data environment and comparisons are made between this study and several existing hybrid VQ/GMM application.

Chapter 6 summarizes and concludes the study and outlines topics for future work. The contributions obtained from this study toward current approach and improvement disclose from this research are clearly stated in this section.

CHAPTER 2

LITERATURE REVIEW

Biometric systems are automated method of verifying or recognizing the identity of the person on the basis of some physiological characteristic, such as a finger print or face pattern and human voice (Kung et al., 2005).

Human voice conveys information about the language being spoken and the emotion, gender and, generally, the identity of the speaker. Speaker recognition is the computing task of validating a user's claimed identity using characteristics extracted from their voices. (Campbell, 1997; Sadaoki, 2004). Voice of a person has many prominent characteristics like pitch, tone which can be used to distinguish a person from the other (Atal, 1976).

Two major field of research in speaker recognition technology are speaker identification and speaker verification (Reynolds. and Heck, 2000). In speaker identification, there is no a priori identity claim, and the system decides who the person is, which group the person is a member of, or (in the open-set case) that the

person is unknown. Speaker verification is defined as deciding if a speaker is whom he claims to be. This is different than the speaker identification problem, which is deciding if a speaker is a specific person or is among a group of persons. In speaker verification, a person makes an identity claim (e.g., by entering an employee number or presenting his smart card) (Rosenberg, 1976; Rosenberg and Soong, 1992; Doddington, 1985).

Speaker recognition can be based on text-dependent or text-independent utterances, depending on whether the recognition process is constrained to a pre-defined text or not (Gish and Schmidt, 1994). The text-dependent systems require a user to re pronounce some specified utterances, usually containing the same text as the training data. There is no such constraint in text independent systems, where the classification is done without prior knowledge of what the speaker is saying. In the text-dependent system, the knowledge of knowing words or word sequence can be exploited to improve the performance. Thus, text-dependent speaker recognition usually gives better performance than text-independent recognition for small amounts of training and testing data (on the order of ten seconds) (Sachin et al., 2008). However, text-independent systems have drawn the attention of the researchers due to its flexibility and practical in real time system. As some researchers have pointed out, the researcher's influence is diffused by the very fact of the demand of forensic science technologies (Meng et al., 2008). Voice recognition for smart voice mail systems and voice identification for building access are two general usages on text-independent systems.

Speaker recognition system basically involves two main phases, the training stage and the testing stage (Richard and Daryl, 1990). These phases involve two main parts:

• Feature Extraction.

• Pattern Classification.

At the time of training, speech sample is acquired in a controlled and supervised manner from the user. The speaker recognition system has to process the speech signal in order to extract speaker discriminatory information from it. This discriminatory information will form the speaker model, which is a process of enrollment speaker data. At the time of testing a speech sample is acquired from the user. The speaker recognition system has to extract the features from this sample and compare it against the models already stored before hand. This is a pattern matching or classification task.

2.1 Speaker Identification

Speaker identification determines which registered speaker provides a given utterance from a set of know speaker. There are two cases in speaker identification which are called “close-set” identification and “open-set” identification (Tomi et al., 2004). In close-set speaker identification system it will choose a speaker in the training set who most matches the unknown speaker as the identification decision without regarding whether he/she is in the training set or not. While in open-set speaker identification system the reference model of the unknown speaker may not exist in the training set, thus an additional decision alternative (the unknown does not match any of the models in the training set) is required.

However, the focus of the research is on a close set environment. Generally

it is assumed the unknown voice must come from a fixed set of known speakers, thus the task is often referred to as closed-set identification

The basic structure for speaker identification is shown in Figure 2.1. The speech signal is first processed to extract features conveying speaker information. In the identification system these features are compared to a bank of models, obtained from previous training processes, representing the speaker set from which we wish to identify the unknown voice. For closed-set identification, the speaker associated with the most likely, or highest scoring model is selected as the identified speaker. This is simply a maximum likelihood classifier (O'Shaughnessy, 1986).

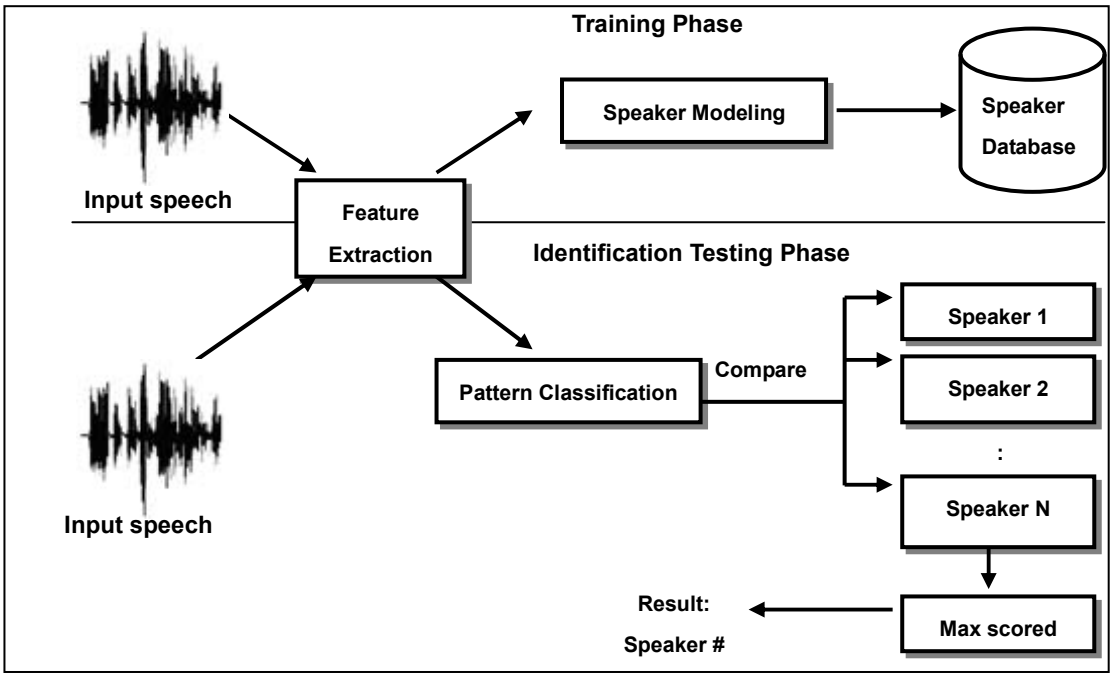


Figure 2.1 The basic structure for speaker identification

The more general problem, speaker identification, may be stated as follows. Out of a total population of *N* speakers, find that speaker whose reference pattern is most similar to the sample pattern of an unknown speaker. Since the sample pattern is compared to each of the *N* reference patterns and since there is a finite probability

of an incorrect decision for each comparison, it is apparent that the overall probability of an incorrect decision must be a monotonically increasing function of N (Rosenberg, 1976).

2.2 Speaker Verification

For the case of speaker verification, the speaker is classified as having the purported identity or not. That is, the goal is to automatically accept or reject an identity that is claimed by the speaker (Doddington, 1998). In this case, the user will first identify herself/himself (e.g., by introducing or uttering a PIN code), and the distance between the associated reference and the pronounced utterance will be compared to a threshold that is determined during training.

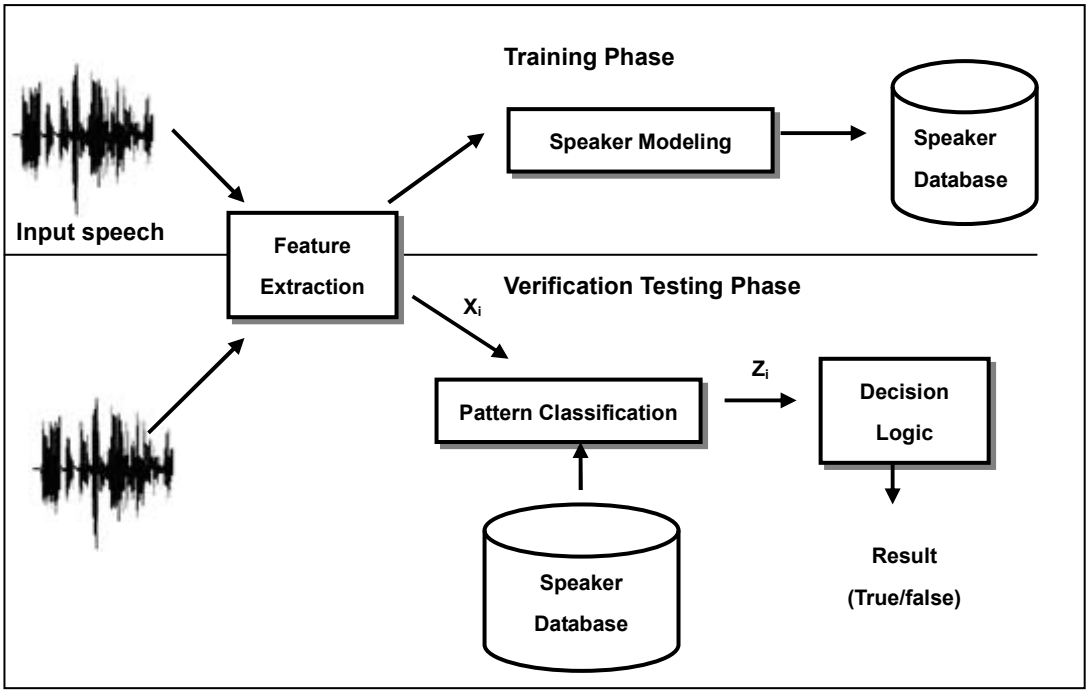


Figure 2.2 The basic structure for speaker verification

The general approach to speaker verification consists of five steps: digital speech data acquisition, feature extraction, pattern matching, making an accept/reject decision, and enrollment to generate speaker reference models (Campbell, 1997). A block diagram of this procedure is shown in Figure 2.2. Feature extraction maps each interval of speech to a multidimensional feature space. (A speech interval typically spans 10–30 ms of the speech waveform and is referred to as a frame of speech.) This sequence of feature vectors X_i is then compared to speaker models by pattern classification. This results in a match score Z_i for each vector or sequence of vectors. The match score measures the similarity of the computed input feature vectors to models of the claimed speaker or feature vector patterns for the claimed speaker. Last, a decision is made to either accept or reject the claimant according to the match score or sequence of match scores, which is a hypothesis testing problem.

2.3 Front-end Processing/Feature Extraction

Speech front-end processing consists of transforming the speech signal to a set of feature vectors (Moretto, 1995). The aim of this process is to obtain a new representation which is more compact, less redundant, and more suitable for statistical modeling (Premakanthan and Mikhad, 2001). Feature extraction is the key to front-end process; it mainly consists in a coding phase.

According to Stolcke et al. (2007), the purpose of feature extraction is to convert the speech waveform to some type of parametric representation (at a considerably lower information rate). A wide range of possibilities exist for parametrically representing the speech signal for the speaker recognition task, such

as Linear Prediction Coding (LPC) (Atal and Hanauer, 1971), Mel-Frequency Cepstrum Coefficients(MFCC), Perceptual Linear Predictive (PLP) and others.

Mel Frequency Cepstral Coefficients (MFCC) (Davis and Mermelstein, 1980) and Perceptual Linear Prediction (PLP) (Hermansky, 1990) are the most popular acoustic features used in speaker recognition. Often it depends on the task, which of the two methods leads to a better performance. Currently, researchers are focusing on improving these two cepstral features (Premakanthan and Mikhad, 2001) or appending new features on them (Waleed, 2007). In fact, it is generally believed that the spectrum smoothing done by MFCC and PLP has some sort of speaker normalization effect.

The selection of feature determines the separability of the speaker, and it also has large influence on the classification step, since the classifier must be turned to the given feature space. Thus, the selection of the features should be carefully considered in the system design. Several analyses have been done for feature extraction technique in order to observe the best technique for transforming the speech signal. Davis and Mermelstein (1980) reviewed the literature of a few feature extraction methods and compared in a syllable-oriented speaker dependent speech recognition system. The experiments were made in a noise-free environment and the segmentation was done manually. They found that features derived using cepstrum analysis outperform those that does not use it and that filter bank methods outperform LPC methods (PLP methods were not included). Best performance was achieved using MFCC.

John and Wendy (2002) report a comparable result of the use of PLP and MFCC. From their research, they figure out MFCC based feature extraction

methods seem to be performing well in most studies. Besides, a theoretical comparison of MFCC and PLP analysis is given in by Milner (2002). The theoretical comparison in (Milner, 2002) continues with a practical implementation. The spectral analysis is followed by channel normalization (both RASTA and CMN are tried) and extraction of dynamic features. The best results are reported for MFCC with RASTA filtration.

Similarly, Schmidt and Thomas (2000) has point out the state-of-the-art speaker recognition systems typically employ the MFCC as representative acoustic feature and GMM as pattern classification method has achieved very good performance which even better than recognition by human. Florian et al. (2005) have substantiated Schmidt and Thomas (2000) argumentation via some experiments. They reports that under clean conditions and when there is no significant mismatch, MFCC features lead to a performance that slightly superior to PLP.

Later on Chakroborty et al. (2008) list three reasons why MFCC method has become so dominant for speaker Identification system. First, MFCC is less vulnerable to noise perturbation, it gives little session variability and is easy to extract. Also, calculation of MFCC is based on the human auditory system aiming for artificial implementation of the ear physiology assuming that the human ear can be a good speaker recognizer too (Vergin, 1999). Further, computation of MFCC involves averaging the low frequency region of the energy spectrum (approximately demarcated by the upper limit of 1 kHz) by closely spaced overlapping triangular filters while smaller number of less closely spaced filters with similar shape are used to average the high frequency zone. Thus MFCC can represent the low frequency region more accurately than the high frequency region and hence it can capture formants which lie in the low frequency range and which characterize the vocal tract resonances (Rabiner and Juang, 2003; Ben and Nelson, 2002).

All these facts suggest that any speaker identification system based on MFCC can possibly be improved. Based on above declare, this research has decided to employ MFCC method as the feature extraction part in order to obtain most advantageous feature vector to adapted in the proposed hybrid pattern classification model.

2.3.1 Mel Frequency Cepstral Coefficients (MFCC)

Mel Frequency Cepstral Coefficients (MFCC) are based on the known variation of the human ear's critical bandwidths with frequency; filters spaced linearly at low frequencies and logarithmically at high frequencies have been used to capture the phonetically important characteristics of speech. The processes to obtain the MFCC can be summarized as in Figure 2.3. The speech input is typically recorded at a sampling rate above 10000 Hz. This sampling frequency was chosen to minimize the effects of aliasing in the analog-to-digital conversion. These sampled signals can capture all frequencies up to 5 kHz, which cover most energy of sounds that are generated by humans. As been discussed previously, the main purpose of the MFCC processor is to mimic the behavior of the human ears. In addition, rather than the speech waveforms themselves, MFCC are shown to be less susceptible to mentioned variations.

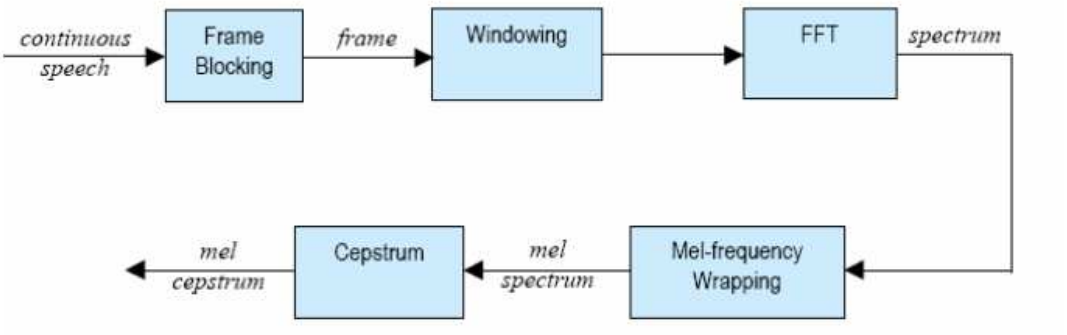


Figure 2.3 Mel Frequency Cepstral Coefficients process

2.3.1.1 Frame Blocking

In this step the continuous speech signal is blocked into frames of *N* samples, with adjacent frames being separated by *M (M < N)*. The first frame consists of the first *N* samples. The second frame begins *M* samples after the first frame, and overlaps it by *N - M* samples. Similarly, the third frame begins 2*M* samples after the first frame (or *M* samples after the second frame) and overlaps it by *N - 2M* samples. This process continues until all the speech is accounted for within one or more frames. Typical values for *N* and *M* are *N* = 256 (which is equivalent to ~ 30 msec windowing and facilitate the fast radix-2 FFT) and *M* = 100.

2.3.1.2 Windowing

The next step in the processing is to window each individual frame so as to minimize the signal discontinuities at the beginning and end of each frame. The concept here is to minimize the spectral distortion by using the window to taper the signal to zero at the beginning and end of each frame. If define the window, where *N* is the number of samples in each frame, then the result of windowing is the signal.

$$y_1(n) = x_1(n)w(n),\ 0 \le n \le N\text{-}1 \tag{2.1}$$

Typically the Hamming Window is used, which is of the form

$$w(n) = 0.54 - 0.46\cos\left(\frac{2\pi n}{N-1}\right), 0 \le n \le N-1 \tag{2.2}$$

2.3.1.3 Fast Fourier Transform

Next step is the Fast Fourier Transform (FTT) which converts each frame of *N* samples in time domain to frequency domain. The frame blocking step that was previously done was to enable the ease of performing of the FFT. The normal FFT equation is given below:

$$X_n = \sum_{k=0}^{N-1} x_k e^{-2\pi jkn / N},\ n = 0, 1, 2, \ldots, N-1 \tag{2.3}$$

2.3.1.4 Mel-Frequency Wrapping

The spectrum obtained from the above step is Mel Frequency Wrapped. The Mel-frequency scale is linear frequency spacing below 1000 Hz and a logarithmic spacing above 1000 Hz. As a reference point, the pitch of a 1 kHz tone, 40 dB above the perceptual hearing threshold, is defined as 1000 Mels. Therefore it is possible to use the following approximate formula to compute the Mels for a given frequency *f* in Hz:

$$\text{Mel}(f) = 2595 * \log_{10}(1 + f / 700) \quad (2.4)$$

The major work done in this process is to convert the frequency spectrum to Mel spectrum. Psychophysical studies have shown that human perception of the frequency contents of sounds for speech signals does not follow a linear scale. Thus for each tone with an actual frequency, *f*, measured in Hz, a subjective pitch is measured on a scale called the 'Mel' scale.

2.3.1.5 Cepstrum

The final step is converting the log Mel spectrum back to time. The result is called the Mel frequency Cepstrum coefficients (MFCC). The cepstral representation of the speech spectrum provides a good representation of the local spectral properties of the signal for the given frame analysis. Mel spectrum coefficients are real numbers. Therefore, is possible to convert them to the time domain using the Discrete Cosine Transform (DCT).

$$\tilde{C}_n = \sum_{k=1}^{K} (\log \tilde{S}_k) coz\left[n\left(k - \frac{1}{2} \right) \frac{\pi}{K} \right], n = 1,2,..,K \quad (2.5)$$

Where S_k is the Mel Scaled Signal got after wrapping. C_n is the Cepstral Coefficient.

2.4 Back-end Processing/ Pattern Classification

Duda et al. (2001) pointed that pattern classification classify data (patterns) based on a priori knowledge or statistical information extracted from the patterns. The patterns classified are usually group of measurements or observations, defining points in an appropriate multidimensional space (Schlesinger and Hlavác, 2002).

The pattern classification task of speaker identification involves computing a match score, which is a measure of the similarity of the input feature vectors to some model (Bo et al., 2006). Speaker models are constructed from the features extracted from the speech signal. To register users into the system, a model of the voice, based on the extracted features, is generated and stored (possibly on an encrypted smart card). Then, to authenticate a user, the matching algorithm compares the incoming speech signal with the model of the claimed user.

The different of modeling approaches may be divided into two distinct categories: discriminative and generative model. Discriminative models are optimized to minimize the error on a set of training samples (Hong and Kwong, 2004; Beyerlein, 1998). Classifiers that are discriminative models include multilayer

perceptions and support vector machines.

Generative models are probability density estimators that attempt to capture all of the underlying fluctuations and variations of the data (in this case the speaker's voice) (Raina et al., 2004). Generative models include Gaussian mixture models, hidden Markov models (Jaakkola and Haussler, 1998). Template models are subsets for generative model. For template models, the pattern classification is deterministic. The template model and its corresponding distance measure is perhaps the most intuitive method. The template method can be dependent or independent of time. An example of a time-independent template model is Vector Quantization modeling. Besides, Dynamic Time Warping also one of the famous models applied in template model.

This research will focus on the design of distributed VQ as pre-processing for GMM model. The next section gives a brief overview on evolution of pattern classifier. It explains the pattern classification techniques and briefly discusses the improvement done in previous research. An analysis of the literature has done in order to emerge the design of hybrid distributed VQ/GMM model. This report also reviews some of the previous hybrid VQ/GMM model and its. Next, remarks on some of the idea for solving similar problems have discussed and the adequate solution observed. Finally, distributed data training is proposed on VQ as a pre-process for GMM modeling to minimize computational time for identification.

2.4.1 Evolution of Pattern Classification Technique in Speaker Identification

Research in automatic speech and speaker identification by machines has attracted a great deal of attention for five decades. From the historic perspective, Lawrence kersta from Bell Labs made the first major step towards speaker identification by computers in the early 1960s where he introduced the term voiceprint for a spectrogram, which was generated by a complicated electro-mechanical device. The voiceprint was matched with a verification algorithm that was based on visual comparison (Kersta, 1978).

An older technique that has fallen out of favor since the advent of Hidden Markov Model is Dynamic Time Warping (DTW). The technique, DTW, was introduced to the speech community by Sakoe and Chiba (1971). The DTW algorithm is able to find the optimal alignment between two time series. It is often used to determine time series similarity, classification, and to find corresponding regions between two time series (Sakoe and Chiba, 1978).

In the 1970s, the use of pattern recognition ideas was introduced by Veichito and Zagoruyko(1970) in Russia. As an alternative to the template-matching approach for text-dependent speaker identification, the Hidden Markov technique (HMM) was introduced and it was a focus of research in the 1980s (Ferguson, 1980). At the same time, an algorithm named LBG algorithm for vector quantizes design was first carried out by Linde et al., (1980). Vector Quantization (VQ) in processing speaker data was develops by Tremain (1982). He led research in low bit rate speech coding designs by developing vector quantization approaches using split codebooks and tree-based algorithms for fast search of large codebooks. These techniques achieve equivalent 2400-bps speech intelligibility at only a 600-bps rate

(Campbell., 1996; Tremain, 1982). Later on, Soong et al. (1985) have successfully applied VQ method into text-independent speaker identification system in short-time spectral features.

In the 1990s, a number of innovations took place in the field of pattern recognition. The idea of single-state HMM, which is now called Gaussian mixture model (GMM) was investigated to solve text-independent speaker identification problems. GMM was the focus of the research trends of enhancing speaker identification system after Reynolds (1992) proves its effective performance in text-independent speaker identification. Besides, the problem of pattern recognition was transformed into an optimization problem solution involving minimization of the empirical identification error (Doddington et al, 2000).

After years 2000, a family of new classification techniques has recently been proposed. These bring a development of artificial neural network (ANN) as a replacement for pattern classification technique (Farrell et al., 1994). Recently, support vector machine (SVM) developed by Vapnik et al. (1997) have become an alternative solution for classification speaker data. This is due to the implement of binary classifier that claims as more simple in calculation (Vincent, 2003). However, GMM still act as the main character in pattern classification for speaker identification by due to its stability of maintaining large speaker data. Although many new technological promises have been offered, a number of practical limitations have also hindered while implementing in applications and services.

2.4.2 Dynamic Time Warping (DTW)

Dynamic time warping (DTW) is an algorithm for measuring similarity between two sequences which may vary in time or speed (Sadaoki, 1991a). It is a successful isolated-word processing technique often associated with small fixed-vocabulary speech recognition. This technique was first applied by Sadaoki (1981) in speaker verification.

- Text dependent with a predefined password
- Text dependent with a specific password for each customer

The first two points above can be summarized as fixed phrase verification, where a predefined phrase is used both during the training and the verification periods. For these cases the DTW approach is mostly used. According to Gold et al. (1999) "... the password of each user is simply represented as a small number of acoustic sequence templates corresponding to pronunciation of the password. ... the score associated with a new utterance of the password is computed by means of dynamic programming ... against the reference model(s)..."

DTW is a classification approach based on distances in feature space, makes use of the fact that in training, the same pass phrase is spoken as in test (Shahin and Botros, 1998). DTW compares the test vector sequence to a stored sequence from training directly, taking into account that two utterances of the same word or phrase are never exactly identical as distinct phonemes can be spoken shorter or longer. In order to cope with that, a time alignment of test and training sequences is found which is optimal in the sense that there is no other alignment yielding a smaller

overall distance and fulfilling certain restrictions from training directly, taking into account that two utterances of the same word or phrase are never exactly identical as distinct phonemes can be spoken shorter or longer (Ravi et al, 2002).

2.4.2.1 DTW in Speaker Identification

According to Sadaoki (1991b) teory , in a speaker identification system, the training data are used as a initial template, and the testing data is time aligned by DTW. DTW is a method that allows a computer to find an optimal match between two given sequences. The average of the two patterns is then taken to produce a new template to which a third utterance is time aligned. This process is repeated until all the training utterances have been combined into a single template. The idea of the DTW technique is to match a test input represented by a multi-dimensional feature vector T= [$t_1, t_2 \ldots t_I$] with a reference template R= [$r_1, r_2 \ldots r_j$]. While aim of DTW is to find the function w(i). as shown in figure 2.5. (Aronowitz, 2006).

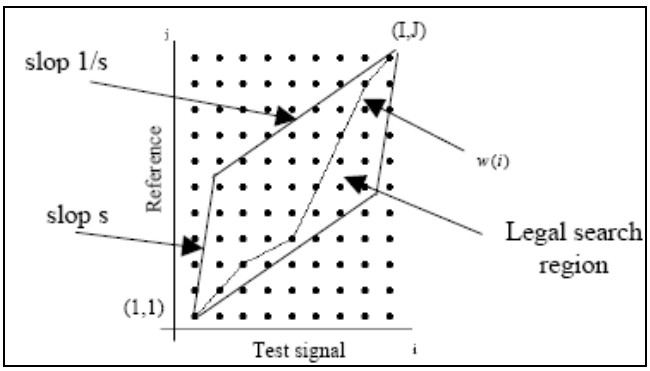


Figure 2.4 DTW model

2.4.3 Hidden Markov models (HMM)

Since 1975 the Hidden Markov Modeling (denoted as HMM) is a technique that has become popular in speech recognition research, introduced by the Russian mathematician A.A. Markov. With HMM-based methods, the statistical variation of spectral features is measured (Doddington, 1998).

HMM is a statistical model in which the system being modeled is assumed to be a Markov process with unknown parameters, and the challenge is to determine the hidden parameters from the observable parameters. The extracted model parameters can then be used to perform further analysis (Kristie, 1999).

HMM has the same advantage for speaker identification as they do for speech recognition (Rabiner, 1989). Remarkably robust models of speech events can be obtained with only small amounts of specification or information accompanying training utterances (Ephraim and Merhav, 2002). Speaker identification systems based on an HMM architecture used speaker models derived from a multi-word sentence, a single word, or a phoneme.

Some experiments have been done by Jarre and Pieraccini (1987) to investigate HMM when apply in speaker identification engine. They reported over the HMM speaker identification application, it uses HMM to encode the temporal evolution of the features and efficiently model statistical variation of the features, to provide a statistical representation of how a speaker produces sounds. During training HMM parameters are estimated from the speech using established automatic algorithms. During identification, the likelihood of the test feature sequence is

computed against the speaker's HMM. Generally, it applies to text-dependent application.

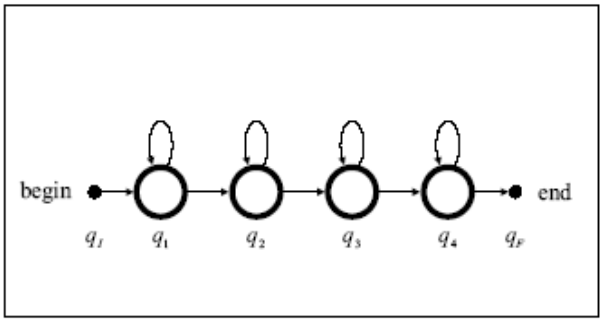
2.4.3.1 HMM in Speaker Identification

The hidden Markov model (HMM) formulation by Baum (1974) may be described as a finite state generator (figure 2.5). In speaker identification, each state, ($q_1 \ldots q_m$) of the HMM may represent phones or other larger units of speech. Temporal information is encoded by moving from state to state along the allowed transitions illustrated. Therefore, the temporal modeling is piecewise stationary. The amount of time spent in each state varies depending upon the data. This allows for variability in speaking rate.

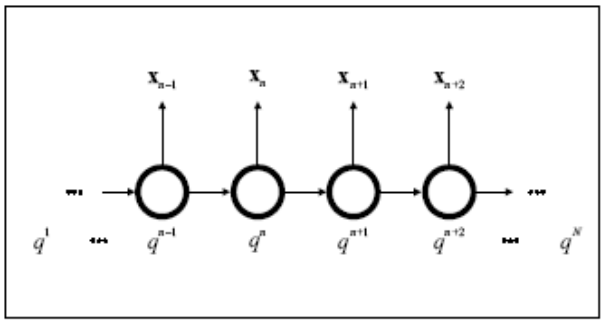
The operation of an HMM is straightforward. Let $\boldsymbol{M}$ denote an HMM such as the one illustrated in figure 2.5a. Consider an utterance, $\boldsymbol{X} = \{\boldsymbol{x}_1 \ldots \boldsymbol{x}_N\}$, to be processed by $\boldsymbol{M}$. Let q_l^n identify the l^{th} state in the Markov chain occupied at time n (i.e. corresponds to the n^{th} frame in X). At time n = 0, the begin state, q_I, is occupied. This state is a non-emitting state (that is, no feature vector is associated with it).

At time n =1 a transition is made to the first state, q_1, and the first feature vector, $\boldsymbol{x}_1$, is emitted with the probability $\boldsymbol{P}(\mathbf{x}_1|\boldsymbol{M}, q_1^1, \theta)$ (figure 2.5b). With each increment of n a transition must be taken along one of the edges of the graph with a

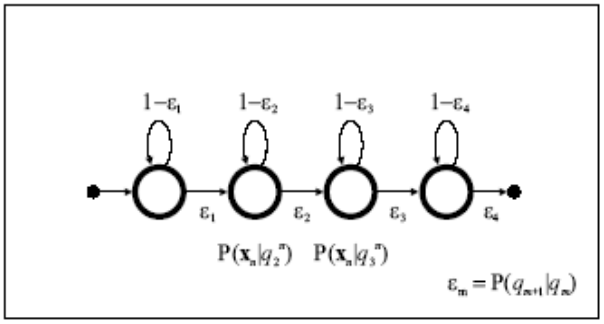
certain probability (figure 2.5c). At time $n = 2$ a transition must be made to state q_2 or back to q_1, and the feature vector, $\mathbf{x}_2$, emitted. This process continues until $n = N + 1$ at which time the end state, q_F, another non-emitting state, must be reached. The probabilities of the HMM process are accumulated to obtain the utterance likelihood, $\boldsymbol{P}(\mathbf{x}|\boldsymbol{M})$. In similar fashion to the GMM, the utterance score is $S(X) = log\ \boldsymbol{P}(\boldsymbol{x}|\boldsymbol{M})$.



a: A simple four state HMM. Each state, q_i, models a different section of the speech signal. Transitions are made from one state to the next along the indicated edges. Self-transitions that begin and end at the same state allow that state to be occupied for a variable amount of time.



b: The state dependence diagram of a first-order HMM. A first order HMM is one in which the state occupied at time n is dependent upon the previously occupied state only. Likewise the symbol, $\mathbf{x}_n$, emitted at time n only depends upon the state, q^n, occupied at that time.



c: The probabilities associated with the HMM.ε is the probability of exiting one state to a different state. Subsequently (1−ε) is the probability of staying in the same state. Upon entering a state the symbol $\mathbf{x}_n$ is emitted with probability $P(\mathbf{x}_n|q_i^n)$. This probability is typically estimated by a Gaussian mixture model. The begin and end states are non-emitting states that do not have an emission probability associated with them.

Figure 2.5 HMM finite state generators

2.4.4 Vector Quantization (VQ)

Data compression techniques have been developed to make the information occupy a small space, thus reducing memory and transmission costs, while at the same time preserving the quality as much as possible. Compression is classified into lossless or lossy, depending on whether or not the reconstruction is an exact replica or an approximation of the input signal, respectively.

Gray (1984) establish in a new method for data compression by using Vector Quantization. According to Gray (1984), Quantization is a lossy source coding technique, where the design of the quantizer determines the loss incurred, subject to certain constraints. It is a technique where the input samples are represented with reduced precision. If the samples of the source are quantized individually, it is scalar quantization; if blocks of samples are quantized together, it becomes Vector Quantization (VQ).

VQ is a data compression technique, producing a reconstruction with as small a distortion as possible. The quality of the reconstruction depends on the amount of data that is discarded. The samples of a source output are grouped into a *k*-dimensional vector, which is the input to a vector quantizer (Gersho, 1986).

The main component of a VQ is a code book that consists of representative vectors called code vectors (Gersho et al., 1992). The elements of the code vector are quantized values of the input samples. The same code book must be maintained both at the transmitter and the receiver. The code book is searched to find the code vector closest to the input vector based on a distortion error measure. The index of

the selected code vector, which is its address in binary form, is transmitted to the receiver. The receiver requires a simple table lookup; the index received is used to select the reproduction code vector that approximates the input vector of *k* samples (Sabin and Gray, 1984; Cuperman, 1986).

2.4.4.1 VQ in Speaker Identification

Vector Quantization (VQ) is a pattern classification technique applied to speech data to form a representative set of features. This set or codebook can be used to represent a given speaker. Among the first apply this technique to speaker identification were Soong et al. (1985) and Buck et al. (1985).

VQ is a process of mapping vectors from a large vector space to a finite number of regions in that space. Each region is called a cluster and can be represented by its center called a codeword. The collection of all code words is called a codebook.

In the training phase, a speaker-specific VQ codebook is generated for each known speaker by clustering his/her training acoustic vectors. The result code words (centroids) are shows in Figure 2.6 by black circles and black triangles for speaker 1 and 2, respectively. The distance from a vector to the closest codeword of a codebook is called a VQ distortion. In the identification phase, an input utterance of an unknown voice is “vector-quantized” using each trained codebook and the total VQ distortion is computed. The speaker corresponding to the VQ codebook with

the smallest distortion is identified.

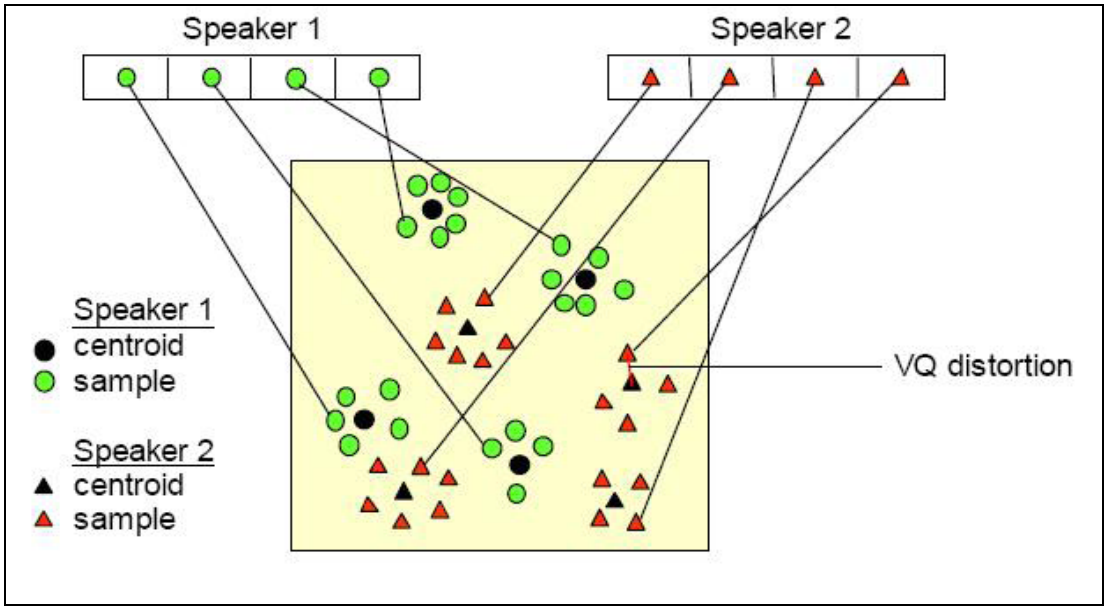


Figure 2.6 Conceptual diagram illustrating Vector Quantization codebook formation

The quality of speech coding with a code book is highly dependent on the similarity between the training set and the coded material can serve as a motivation for the use of code books for speaker identification (Sadaoki, 1991a). In this case, for each speaker, a code book is estimated in training. This code book can be thought of as containing those features as mean vectors which are characteristic for that speaker. Classification of unknown signals is based on the mean quantization error of test feature vectors in regard to the appropriate speaker specific code books, i.e. the quantization error is used as a distance measure.

2.4.5 Gaussian Mixture Models Classification Model

The Gaussian mixture model (GMM) is a density estimator and is one of the most commonly used types of classifier. The mathematical form of an *m* component Gaussian mixture for *D* dimensional input vectors is,

$$P(\mathbf{x}|M) = \sum_{i=1}^{m} a_i \frac{1}{(2\pi)^{D/2}|\Sigma_i|^{1/2}} \exp\left(-\frac{1}{2}(\mathbf{x}-\mu_i)^T \Sigma_i^{-1}(\mathbf{x}-\mu_i)\right) \quad (2.6)$$

Where $P(\mathrm{x}|M)$ is the likelihood of *x* given the mixture model, *M*. The mixture model consists of a weighted sum over *m* unimodal Gaussian densities each parameterized by the mean vectors, μ_i, and covariance matrices, Σ_i. The coefficients, a_i, are the mixture weights, which are constrained to be positive and must sum to one. The parameters of a Gaussian mixture model, a_i, μ_i and Σ_i for $i = 1m$ may be estimated using the maximum likelihood criterion via and the iterative Expectation Maximization (EM) algorithm (Paalanen et al., 2006). In general, fewer than ten iterations of the EM algorithm will provide sufficient parameter convergence.

Training GMM using maximum likelihood leads to a generative model. GMM are analogous to vector quantization in that the mean of each Gaussian density can be thought of as a codebook vector. The GMM combines the robust parametric approach of Gaussian density modeling with the arbitrary data modeling approach of the non-parametric vector quantization model.

2.4.5.1 GMM in Speaker Identification

A detailed discussion on applying GMM to speaker modeling can be found in (Reynolds, 1992). The basic method is straightforward. GMM with diagonal covariance matrices are generally used. Although full covariance matrices may be used if desired, the simplification leads to a model with fewer parameters without sacrificing accuracy. Empirical evidence indicates that the accuracy of a full covariance mixture model can be achieved by a diagonal covariance model with a larger number of mixture components.

In speaker identification, each trained speaker is represented by a GMM model. GMM is trained using maximum likelihood, to estimate the probability density function, $\boldsymbol{P}(\mathbf{x}_i|\boldsymbol{M})$, of the client speaker. The probability, $\boldsymbol{P}(\mathbf{X}|\boldsymbol{M})$, that an utterance, $\mathbf{X} = \{\mathbf{x}_1 \ldots \mathbf{x}_N\}$, is generated by the model, $\boldsymbol{M}$, is used as the utterance score. It is estimated by the mean log likelihood over the sequence,

$$S(X) = \log P(X|M) = \frac{1}{N}\sum_{i=1}^{N} \log P(\mathbf{x}_i|M). \tag{2.7}$$

The speaker's identity is defined by the model that produced the maximum probability, i.e.

$$i^* = arg \max_{1 \le i \le N_s} P(X|\lambda_i), \tag{2.8}$$

where n_s is the total number of trained speakers. The identification error rate (IER) is a measure of how well an identification system can identify speakers. It is simply defined as

$$IER = \frac{N_{ii}}{N_{ti}}, \tag{2.9}$$

where n_{ii} is the number of incorrect identifications and n_{ti} is the total number of

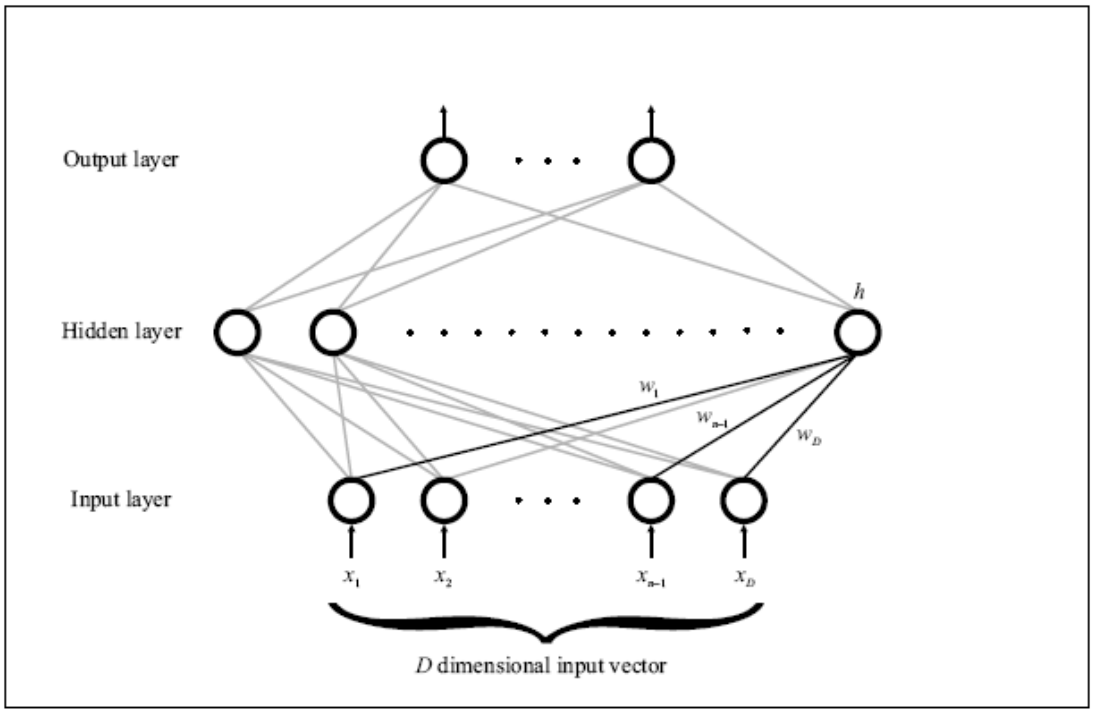
identifications performed. A system can be assumed to be good at identifying speakers if the IER is relatively low (approaching zero). On the other hand if the IER approached one then the system performs poorly in an identification role. The ability of an identification system relies heavily on the number of speakers in the enrolled population. As the number of enrolled speakers increases, the ability of the system to differentiate speakers decreases. This results in an increase in the IER.

2.4.6 Neural Networks for Classification

An artificial neural network is a powerful tool for regression and classification (Bishop, 1995). There are many types of neural network but in this literature review only focus on Multilayer Perceptron (MLP) that has been applied to speaker identification. MLP is a feed-forward network that incorporates little or no temporal information.

The architecture of an MLP with one input layer, one hidden layer and one output layer is illustrated in figure 2.7. Each node computes a linear weighted sum over its input connections, where the weights of the summation are the adjustable parameters. A transfer function is applied to the result to compute the output of that node. Commonly used transfer functions include linear, tanh, sigmoid and softmax functions. The weights of the network are estimated by gradient descent in a process called back-propagation. It is well known that an MLP with a non-linear transfer function and sufficiently large number of nodes in the hidden layer may approximate any functional mapping from input to output. This is why MLP are considered a useful tool. Furthermore, with appropriate constraints, an MLP may

be used to estimate posterior probabilities directly (Baum and Wilczek, 1988; Gish, 1990; Bishop, 1995).



A fully connected three layer multi-layer perceptron. The nodes in each layer are connected to all of the nodes of the previous layer and gives its output to all of the nodes of the next layer. The output, h, of the node indicated is given by the equation,

$$h = f\left(\sum_{i=1}^{D} w_i x_i\right)$$

where w_i are the connection weights, x_i are the outputs of the nodes of the previous layer and f is the transfer function

Figure 2.7 A multilayer perceptron

An MLP for speaker identification would have only one output node since the task is only to score the frames of an utterance. In the same fashion as the GMM, the utterance score is the mean classifier output over the complete utterance (Oglesby and Mason, 1990).

2.4.6.1 Neural Networks in Speaker Identification

MLP was applied to speaker identification as following step. First, the feature vectors are gathered for all speakers in the population. The feature vectors for one speaker are labeled as "one", and the feature vector for remaining speakers are labeled as "zero". An MLP is then trained for that speaker using these feature vectors. The MLP's for all speakers in the population are trained using this method.

Ideally, test vectors for a specific speakers should have a "one" response for that speaker's MLP, whereas test vectors from different speakers should have a "zero" response. For speaker identification, all test vectors are applied to each MLP, and outputs of each are accumulated. The speaker is selected as corresponding to the MLP with the maximum accumulated output (Lung, 2007).

One problem that is encountered during MLP training for large speaker populations is that the majority of the training vector have inhibitory (zero) labels, and only a small percentage of the training vectors have excitatory (one) labels. Hence, the classifier tends to learn "everything" in inhibitory. One method to alleviate this problem add noisy duplicates of excitatory vector to the training set to even out the distribution of excitatory and inhibitory vectors. However, for a large number of speakers, this results in an unwieldy amount of training data that can hinder the convergence of the classifier (Ouzounov, 2006).

The second problem encountered was the optimal MLP architecture, number of notes and hidden layers. To solve a particular problem, MLP architecture must be selected by trial and error which is a drawback. In addition, the training time

required to solve large dataset problems can be excessive, and the algorithm is vulnerable to converging to a local minima instead of the global optimum (Ouzounov, 2007).

2.4.7 Support Vector Machines (SVM)

Another alternative approach to develop classifier is Support Vector Machines (SVM). SVM is state-of-the-art tools for linear and nonlinear knowledge discovery (Scholkopf and Smola, 2002; Vapnik, 1995). The powers of SVM lie in their ability to transform data to a higher dimensional space and to construct a linear binary classifier in this space (Vincent and Steve, 2003). Unlike others, such as ANN or some modifications of HMM that minimize the empirical risk on the training set, SVM also minimize the structural risk (Vapnik, 1995), which results in a better generalization ability. In other words, given a learning problem and a finite training database, SVM generalize better than similar ANN because they properly weigh the learning potential of the database and the capacity of the machine (Solera et al., 2007).

SVM is a binary classification method that finds the optimal linear decision surface based on the concept of structural risk minimization (Raghavan et al., 2006). The decision surface is a weighted combination of elements of a training set. These elements are called support vectors, which characterize the boundary between the two classes. Let the two classes of the binary problem be labeled +1 and -1.

For the purpose to characterize the boundary between the two classes, maximizing the margin is needed. Maximizing the margin are the process find the "middle-line" consider two parallel lines both of which separate the two classes without error. Several steps need to be determining the linear separator (Figure 2.8a, 2.8b, 2.8c) (Burges, 1998):

- Find closest points in convex hulls
- Plane bisect closest points
- Maximize distance between two parallel supporting planes

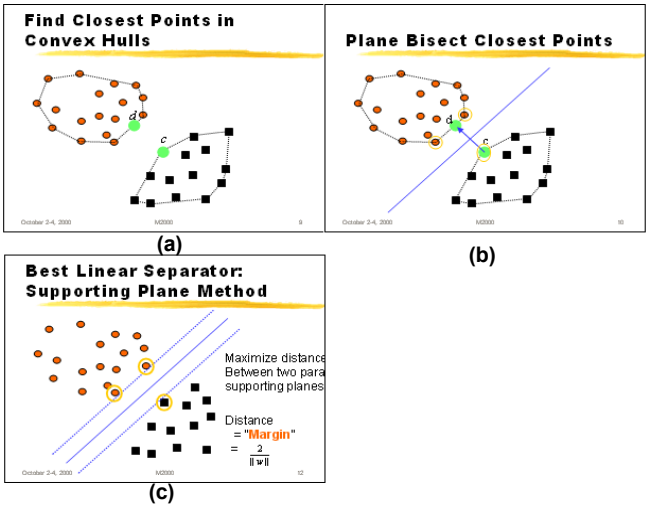


Figure 2.8 Steps for Binary Linear Decision Boundary

2.4.7.1 SVM in Speaker Identification

During speaker identification process, classifying the feature which derived from the transformation of feature extraction directly will not immediately works when using SVM (Campbell et al., 2006). It is because SVM only can process

fixed-length input, whereas speech signals are non-stationary. Therefore, some pre-processing need to done for categorizes the feature and scaling them.

SVM requires that each data instance is represented as a vector of real numbers. Hence, if there are categorical attributes, the first step is convert them into numeric data. Vincent and Steve (2003) recommend using *m* numbers to represent an *m*-category attribute. Only one of the *m* numbers is one, and others are zero. For example, a two-category attribute such as {speaker, imposter} can be represented as (0,1) and (1,0).

Scaling them before applying SVM is very important. The main advantage is to avoid attributes in greater numeric ranges dominate those in smaller numeric ranges. Another advantage is to avoid numerical difficulties during the calculation. Because kernel values usually depend on the inner products of feature vectors, e.g. the linear kernel and the polynomial kernel, large attribute values might cause numerical problems (Wan and Renals, 2005; Osuna and Girosi, 2005). Wan and Renals (2005) recommend linearly scaling each attribute to the range [−1, +1] or [0, 1].

2.5 Evaluation on Several Pattern Classification Techniques

The aim of this section is to provide an evaluation of the usage of several pattern classification techniques that applied in speaker identification. The advantages and disadvantages of each technique are reviewed and this research looks

forward in the enhancement for pattern classification which has been done in the past.

DTW is an algorithm for measuring similarity between two sequences which may vary in time or speed. It is a first pattern classification technique has been applied in automatic speaker identification, to cope with different speaking speeds (Myers and Rabiner, 1981). As stated as Sadaoki (1997), DTW is a simple and less computation method because it allows a computer to find an optimal match between two given sequences (e.g. time series) with certain restrictions. However, because of it characteristic of aligning two templates at equivalent point in time, it only suitable applies in text-dependent speaker identification.

Another text-dependent speaker identification technique is HMM. A HMM can be considered as the simplest dynamic Bayesian network. Kristie et al. (1999) declare that one of the most important advantages of HMM is that they can easily be extended to deal with classification tasks. Rabiner (1989) prove the HMM effectiveness via an experiment on speech data. He comments that because each HMM uses only positive data, they scale well; since new words can be added without affecting learnt HMM. It is also possible to set up HMM in such a way that they can learn incrementally. The basic theory of HMM is also very elegant and easy to understand. This makes it easier to analyze and develop implementations for (Ephraim and Merhav, 2002). However, Dymarski and Wydra (2008) disputed Ephraim and Merhav (2002) idea due to the number of parameters that need to be set in an HMM is huge. For example, for the very simple three-state HMM, there are a total of 15 parameters that need to be evaluated. For a simple four-state HMM, with five continuous channels, there would be a total of 50 parameters that would need to be evaluated.

Notwithstanding DTW and HMM shows considerable advantage in classification, yet these two techniques are suitable in text-dependent environment. In fact, real time system always request flexible and adaptable for the input speech in order to make the user convenient. Thus, pattern classification methods for text-independent speaker identification become a demand.

VQ is a classical quantization technique from signal processing which allows the modeling of probability density functions by the distribution of prototype vectors. It was originally used for data compression. Soong et al. (1985) are the first research group trying to adapt VQ for text-independent speaker identification system. They are success with achieved over 98% accuracy rates for speaker identification. VQ procedure is not guaranteed to find the best set of clusters but in practice it work very fast and can get good result in short-time acoustic feature classification. Nevertheless, VQ are based on short-time acoustic feature, when the feature set become large, it become hard to computed the distance between codebook, thus it cannot get very accurate result (Karpov et al, 2004).

It has been shown that VQ is very effective for speaker recognition. Although the performance of VQ is not as good as that of GMM (Reynolds and Rose, 1995), VQ is computationally more efficient than GMM. The advantages of using a GMM as the likelihood function are that it is computationally inexpensive, is based on a well understood statistical model, and, for text-independent tasks, is insensitive to the temporal aspects of them speech, modeling only the underlying distribution of acoustic observations from a speaker. GMM provides a probabilistic model of the underlying sounds of a person's voice. It is computationally more efficient than HMM and has been widely used in text-independent speaker recognition.

GMM, as introduced by Reynolds (1995), perform very well but training requires a lot of time and they get numerically unstable when trained with small amount of data. The main problem is the inversion of the (underestimated) covariance matrices. Pure vector quantization, e.g. using k-means clustering or the LBG algorithm by Linde, Buzo and Gray (Linde et al., 1985), on the other hand is numerically stable and rather fast, but the performance in speaker recognition is not as good as for GMM (Miyajima , 2001a). The use of GMM are most common due to it can be performed in a completely text independent situation. Besides, GMM are base on probabilistic framework, it provide high-accuracy recognition. Currently GMM have shown themselves to be adaptable to a wide variety of situations (Hsieh et al., 2003).

Other than GMM and VQ, artificial neural networks (ANN) which base on semi-parametric model, try to attempt to go against the tide of history of GMM as dominant of pattern classification method for speaker recognition (Franzini et al, 1989). And yet, the ANN has many parameters to fit, which can make learning more difficult, and may require special algorithms to normalize the parameters (Daniel et al., 2007). Besides, recurrent networks are only suitable for implementing short-term memories. It has been proven that training recurrent networks becomes increasingly difficult as the length of the sequences and the duration of temporal dependencies increases. This is primarily an effect of error propagation in gradient descent methods (Franzini et al, 1989).

Unlike others, such as ANN or some modifications of HMM that minimize the empirical risk on the training set, SVM also minimize the structural risk (Vapnik, 1995), which results in a better generalization ability. In other words, given a learning problem and a finite training database, SVM generalize better than similar ANN because they properly weigh the learning potential of the database and the

capacity of the machine (Solera et al., 2007). Support Vector Machines have become extremely successful discriminative approaches to pattern classification and regression problems. Excellent results have been reported in applying SVM in multiple domains. However, they require the use of an iterative process such as quadratic programming to identify the support vectors from the labeled training set of samples. When the number of samples in the training set is huge, sometimes it is impossible to use all of them for training.

Due to certain practical limitations the SVM has not gained widespread usage in mainstream applications. Initial speaker identification work using SVM by Schmidt and Gish (1996) highlighted the main problem: SVM become inefficient when the number of training frames is large. Besides, Vincent (2003) declare that SVM in speaker recognition need a normalization method to transform the signal in to fixed length due to SVM only can process fixed-length input, whereas speech signals are non-stationary. Therefore, SVM need to categorize the feature and scaling them before processing. It causes the heavy load for computation time.

Among these pattern classification approach, the use of GMM are most common due to it can be performed in a completely text independent situation. Besides, GMM are base on probabilistic framework, it provide high-accuracy recognition (Bruneaua et al., 2009). It becomes a dominant character for speaker identification techniques. Recently, many researchers pay their attention to investigate the use of GMM in speech processing in order to get better result. However, for speaker identification task, each speaker data is modeled by a GMM, and during testing phase, each GMM is calculated independently to estimate the parameters and compare with all other GMM to find the best match score. It sounds efficient and useful to speaker identification application, but in practice, it result was causing drawback of computational time. Alternative methods must be

sought in order to reduce computational time.

2.6 Comparison on Several Pattern Classification Approaches

The last few sections discussed several approaches in pattern classification. The following Table 2.1shows the comparison are made between those approaches when apply in speaker recognition.

Table 2.1: Comparison on several pattern classification approaches

Year	Researcher	Approach	Advantages/ Disadvantages
1971	Sakoe and Chiba	Dynamic Time Warping - algorithm for measuring similarity between two sequences which may vary in time or speed.	Associated with small fixed-vocabulary speech recognition, not suitable for huge data processing.
1985	Soong et al	Vector Quantization - process of mapping vectors from a large vector space to a finite number of regions.	Work fast in short-time acoustic features but achieve low accuracy rate. Hard to handle large scale of data.
1989	Rabiner	Hidden Markov Models - system being modeled is assumed to be a Markov process with unknown parameters.	The number of parameters that need to be set in an HMM is huge, only work at text

			dependent environment.
1992	Reynolds	Gaussian Mixture Models - using maximum likelihood score to match similar speaker.	Perform well but training times require are time consuming. Provide high-accuracy recognition.
1995	Bishop	Artificial Neural Network - computational model based on Biological neural networks	Training recurrent networks becomes increasingly difficult as the length of the sequences and the duration of temporal dependencies increases.
2003	Vincent and Steve	Support Vector Machines - based on the principle of structural risk minimization, consist of binary classifiers that maximize the margin between two classes.	Shown good promise as a basis for discriminative training, become inefficient when the number of training frames is large, only work on fixed length data, while speech signal is non-stationary.

2.7 Recent Work Progress on GMM in Speaker Identification

Over the past several years, GMM have become the dominant approach for modeling in text-independent speaker identification applications. This is evidenced by the numerous papers from various research sites published in major speech conferences such as the International Conference on Acoustics Speech and Signal Processing (ICASSP), the European Conference on Speech Communication and Technology (Eurospeech), and the International Conference on Spoken Language Processing (ICSLP), as well as articles in ESCA Transactions on Speech Communications and IEEE Transactions on Speech and Audio Processing (NIST, 2008).

Gaussian mixture models (GMM) has proved to be an effective probabilistic model for speaker identification, and has been widely used in most of state-of-the-art systems. Reynolds and Rose (1995) were first implemented it in speaker identification system and attains 96.8% identification accuracy. The Gaussian mixture speaker model is experimentally evaluated on a 49 speaker conversational speech database containing both clean and telephone speech. The experiments examine algorithmic issues such as model initialization, variance limiting, and model order selection. The experiments also examine the GMM speaker identification performance with respect to an increasing speaker population

Most interestingly, they found a contrast between VQ pattern classification, which is VQ are suitable for handle small range data where GMM shows it stability and reliability in handle huge speaker data set. In fact, Speaker identification was significantly improved as it elastic for real time system.

GMM-based systems applied to the annual NIST (National Institute of Standards and Technology) Speaker Recognition Evaluations (SREs) have consistently produced state-of-the-art performance (NIST, 1996). In particular, GMM-based system developed by MIT Lincoln Laboratory has been the basis of the top-performing systems since 1995 (Reynolds and Rose, 1995).

In years 2000, Reynolds et al. (2000) have built up a system around the likelihood ratio test for verification, using simple but effective GMM for likelihood functions, a universal background model (UBM) for alternative speaker representation, and a form of Bayesian adaptation to derive speaker models from the UBM. Later in years 2001, Miyajima et al. (2001b) presents a new approach to modeling speech spectra and pitch for text-independent speaker identification using Gaussian mixture models based on multi-space probability distribution (MSD-GMM). This improve GMM model performs well for speaker identification task by allows us to model continuous pitch values for voiced frames and discrete symbols representing unvoiced frames in a unified framework. Experimental results show that the MSD-GMM can efficiently model spectral and pitch features of each speaker and outperforms conventional speaker models in term of accuracy.

Subsequently, Sturim et al. (2002) from Reynolds research group further carried out their work on GMM in obtaining an approach to close the gap between text-dependent and text-independent speaker verification performance. They performed an automatic text-constrained GMM-UBM system. This system are created using word segmentations produced by a LVCSR (large vocabulary continuous speech recognition) system on conversational speech allowing the system to focus on speaker differences over a constrained set of acoustic units. Results on the 2001 NIST extended data task show this approach can be used to produce an equal error rate of <1%.

Similarly, Honga and Kwong (2005) improve baseline GMM by propose a discriminative training approach on it. They point out that the estimation of model parameters is generally performed based on the maximum likelihood (ML) criterion. However, this criterion only utilizes the labeled utterances for each speaker model and very likely leads to a local optimization solution. To solve this problem, they propose a discriminative training approach based on the maximum model distance (MMD) criterion.

Most of the above enhancements are focus on improving accuracy rates on several of situation for speaker verification or identification task. There is no doubt that most of them bring a great improvement over GMM approach and lead it in a more maturity pattern classification techniques. However, since GMM are based on statistical model which it perform a comparison of all log likelihood score for each input speaker test. This incurrence heavy computational time when speaker's database become large. As a consequence, a series of speed up GMM model was introduce in the speaker recognition task.

Younjeong et al. (2006) propose the method that estimates the optimal number of Gaussian mixtures based on incremental k-means for speaker identification. Over this method, the initialization with the optimal number of mixtures is done by adding dynamically the number of mixtures one by one until the mutual relationship between any two mixtures becomes dependent. The experimental results show that the proposed method is an effective and fast algorithm to find accurate parameters with obtaining the optimal number of mixtures for GMM.

Besides, Zhenyu et al. (2006) have suggested propose a tree-based kernel selection (TBKS) algorithm as a computationally efficient approach to the Gaussian

mixture model–universal background model (GMM–UBM) based speaker identification. As a result of this TBKS process, computational complexity can be significantly reduced. They improve the efficiency of the proposed system further by applying a previously proposed observation reordering based pruning (ORBP) to screen out unlikely candidate speakers. By integrating TBKS and ORBP together they speed up the computational efficiency by a factor of 15.8 with only a very slight degradation of identification performance, i.e., an increase of 1% of relative error rate, compared with a baseline GMM–UBM system.

Overall, they are tremendous amelioration research in investigate better GMM techniques in speaker recognition since the year 1995 until now. By far, the accuracy rates of the GMM pattern classifier have great improvement. However, there are still lacks of focus in speed up the computational efficiency for GMM classifier. The current we have by Zhenyu et al. (2006) and Younjeong et al. (2006) have prove their efficiency in computational times, but the accuracy rate are not guarantee. Pattern classification engine for speaker recognition should capable to manage and process huge speaker data sets in a short time limit and obtain high accuracy rate. Therefore, there are still more room for improvement on it.

2.8 Recent Work on Hybrid Modeling

Many research efforts have been done in the speaker recognition pattern classification techniques adaptation in the past. The scope of discussion here will only focus on text-independent environment since this research is based on text-independent scheme. Basically current research are divided into two direction

which the first is aimed to improve the accuracy rate of the identification, while the second is aim to reduce the processing time.

Several attempts have been made to improve the accuracy rate. Minghui et al. (2006) use hybrid GMM and SVM as a pattern classifier. They adapt the GMM with Universal Background Model (UBM) for feature extraction and the SVM work as pattern recognizer in text-independent speaker verification. The UBM is a large GMM trained to represent the speaker-independent distribution of features. From their research, they found that extracting features which cover not only centroids but also variances for SVM can bring a significant improvement in accuracy rate because of GMM clustering face the data densities neglected problem. Therefore, they use adapted GMM to extract a small quantity of typical feature vectors from large numbers of speech data and later on they bring the data to pattern classification process which use SVM as classifier.

Besides, Fine et al. (2001) describe a novel scheme which employs an SVM classifier as an “advisor” to the GMM classifier in uncertain cases. The utility of the combined generative/discriminative approach is demonstrated on standard text-independent speaker verification and speaker identification tasks in matched and mismatched training and test conditions. Instead of combining the scores from the GMM and SVM classifier, they use SVM advice on GMM confusions. The utility of the advisory system is presented by the relative improvement in identification rate over the baseline system. Most interestingly, they gained 7-10% improvement over the identification rates.

Fenglei and Bingxi (2003) use the output of the Gaussian mixture model to adjust the probabilistic output of the support vector machine. They Unifies the

GMM logarithm according to the attribute ability with the SVM logarithm and the separating strong capacity characteristic, carries on the adjustment, introduces GMM to the SVM output to realize probability output of SVM. As the result, the rate of the system which established by SVM-GMM mixture model are higher than traditional SVM model.

To reduce the processing time, recent evidence suggests that combining data compression techniques with statistical method or machine learning method. According to Osuna and Girosi (2005), data compression method did not use all the feature vector for training, these method like VQ, just made a prediction on the feature vector and the training are just run on selective feature data. Therefore, system which is applying data compression techniques such as VQ is using less computational time if compare with system which applying statistical or machine learning method (Guangyu and Micheal, 2005).

Wang et al. (2005) combines VQ K-means clustering and SVM to speed up the real-time learning whereas Scholkopf and smola (2002) discussed the combination between VQ and SVM in their book. Temko et al. (2007) apply VQ to score a fast algorithm of data reduction based on proximal SVM. The result shows applying VQ as pre-processing method for training data have successfully reduce the time consuming issue for most of the conventional pattern classification method.

2.9 Summary

This chapter reviewed several pattern classification techniques for speaker identification and the brief about speaker identification environment. The literature review shows that recent development on GMM for speaker identification is still insufficient to provide a satisfying result in achieving accuracy while maintaining the time processing. Later on, some studies on GMM enhancement towards dominance technique for speaker identification are discussed. Then, the next topic looks forward into some research trend in hybrid pattern classification modeling. Coming chapter will discuss on the incentive force of hybrid VQ/GMM model and review some hybrid VQ/GMM model adaptation in the past few years. Finally, the whole research methodology is being presented.

CHAPTER 3

RESEARCH METHODOLOGY

The previous chapter of literature review provides a knowledge foundation to support and motivate the research study. This chapter focuses on the operational framework and the design methodology that were carried out to achieve the objectives of this study. The scope of discussion here only covers the design of the hybrid modeling for pattern classification part.

3.1 The Incentive of Hybrid VQ/GMM Model

As discussed earlier in chapter 1, this research intends to reduce the processing time of conventional GMM approach and maintain the accuracy rate under incremental amounts of training data. Based on the previous research works (see chapter 2), it is found that GMM is the dominance of the pattern classification techniques used for text-independent speaker identification. It provides high

accuracy rates and able to manage huge speaker data set while maintaining its stability of classification. Besides, GMM works well on text-independent environment. GMM, however, requires longer computational time to operate.

Recently, many researchers pay their attention to investigate the use of GMM in speech processing in order to get better result. However, for speaker identification task, each speaker data is modeled by a GMM, and during testing phase, each GMM is calculated independently to estimate the parameters and compare with all other GMMs to find the best match score (Mayajima et al., 2001b). Mayajima et al. (2001a) analyzed the data using GMM method. They concluded that speaker identification system using conventional GMM method was time consuming, mainly due to its need to compare all the data in the speaker database with statistical calculation. To reduce the issue of computational time, Mayajima et al. (2001b) in their research work have proposed the implementation of multi-space probability distribution on GMM model. The proposed method was used as pre-processing process for training data and distributed data followed by their group. Similarity, Tomi et al. (2009) emphasized the need of finding alternative solution for GMM time consuming issue. Tomi and his research members have suggested that a hybrid model using VQ and GMM method is able to solve the above mention problem.

A summary is drawn based on the previous findings, there is still a demand for constructing a better GMM based pattern classification techniques with fast computation for large database and high accuracy over incremental data. At the moment, in chapter 2.8, some analyses of the hybrid modeling are done. Previous studies show that hybrid GMM can be successfully carried out to maintain accuracy rates (Fenglei and Bingxi, 2003; Minghui et al., 2006). According to Minghui et al. (2006), GMM method achieved the most accurate result for speaker identification

system because of its applying statistical calculation to compare all speakers data instead of selective apart of possible speakers data. On the other hand, some researchers who work on reducing computational times will apply VQ method (Scholkopf and Smola, 2002; Temko et al.,2007). After analysis by experimental results, they claimed that VQ approach is not guaranteed to find the best set of clusters but in practice it works very fast.

The findings above support motivation of this research to construct a hybrid VQ/GMM model for text-independent speaker identification which can maintain the accuracy rates and reduce the processing times. According to Scholkopf and Smola (2002), VQ method worked very well as pre-classifier to select a small range of data with short time limit for the classification machine. Besides, Temko et al. (2007) noted that although VQ accuracy is relatively poor compared with other methods, it is still suitable for a pre-classifier engine.

In proposed modeling, GMM model remains the main classifier and VQ model acts as an advisor to make an advance for the classification result. The results of the research show that this hybrid VQ/GMM can achieve fast computation. In conclusion, it was decided that the best design for adaptation was using VQ as a pre-classifier to select a group of possible model for GMM pattern matching.

3.2 Recent Works on Hybrid VQ/GMM Modeling

Before further discussing the proposed hybrid VQ/GMM model, reviews have been done on some recent works regarding the hybrid VQ/GMM model. There are many forms of GMM and other pattern classification techniques adaptation in the past and yet there are scantiness amount for VQ/GMM adaptation. In hybrid VQ/GMM, there are some researchers used VQ as optimization function to reduce the Expectation Maximization algorithm in order to improve the training speed. Besides, some researchers employed GMM as a post-processor after VQ cluster the speech signal into regions.

Qiguang et al. (1996) were the pioneers who came out with the idea of hybrid VQ/GMM in speaker recognition field. Conventional GMM generates a Gaussian mixture model for each enrolled speaker. The model statistics is estimated using acoustic features covering the entire acoustic space. They argued that the statistics can be better estimated by first clustering (vector-quantizing) the acoustic space into several subspaces. Each subspace is then represented by a number of Gaussian mixture models whose parameters are determined using only those relevant acoustic features belonging to the subspace. They therefore recommended vector-quantization based Gaussian mixture models (VQGMM) and the system has recently been used in 1996 NIST Speaker Identification Evaluation. From the official evaluation results, the system generally produces top scores among all the participating sites. For some test subsets (short utterances), the VQGMM system yields the best scores.

In 2000, Pelecanos et al. (2000) proposed a method to include the contribution of adjacent regions but using computationally efficient single Gaussian components

to establish the model. Here Vector Quantization is used to separate speech vector into their corresponding regions and a single multi-dimensional Gaussian is calculated for each. A substitute GMM is formed that consider density contribution information from adjacent regions by compiling information available from the mixtures and the number of point in each region. They declared the use of their suggested model named Vector Quantization Gaussian Modeling (VQG) is able to provide a rapid means for training to form a reliable speaker verification system. As a result, they successfully reduced 20% for the period of GMM training.

Gurmeet et al. (2003) introduced the use of Vector Quantization algorithm, namely Linde, Buzo, Gray (LBG) algorithm for training Gaussian mixture speaker models as a replacement for Expectation Maximization (EM) algorithm to reduce computational complexity. EM algorithm normally used for GMM training to find a local maximum value. However, if the speaker data become too large, it faces the time consuming problem. This is because EM algorithm is an optimization algorithm. In order to obtain a global maximization, EM should run many times from varied starting points (Hong et al., 2004). Therefore, Gurmeet et al. (2003) replaced the EM algorithm with LBG algorithm which is less calculation. From experiment, they found that by replacing the LBG algorithm, the complexity of calculation can be reduced by 50% if compared to the EM algorithm which is original GMM function. The reason is LBG algorithm is a data compression technique and it just utilize apart of feature vectors for calculation of the classification. Gurmeet et al. (2003) have successfully proved that they deliver comparable performance as the EM algorithm and significantly reduced computational complexity.

In 2006, Marie (2006) suggested the use of VQ as pre-classifier which generates a set of possible result using a novel application of Gaussian selection, and

a transformation of the traditional tail test statistic which lets the implementer specify the tail region in terms of probability. The system is trained using parameters of individual speaker models and does not require the original feature vectors, even when enrolling new speakers or adapting existing ones. As the correct class label need only be in the possible result set, it is possible to prune more Gaussians than in a traditional Gaussian selection application. The possible result set is then evaluated using individual speaker models, resulting in an overall reduction of workload. The use of the Gaussian-selection-based pre-classifier leads to an overall reduction of system complexity. For the non-UBM trials, the pre-classifier performed reasonably well with the confidence interval of 95 % and showing speedups of about 4 times.

Among the four hybrid VQ/GMM models that have been discussed earlier, Qiguang et al. (1996) focused on the accuracy rates and Pelecanos et al. (2000) focused on reducing the training time when enroll a new user. Gurmeet et al. (2003) focused on minimizing the complexity of the EM algorithm in order to increase the speed of the identification process. Marie (2006) on the other hand proposed a model which is quite similar with proposed research mentioned in this thesis (using VQ as pre-classifier). Besides, she shows about 4 times speedups for the identification process. In summary, the research experiment will discuss on the differences between the model proposed by Marie (2006) and model proposed by Gurmeet et al. (2003) since both focus on using VQ/GMM modeling for reducing time processing for speaker recognition.

3.2.1 Constrain of Hybrid VQ/GMM Modeling

Previous studies have derived the idea of constructing a hybrid modeling using VQ as pre-classifier to find out a set of initial speaker model to use as input of GMM classifier. However, there is a difficulty in this hybrid modeling and it's reflected the accuracy of the identification rate. After the first stage of the proposed method, it can efficiently decrease the processing time, but the accuracy rate cannot be guarantee. This is due to VQ can only perform well in small range of data (Jialong, 1999). VQ is based on short-time acoustic feature. When the feature set become large, it becomes hard to compute the distance between codebook, resulting in inaccurate results (Karpov et al., 2004). Consequently, alternative methods must be sought in order to reduce training data for VQ approach, which is also the second focus of this research.

3.2.2 Review of VQ Problem Solutions

Based on the studies on constrain of hybrid VQ/GMM modeling, a set of solution that suggested by other researchers to solve VQ problem have been taken into consideration for this research. In VQ system, reducing the spectral distortion to 1 dB requires a large codebook, which leads to intractable complexity. Besides, because of the large codebook, it also results in less accuracy in identification result (Hautamäki et al., 2008). By adding suitable structure to the codebook, both of the memory requirements and computational complexity can be reduced significantly (Phamdo et al., 1991). Two attractive structures are multistage codebooks and split codebooks (Paliwal and Atal, 1991). The performance of a multi-stage VQ can be

improved using a multi layer tree search procedure.

The first distributed train idea was carried by Nakai et al. (1992). They claim that by dividing training vectors into small groups will lead a fast computation process. Nakai et al. (1992) pointed out that the LBG algorithm requires a lot of computation as the training vectors increase, and proposed a fast VQ algorithm for a large amount of training data. This algorithm consists of three steps: first, divide training vectors into small groups; second, quantize each group into a few code words by the LBG algorithm; finally, construct a codebook by clustering these code words using the LBG algorithm again. They also reported that the distortion error of the algorithm can be reduced by adapting an effective data-dividing method. In experiments of quantizing 17500 training vectors into 512 code words, this algorithm requires only 1/6 computation time compared with the conventional algorithm, while the increase of distortion is only 0.5 dB.

Acero et al. (1996) found that a significant reduction in memory in speech processing using Sub-partitioned vector quantization techniques. Whereas, Guangyu and Michael (2005) suggested an Adaptive Discriminative VQ technique to divide feature vector space into subspace before training VQ approach. In 2007, Wan et al. (2007) proposed a multi-band 2-stage vector quantization (VQ) as the recognition model. From analysis done, it leads to the distributed train approach to be utilized in the proposed hybrid VQ/GMM solution.

3.2.3 Distributed VQ Training

As discussed earlier in chapter 2, VQ technique clusters the signal based on short-time acoustic feature. It faces difficulty when handling a large feature set due to VQ technique run nearest neighbour search to find the centroid in the current codebook that is closest. These nearest neighbour search are based on estimation. Therefore, if the feature data sets are too large, the estimation location for centroid by nearest neighbour search will be far from veracity (Karpov et al., 2004).

In view of this, this research has suggested to adapt the conventional VQ clustering method to a novel distributed VQ clustering method. This novel method aims to separate the huge group of speaker data into some smaller subgroups for VQ training. This research attempts to distribute speaker data according to the speaker attribute by decision tree. The measure unit using for decision tree are speaker's pitch. Decision trees are commonly used in operation research, specifically in decision analysis, to help identify a strategy in order to reach a goal. Through this tree model, it helps making decisions on each input speaker and its subgroup based on their personal attribute (gender and pitch). Consequently, the VQ clustering method only needs to train and test the speaker data inside the subgroup instead of the whole dataset.

3.3 Operational Framework of Research Model

This research will focus on pattern classification area for speaker identification. Hybrid distributed train VQ/GMM model is chosen as the subject of improvement. The decision tree approach has been applied for the purpose of distributing data to reduce number of training for VQ pre-classifier. The initial results gained from VQ pre-classifier will be tested on GMM approach. The framework of the hybrid distributed VQ/GMM model is shown in Figure 3.1.

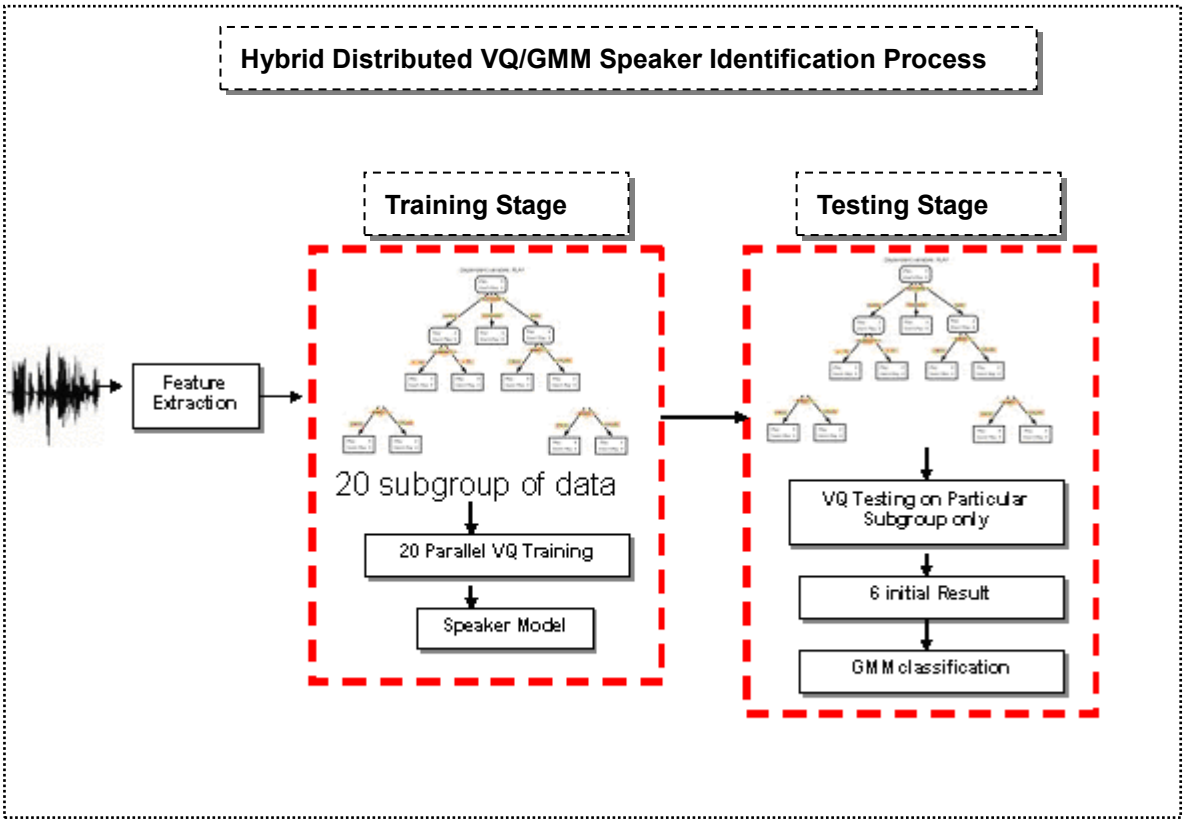


Figure 3.1 Hybrid distributed VQ/GMM speaker identification process

Figure 3.1 shows hybrid distributed VQ/GMM speaker identification process in general. Proposed model utilized decision tree approach to classify the data into

smaller subgroup depends on its personal attribute. The research hypothesis claims that training data in smaller subgroup will improve the time efficiency. Thus, a decision tree function which separates the speech signal according to their gender information via pitch is added as pre-processing for hybrid VQ/GMM approach.

In the research, the design of decision tree function use to distributed train data for VQ technique. These bring VQ only clustering data at small amount instead of the whole dataset. The distributed train idea for VQ have successfully solved the VQ constrains as VQ technique is only able to work at small range of data. Besides, through this distributed train idea, it can fix identification errors in huge database by separate out confusable speakers prior and training under different codebook. Finally, with distributed VQ pre-classifier, a set of initial result have been estimated and these results were tested by GMM classification to achieve final identification result.

The following are the process flow for speaker identification system using hybrid distributed VQ/GMM model.

(i) During training phase, speech signal will transform to a set feature vectors via feature extraction process.

(ii) Each set of feature data convey speaker's information like gender and pitch tone. Decision tree is using gender and pitch analysis result as unit of measure to distributing speaker data into 20 subgroups.

(iii) VQ clustering training speaker data in particular subgroup only. Speaker data in other subgroups will be trained by another VQ model.

(iv) When training process is completed, all speaker data will save as speaker model to store at database.

(v) During testing, decision tree will analyze speaker data and assign this speaker data into a rational subgroup.

(vi) VQ will run to the nearest neighbour search to estimate a set of possible speaker identity.

(vii) The initial result will then be finalized by GMM classification model.

3.4 Design Methodology

There are several issues have to be considered before constructing the hybrid modeling:

(i) Adequate dataset used to train and test.

(ii) Level of decision tree function that required.

(iii) Number of the distributed subgroup.

(iv) Amount of the initial result estimated from VQ

Figure 3.2 shows the methodology developed in this study using hybrid distributed VQ/GMM model. The methodology is divided into following phases: pre-processing phase, distributed VQ training phase, distributed VQ classification phase and GMM identification phase.

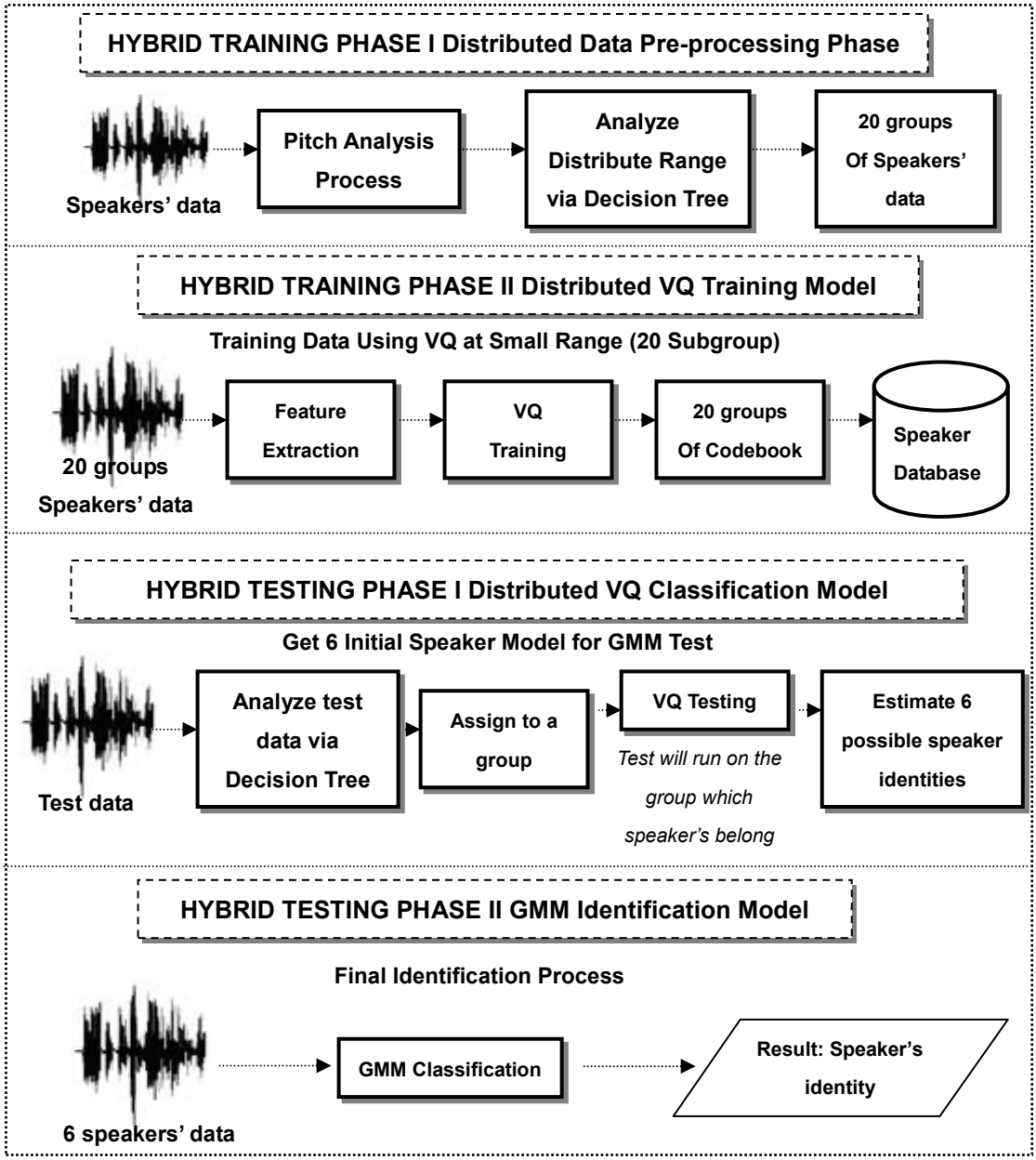


Figure 3.2 Design methodology for hybrid distributed VQ/GMM model

3.4.1 Data Collection (Speech Corpus)

The dataset used must be sufficient in order to fulfill the research objective, which is able to handle large datasets. There are several speech corpuses available for speaker identification. Among them are TIMIT, YOHO and KING. In order to standardize this research outcome and benchmark with other existing attempts, the dataset chosen for testing should be familiar.

The first issue should be considering here is this research requires a large dataset. The second issue is this dataset must contain male and female speaker as one of the research objective is to distribute the speaker into smaller range by using decision tree. Gender and pitch are the unit of measure for decision tree approach.

The KING corpus was created for research in the area of speaker identification. It was collected partly in New Jersey and partly in San Diego in 1987. There are twenty-six San Diego speakers and twenty-five New Jersey speakers with all speakers are male. There are ten sessions for each speaker, and each session was recorded in both a wide-band (wb) and a narrow-band (nb) channel.

The YOHO Speaker Verification Corpus supports development, training and testing of speaker verification systems that use limited vocabulary. The particular vocabulary employed in this collection consists of two-digit numbers ("thirty-four", "sixty-one", etc), spoken continuously in sets of three (e.g. "36-45-89"). There are 138 speakers (108 male, 30 female); for each speaker, there are 4 enrollment sessions of 24 utterances each, and 10 verification sessions of four utterances each, for a total of 136 utterances in 14 sessions per speaker (Higgins et al., 1992).

Whereas the TIMIT corpus has been designed to provide speech data for the acquisition of acoustic-phonetic knowledge and for the development and evaluation of automatic speech recognition systems. However, it is possible to use TIMIT in speaker identification research because it contains 10 utterances of difference sentences for each speaker. TIMIT has a total of 6300 sentences, 10 sentences spoken by each of 630 speakers from 8 major dialect regions of the United States. There are 438 male data and 192 female data in TIMIT corpus. Table 3.1 shows the number of speakers for the 8 dialect regions, broken down by sex (Louis et al., 1996).

Table 3.1: Dialect distribution of speakers

Dialect Region(dr)	#Male	#Female	Total
1	31(63%)	18(27%)	49(8%)
2	71(70%)	31(30%)	102(16%)
3	79(67%)	23(23%)	102(16%)
4	69(69%)	31(31%)	100(16%)
5	62(63%)	36(37%)	98(16%)
6	30(65%)	16(35%)	467(%)
7	74(74%)	26(26%)	100(16%)
8	22(67%)	11(33%)	33(5%)
Total :	438(70%)	192(30%)	630(100%)

After analysis, it is found that both KING and YOHO corpus are not suitable for this research. This is because KING corpus only contain male speaker whereas YOHO corpus are using limited vocabulary which could not support text independent speaker identification. The dataset used in all the experiments in this study is the TIMIT corpus. This corpus is chosen as it fulfils two issues that have been discussed above which is huge dataset and contain male and female speaker.

Furthermore, TIMIT work in text independent environment.

3.4.2 Distributed Data Pre-processing Phase

Speakers' data are distributed in this pre-processing stage using decision tree approach. Decision tree analysis is a formal, structured approach which eases the knowledge-acquisition for decision making. Decision trees can decompose a complex problem into smaller and more manageable undertakings which allowing the decision makers to make smaller determinations along the way to achieve optimal overall decisions (Almuallim et al., 2002).

A decision tree algorithm is used to construct a decision tree classifier for determining an appropriate class (among a predetermined set of classes) for a given rule to be tested (Deboys, 2004). According to Deboys (2004), the standard approach for the induction of rules involves dividing the data into two sets, then performing training on the first set, and finally testing the induced knowledge on the second set. One can repeat this process a number of times with different splits, then calculate the average of the obtained results to estimate the rules' performance on possible new test data. In the proposed modeling, decision tree is used to separate full set of TIMIT corpus into some smaller subgroups. Therefore, there are several issues need to be determined before using decision tree to distribute data. These include:

(i) Rules of decision tree.

(ii) Amount of the distributed subgroup.

(iii) Level of decision tree.

3.4.2.1 Rules of Decision Tree – Pitch Analysis

The rules of decision tree are determined by pitch conveyed by speakers' data. Pitch represents the perceived fundamental frequency of a sound. Nearly all information in speech is in the range 200 Hz to 8 kHz. Humans discriminate voices between males and females according to the frequency (Vergin et al., 1996). Females speak with higher fundamental frequencies than males. The frequency for adult male is ranged from 50 Hz to 250 Hz, with an average value of 120Hz. For an adult female, the upper limit of the frequency range is much high, as high as 500 Hz. Therefore, by analyzing the average pitch of the speech samples, algorithm for a gender classifier is derived (Childers and Ke, 1991).

Pitch is defined as the fundamental frequency of the excitation source. Hence an efficient pitch extractor and an accurate pitch estimate calculated can be used in an algorithm for gender identification (Marston, 1995; Gold and Rabiner, 1969).

3.4.2.2 Subgroup Data Distribution

In the proposed research, decision tree is utilized to distribute speaker data into smaller subgroups. The information about amount of the subgroups is the primary key to set rules and design decision tree level. However, the ability of VQ will be the main concern on this issue. This is because VQ technique faces difficulty when handling large feature sets and decision tree is the pre-processing to

separate these large feature sets into smaller subgroups.

Vector Quantization is a mapping of a large to smaller set of values based on partitioning concept. VQ model maps 2-dimensional vectors in the vector space R^k into a finite set of vectors *Y*. *K* represents iteration done to find the location of centroid. Each centroid represents a code vector, and the set of all code vectors is called a codebook (Chatterjee et al., 2008).

Iteration procedure is recursively performed until the ratio of two consecutive average distortions between the improved code words and training vectors is less than a certain threshold value. LBG algorithm is the VQ algorithm using 2-dimensional vector. It should be noted that the LBG-VQ design algorithm is iterative and it requires an initial codebook *C (0)*. This initial codebook is obtained by the splitting method in which an initial code vector is set as the average of the entire training sequence. This code vector is then split into two. The iterative algorithm is run with these two vectors as the initial codebook. The final two code vectors are split into four and the process is repeated until the desired number of code vectors is obtained (Shen et al., 2003).

As a general rule, VQ that is used in practice has code vector of 32 or less, because complexity, memory, and performance tradeoffs are generally most attractive in this range. That means that iteration is equal to 5 (2^5=32) (Chatterjee et al., 2008). So and Paliwal (2007) have been inspired by a review from Mohammad and Mark (2005) regarding the codeword/centroid adjustment for large VQ. Mohammad and Mark (2005) experienced that the more iteration conducted, the worst the clustering result. This is because the code words for each iteration 'move' through contiguous regions based on estimation. This implies that if there are bad

initialization centroids, it could lead to the impossibility of finding accurate speaker data.

In addition, analysis was carried out by Chen (2004) to review how many partitions should be separated in order to get the ideal identification rates as VQ only suitable work on small range of data. From the analysis of the author, from the first iteration until 10^{th} iteration, among the best data obtain is iteration from range 1-5. Starting from 6^{th} iteration, some locations of the centroid are hard to predict.

The above mention study was designed to determine the amount of the subgroup which distributed by Decision Tree. This is an important measure to construct decision tree. From analysis above, the best range to design VQ code book was in 5^{th} iteration and the code vector should be around 32 or less. In classification of speaker identification, codebook that contain 32 code vectors are equal to 32 set speakers data in a group which is used to train and test. In order to obtain this best set, an analysis is done on TIMIT database, which contain 630 data. Following is the equation to achieve the amount of how many subgroups have to distribute:

$$Ceil\ (N/d) = Z \tag{3.1}$$

where N is the sum of speaker data in data set and d is the best range of code vector. From the calculation, the amount of the subgroups (Z) are decided to be 20 for each subgroup contains 32 or less speakers' data, which is the best range for VQ classifier.

3.4.2.3 Level of Decision Tree

In the case of determining the appropriate level(s) to be assigned to each search profile, the rules are important because they will decide how many levels in the decision tree (Elif and Michele, 2008). Rules are made up of two major parts: a first part that comes between the operators '*IF*' and '*THEN*', expressing a condition, and a second part which comes after the operator '*THEN*' defining a class that, in this proposed research, corresponds to one or more levels of detail. Figure 3.3 presents the rules for the decision tree in this research.

Analyze TIMIT dataset, WHILE Not end of TIMIT dataset

IF Pitch tone = Male

THEN collect all male data, divide into 2 subgroups (M1, M2)

IF M1> 32 (the best range for VQ vector)

THEN Calculate N= Ceil (M1/32)

Divide M1 into N group

IF M2> 32 (the best range for VQ vector)

THEN Calculate N= Ceil (M2/32)

Divide M1 into N group

IF Pitch tone = Female

THEN collect all female data, divide into 2 subgroups (F1, F2)

IF F1> 32 (the best range for VQ vector)

THEN Calculate N= Ceil (F1/32)

Divide F1 into N group

IF F2> 32 (the best range for VQ vector)

THEN Calculate N= Ceil (F2/32)

Divide F1 into N group

Figure 3.3 Rules for decision tree to distribute speaker dataset

As shown in figure 3.3, the decision tree in 1st level is used to decide the dataset categorized into male or female group. Second level of decision tree collects all data and divides into 2 subgroups for each male and female group. In 3rd level, the sum of each subgroup, for example *M1* is used to apply in division of 32, which is the amount of best range for vector code books. Finally, the result shows that *M1* should divide into *N* group. In conclusion, the levels of the decision tree are mainly depending on decision rules set. In the proposed research, based on the rules for distributed data; the levels of decision tree are 3. However, the level can be more depending on the restructure of the rules (Deboys, 2004).

3.4.2.4 Process Flow of Data Distribution

The next step after design decision tree is the process flow of data distribution. As shown in Figure 3.2, all speakers' data will pass through a process named pitch analysis during distributed data pre-processing phase. This process aims to analyze the pitch frequency conveyed by speech data. These pitch frequency are used as rules for decision tree for distributing speaker data in to smaller subgroups, dividing it into 20 groups of speakers' data

3.4.3 Distributed VQ Training Model

The second process after distributing data is VQ training. Figure 3.2 shows the process flow of this distributed VQ training model. VQ training provides training for all speaker data in all subgroup. VQ training aims to save all speaker data into speaker database in order to compare with test data during testing process.

During training process, VQ codebook will represent the speaker feature from the training data. The speaker identification engines are dependent on the codebook to identify a speaker. In VQ training phase, Vector Quantization is executed using feature data as input. It is followed by the speaker identification engine which will run the nearest-neighbour search to find the codeword in the current codebook, causing vector to the corresponding cell. The result of clustering is a set of M vector, $C = \{c_1, c_2, \ldots c_m\}$, called a codebook of the speaker. Then, it finds centroids and update for each speech signal and the codebooks are created.

In proposed model, there were 20 groups of codebook stored as speaker model due to the implementation of distributed train. The VQ approach used to train and test speaker data was LBG algorithm.

3.4.3.1 LBG Algorithm

LBG algorithm was proposed and extensively studied using the k-means clustering approach. It also referred as Generalized Lloyd algorithm (Nakai et al., 1992). It should be noted that the LBG-VQ design algorithm is iterative and requires an initial codebook *C (0)*. This initial codebook is obtained by the splitting method in which an initial code vector is set as the average of the entire training sequence. This code vector is then split into two. The iterative algorithm is run with these two vectors as the initial codebook. The final two code vectors are split into four and the process is repeated until the desired number of code vectors is obtained (Lupini and Cuperman, 1995; Haber and Seidel, 2000). The algorithm designed for this study to minimize the codebook production time can be summarized in Figure 3.4.

Intuitively, the LBG algorithm designs an *M*-vector codebook in stages. It starts first by designing a 1-vector codebook, then uses a splitting technique on the code words to initialize the search for a 2-vector codebook, and continues the splitting process until the desired *M*-vector codebook is obtained. Figure 3.4 shows the detailed steps of the LBG algorithm. "*Cluster vectors*" is the nearest-neighbor search procedure which assigns each training vector to a cluster associated with the closest codeword. "*Find centroids*" is the centroid update procedure. "*Compute D (distortion)*" sums the distances of all training vectors in the nearest-neighbor search so as to determine whether the procedure has converged.

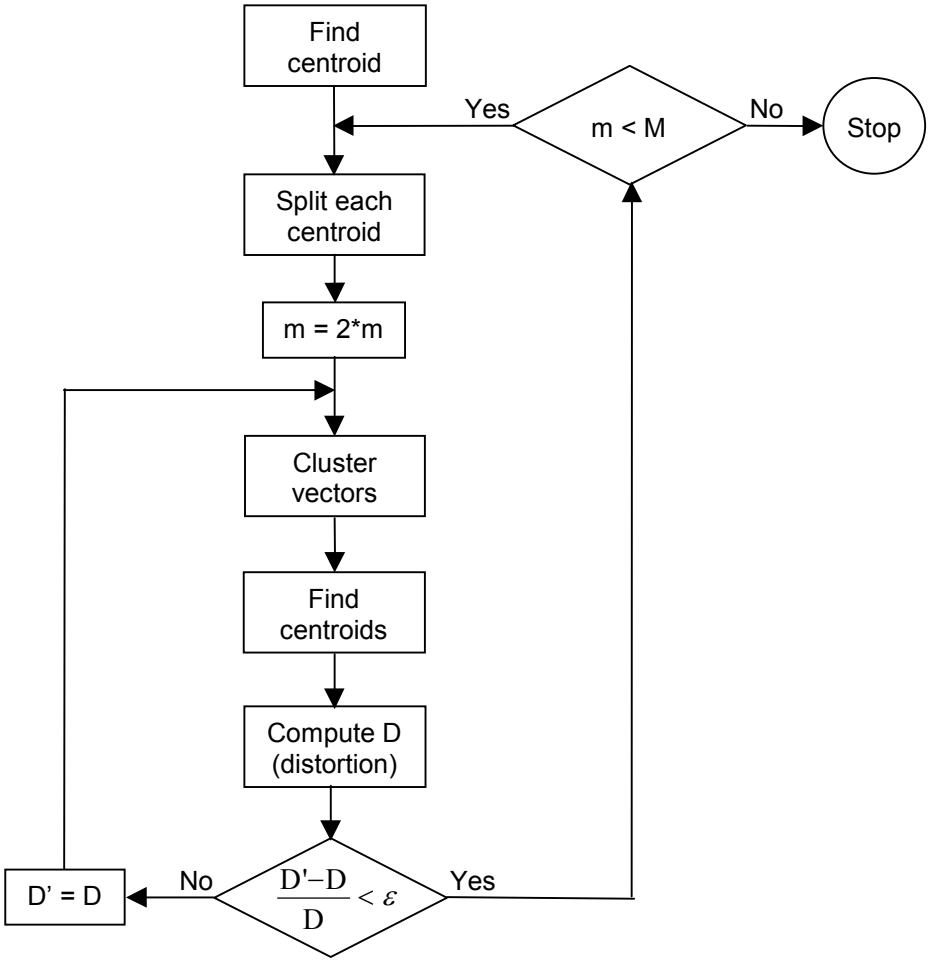


Figure 3.4 Flow diagram of the LBG algorithm (Adapted from Rabiner and Juang, 1993)

3.4.4 Applying Distributed VQ Clustering as Pre-classifier for GMM

VQ pattern classification has its own characteristic which is based on the data compression techniques. It defines a centroid for each speaker data and the calculation is only based on the particular centroid and it ignores other data convey by the speaker data (Elmisery et al., 2005). Figure 3.5 shows 3 selected centroids

for code vector representation.

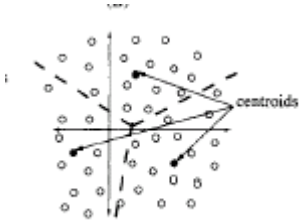


Figure 3.5 Centroids and its code vectors

In VQ testing process, a function will compute the Euclidean distance (minimum distance) between training data and testing data. The system will identify which calculation yields the lowest value and checks this value against a constraint threshold. If the value is lower than the threshold, the system outputs an answer. While on the contrary， the system will continue the iteration for searching lowest value to find the best data. The lowest value estimates are calculated from counts as follows:

$$\frac{D'-D}{D} < \varepsilon, \quad \varepsilon = 0.01 \tag{3.2}$$

where D' are distances of current training vectors and D represent previous vectors that hold lowest distance value. ε is the threshold for calculate Euclidean distance. In general, ε are selected from range 0.01 until 0.05. Based on the review of Soong et al. (1985), threshold with minimum value will lead to more complex calculation. But in fact, it brings better accuracy. Therefore, this study consider ε=0.01 as threshold.

The match score between the unknown speaker's feature vector $X = (x_1, \ldots . x_t)$ and a given codebook $C = \{c_1, c_2, \ldots c_m\}$, is computed as the average quantization distortion:

$$D_{\text{avg}}(X, C) = \frac{1}{T} \sum_{i=1}^{T} e(\boldsymbol{x}_i, C) \tag{3.3}$$

where $e(x_i, c) = \min_{c_j \in c} \| x_i - c_j \|^2$, and $\|x_i - c_j\|$ denotes the Euclidean norm (Linde et al.,1980).

For each speaker identification process, VQ will compare the training centroid with all other centroids. The decision will be made according to the nearest distance of testing model and training model (Tomi et al., 2006). In the case of VQ clustering data based on its centroids, the decision made has provided estimation for the classification result. This result cannot guarantee high accuracy simply because estimation will made on the most similar speaker (Chatterjee et al., 2008). Hence, VQ is not suitable as a final decision of pattern classification method for a hybrid modeling. Thus, this research utilizes VQ superiority to estimate a series of initial result and bring it to the final classification stage by using GMM approach to determine the speaker identity. The time of processing data will strictly reduce if compare with baseline GMM approach because proposed model only testing on initial estimation result rather than testing whole set of data in speaker database.

3.4.4.1 Data Selection for Initial Test Set

VQ classifier acts as pre-classifier for GMM model in proposed research. During testing phase, VQ calculates the distance between train data and test data. When comparison done, VQ will estimate 6 possible speaker identities.

These initial results will then be finalized by GMM classification to determine speaker's identity.

There have been some survey to derive the selection of these 6 initial test set. Chang et al. (2006) analyzed the process flow of LBG algorithm and concluded that LBG algorithm consider adjacent neighbors only in determining a codeword. According to Chang et al. (2006), VQ will produce a set of possible code vector which have similarity with the test data for classification. These possible code vectors were generated by nearest-neighbour search. However, because of LBG algorithm is consider adjacent neighbors only in determining a codeword, it results in the 1st and 2nd iteration process of finding nearest centriods be likely to match with test data. LBG algorithm is based on binary split when it is run nearest-neighbour search. When process to 1st iteration, there are 2 centriods determined. For the 2nd iteration, another 4 locations of the centriod will be gained (Li and Chan, 2004).

Mowlaee et al. (2008) agreed with Chang et al. (2006) findings by implementing this estimation on Split-VQ to achieve Monaural Sound Separation. Besides, Mowlaee et al. (2008) claimed that, the results of VQ methods are greatly affected by the estimation similarity of the codebook. Therefore, initialization of the codebook and its nearest centroids will become target to achieve final result. Similarity, Wan et al. (2008) applying Chang et al. (2006) theory on a research to predict possible speaker data based on two-stage Vector Quantization. Yet they have gained a great result when applying this estimation theory. Based on some reasonable findings, we decided to derive the selection of 6 initial test set from VQ clustering.

3.4.5 Speaker Identification using GMM Likelihood Ratio Algorithm

After VQ acts as pre-classifier to estimate a set of possible speaker data as initial result, GMM model is implemented to calculate the final result among 6 speakers' data. Gaussian mixture model (GMM) is a density estimator and is one of the most common classifiers in text- independent speaker identification. In this method, the distribution of the feature vector *x* is modeled clearly using a mixture of M Gaussians as stated above.

$$P(\mathbf{x}|M) = \sum_{i=1}^{m} a_i \frac{1}{(2\pi)^{D/2}|\Sigma_i|^{1/2}} \exp\left(-\frac{1}{2}(\mathbf{x}-\mu_i)^T \Sigma_i^{-1}(\mathbf{x}-\mu_i)\right) \quad (3.4)$$

The term of mu_i, Σ_i expressed in Eq. (3.4) represents the mean and covariance of the ith mixture. Given the training data , speaker represented *y* models $\lambda_1, \lambda_2 \ldots \lambda_n$, and the number of mixture *M*, the parameters μ_i, Σ_i, a_i is learnt using expectation maximization. During recognition, the input speech extracts a sequence of features $x_1, x_2 \ldots x_L$. The distance of the given sequence from the model is obtained by computing the log likelihood of given sequence given the data. The identification process is to fid the speaker model which has the maximum posteroir probility features $x_1, x_2 \ldots x_L$. A detailed discussion on applying GMM to speaker modeling can be found in the work of Reynolds (1995).

In a GMM speaker identification system, each speaker is represented by a mixture of means, variances and weights $\left(\lambda = \{w_i, \mu_i, \sum_i\}\right)$ where $0 \le w_i \le 1$ and the sum of the mixture weights equals to 1. To obtain the parameters for the individual speaker models the GMM needs to be trained. The training of the models is done by estimating of the parameters of the GMM from speech collected from each participating speaker. Maximum likelihood (ML) estimation is used to estimate the parameters. The aim of the ML estimation is to find the parameters

that maximize the likelihood of the GMM (Bilmes, 1998).

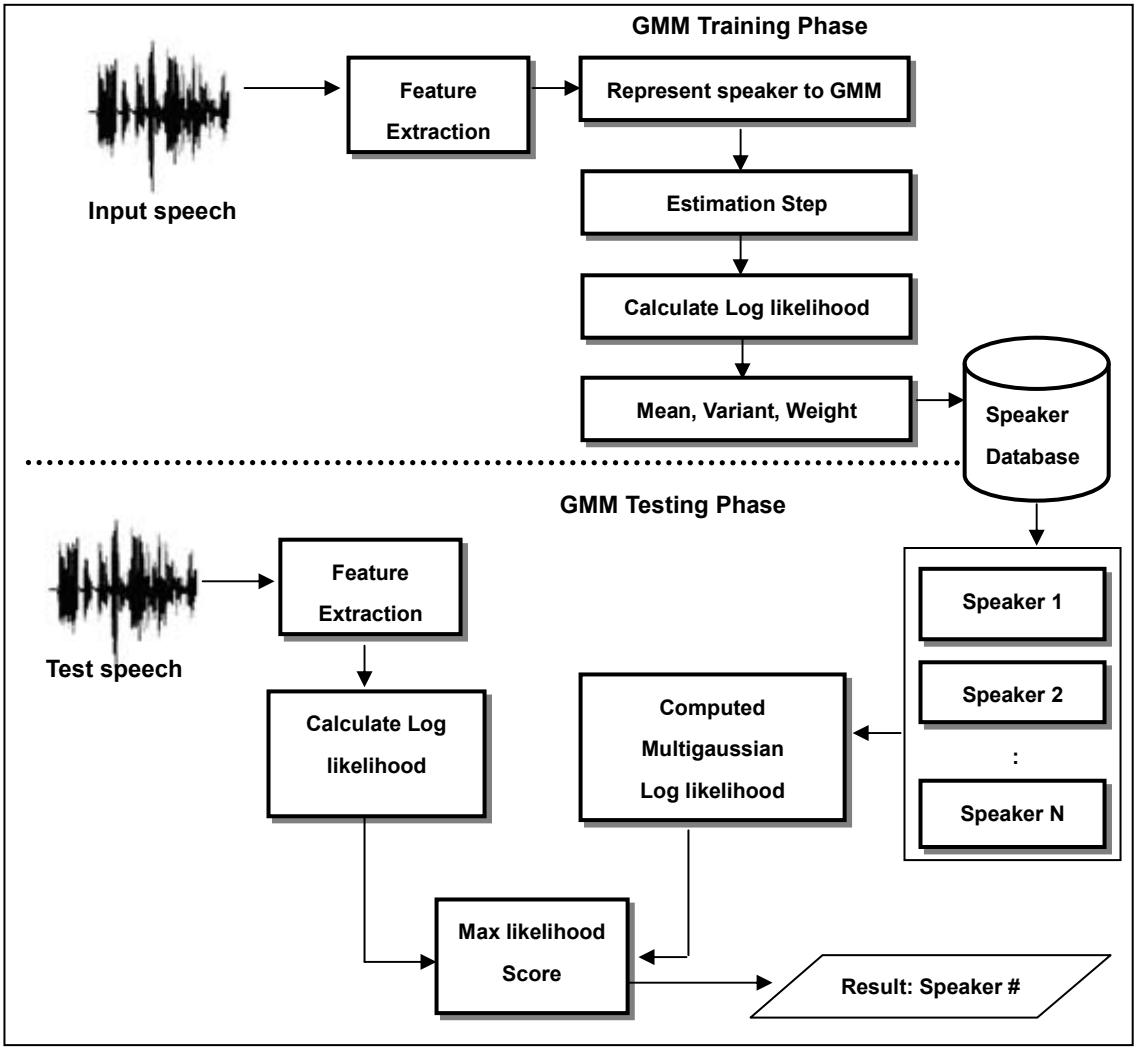


Figure 3.6 A process flow of GMM approach in training and testing phase for speaker identification system

Figure 3.6 shows a process flow for GMM approach in speaker identification training phase and testing phase. In GMM training phase, an MFCC output will return as GMM input after computing signal Mel-frequency cestrum coefficients. For speaker identification, each speaker is represented by a GMM and is referred to his/her speaker model. The Expectation-Maximization (EM) algorithm is applied in the estimation step. The EM algorithm starts with an initial parameter set of the

model and then calculates a new set of parameters. Then, the likelihood of the training data were generated by the current estimate of the model, defined by the new parameter set, is higher than the previous estimate. Then the new estimated model is used as the starting point of the next iteration. This process will be repeated until some convergence threshold is reached (Geoffrey and Thriyambakam, 1998). GMM classification engine will calculate log likelihood score for all training speaker data and save it into a speaker model. While in testing phase, a comparison about training speaker and testing speaker will be done. GMM classification engine will make a decision followed by maximum posteriori probability. The model with the highest likelihood score will be verified as the identity of the speaker.

3.5 Summary

This chapter discussed the needs of hybrid distributed VQ/GMM modeling and pointed out some constraints of hybrid VQ/GMM without distributed training data. The detailed design methodology and theoretical framework of this study will be described on the next chapter. It consists of following procedures: analyze the ability of the distributed VQ, design methodology, data collection and finally the output which is the hybrid distributed VQ/GMM design framework. Besides, reviews have been conducted based on the previous research works to determine the important parameters for constructing proposed hybrid framework. As a consequence, next chapter will further discuss on the implementation of the proposed hybrid distributed VQ/GMM modeling on speaker identification.

CHAPTER 4

HYBRID DISTRIBUTED VQ/GMM MODELING

Corresponding to the time consuming issue of conventional pattern classification techniques, it is hard to produce classifier that capable to manage huge speaker dataset in short period of time. Therefore, a novel hybrid model which takes the advantages of 2 typical pattern classification approaches for text-independent speaker identification is proposed in this study. These two techniques are VQ and GMM.

This study suggested applying VQ as a pre-classifier to select a set of initial speakers that possible to the test data. However, due to the reason of VQ have its limitation of clustering huge dataset; distributed data training is adapted into VQ training process in order to train VQ centroids in a set of small data range instead of the whole dataset.

Through utilizing VQ approach hybrid with GMM statistical computation, this hybrid distributed VQ/GMM model attempts to achieve improvement in terms of

reduce the processing time for speaker identification process. This chapter details the step to construct this hybrid model.

4.1 Overview of the Design Framework

The hybrid distributed VQ/GMM model consists of 4 major phases, which it contains 2 training phases and 2 testing phases. Training phases are used to enroll user into the speaker data base while testing phases aim at identify speaker from trained speakers Followed are the details for each phase:

(i) Hybrid Training Phase I: Distributed Data Pre-processing Phase

(ii) Hybrid Training Phase II: Distributed VQ Training Model

(iii) Hybrid Testing Phase I: Distributed VQ Classification Model

(iv) Hybrid Testing Phase II: GMM Identification Model

4.2 Hybrid Training Phase I: Distributed Data Pre-Processing Phase

This is the first stage of the hybrid training phase. Basically it is a pre-processing process for speech data analysis. The aim of analyses these speech data is to distribute them into smaller subgroups based on its pitch attribute. The research claims that training data in a smaller sub-group instead of the full set of data will lead to a time optimization. The overall process of this training phase I are shows in figure 4.1.

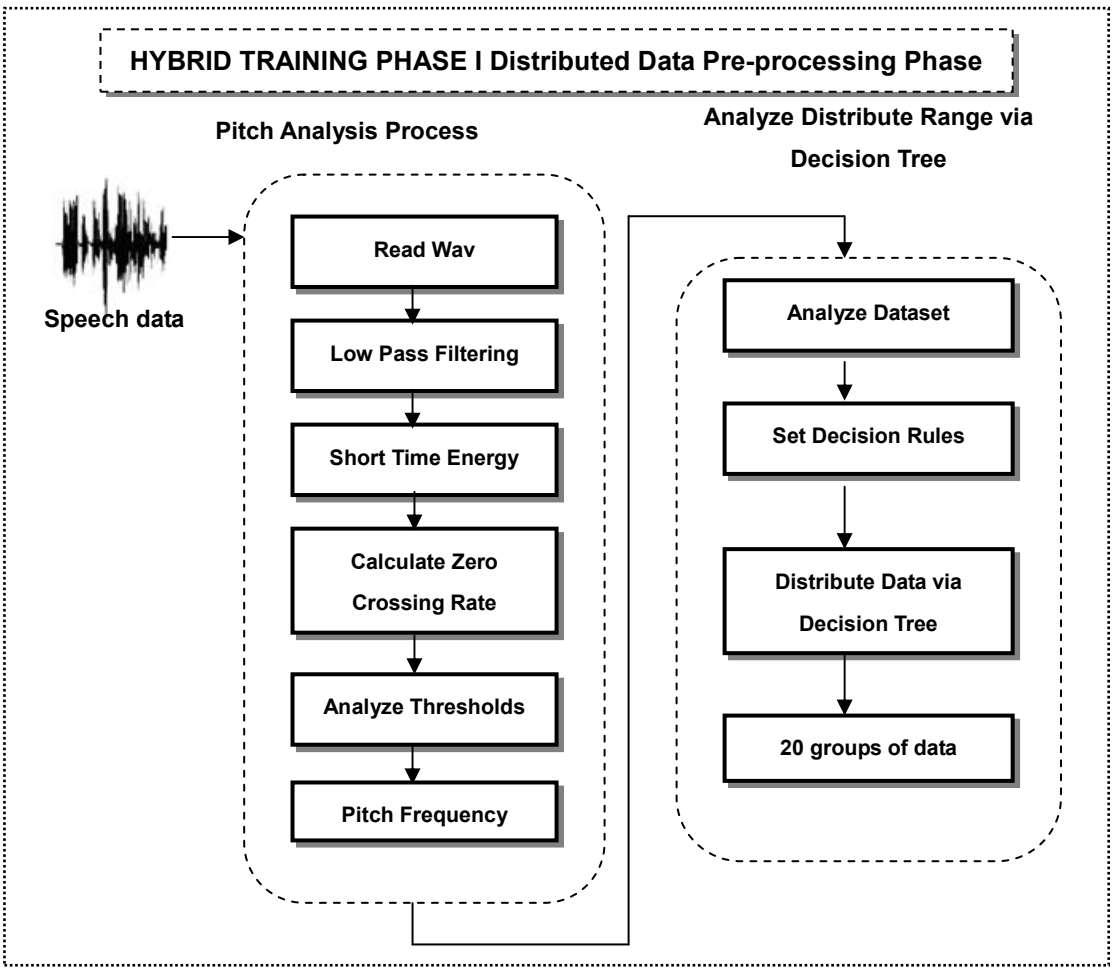


Figure 4.1 Hybrid distributed VQ/GMM modeling - Training phase I

As observed in figure 4.1, the training phases consist of 2 sub-processes. First is the pitch analysis process and second is the decision tree process which aim to assign each speaker data into smaller subgroups.

4.2.1 Pitch Analysis Process

Pitch represents the perceived fundamental frequency of a sound. The human auditory perception system may also have trouble distinguishing frequency

differences between notes under certain circumstances. According to ANSI acoustical terminology, it is the auditory attribute of sound according to which sounds can be ordered on a scale from low to high. In the same theory, human voices convey difference pitch frequency.

Several works have implemented pitch extraction algorithms based on computing the short-time autocorrelation function of the speech signal. First, the speech is normally low-passed filtered at a frequency of about 1 kHz, which is well above the maximum anticipated frequency range for pitch. Filtering helps to reduce the effects of the higher formats and any extraneous high-frequency noise (Julio and Juan, 2005). The signal is windowed using an appropriate soft window (such as Hamming) of duration 20 to 30 ms and a typical autocorrelation function is shown on figure 4.2.

```
// Return RC low-pass filter output samples y, given input samples x
// time interval dt(20ms), and time constant RC (30ms)
 function lowpass(real[0..n] x, real dt, real RC)
   var real[0..n] y
   var real α := dt / (RC + dt)
   y[0] := x[0]
   for i from 1 to n
       y[i] := α * x[i] + (1-α) * y[i-1]
   return y
```

Figure 4.2 Pseudo code for low pass filtering

The autocorrelation function gives a measure of the correlation of a signal with a delayed copy of itself. In the case of voiced speech, the main peak in short-time autocorrelation function normally occurs at a lag equal to the pitch-period. This peak is therefore detected and its time position gives the pitch period of the

input speech.

Second process in pitch analysis is calculating the short-time energy function of a speech file (Harb et al., 2001). The short-time energy function of speech is computed by splitting the speech signal into frames of *N* samples and computing the total squared values of the signal samples in each frame. Splitting the signal into frames can be achieved by multiplying the signal by a suitable window *W[n], n*=0, 1, 2…, *N*-1, which is zero for *n* outside the range (0, *N-1*). A simple function given to extract a measure related to energy can be defined as

$$W[n] = \sum_{n}^{N-1} |x[n]| \bullet W[n-m] \tag{4.1}$$

The energy of the voiced speech is generally greater than that of unvoiced speech.

Zero-crossing rate (ZCR) is a measure of the number of times in a given time interval (frame) that the amplitude of the speech signals passes through the zero-axis. ZCR is an important parameter for voiced/unvoiced classification and end-point detection as well as gender classification as the ZCR for female voice is higher than that for male voice.

Given in (Harb et al., 2001) the proposed variable to do gender classification is defined by a function comprising the mean of ZCR and the center of gravity of the acoustic vector. The logic is that the center of gravity for a male voice spectrum is closer to low frequencies and that of female is to higher frequencies.

$$W = \frac{\sum_{f=1}^{5} X_f}{\sum_{f=35}^{40} X_f} \bullet \frac{1}{Mean(ZCR)} \bullet \frac{\sum_{f} X_f \bullet f}{\sum f} \tag{4.2}$$

where Mean (ZCR) is the mean of ZCR in 1s and X_f is frequency coefficient of "*f*". The *W* should be higher for male voices.

4.2.2 Threshold Analysis

An initial estimate of the average pitch was calculated across the regions of interest identified by a pattern matcher. The estimate is refined by calculating a new average from pitch estimates within a percentage of the original average. Thus this removes the outliers produced by pitch doubling, tripling and error in region classification. This technique using pitch can be used in isolation for gender identification by comparing the average pitch estimate with preset threshold. Estimates below the threshold are identified as male and those above as female (Parris and Carey, 1996).

4.2.3 Distributed Data Process

In a distributed data process, speaker data is separated by a decision tree model. Frequencies of each speaker are obtained from pitch frequency process. These frequencies data are used to set the rules for decision tree modeling in order to distributed data into groups. Subsection 3.4.2 has been discussed about how to design the decision tree and how large is the range for a particular subgroup. Next will presenting the pseudo code for decision tree model to distributed speaker data.

Figure 4.3 shows the pseudo code for decision tree model to distributed speaker data into smaller subgroup. The aim of this model is to provide a smaller range for VQ clustering to cluster speaker data. As shown from figure 4.3, the first step of the decision tree was collecting all train data. Secondly, data are divided into 2 groups using gender as the unit of measure. Then, on each group, data are separated using binary decision rules. In this stage, there are 4 groups of data have been obtained. Next, on each groups, decision tree will calculating the best range for VQ and distributed data into smaller groups based on this range. Finally, data are presenting in the form of group and these groups of data will be utilizing for VQ training.

```
Analyze TIMIT dataset, WHILE Not end of TIMIT dataset
IF Pitch tone = Male
        THEN collect all male data, divide into 2 subgroups (M1, M2)
        IF M1> 32 (the best range for VQ vector)
                THEN Calculate N= Ceil (M1/32)
                        Divide M1 into N group
        IF M2> 32 (the best range for VQ vector)
                THEN Calculate N= Ceil (M2/32)
                        Divide M1 into N group
IF Pitch tone = Female
        THEN collect all female data, divide into 2 subgroups (F1, F2)
        IF F1> 32 (the best range for VQ vector)
                THEN Calculate N= Ceil (F1/32)
                        Divide F1 into N group
        IF F2> 32 (the best range for VQ vector)
                THEN Calculate N= Ceil (F2/32)
                        Divide F1 into N group
```

Figure 4.3 Pseudo code of decision tree in distributing speakers' data

4.3 Hybrid Training Phase II: Distributed VQ Training Model

This section discussed the second phase of the hybrid model for Training data. The model used is baseline VQ model. However, the main difference here are the VQ training range is minimized into 20 subgroups and training process will be done independently in each particular subgroup only.

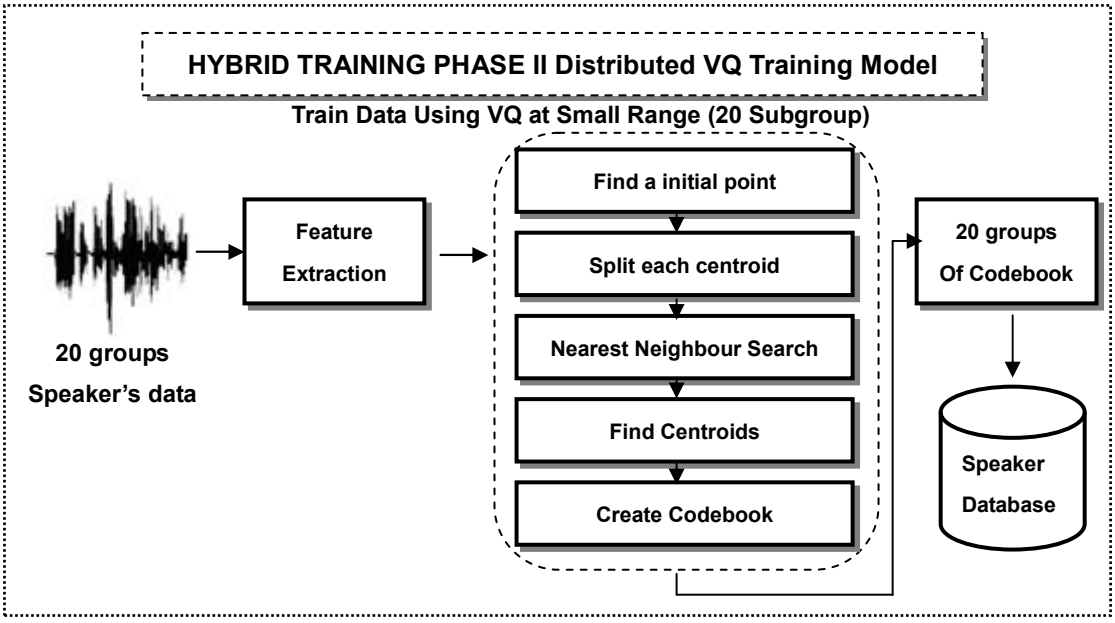


Figure 4.4 Hybrid distributed VQ/GMM modeling - Training Phase II

Figure 4.4 shows the process of VQ training. In general, for VQ training phase, Vector Quantization is executed using feature data as input. Later on, the speaker identification engine will run the nearest-neighbour search to find the centroids in the current codebook and assign that vector to the corresponding cell. The result of clustering is a set of M vector, $C = \{c_1, c_2, \ldots c_m\}$, called a codebook of the speaker. Then, its find centroids and update for each speech signal and the codebooks are created (Pelecanos et al., 2000). After these 20 groups of distributed data training by VQ, 20 codebooks will be stored in the speaker database. Figure

4.5 shows the distributed VQ algorithm for data training. Algorithm started with randomly finds an initial centroid among feature vectors. Then it started to split and find nearest centroid via nearest-neighbour search. This is done by iteration. The iteration is stopped when all the available data are mapped into the codebook.

1 **Input** : Speech feature

2 **Design a 1-vector codebook**; this is the centroid of the entire set of training vectors (hence, no iteration is required here).

3 **Double the size of the codebook** :splitting each current codebook $\mathbf{y}_n$ according to the rule

$$\mathbf{y}_n^+ = \mathbf{y}_n(1+\varepsilon)$$

$$\mathbf{y}_n^- = \mathbf{y}_n(1-\varepsilon)$$

where n varies from 1 to the current size of the codebook, and ε is a splitting parameter (ε=0.01).

4 **Nearest-Neighbor Search**: for each training vector, find the codeword in the current codebook that is closest (in terms of similarity measurement), and assign that vector to the corresponding cell (associated with the closest codeword).

5 **Centroid Update**: update the codeword in each cell using the centroid of the training vectors assigned to that cell.

6 **Iteration 1**: repeat steps 3 and 4 until the average distance falls below a preset threshold

7 **Iteration 2**: repeat steps 2, 3 and 4 until a codebook size of M is designed.

Figure 4.5 Distributed VQ algorithm for training data

4.4 Hybrid Testing Phase I: Distributed VQ Classification Model

The testing phase of speaker identification process is to identify a user's identity. However, in this research, the hybrid modeling consist of 2 testing phases, which the first phase employs VQ model as an estimation technique to select a number initial data. As discussed in subsection 3.4.4.1, 6 best speaker models are chosen here. The VQ technique here remains as pre-classifier to find a small range of possible data. These speaker models will become the input for phase II to run final testing.

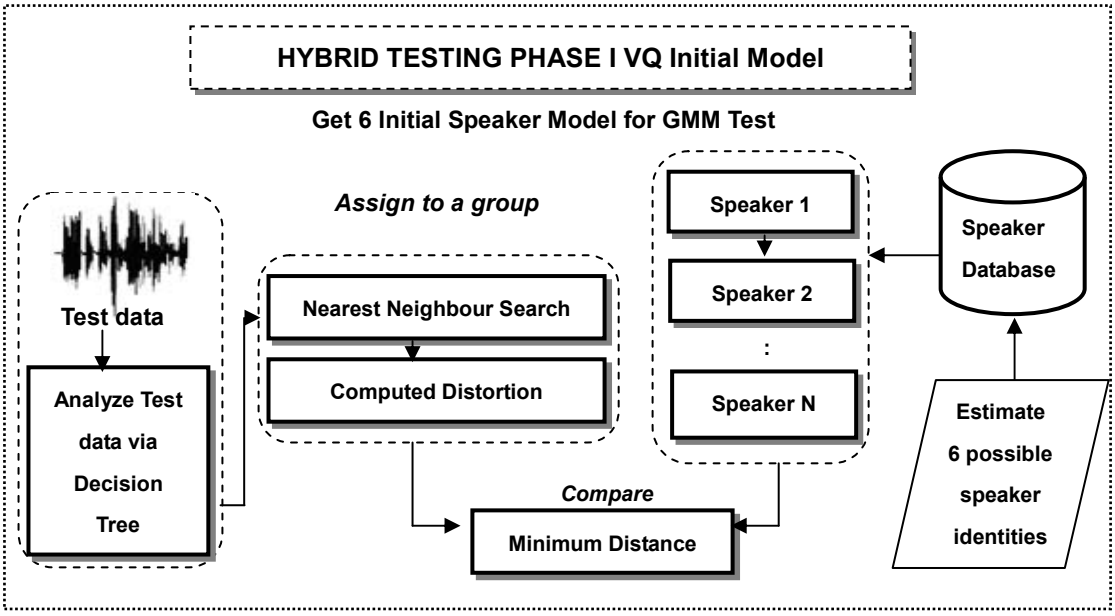


Figure 4.6 Hybrid distributed VQ/GMM modeling - Testing Phase I

Figure 4.6 shows the design of VQ process act as pre-classifier to get 6 initial data and store in speaker database. First, test data will go thru decision tree analysis in order to decide it belonged to which group in 20 groups. Secondly, nearest-neighbour search will run on test data to decide the location of the centriods. Then the distance between of the test centroids and data stored in speaker database

are calculated. Finally, 6 initial data that hold minimizes distance are selected. These data are considering 6 most possible identities for the test data. Figure 4.7 shows the process of decision tree assign test data into a group. The process started with analysis the test data. Next, the data is tested by If-Else condition in order to assign to an appropriate subgroup. Finally, data in the subgroup are compared with test data. This comparison function is done by VQ estimator. Finally, a set of initial data are chosen by VQ estimator.

Analyze Test Data, **WHILE Not end of Test data**
IF Pitch tone = Male
    THEN test
    IF = group M1 (Male subgroup)
        THEN Test which smaller subgroup should be assign (7 groups)
            Assign to appropriate group
      IF = group M2 (Male subgroup)
        THEN Test which smaller subgroup should be assign (7 groups)
            Assign to appropriate group
IF Pitch tone = Female
    THEN Test
    IF = group F1 (Female subgroup)
        THEN Test which smaller subgroup should be assign (3 groups)
            Assign to appropriate group
      IF = group F2 (Female subgroup)
        THEN Test which smaller subgroup should be assign (3 groups)
            Assign to appropriate group

Figure 4.7 Pseudo code for decision tree to assign speakers' data into groups

4.4.1 Nearest Neighbour Search and Distance Calculation

In testing phase, nearest-neighbour search function computes the Euclidean distance between training data and testing data. After comparison made on all available speaker data, this function will identify which calculation yields the lowest value. The speaker data which obtains the lowest value will be identifying as speaker's identity. The match score between the unknown speaker's feature vector $X = (x_1, \ldots x_t)$ and a given codebook $C = \{c_1, c_2, \ldots c_m\}$, is computed as the average quantization distortion:

$$D_{\text{avg}}(X, C) = \frac{1}{T} \sum_{i=1}^{T} e(\boldsymbol{x}_i, C) \tag{4.3}$$

Where $e(x_i, c) = \min_{c_j \in c} \| x_i - c_j \|^2$, and $\| x_i - c_j \|$ denotes the Euclidean norm (Shen et al., 2003).

4.4.2 Best Speaker Model Selection

After computing the distortion, VQ clustering process compare each speaker model based on the distance of each speaker centroids. The VQ engine will determine the speaker data as the identity of test user based on the calculation of minimum distance. Nevertheless, this study adapts some changes in this part in order to select 6 best speaker models for possible identity. Figure 4.8 shows VQ algorithm for the calculation of minimum distances and selections of 6 best speaker models. As shown in the figure, the 6 best speaker models are selected by the 6 nearest distance between the test data.

1	**INPUT**: Test data
2	Find location of centroids
3	Compute Distance between test data and each train data in subgroup
	The Euclidean distance D between two vectors *X* and *Y* is:
	D = sum((x-y).^2).^0.5
4	Store each distance value into array. Arrange data by sorting.
5	Select the 6 speakers hold minimizes distance.
6	**OUTPUT**: 6 possible speakers' data

Figure 4.8 Selection of 6 best speaker models

4.5 Hybrid Testing Phase II: GMM Identification Model

This section discusses the final stage of hybrid modeling. Basically in the stage II of testing phase, this study proposed to utilize the power of the GMM as final classification technique. The reason here is because of GMM obtain high accuracy rates on classification (Xiang and Berger, 2003). The details of the GMM study are shown in chapter 2. Figure 4.9 shows the process of the GMM in classification. Instead of comparing with whole set of data, this study propose to use 6 initial data which is obtained from VQ pre-classifier phase.

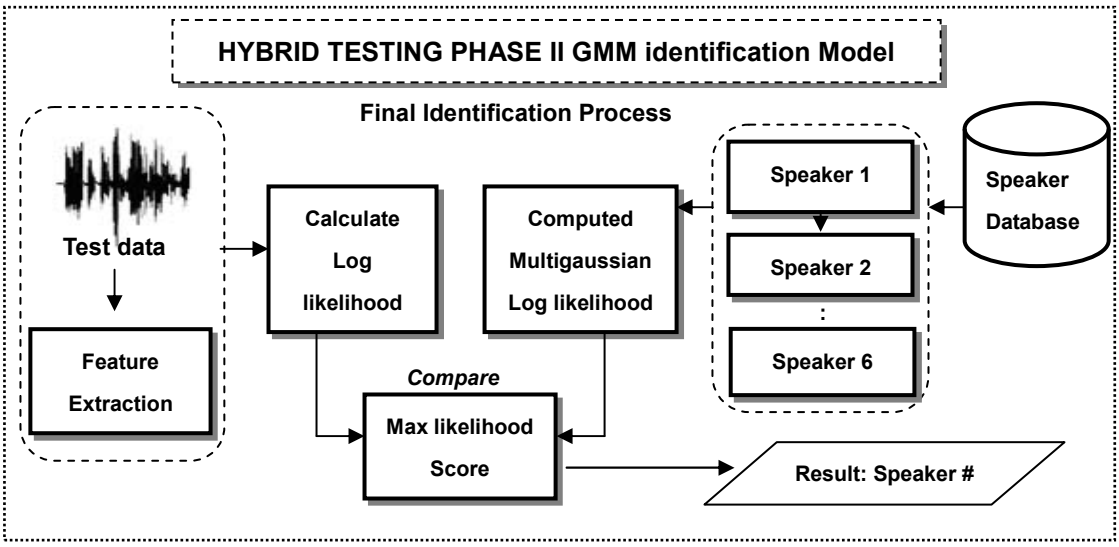


Figure 4.9 Hybrid distributed VQ/GMM modeling - Testing Phase II

1 **INPUT**: 1 Test data ($\vec{x}$, feature Vector)

2 Extract data into Gaussian (G_s) $(w_i, \overline{\mu}_i, \Sigma_i)$

w_i is weight of the Gaussian, $\overline{\mu}_i$ is mean of i^{th} mixture, Σ_i is covariant matrix.

3 Calculate log-likelihood score

$$L(X, G_s) = \sum_{t=1}^{M} P(\vec{x}_t \mid G_s) \quad , X = \{\vec{x}_1, \vec{x}_2, \cdots, \vec{x}_p\}$$

M is total number of the feature.

4 Calculate multi Gaussian log-likelihood score for 6 initial data (Repeat steps 1-3)

5 Compare likelihood score between test data and 6 initial data

6 Get highest score, determine test speaker identity

7 **Output** : Speaker identity

Figure 4.10 Algorithm of GMM test on 6 initial data

Figure 4.10 shows the algorithm for GMM to calculate the multi Gaussian value and test on 6 initial data. Test data is represented in a form of Gaussians and the GMM model calculate the log likelihood score for test data (Reynolds, 1995). GMM model computed the multi Gaussian log likelihood on the 6 initial speaker models. Finally, a comparison is made between the test data and 6 initial data.

4.6 Summary

This chapter reports on the implementation of the hybrid distributed VQ/GMM model. The model is designed to be implemented modularly by four phase. This chapter illustrates how workflow and progress has been carried out in details for this study. The next chapter will discuss on experiments in order to carry out some finding based on the hybrid distributed VQ/GMM model.

CHAPTER 5

RESULTS AND ANALYSIS

Experimentation is an important procedure in which techniques and algorithms are explored and assessed. This will determine the viability of proposed model for speaker identification.

This chapter presents some result obtained from evaluation of hybrid distributed VQ/GMM model which is the focus of this study. Experiments were performed on text-independent environment using standard DARPA TIMIT Acoustic-Phonetic Continuous Speech Corpus.

This study proposed to utilize distributed data training on VQ as a pre-classifier in order to estimate a set of initial result. GMM will work as step after distributed VQ to obtain final decision. Such hybrid model allows one to reach an acceptable compromise between the identification quality (accuracy rate) and time (speed of processing time). The proposed modeling aims to reduce conventional GMM computational time for speaker identification.

This chapter further discusses the proposed modeling that is carried out along with the experimental results. Detailed information about the experimental setup is explained. The chapter consists of 4 major experiments. The first experiment based on the evaluation and comparison performance of baseline VQ and GMM model. The second experiment is followed by an evaluation on hybrid VQ/GMM design with and without distributed data training. Thirdly, the experiment focuses on comparing the baseline method and hybrid distributed VQ/GMM model. Subsequently, the result between proposed model and previous design hybrid VQ/GMM model are evaluated. Finally, a summary and drawback of the experiments are discussed in this chapter.

5.1 Evaluation Measures

The performance of speaker identification system is based on the accuracy rate and the computational time from training to testing. The accuracy rate is measured by how many speakers are correctly identified over the whole database. It can defined as

$$AccuracyRates(\%) = \frac{M - N}{M} \times 100 \tag{5.1}$$

where M is the sum of the speaker store in database and N is the amount speaker who wrongly been identified.

Besides accuracy, this study focused on the processing time for speaker identification. It divides by two categories, which consists of training times and testing times. Training time is the time used for registering all speaker models in

the database, whereas testing time is counted once the user input a speech data until obtain the result of identification.

5.2 Experimental Setup and Condition

Experiments are conducted on clean speech condition. In order to get a fair comparison between each types of classifier, each of them is properly selected from the same datasets. Pre-processing is used to enhance the feature data through a set of preliminary experiments.

Evaluations are performed on TIMIT speech database. The TIMIT corpus of read speech has been designed to provide speech data for development and evaluation of automatic speech recognition systems. Besides, the large number of distinct speakers present in the corpus makes it suitable for evaluation speaker identification system as well.

The experimental tests were performed on a PC running Windows XP Professional SP2, with 512 of RAM and Intel Pentium M Processor 1.50GHz CPU.

5.3 Amount of Data Chosen in Experiments

There are various aspects need to be considered when constructing an experiment. Data sampling is the part of statistical practice concerned with the selection of individual observations intended to yield some knowledge about a population of concern, especially for the purposes of statistical inference. Planning ahead ensures that the experiment is carried out properly and the results reflect the real world, in the best possible way.

The most significant element in this study is the amount of data chosen for experiments. These data sampling are used to draw conclusion about the ability of the proposed model compare to the other models to handle speakers' data from small range to big range. The data range 10, 50 and 100 is chosen to represent abovementioned topic. In these chosen data range, both of them contained female and male speaker equality. Other than these incremental data sampling, a full sets TIMIT corpus which contain 630 speakers' data have been utilized in the experiments to measure the stability of each pattern classification model to handle large dataset.

5.4 Experiment I: Conventional Baseline System

To date, a number of studies have investigated on the performance of text-independent TIMIT database. The results of these studies have been widely used as the benchmark for direct comparison against other approaches. The research

conducted by Reynolds et al. (2000) and Soong et al. (1985) are still highly considered as the classic piece among similar works though some more recent researches have reported better results.

As discuss earlier in chapter 2, VQ and GMM are two conventional baseline systems, which are suitable to apply in text-independent speaker identification system. Vector quantization speaker modeling was popular in the 1980s and 1990s (He and Zhao, 2000; Soong et al., 1985), but after the introduction of the background model concept for GMMs (Reynolds et al., 2000), GMM has become the dominant approach. According to Tomi et al. (2009), VQ approach achieves speed-up in training compared to GMM but with incomparable accuracy. This experiments focus on evaluating ability of the VQ and GMM baseline model for handling increasing and large dataset.

5.4.1 GMM Baseline Speaker Identification Performance

In GMM speaker identification task, each speaker data is modeled by a dedicated Gaussian model. In training phase, Gaussian model which represent speaker will store as speaker model in database. Subsequently, for each testing data, a model is constructed using the Maximum a Posterior (MAP) adaptation method. During testing phase, each Gaussian model is calculated based on MAP independently to estimate the parameters and compare with other Gaussian model to find the best match score.

The aim of the experiment is to evaluate the usage of GMM as speaker identification pattern classification techniques. A series of data contain 10, 50 and 100 data are used in the experiment to bring up the impact of increasing data on accuracy rates. Experiment is conducted on a classified full TIMIT database which contains 630 data in order to assess GMM, which is able to handle a set of large data with stability performance.

Figure 5.1 shows the result of increasing number of the speakers on performance of the GMM speaker identification system. With this sizeable increase, accuracy rates start off with 100%, and slowly declines to approximately 97.1%. From observation, GMM speaker verification accuracy rate is decreased when the training data increase. A possible explanation for this might be GMM has ignored some of the similar knowledge of the underlying phonetic content of the speech. Therefore it does not take advantage of all the available information. There are similarities between the attitudes expressed by Reynolds et al. (2000) in his research of applying adapted GMM.

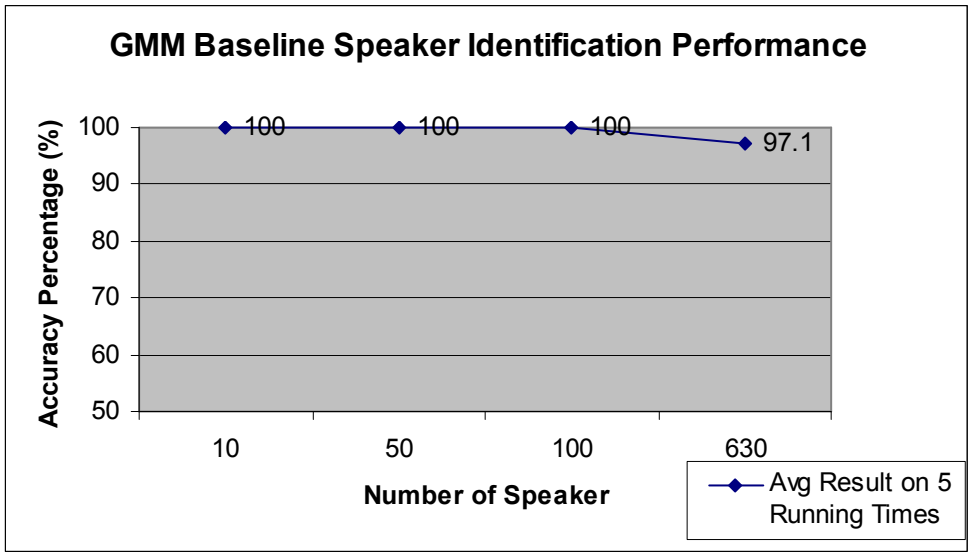


Figure 5.1 GMM baseline speaker identification performance

Although this baseline recognizer is not the best among similar systems, its performance is satisfied for overall when comparing to those representative results produced by Sturim et al.(2002), Reynolds et al. (2000) and Zhao (1999). Therefore, it can be considered as an appropriate recognizer of which the role is to provide a standard baseline that allows proper experiments to be conducted. The most important finding to appear from the data is the GMM prove its ability to handle large dataset in a very high accuracy rate.

Table 5.1 shows the time used for training process and testing process for GMM approach in speaker identification. Focusing now on the performance of the classifier versus time consuming issue, as shown in table 5.1, testing process required more computational time if compared with training process. A possible explanation for this result might be the training process for GMM speaker identification only required calculating log-likelihood score for each trained data to store in database. Whereas testing process include computed multi Gaussian log-likelihood score for all testing data and compare with each log-likelihood score stored in database (Tomi et al., 2009).

Table 5.1: Time use for training and testing for GMM (seconds)

Number of Speaker	10	50	100	630
Time for Training data	30.32	134.37	250.08	4021.65
Time for Testing data	56.19	267.99	569.40	8947.75

As shown in the table, the time needed for testing will increase drastically when the data become larger. These findings has agreed by Reynolds (2000) and Honga et al. (2005) in their research which showed time consuming issue while applying GMM statistical modeling. The reason that leads to this significant increase is because of GMM characteristic will compare all speaker models that store in the database.

5.4.2 VQ Baseline Speaker Identification Performance

VQ is a pattern classification technique applied to speech data to form a representative set of features. Among the first applied of this technique to speaker verification were Soong et al (1985) and Buck et al (1985). It maps vectors to smaller regions called cluster. These cluster's center, centroid, are collected and will make up a codebook. The VQ codebook will represent the speaker feature from the training data. The speaker identification engine depends on the codebook to identify a speaker.

The second evaluation use VQ as pattern classification techniques. Focusing now on the performance of the VQ classifier in handling increasing data, as shown in figure 5.2, accuracy starts off highly 90% for 10 data sampling chosen, and slowly declines to approximately 66% for 100 data sampling chosen. Steep degradation is observed when VQ classifier is tested using full set of TIMIT dataset. It only gained 59.5% of correct identity compared with all data.

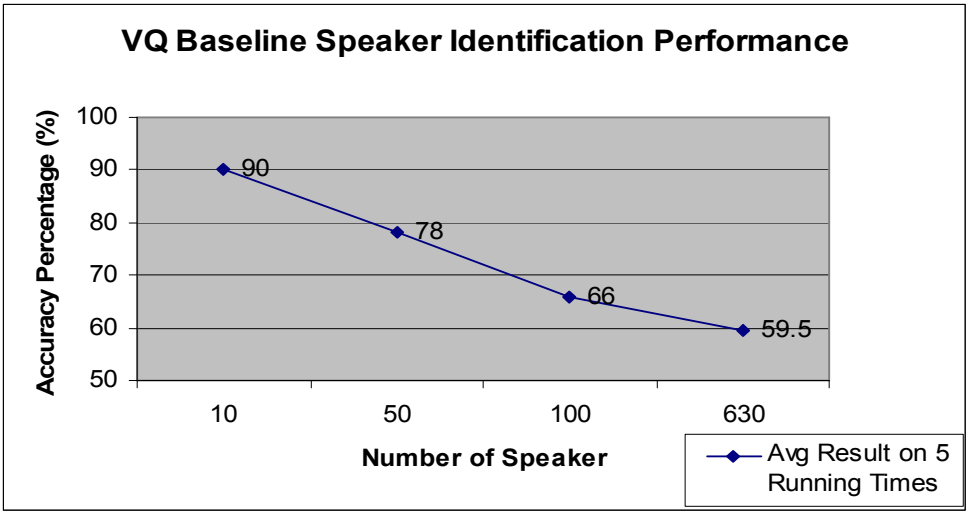


Figure 5.2 VQ baseline speaker identification performance

What is interesting in these findings from figure 5.2 is when VQ classifier handling large set of data, the accuracy rates becomes unreliable. Strong evidence on its disadvantages appear when VQ train and test on the full set of data. This proves Guangyu and Michael (2005) claims about VQ used to work on small range of data.

There is a possibility for this evaluation result is VQ classifier are design to handle short-term spectrum compressed to less than 20 bits per frame (Soong et al., 1985). As Chang and Hung (2006) agree, VQ still can obtain satisfy result when training and testing in small range of data. This is due to its estimation characteristic to estimate possible data using nearest-neighbour search on feature vector. This estimation is based on a centroid that represented speaker identity. While the data become large, after several iteration run nearest- neighbour search, estimation of centroid are deflect from the correct data (Chen, 2004; Guangyu and Michael, 2005). These results are consistent with other studies which found VQ model is suitable for estimation data, not final classification engine (Jialong et al., 1999).

Table 5.2 shows the time use for training process and testing process for VQ approach in speaker identification. As the table shows, the time need for testing will increase drastically when the data become large. The most important finding that can be seen from these results is that the processing time of classification is closely related to the amount of data sampling. This is due to VQ computing the distance between all speaker data to determine the speaker identity.

Table 5.2: Time use for training and testing for VQ (seconds)

Number of Speaker	10	50	100	630
Time for Training data	18.4866	113.133	211.985	1922.56
Time for Testing data	25.9573	131.669	257.681	4975.38

5.4.3 Comparison between VQ and GMM Performance

A comparison has done between VQ and GMM to evaluate their performance over accuracy and time domain. The aim of the experiment is to substantiate the previous studies knowledge for baseline classification.

Figure 5.3 shows a comparison performance between VQ and GMM baseline as a function of number of speaker. As can be seen, GMM are superior in handling large data compared to VQ technique. With increasing number of speaker from 10 to 630, it is found that the accuracy of VQ dropped remarkably while GMM remained almost unchanged, except when a huge amount of speaker is applied. It has proven that GMM displayed greater stability even the dataset are huge. These findings were in good agreement with outcomes reported by many researchers (Chatterjee et al., 2008; So and Paliwal, 2007; Tomi et al., 2006). It is known that VQ is powerless in handling data with more than 50 speakers. This observation is consistent with the researchers who have made claim on VQ model for using 32 code vectors or less. Chatterjee et al. (2008) reported VQ model in 32 code vectors is the best range of clustering data. This is because the complexity, memory, and performance for VQ model are generally most attractive in this range (So and Paliwal, 2007).

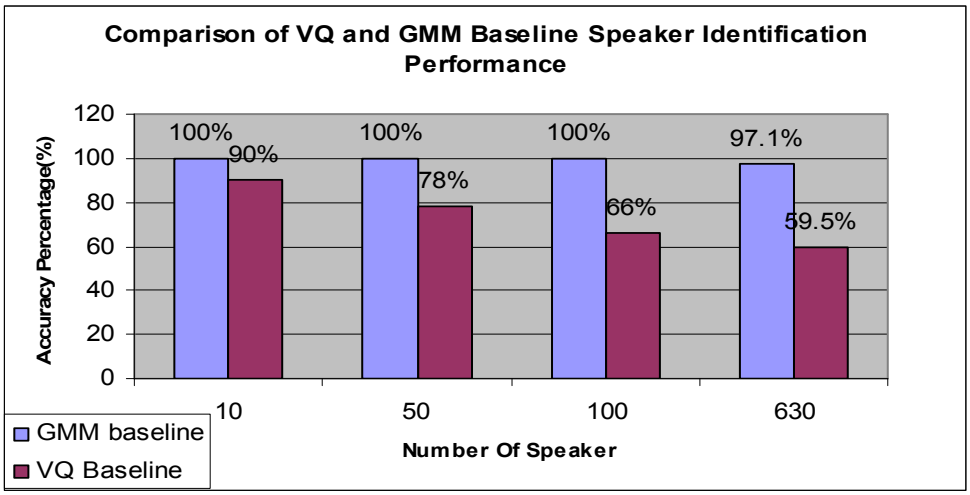


Figure 5.3 Comparisons of VQ and GMM baseline speaker identification performance

The purpose of this part is to evaluate time required for training and testing for VQ baseline and GMM baseline method. The comparison is conducted on the basis of 630 speaker data and the results are presented in Table 5.3. Clearly, it is shown that the time for VQ for training process and testing process are significantly lower than GMM method. This is due to the fact that VQ are based on data compression concept. They will only calculate data based on centroids (by ignoring other data). Compared to VQ, GMM will compare all data available simply because the technique is based on the concept of statistical probability (Tomi et al. 2009).

Table 5.3: Time use for training /testing for VQ and GMM on full TIMIT data

Method	Time for Training data(sec)	Time for Testing data(Sec)
VQ	1922.56	4975.38
GMM	4021.65	8947.75

From the comparisons made, it can be concluded that VQ are not appropriate in handling large dataset even though it provides fast computation. Practically,

most of the speaker identification techniques are applied in security system. Thus, the accuracy rate is vital. However, it is possible if VQ adapted in other pattern classification techniques as a pre-classifier to make estimation on possible vector. Further investigations on the issue of adaptation data will be discussed in the following sub-chapters.

5.5 Experiment II: Hybrid VQ/GMM Modeling

This section discusses the impact of distributed data training on hybrid VQ/GMM modeling. As mentioned in chapter 1, this study aims to design a hybrid model to solve the time-consuming problem encountered in the conventional GMM.

The findings from section 5.4 showed that GMM are able to manage huge speakers' dataset but require longer computational time. Whereas VQ does not face time consuming issue but having limitation on classify data in large set. The relationship between the two characteristics is interesting because one of them (GMM) provide stability in classify data whereas another one (VQ) solve the time consuming issue. Thus, the idea of integrating the positive features of VQ and GMM has been proposed.

This study derives the idea of constructing a hybrid modeling using VQ as pre-classifier to find out a set of initial speaker model so that it can be used as input of GMM classifier. However, based on the findings in section 5.4, VQ are powerless when handling large set of data. These findings are in line with

outcomes reported by Guangyu and Michael (2005). Similarly, Wan et al. (2007) also suggested to divide feature vector via multi-band 2-stage VQ to solve VQ limitation. Consequently, it leads distributed train approach to be utilized in the proposed hybrid VQ/GMM solution.

Since the proposed hybrid VQ/GMM model applied distributed data training, it is important to determine performances of the distributed data training in VQ/GMM. Due to this, a comparison between hybrid VQ/GMM and hybrid distributed VQ/GMM model will be conducted in this section.

5.5.1 Performance of Hybrid VQ/GMM Model without Distributed Training

Experiments were carried out to investigate the necessity of distributed training applied on VQ/GMM model. To begin the evaluation, a hybrid VQ/GMM model without distributed data training has been constructed to evaluate the performances of the model.

VQ method is based on the calculation of the distance measures. The speaker models are identified based on the nearest or minimum distance. For direct adapted VQ/GMM model without distributed data training, this study applied VQ approach by LBG algorithm to train and selected a set of speaker that contain some similarity in terms of the nearest distance. Six sets of the initial data will be selected by VQ. GMM model will then make a comparison between the test data and these six initial estimation data. The comparison between the six set of data and

test data are tested by calculating each log likelihood score. The initial data which gain a maximum score will be determined as the speaker's identity.

The emphasis of this hybrid VQ/GMM model is utilizing VQ as pre-classifier to estimate an initial set of possible speakers' data. 6 initial data have been selected in pre-classifier process. As mentioned in subsection 3.4.4.1, LBG algorithm only considers adjacent neighbors in determining a codeword (Chang e al., 2006). Therefore, results in the 1st and 2nd iteration process of finding nearest centriods will be likely to match with test data (Mowlaee et al., 2008). In general, LBG algorithm is based on binary partition rule where each of the iteration will split into two nearest point. As a consequence, after 1st and 2nd iteration, 6 locations of the nearest centroids can be determined.

Figure 5.4 shows the impact of speaker number on the performances of the hybrid VQ/GMM model without adopting distributed data training. This hybrid model was trained and tested on different speaker numbers, in the range of 10 to 630. Full TIMIT dataset were employed in the testing process in order to assess the stability of hybrid VQ/GMM model when handling large dataset.

The purpose of increase in testing data is to verify the ability of the hybrid model for handling large set of data. The evaluation is based on its stability of classification data. As can be seen from Figure 5.4, accuracy rate was very high when the model was tested on 10 data. The rates however started to decline with increasing speaker data and eventually displayed 56.7% of accuracy rate when it was tested on full TIMIT data.

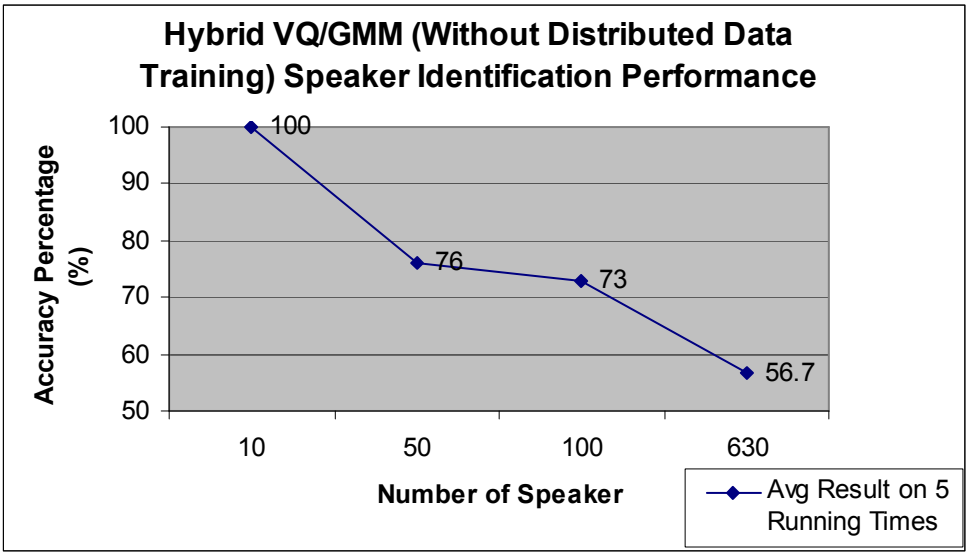


Figure 5.4 Performance of hybrid VQ/GMM model (without distributed data training)

Theoretically, it is known that adapted VQ as pre-classifier for GMM model can perform much result. However, the findings shown by this hybrid design fail to reach the research objective of handling large set of data with stability performance. This variation has been explained in previous section (see section 3.2.1). As previously discussed, VQ is based on short-time acoustic feature. When the feature set becomes too large, it is hard to compute the distance between codebook, leading to inaccurate results (Karpov et al., 2004). These findings were agreed by Nakai et al. (1992), who were the pioneers to develop distributed data training on VQ model. Furthermore, this distributed data training on VQ model are also supported by Guangyu and Michael (2005). Based on the discussion, it can be concluded that direct adapted VQ as pre-classifier for GMM model was failed to obtain a satisfy result when handling large dataset. This is due to its poor performance in terms of accuracy rate.

5.5.2 Computational Time for Hybrid VQ/GMM Model (without Distributed Training)

Since this study focuses on time consuming issue, it is important to evaluate the computational time taken for hybrid VQ/GMM model. This hybrid VQ/GMM model is excluded from distributed data training; it is just direct adaptation of VQ/GMM. In order to assess the performance of hybrid VQ/GMM in solving time consuming issue, it is important to make a comparison between the processing time of different models, i.e. VQ, GMM and hybrid VQ/GMM model.

Table 5.4 shows the result of the processing time for 3 different models. In general, testing time is longer than training time. It is because testing is required to compare each speaker data via algorithm whereas training progress only save variable that convey by speaker's feature into database. No statistical calculation was considered in speaker identification training process (Shen and Reynolds, 2008).

Table 5.4: Computational time use for full TIMIT data training /testing for VQ, GMM and hybrid VQ/GMM model (without distributed data training)

Method	Time for Training(Sec)	Time for Testing(Sec)
VQ	1922.56	4975.38
GMM	4021.65	8947.75
VQ/GMM (Without distributed data training)	1922.56	4646.88

From the table, one can observe that hybrid VQ/GMM model performed faster in comparison with conventional GMM. Besides, it obtained a small range of enhancement for VQ testing time. There are several possible explanations for this result. Firstly, hybrid VQ/GMM model are utilizing VQ algorithm to training,

therefore, the training time for hybrid VQ/GMM are same with VQ training process. While testing process, it utilizes VQ to estimate a set of initial data. GMM model are comparing all test data with these initial set. With these initial set, GMM only compares data in these range instead of all data between test data and train data. Thus, it causes testing time for hybrid VQ/GMM model lesser than conventional GMM model. Secondly, the result shows a small range of enhancement for VQ testing time because VQ method in hybrid model is utilized as estimation function, not classification engine. According to Qiguang et al. (1996), using VQ as estimation function required less computational time if compared with utilizing it as classification engine. This is because there are less iterations of nearest-neighbour search process.

On average, this hybrid VQ/GMM model has successfully reduced the processing time for VQ and GMM conventional techniques. However, this design cannot take into consideration because it fails to obtain good result in accuracy rate. This findings were supported by Karpov et al. (2004) in which they showed that distributed data training should be applied on VQ in order to fix the VQ limitation on handling large data. Due to this, this study constructs a hybrid distributed VQ/GMM model as a substitution for conventional speaker identification system.

5.5.3 The Performance of Hybrid Distributed VQ/GMM Model

Based on findings obtained from sections 5.5.1 and 5.5.2, this study has justified the importance of applying distributed data training on VQ as GMM pre-classifier. This experiment aims to evaluate the performance of the hybrid

VQ/GMM modeling which adapted with distributed data training on VQ model. Based on the report of Jialong (1999), VQ method can perform well in small range of data. Therefore, to make use of VQ method, large dataset have to be distributed into smaller subgroup (Shen et al., 2003; Tae et al., 2002).

Based on these findings, hybrid distributed VQ/GMM model has been proposed for speaker identification pattern classification. A decision tree modeling is used to distribute data for VQ model. Speakers' data are distributed based on pitch frequency conveyed by every speaker. Next, VQ works as pre-classifier to train each subgroup and estimate a set of initial speaker data for GMM testing.

Figure 5.5 shows the accuracy rate of hybrid distributed VQ/GMM on identifying speaker data. As can be seen, the accuracy rates remained unchanged even with the huge dataset. It decreased slightly from 100 to 98.9% with increasing data from 10 to 630. It is found that this hybrid distributed VQ/GMM design produce better result compared to the direct adapted VQ/GMM design (without distributed data training) for handing large set of data. The finding of this experiment support the research claim which is using distributed train for VQ modeling can improve accuracy and time efficiency.

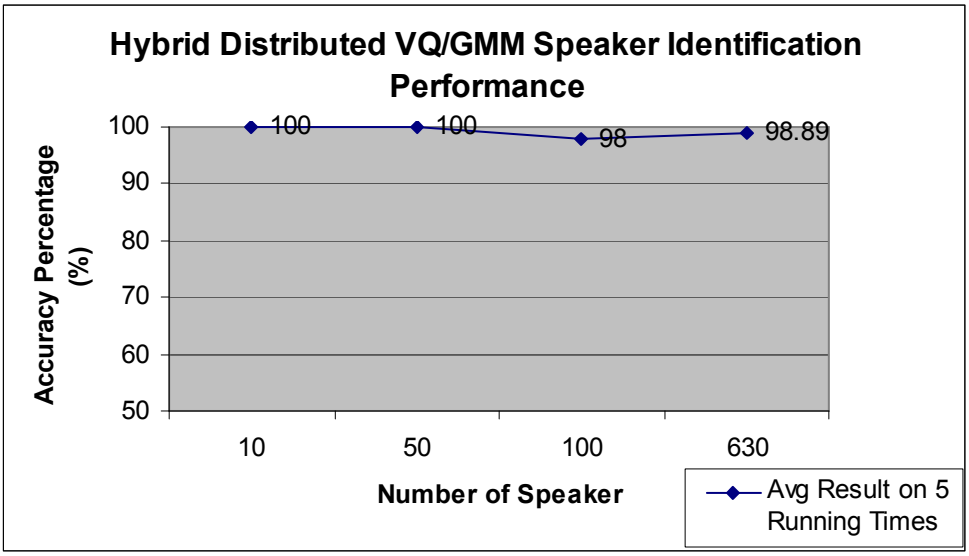


Figure 5.5 Performance of hybrid distributed VQ/GMM model

As mentioned in subsection 3.2.2, direct adaptation using VQ/GMM is often less effective because the VQ encountered difficulties to handle large dataset. Erroneous centroid will be generated from the nearest-neighbour search process (Karpov et al., 2004). Admittedly, with distributed VQ/GMM adaptation, classification result performed marginally better in all size of data sampling. A possible explanation for this might be this study has a well design decision tree approach to distribute data into a well-pleasing range for VQ clustering model. There are other possible explanation which is distributed data training using decision tree has successfully separated huge data into smaller range and reduced computational complexity for VQ model (Chatterjee et al., 2008).

5.5.4 Computational Time for Hybrid Distributed VQ/GMM Model

A comparison has been made between direct adaptations VQ/GMM model and hybrid distributed VQ/GMM model. Table 5.5 shows the result of the processing time for 2 different models. It is obvious that with applying distributed train on VQ techniques, the processing time are less than direct adaptation of VQ/GMM model. This is because distributed training on data has effectively reduced computational complexity with separating all data into smaller subgroup. Thus, training and testing data were run on each subgroup instead of train and test full set data. These results are consistent with the previous studies discussing that split VQ on GMM is capable of dealing with time-varying components for speaker identification (Mowlaee et al., 2008).

Table 5.5: Computational time use for full TIMIT data training /testing on 2 hybrid VQ/GMM model

Method	Time for Training(Sec)	Time for Testing(Sec)
Hybrid VQ/GMM (Without distributed data training)	1922.56	4646.88
Hybrid Distributed VQ/GMM	1664.96	1761.64

5.6 Experiment III: Evaluation on Hybrid Model Vs Baseline Model

The third experiment is focused on comparison of hybrid distributed VQ/GMM modeling with 2 baseline pattern classification models, i.e. VQ and GMM baseline model. The aim of this experiment is to evaluate the identification performance of the purposed model and explore the improvement done on time

consuming issue. The evaluation is based on the processing time and the accuracy rates.

5.6.1 Evaluation on Processing Time

The experiment is executed on full set of data which is consisting of 630 speaker data. Table 5.6 shows the result of time use for training and testing for each type of pattern classification techniques. As shown on the table, hybrid distributed VQ/GMM model used about 1761.64 seconds to identify all speaker identity. It indicated that about 2.79 seconds are required to identify 1 speaker on average. This result shows the hybrid distributed VQ/GMM model is 5.1 times faster than GMM method for speaker identification testing process. While for VQ, it shows the proposed model are only 2.8 times faster than VQ method in testing process. In other words, 64.46% of times are reduced compared to VQ baseline method of 80.31%.

Table 5.6: Processing time for VQ, GMM and hybrid distribute VQ/GMM on full set data

Method	Time for Training(Sec)	Time for Testing(Sec)
VQ	1922.56	4975.38
GMM	4021.65	8947.75
Hybrid distributed VQ/GMM	1664.96	1761.64

In brief, from the whole training and testing process, the hybrid distributed VQ/GMM model have reduced more than 73% of processing times compared with GMM baseline method. Whereas, compared with VQ method, it shows the

processing time is decreased to half. There are a number of different factors affecting the results. A possible explanation for this might be distributed VQ as pre-classifier has successfully reduce the statistical calculation process. With this pre-classifier, classification model are able to skip the process of calculation statistical formula and making comparison on each data. The classification model just needs to work on initial data. Another possible explanation for this might be distributed data in smaller range in the same time cutting down iteration process for nearest-neighbour search. These will bring up more simple calculation for VQ. This is because VQ are based on iteration running nearest-neighbour search to decide location of the centroids. More iteration indicates more complexity in calculation (So and Paliwal, 2007).

5.6.2 Evaluation on Accuracy Rates

Figure 5.6 shows the performance of 3 type of pattern classifiers which is VQ baseline model, GMM baseline model and the hybrid distributed VQ/GMM model. From the figure, the hybrid model has proven its ability to handle large set of data. The accuracy rates obtained from the hybrid method for the full set data test are even better than the baseline GMM. The improvement is 1.79% increase for the identification rates. It is surprising that using hybrid distributed VQ/GMM model, the performance is much better than conventional GMM model. Based on the finding, it can be understood that distributed data training have separated some of the undefined speaker data into different subgroup, leading to higher accuracy.

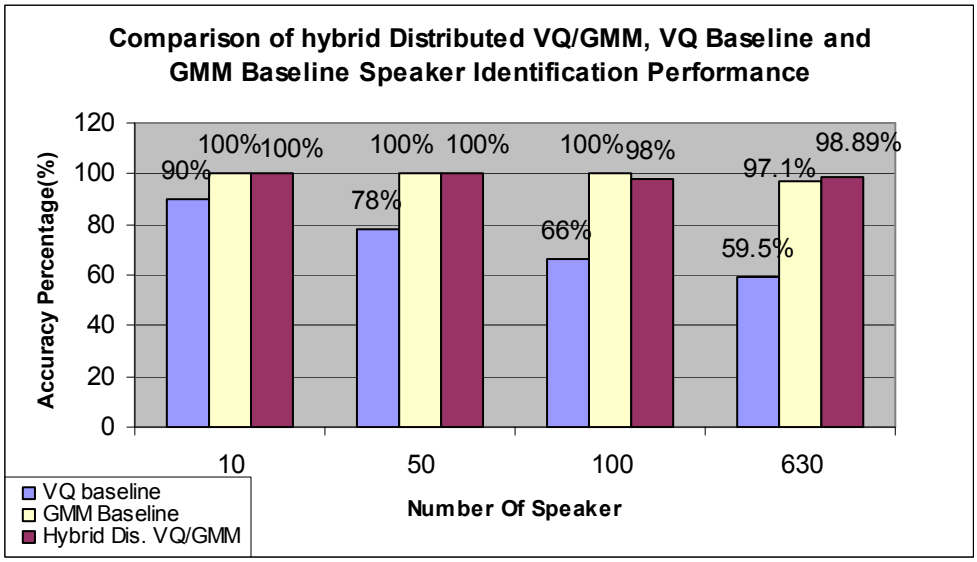


Figure 5.6 The Performance of 3 type of pattern classification models

These findings have supported that hybrid distributed VQ/GMM is suitable to apply as pattern classification model for speaker identification and able to act as an alternative for conventional since it can perform faster and more accurate than traditional method.

5.7 Experiment IV: Evaluation on Other Hybrid VQ/GMM Model

The fourth set of experiment focuses on the evaluation of other hybrid VQ/GMM models proposed by other researchers. Two hybrid models were chosen because they have same research objective which is to reduce the processing time for speaker identification task.

5.7.1 VQ Pre-classifier for Gaussian Selection Model

The first model for evaluation is based on VQ Pre-classifier for Gaussian Selection (VQPGS) Model (Marie, 2006). Marie proposed a pre-classifier by VQ which generates an N-best hypothesis using a novel application of Gaussian selection. The system is trained using parameters of individual speaker models and does not require the original feature vectors, even when enrolling new speakers or adapting existing ones. However, the speaker data are needed to be in the *N*-best hypothesis set. The *N*-best hypothesis set is then evaluated using individual speaker model, resulting in an overall reduction of workload.

Figure 5.7 shows the overall idea of VQPGS model proposed by Marie (2006). The experiment results of the research work are included in appendix B. Although the idea is quite similar with the idea of using VQ as pre-classifier, but the main difference is VQ is used to predict the Gaussian for testing propose instead of initiating speaker model. Moreover, the proposed study applied distributed train on VQ clustering. Figure 5.7 also provides an overview of the VQPGS model. Acoustic feature vectors are quantized to a codeword in a global codebook that is created by clustering the means of the GMMs themselves. Each codeword has a Gaussian short list associated with it which may contain Gaussians from multiple models. The lengths of the short lists are influenced by the percentage of the Gaussians labeled as tails by ρ. As ρ increases, the short list size decreases, possibly at the expense of accuracy. This provides a time versus accuracy performance trade off familiar to implementers of Gaussian selection. As the short list size decreases, small differences in the probability may be enough to change the decision from the correct class to an incorrect one. In a standard Gaussian selection implementation, one would need to increase the threshold once this behavior began to occur.

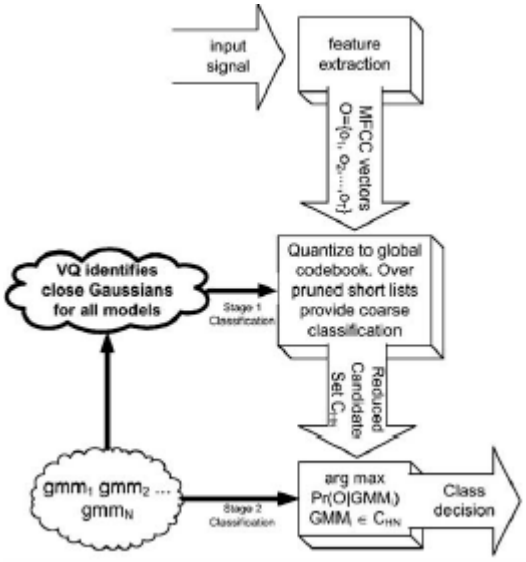


Figure 5.7 Structure of VQ pre-classifier for Gaussian selection model (Marie, 2006)

By using an *N*-best pre-classifier, as long as the correct class is ranked in the highest *N* likelihood scores, the second stage can provide a more thorough analysis and the final decision may indeed be the correct class. The second stage permits the first stage to aggressively set ρ to values that would otherwise lead to unacceptable performance. Like any other pre-classifiers, the work done in the first stage must be significantly less than evaluating the complete set if the goal is to reduce computation time.

First stage classification begins by quantizing each input vector to the nearest codeword. Then for each model, the likelihood of the input vector o_t is computed given the Gaussians on the short lists. The small probability due to the mixtures for which o_t lies on their tails is approximated by the likelihood of the codeword given the culled mixtures. This value is pre-computed and is retrieved by table lookup during recognition. The likelihoods for each observation are merged on a per class

basis in the standard way, (e.g. log sum) and then ranked to determine the *N*-best set.

Figure 5.8 shows the performance of VQPGS (Marie, 2006). Generally, VQPGS model exhibited promising results in pattern classification. It is able to maintain the stability of classifies data under increased data environment. The result is just a little worse than baseline GMM. Accuracy starts off highly 100%, and slowly declines to approximately 95.07%. As can be observed, even accuracy rate has decreased when the training data increased; the model is still able to perform well.

The advantages of VQPGS model is the amount of Gaussians used to evaluate speaker data are based upon previous training knowledge. However, this advantage is in condition: if there are enough codewords provided to model the feature space. Given enough data, Bayesian models converge to a maximum likelihood estimate. However, there is no guarantee that corresponding mixtures of a UBM and its derived model will have diverged enough to make the heuristic effective in some cases. Therefore, the accuracy rates of VQPGS model are less comparable with GMM model which always obtain more that 97% accuracy rates.

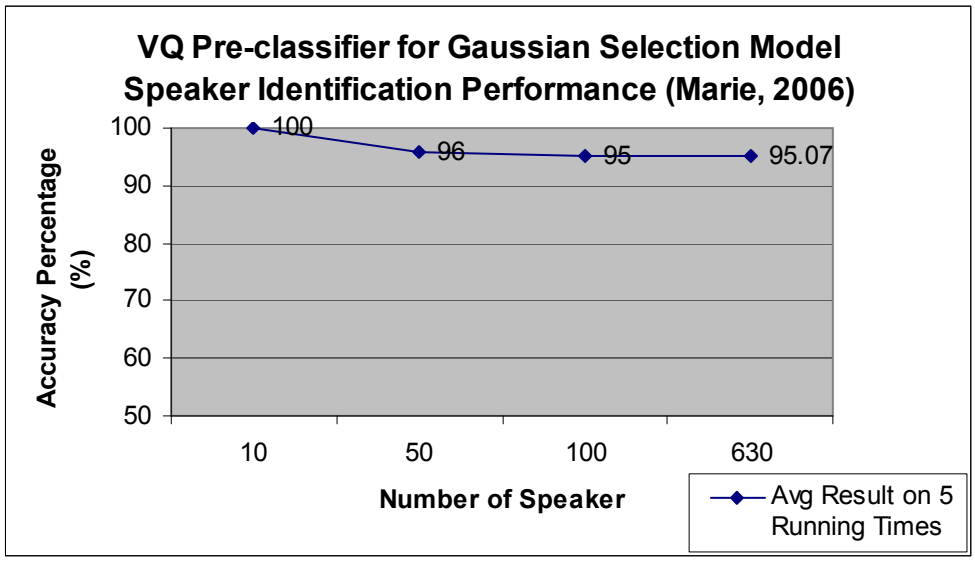


Figure 5.8 Performance of VQ pre-classifier for Gaussian selection

5.7.2 Evaluation on VQ Pre-classifier for Gaussian Selection Model

The aim of this experiment is make a comparison between hybrid distributed VQ/GMM model (proposed model) and VQ Pre-classifier for Gaussian Selection Model (VQPGS). Figure 5.9 presents the performance of both type models. On average, the hybrid distributed VQ/GMM model performs better in terms of accuracy. A possible explanation for this might be that the proposed models have applied distributed data training in VQ clustering phase. As discussed earlier in subsection 3.2.3, VQ clustering result will deflect when handling large set of data. By applying distributed data training on VQ, data range become smaller and leads to less deflection. Thus, it is undoubted that hybrid distributed VQ/GMM model is more efficient than VQPGS model.

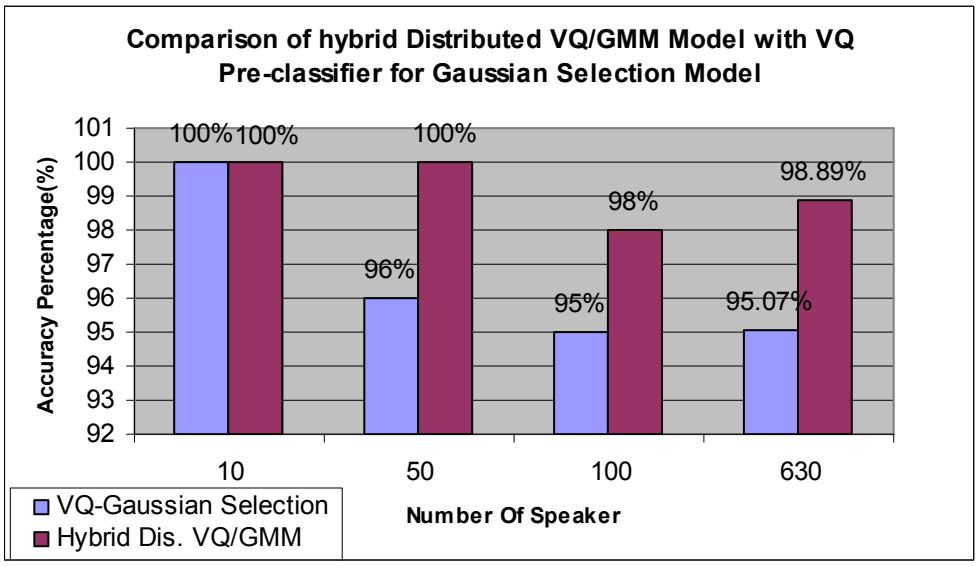


Figure 5.9 Comparison of hybrid distributed VQ/GMM model with VQ pre-classifier for Gaussian selection model

Table 5.7 shows the time required for training and testing for VQ and GMM baseline system, the hybrid distributed VQ/GMM model and VQPGS model. As can be seen, the VQPGS Model used about 3221.60 seconds to identify all speaker identity. In other words, about 5.11 seconds is required for 1 speaker. This result shows VQPGS model are 2.78 times faster than GMM method on speaker identification testing process. While compared with VQ, it is found that VQPGS Model is 1.54 times faster on testing process. It seems possible that this result due to VQPGS model only done statistical calculation on selected Gaussian. Thus, it does not utilize all available data like conventional baseline model (Marie, 2006). Therefore, it always results in faster performance.

Table 5.7: Time used for training /testing on 4 difference models

Method	Time for Training(Sec)	Time for Testing(Sec)
VQ	1922.56	4975.38
GMM	4021.65	8947.75
VQ Pre-classifier for Gaussian Selection Model	1724.8o	3221.60
Hybrid distributed VQ/GMM	1664.96	1761.64

Although VQPGS Model has performed better than conventional baseline model, if compared with hybrid distributed VQ/GMM model, VQPGS Model requests longer computational time. The reason is VQPGS model will classify all data by selected Gaussian. Unlike VQPGS model, hybrid distributed VQ/GMM model classified data in particular subgroup only. Therefore, when handling large dataset, hybrid distributed VQ/GMM model always displays better performance as it ignores other data which is not in group. There are similarities between the attitudes of hybrid distributed VQ/GMM model in this study and those described by So and Paliwal (2007).

Overall, hybrid distributed VQ/GMM model has proved that it can perform better than VQPGS model in term of processing time. Interestingly, these findings indirectly support that the distributed train idea can minimize the processing time.

5.7.3 LBG Training for GMM Model

The use of Gaussian Mixture Model for speaker identification has gained widespread popularity in recent years. This is due to the fact that Gaussian mixture

modeling is a powerful tool for representing virtually any distribution. However, based on Gurmeet et al. (2003) findings, the expression of GMM is simple, the training of a GMM, i.e. finding a model given the feature vectors, is rather complex and time consuming due to its computational complexity and iterative nature.

Training of a GMM is generally accomplished by the EM algorithm (Moon et al., 2003), which guarantees convergence to a local maximum. However, the high computational complexity of the algorithm necessitates high hardware cost as well as large training time. These issues have brought Gurmeet et al. (2003) to introduce LBG algorithm for training Gaussian mixture speaker (LBG-GMM) models as a replacement for Expectation Maximization (EM) algorithm to reduce computational complexity. EM algorithm is typically used for GMM training to find a local maximum value. However, if the speaker data become large, it still faces the time consuming problem. Therefore, Gurmeet replaced the EM algorithm with LBG which is from the VQ training. Based on the outcomes, the researcher found that by adopting the LBG algorithm, the complexity of calculation can be reduced 50%. These results reported by Gurmeet et al. (2003) can be found in appendix C.

Figure 5.10 shows the performance of LBG Training for GMM Model (LBG-GMM). This model was proposed by Gurmeet et al. (2003). As shown in the figure, LBG-GMM model generally can obtain a very good result in pattern classification technique due to its ability to maintain the stability of classify data under increasing data environment. The result is even better than baseline GMM. Accuracy starts off highly 100%, and slowly declines to approximately 99.52%. As can be observed, even though the accuracy rate has decreased with increasing training data, it is still able to exhibit good result if compared to GMM.

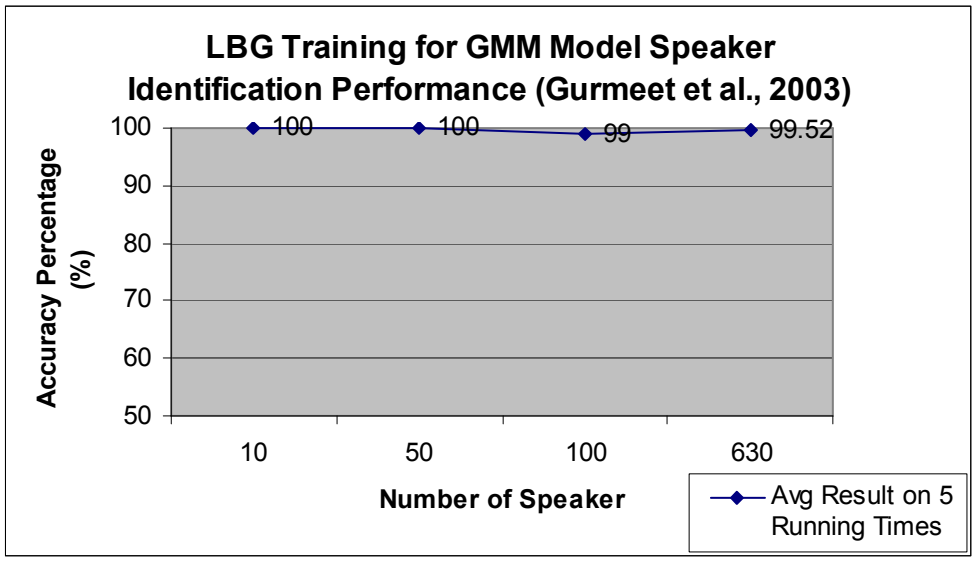


Figure 5.10 Performance of LBG Training for GMM model

5.7.4 Evaluation on LBG Training for GMM Model

This section performs an evaluation about accuracy and processing time issue between hybrid distributed VQ/GMM model and LBG-GMM model. From Figure 5.11, it is shown that the LBG-GMM model displayed better result in handing full set of data than VQ/GMM model. The processing time of these two models are shown in Table 5.8. Clearly, LBG-GMM model just reduce a small amount of testing time. A possible explanation of this result might be LBG-GMM model was focuses on adaptation LBG algorithm into GMM model. By this type of adaptation, the complexity of Expectation-maximization algorithm can successfully reduced. However, the step of training data and testing data remained the same as GMM model. Due to above reason, LBG-GMM model is only provided a classification model with less memory usage required. It does not help much in time consuming

issue (Gurmeet et al., 2003).

However, there is a possible explanation for why LBG-GMM model testing times are less than conventional GMM model. This might be LBG-GMM model has successfully reduced the vector used for testing, thus, the threshold of the classification also decreases, resulting in the data easier to classify. Moreover, due to less calculation involved, it takes less time compared with baseline GMM system. However, the time taken for processing was longer than hybrid distributed VQ/GMM. This is because LBG-GMM model works on full set data whereas hybrid distributed VQ/GMM perform the work on part of related data only.

From above discussion, it can be concluded that LBG-GMM model can work better in terms of accuracy. However, there are still advantages for applying hybrid distributed VQ/GMM model since it does not raise time consuming issue.

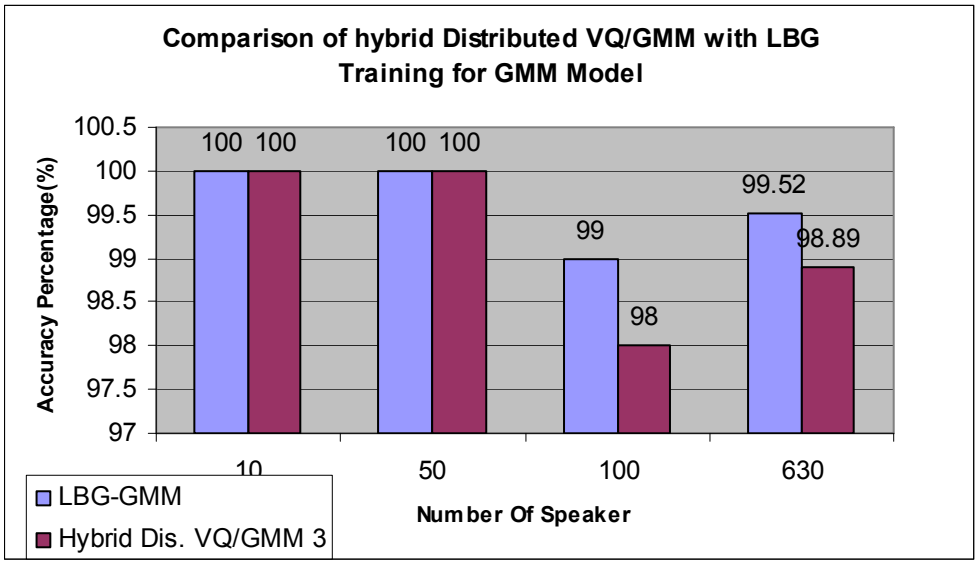


Figure 5.11 Comparison of hybrid distributed VQ/GMM with LBG training for GMM model

Table 5.8: Comparison on 4 difference models (processing time)

Method	Time for Training(Sec)	Time for Testing(Sec)
VQ	1922.56	4975.38
GMM	4021.65	8947.75
LBG training for GMM Model	4027.98	8765.78
Hybrid distributed VQ/GMM	1664.96	1761.64

5.8 Summary

The principal contributions of this experiment are presented a series of evaluation and comparison performance. From the findings of the experiment, the proposed model - hybrid distributed VQ/GMM has been proven to be a powerful tool for text-independent speaker identification system. It has successfully achieved the goal of this research which is solving the time consuming issue for GMM model. Although hybrid distributed VQ/GMM that applied in this study has performed well for several comparisons in experiment, it retains some constraints. Next chapter will discuss further research direction which could possibly reduce these constraints.

.

CHAPTER 6

CONCLUSION AND FUTURE WORK

6.1 Summary

The demand of developing a pattern classification model to handle large dataset with less time consuming is high for speaker identification applications. Recent development in classifying speakers' data from a group of speakers is still insufficient to provide a satisfying result in achieving high performance pattern classification engine. In a real time speaker identification system, the problems such as the learning ability of pattern classification model and high discrimination power need to be solved so as the system is able to handle large dataset at a higher accuracy rate while less time consuming.

GMM method is the dominance of the pattern classification techniques for text-independent speaker identification. It provides high accuracy rates and able to manage huge speaker data set while in the same time maintains the stability of classification. Recently, many researchers have paid great attention on the

investigation of the GMM in speech processing with the aim of getting better results. Although GMM is able to enhance the approaches, the issue of time consuming still is a major concern. This is because the operation of GMM is strongly dependent on statistical modeling; leading to more time is required to obtain results.

This research intends to develop a novel hybrid distributed VQ/GMM model for text-independent speaker identification in order to reduce the processing time for conventional GMM approach without compromising the accuracy rate. This study presents a distributed data training for VQ pre-classifier to separate the large group of speaker data into some smaller subgroups. Pitch frequency is used as measure unit for the decision tree to distribute speaker data into smaller groups. This distributed data training for VQ has successfully solved the VQ constrain to work with small data range. Finally, with distributed VQ pre-classifier, a set of initial result have been estimated and this result was tested by GMM classification to achieve final identification result.

VQ approach is not guaranteed to find the best set of clusters but in practice it works very fast. Therefore, VQ method can work very well as pre-classifier because proposed model utilized VQ to estimate an initial set of possible data. By utilizing VQ as pre-classifier, it is found that the proposed method could skip the step (i.e. statistical calculation) conducted by conventional GMM method. Instead of comparing all data, GMM classifier only works on the initial set select by VQ pre-classifier. The proposed model thus offers the significant advantage of minimizing the time of processing.

The efficiency of this hybrid distributed VQ/GMM modeling is evaluated by computational time and accurate result compared to conventional GMM model.

Experimental results showed that the hybrid distributed VQ/GMM yields better accuracy. It reduced more than 80% processing time and exhibited 5.08 times faster compared to GMM baseline techniques. From the experiments, a good way of applying hybrid method between VQ and GMM has been observed. Overall, the experiments have shown that this hybrid distributed VQ/GMM model performed efficiently and competitively on classify speakers' identity. Moreover, it is capable to handle large speakers' dataset for text-independent speaker identification system. Since this study is designed for speaker identification system, it offers benefit to enhance the performances of identification process on accessing control system. Besides, it also leads an alternative to pattern classification research for speech processing. As a conclusion, this proposed hybrid model has been successfully proven less time consuming and easier to implement in comparison with conventional GMM methods.

6.2 Future Works

Several issues concerning the construction of the hybrid distributed VQ/GMM model in order to deserve further investigations, developments and experiments. In this section, several important directions for further research are being suggested.

First is utilizing this hybrid distributed VQ/GMM model to deal with other types of speech corpus. As discussed in previous chapters, this study is using TIMIT corpus to conduct the experiment. TIMIT corpus is a well known and standard corpus taken from United Stated (Louis et al, 1996). However, people from different races may convey different pitch tone in their dialect and the most

important is this study has utilized pitch tone as unit of measure to distribute speakers' data. Therefore, to improve the robustness of the method described in this study, more varieties of speech corpus are needed to set better rules for decision tree.

This study was conducted in noisy less speech environment. The next challenge regards to the attempt to adapt proposed model into noisy speech environment. Based on general understanding, real time speaker data contains some noisy speech data. For example, the noise of a car passing by or raining sound. Therefore, a noise removal process is needed to determine the speakers' features accurately and minimize the errors in identification.

Further investigations are also needed on decision tree approach. Decision trees have been well studied and widely used in knowledge discovery and decision support systems. Decision Tree algorithms can grow each branch of the tree just deep enough to perfectly classify the training examples. While this is sometimes a reasonable strategy, it can also lead to difficulties when there is noise in the data, or when the number of training examples is too small to produce a representative sample of the true target function. In either of these cases, this simple algorithm can produce trees that over-fit the training examples. Over-fit is a condition where a model is able to accurately predict the data used to create the model, but does poorly on new data presented to it. Hence, selecting rules for decision tree should be flexible to all type of speakers' data. There is some solution which is possible to solve this issue. For example, apply post-pruning for decision tree. Following are rules for applying post – pruning approach:

(i) Convert decision tree to equivalent set of rules.

(ii) Prune each rule independently of others, by removing any preconditions that result in improving its estimated accuracy.

(iii) Sort final rules into desired sequence for use.

Another important area of further research consists of pre-processing and enhancement phases. Speech front-end processing consists of transforming the speech signal to a set of feature vectors (Moretto, 1995). The aim of this process is to obtain a new representation which is more compact, less redundant, and more suitable for statistical modeling. It is favorable if speaker information can be extracted and preserved in good condition at the initial phase to avoid misinterpretation of data.

REFERENCES

Acero, A., Chow, Y. L. and Lee, K. F. (1996). *US Patent No.5535305*. US : Apple Computer, Inc.

Almuallim, H., Kaneda, S. and Akiba, D. (2002). *Development and Applications of Decision Trees*. In Leondes, C. T. (Ed.) *Expert Systems : The Technology of Knowledge Management and Decision Making for the 21st Century*. (pp. 53-77). London : Academic Press.

Atal, B. S. (1976). Automatic Recognition of Speakers from Their Voices. *Proceedings of the IEEE*. 64 (1976), 460 - 475.

Atal, B. S. and Hanauer, L. S. (1971). Speech Analysis and Synthesis by Linear Prediction of the Speech Wave. *Journal of the Acoustical Society of America*. 50, 637-655.

Auckenthaler, R., Parris, E. S. and Carey, M. J.(1999). Improving A GMM Speaker Verification System by Phonetic Weighting. *IEEE International Conference on Acoustics, Speech, and Signal Processing*. 15-19 March. USA, 313-316.

Aronowitz, H. (2006). *United States Patent 7050973*. Santa Clara, CA, US : Intel Corporation.

Baum, L. E. (1974). An Inequality and Associated Maximization Techniques In Statistical Estimation of Probabilistic Functions of Markov Processes. *American Mathematical Society Bulletins*, 73, 360-363.

Baum, E. B. and Wilczek, F. (1988). *Supervised Learning of Probability Distributions by Neural Networks*. In Anderson, D. Z. (Ed.) *Neural Information Processing Systems*. (pp. 52-61). USA : American Institute of

Physics.

Ben, G.. and Nelson, M. (2002). *Speech and Audio Signal Processing*. (2nd ed). USA : John Willy & Sons.

Beyerlein, P. (1998). Discriminative Model Combination. *Proceedings of the 1998 IEEE International Conference on Acoustics, Speech and Signal Processing*. 12-15 May. Seattle, WA, USA, 481-484.

Bo, Q., Yanping, L., Limin, X. and Zhenmin, T. (2006). A Method of Biomimetic Pattern Recognition for Speaker Recognition. *International Conference on Intelligent Information Hiding and Multimedia Signal Processing*. December 2006. USA, 317 - 320.

Bilmes, J. A. (1998). *A Gentle Tutorial of the EM Algorithm and its Application to Parameter Estimation for Gaussian Mixture and Hidden Markov Models*. Technical Report, University of Berkeley.

Bimbot (2005). A Tutorial on Text-Independent Speaker Verification. *Journal of Application Signal Process*. 4, 430-451.

Bishop, C. M. (1995). *Neural Networks for Pattern Recognition*. (1st ed.) New York : Oxford University Press.

Bruneaua, P., Gelgona, M. and Picarougnea, F. (2009) . Parsimonious Reduction of Gaussian Mixture Models with a Variational-Bayes Approach. *Journal of Pattern Recognition*. 43(3), 850-858.

Buck, J. T., Burton, D. K. and Shore, J. E. (1985). Text-Dependent Speaker Recognition Using Vector Quantization. *Proceedings of ICASSP-85*. 1,391-394.

Burges, C. J. C. (1998). A Tutorial on Support Vector Machines for Pattern Recognition. In Usama, F. (Ed.) *Data Mining And Knowledge Discovery*. (pp. 121-167). Bostom, Kluwer Academic Publishers.

Campbell, J. (1996). In Memory of Thomas E. Tremain 1934-1995. *IEEE Transactions on Speech and Audio Processing*. 4(1), 1-12.

Campbell, J.P. (1997). Speaker Recognition: A Tutorial. *Proc. of the IEEE*. 85(9), 1437-1462.

Campbell, W. (2002). Generalized Linear Discriminant Sequence Kernels for Speaker Recognition. *Proc. of IEEE International Conference on Acoustics, Speech, and Signal Processing*. 1(1), 161-164.

Campbell, W. M., Campbell, J. P., Reynolds, D.A., and Singer, E. (2006). Support Vector Machines for Speaker and Language Recognition. *Journal of Computer Speech Language*. 20(2–3), 210–229.

Chakroborty, S., Roy, A. and Saha, G.. (2008). Improved Closed set Text-Independent Speaker Identification by Combining MFCC with Evidence from Flipped Filter Banks. *International Journal of Signal Processing*. 4(2),114-122.

Chang, C. C, and Hung, K. L. (2006). An Efficient Euclidean Distance Estimation Scheme for Accelerating VQ Encoding. *First International Conference on Innovative Computing, Information and Control*. 30 August - 1 September. Beijing, China, 194-196.

Chatterjee, S. and Sreenivas, T.V. (2008) Switched Conditional PDF-Based Split VQ Using Gaussian Mixture Model. *IEEE Signal Processing Letters*. 15(1), 91-94.

Chen, P. Y. (2004). An Efficient Prediction Algorithm for Image Vector Quantization. *IEEE Trans. Man Cybernet*. 34 (1), 740–746.

Cheung, C. L. (2004). *GMM Based Speaker Recognition for Mobile Embedded Systems*. Doctor Philosophy. Chinese University of Hong Kong, People's Republic of China.

Childers, D. G.. and Ke, W. (1991). Gender Recognition from Speech. Part II: Fine analysis. *Journal of the Acoustical Society of America*. 90(4), 1841-1856.

Cuperman, V. (1986). Vector Transform Quantization for Speech Coding. *Proceedings of IEEE Globecom*. 23 - 25 December. Houston, USA, 792-796.

Daniel, J. (2004). Enhancement of GMM Speaker Identification Performance Using Complementary Feature Sets. *7th Africon conference in Africa*. 15-17 September 2004. Gaborone, Botswana, 1273-1278.

Daniel, R.C., Julian, F. A., Joaquin, G. R. and Javier, O. G. (2007). Speaker Verification Using Speaker and Text-Dependent Fast Score Normalization. *Pattern Recognition Letters archive*. 28(1), 90-98.

Davis, S. B. and Mermelstein, P. (1980). Comparison of Parametric Representations For Monosyllabic Word Recognition in Continuously Spoken Sentences. *IEEE Trans. on Acoustic, Speech and Signal Processing*. 28 (4), 357-366.

Deboys J. (2004). Decision Pathways in Patent Searching and Analysis. *World Patent Information*. 26(1), 83–90.

Doddington, G. R. (1985). Speaker Recognition - Identifying People by Their Voices. *Proc. IEEE*. 73(11), 1651-1664.

Doddington, G. (1998). Speaker Recognition Evaluation Methodology - An Overview and Perspective. *Workshop on Speaker Recognition and its Commercial and Forensic Applications*. 20-23April. Avignon, France, 60-66.

Doddington, G. R., Przybocki, M. A., Martin, A. F. and Reynolds, D. A. (2000). The NIST Speaker Recognition Evaluation Overview, Methodology, Systems, Results, Perspective. *Journal of Speech Communication*. 31(2-3), 251-254.

Duda, R. O., Hart, P. E. and Stork, D. G.. (2001). *Pattern Classification* (2nd ed.). Canada : John Wiley and Sons Inc.

Dymarski. P. and Wydra, S. (2008). Large Margin Hidden Markov Models In Command Recognition and Speaker Verification Problems. *15th International Conference on Systems, Signals and Image Processing*. 25-28 June. Bratislava, Slovakia , 221 - 224.

Elif, D. and Michele, F. (2008). Decision Tree Analysis as a Tool to Optimise Patent Current Awareness Bulletins. *World Patent Information*. 30 (2008), 212–219.

Elmisery, F. A., Khaleil, A. H., Salama, A.E. and El-Geldawi, F. (2005). An FPGA Based Vector Quantization for Speaker Identification. *17th International Conference on Microelectronics.* 13-15 December. Islamabad, Pakistan, 130-132.

Ephraim,Y. and Merhav, N. (2002). Hidden Markov processes. *IEEE Trans. Inform. Theory*. 48 (6), 1518-1569.

Farrell, K. R., Mammone, R. J. and Assaleh, K.T. (1994). Speaker Recognition Using Neural Networks and Conventional Classifiers. *IEEE Trans. Speech Audio Process*. 2(1), 194–205.

Ferguson, J. (1980). Hidden Markov models for speech. *Article of Institute for Defence Analysis, Communications Research Division.* p. 143-179.

Fenglei, H. and Bingxi, W. (2003). Text-Independent Speaker Recognition Using Probabilistic SVM with GMM Adjustment. *Proceedings of International Conference on Natural Language Processing and Knowledge Engineering.* 26-29 October. Beijing, China, 305 - 308.

Fine, S., Navratil, J. and Gopinath, R. A. (2001). Enhancing GMM Scores Using Svm "Hints". *Proceedings of Eurospeech 2001*. 3-7 September. Aalborg, Denmark, 1757-1761.

Florian, H., Georg, S., Christian, H. and Fabio, B. (2005). Revising Perceptual Linear Prediction. *Proceedings of the 9th European Conference on Speech Communication and Technology*. 4-8 September. Lisbon, Portugal, 2997-3000.

Franzini, M.A., Witbrock, M.J. and Lee, K. F. (1989). Speaker-Independent Recognition of Connected Utterances Using Recurrent and Non Recurrent Neural Networks. *Proc. Int'l Joint Conf. Neural Networks*. 2(2), 1-6.

Fussell, J. W. (1991). Automatic Sex Identification from short Segment of Speech. *Proceedings of the International Conferences of Acoustics, Speech, and Signal Processing*. 14-17 April. Toronto,Canada, 409-412.

Gersho, A. (1986). *Vector Quantization: A New Direction in Source Coding*. In Biglieri, E. and Prati, G (Ed.). *Digital Communication*. (pp. 267-281). North- Holland : Elsevier Science Publishers.

Gersho, A., Wang, S. and Zeger, K. (1992) *Vector Quantization Techniques in Speech Coding*. In Furui, S. and Sondhi, M. M. (Ed.). *In Advances in Speech Signal Processing*. (pp. 49-84). New York : Mar- cel Dekker.

Geoffrey, M. and Thriyambakam, K. (1996). *The EM Algorithm and Extensions.* (1st ed). New York : John Wiley & Sons.

Gish, H. (1990). A Probabilistic Approach to the Understanding and Training of Neural Networks. *Proc. ICASSP*. 1(1), 1361-1364.

Gish, H. and Schmidt, M. (1994, October). Text-Independent Speaker Identification. *IEEE Signal Processing Magazine*. p.18 - 32.

Gold, B. and Rabiner, L.R (1969). Parallel Processing Techniques for Estimating Pitch Periods of Speech In Time-Domain. *Journal of the Acoustical Society of America*. 46 (2B), 442-448.

Gold, B. and Morgan, N. (1999). *Speech and Audio Signal Processing*. (1st Ed.). Europe : John Wiley & Sons, Inc.

Gray, R.M. (1984). Vector quantization. *IEEE ASSP Magazine* . 1(1), 4–29.

Guangyu, Z. and Michael, W.B. (2005). Speaker Identification Based On Vector Quantization with Adaptive Discriminative Techniques. *48th Midwest Symposium on Circuits and Systems*. 7-10 August. Ohio, US, 1851 - 1854.

Gurmeet, S., Panda, A., Bhattacharyya, S. and Srikanthan, T. (2003). Vector Quantization Techniques for GMM Based Speaker Verification. *IEEE International Conference on Acoustics, Speech, and Signal Processing*. 6-10 April. United States, 65- 68.

Haber, J. and Seidel, H.P. (2000). Using an Enhanced LBG Algorithm to Reduce the Codebook Error In Vector Quantization. *Proceedings Computer Graphics International*. 19-24 June. Geneva, Switzerland, 99 - 104.

Harb, H., Chen, L. and Auloge, J. (2001). Speech/ Music/ Silence and Gender Detection Algorithm. *Proceedings of the 7th International conference on Distributed Multimedia*. 26-28 September. Taipei, Taiwan, 356-362.

Hautamaki, V., Tomi, k., Karkkeinen, I., Tuononen, M., Saastamoinen, J. and Franti, P. (2008). Maximum a Posteriori Estimation of the Centroid Model for Speaker Verification. *IEEE Signal Processing Letter.* 15(1), 162–165.

He and Zhao (2003). Fast Model Selection Based Speaker Adaptation for Nonnative Speech. *IEEE Trans. Speech Audio Process*. 11 (1), 298–307.

Hermansky (1990). Perceptual Linear Predictive Analysis of Speech. *Journal of Acoust. Soc. Am*. 87(4), 1738–1752.

Higgins, A., Bahler, L., Vensko, G., Porter, J. and Vermilyea, D. (1992). *YOHO Speaker Authentication Final Report*. ITT Aerospace / Communications Division.

Hong, Q.Y., Sam, K. and Wang, H.L. (2004). Optimization of Gaussian Mixture Model Parameters for Speaker Identification. *Springer Link Lecture Notes in Computer Science.* 3103 (1), 1310-1311.

Hong, Q.Y. and Kwong, S. (2004). Discriminative Training for Speaker Identification Based On Maximum Model Distance Algorithm. *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing*. 17-21 May. Montreal, Canada, 25-28.

Honga, Q.Y. and Kwong, S. (2005). A Discriminative Training Approach For Text-Independent Speaker Recognition. *Journal of Signal Processing*. 85(7), 1449-1463.

Hsieh, C. T., Lai, E. and Wang, Y. C.(2003). Robust Speaker Identification System Based on Wavelet Transform And Gaussian Mixture Model. *Journal of Information Science and Engineering*. 19 (1), 267-282.

Jaakkola, T. and Haussler, D. (1998). Exploiting Generative Models in Discriminative Classifiers. *Proc. of Tenth Conference on Advances in Neural Information Processing Systems*. 1(1),156-161.

Jarre, A. and Pieraccini, R. (1987). Some Experiments on HMM Speaker Adaptation. *IEEE International Conference on Acoustics, Speech, and Signal Processing*. Apr 1987. Universita' di Torino, Torino, Italy, 1273 - 1276.

Jialong, He., Li, L., and Palm, G. (1999). A Discriminative Training Algorithm for Vector Quantization Based Speaker Identification. *IEEE Transactions on Speech and Audio Processing*. 7(3), 353 - 356.

Jingwei, L., Qiansheng, C., Zhongguo, Z. and Minping, Q. (2002). A DTW-based Probability Model for Speaker Feature Analysis and Data Mining. *Pattern Recognition Letters*. 23(11), 1271-1276.

Jiuqing, D. and Qixiu, H. (2003). Open Set Text-Independent Speaker Recognition Based on Set-Score Pattern Classification. *2003 IEEE International Conference on Acoustics, Speech, and Signal Processing*. 6-10 April. Hong Kong, 73-76.

John, H. and Wendy, H. (2002). *Speech Synthesis and Recognition*. (2nd ed.) Bristol, PA, USA : Taylor & Francis, Inc.

Judith, A. M. (2000). Voice Biometrics. *Journal of Communications of the ACM*. 43(9), 66-73.

Julio, G. and Juan, C. O. (2005). Gender and Speaker Identification as a Function of the Number of Channels in Spectrally Reduced Speech. *Journal of the Acoustical Society of America*. 118(1), 461-470.

Karpov, E., Kinnunen, T. and Fränti, P. (2004). Symmetric Distortion Measure for Speaker Recognition. *Proc. 9th International Conference Speech and Computer*. 20-22 September. St. Petersburg, Russia, 366-370.

Kersta, L. G. (1978). Voiceprint Identification Infallibility. *Journal of the Acoustical Society of America.* 34(1), 171-179.

Kershaw, D.J. (1997). *Phonetic Context-Dependency in a Hybrid ANN/HMM Speech Recognition System*. Doctor Philosophy. University of Cambridge.

Kristie, S., Andrew, M. and Roni, R. (1999). Learning Hidden Markov Model Structure for Information Extraction. *AAAI 99 Workshop on Machine Learning for Information Extraction*. 18–22 July. Orlando, Florida, 235-241.

Kung, S.Y. , Mak, M.W. and Lin, S.H. (2005). *Biometric Authentication: A Machine Learning Approach*. (1st ed.). New Jersey, USA: Prentice Hall.

Lawrence, R. R. (1989). A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition. *Proceedings of the IEEE 1989.* 77(2), 257 - 286.

Lang, K., Waibel, A. and Hinton, G. E. (1990). Time-Delay Neural Network Architecture for Isolated Word Recognition. *Journal of Neural Networks*. 1(1), 23-43.

Linde, Y., Buzo, A. and Gray, R. (1980) An Algorithm for Vector Quantizer Design. *IEEE Transactions on Communications*. 28(1), 84-95.

Li, C.Y. and Chan, C. C. (2004). *A Prediction Scheme for Image Vector Quantization Based on Mining Association Rules*. In Li, C.Y. and Chan, C. C. (Ed.) *Book Series Lecture Notes in Computer Science.* (pp. 230-240) Heidelberg: Springer Berlin.

Louis, C. W., Xue, W. and Louis, F. M. (1996). Modelling of Phone Duration (using the TIMIT database) and its Potential Benefit for ASR. *Journal of Speech Communication.* 19 (2), 161-176.

Lupini, P. and Cuperman, V. (1995). Non-Square Transform Vector Quantization for Low-Rate Speech Coding. *Proc. IEEE Speech Coding Workshop.* 20-22Sept. Annapolis, Maryland, 87-89.

Lung, S.Y. (2007). Efficient Text Independent Speaker Recognition with Wavelet Feature Selection Based Multi-layered Neural Network using Supervised Learning Algorithm. *Journal of Pattern Recognition*. 40(12), 3616-3620.

Marie, R. (2006). Gaussian Selection Based Non-Optimal Search for Speaker Identification. *Journal of Speech Communication*. 48(1), 85-95.

Marston, D. F. (1995). Gender Adapted Speech Coding. *IEEE Trans. Acoustics, Speech and Signal Processing*. 1(1), 357–360.

Meng, Z., Jianhua, T., Jilei, T. and Xia,W. (2008). Text-Independent Voice Conversion Based On State Mapped Codebook. *IEEE International Conference on Acoustics, Speech and Signal Processing*. March 31 - April 4. Las Vegas, Nevada, U.S.A.,4605 - 4608.

Minghui, L., Yanlu, X., Zhiqiang, Y. and Beiqian, D. (2006). A New Hybrid GMM/SVM for Speaker Verification. *Proc. of the ICPR 2006*. 4 (1), 314-317.

Miyajima, C., Hattori, Y.,Tokuda, K., Masuko, T., Kobayashi, T. and Kitamura, T. (2001a). Text Independent Speaker Identification Using Gaussian Mixture Models Based On Multi-Space Probability Distribution. *IEICE Transactions on Information and Systems*. 84(7) ,847-855.