Speaker Recognition Algorithm to Facilitate Lab Assistance

By Balsam Zakaria Ishaq Khojah

A thesis submitted for the requirements of the degree of Master of Computer Science

FACULTY OF COMPUTING AND INFORMATION TECHNOLOGY
KING ABDULAZIZ UNIVERSITY
JEDDAH – SAUDI ARABIA
Safar 1437H- December 2015G

بسم الله الرحمن الرحيم

قال تعالى ﴿ الحَمْدُ لِلَّهِ الَّذِي هَدَانَا لِهَذَا وَمَاكُنَّا لِنَهْتَدِيَ لَوْلَا أَنْ هَدَانَا اللَّهُ ﴾
سورة الأعراف آية 43

Speaker Recognition Algorithm to Facilitate Lab Assistance

By Balsam Zakaria Ishaq Khojah

A thesis submitted for the requirements of the degree of Master of Computer Science

Supervised By
Dr. Wadee Saleh Alhalabi

FACULTY OF COMPUTING AND INFORMATION TECHNOLOGY
KING ABDULAZIZ UNIVERSITY
JEDDAH – SAUDI ARABIA
Safar 1437H- December 2015G

خوارزمية تحديد المتحدث التي تستخدم للمساعدة في المعامل والتعليم عن بعد

بلسم زكريا إسحاق خوجة

بحث مقدم لنيل درجة الماجستير في علوم الحاسبات

إشراف:
د. وديع صالح الحلبي

كلية الحاسبات وتقنية المعلومات
جامعة الملك عبدالعزيز
جدة – المملكة العربة السعودية
ديسمبر 2015 م – صفر1437 هـ

Speaker Recognition Algorithm to Facilitate Lab Assistance

By
Balsam Zakaria Ishaq Khojah

This thesis has been approved and accepted in partial fulfillment of the requirements for the degree of Master of Computer Science

EXAMINATION COMMITTEE

	Name	Rank	Field	Signature
Internal Examiner	Dr. Fadi Fouad Foz	Professor	Computer Science	
External Examiner	Dr. Ali Hussein Morfeq	Assistant Professor	Electrical Engineering and Computer Engineering	
Advisor	Dr. Wadee Saleh Alhalabi	Assistant Professor	Computer Science	

KING ABDULAZIZ UNIVERSITY
Safar 1437H- December 2015G

خوارزمية تحديد المتحدث التي تستخدم للمساعدة
في المعامل والتعليم عن بعد

إعداد
بلسم زكريا إسحاق خوجة

تمت الموافقة على قبول هذه الرسالة استكمالا لمتطلبات درجة الماجستير في علوم الحاسبات

لجنة المناقشة و الحكم على الرسالة

	الاسم	المرتبة العلمية	التخصص	التوقيع
عضو داخلي	د. فادي فؤاد فوز	أستاذ	علوم حاسبات	
عضو خارجي	د. علي حسين مرفق	أستاذ مساعد	الهندسة الكهربائية وهندسة الحاسبات	
مشرف رئيس	د. وديع صالح الحلبي	أستاذ مساعد	علوم حاسبات	

جامعة الملك عبدالعزيز
ديسمبر 2015 م – صفر 1437 هـ

Dedicated to

Most important person in my life, my mother

ACKNOWLEDGEMENTS

In the Name of Allah, the Most Merciful, the Most Compassionate, all praise be to Allah, the Lord of the Worlds, and prayers and peace be upon Mohamed, His servant and messenger.

First and foremost, I must acknowledge my limitless thanks to Allah, the Ever-Magnificent, the Ever-Thankful, for His help and blessing. I am sure that this work would have never been completed without His guidance.

I am grateful to the people who worked hard with me from the beginning to the completion of this present research, particularly my supervisor, **Dr. Wadee Alhalabi**, your efforts, motivation, encouragement, patience and extraordinary support were the light that directed my way to reach my dream. Working with you was my honor and I could not be able to finish this thesis without your professional guidance.

My great father, thank you for believing in me in every step of my life and especially in the last years of my education. Without your advice, I would not be here.

My precious mother, your warm heart, unconditional love and prayers are priceless. I could not ask for more.

My husband, you shared with me the ups and downs in my work with care, hope and understanding. My academic life won’t be successful without your company.

My sisters and brothers, thank you for your cooperation until I reached thesis completion with pride.

My children, Lujain, Abdullah, and Hibah, you are a blessing from Allah that always motivate me to go beyond my expectations.

Speaker Recognition Algorithm to Facilitate Lab Assistance

Balsam Zakaria Ishaq Khojah

Abstract

The increasing need of human-machine interaction in our daily life leads to rapid advances and developments. Voice biometry is a powerful measurement technology, which can provide rich information useful in a multitude of circumstances. Words are not the only understandable information that can be gathered from speech. Listeners can know gender, age, health situation, emotion state, and speaker identity. The speech processing field is concerned with understanding all these aspects, depending on the application, a multitude of factors being considered. Automatic Speaker Recognition (ASR) in particular is the ability of a program or a device to identify from its utterance who is speaking. ASR is usually divided into Speaker Verification (SV) and Speaker Identification (SI). SV applications relate to the security area because it's scope is verifying if the unknown speaker belongs to the system. On the other hand, SI is the process of identifying an unknown speaker from groups of enrolled speakers. SI is generally used for audio conferencing and similar applications.

Our study is focused on proposing a speaker identification system (SIS) that identifies effectively all registered speakers based on their speech. SIS is composed of two main modules: feature extraction and feature matching. Mel Frequency Cepstrum Coefficients (MFCC's) is used for extracting features from speech signal. For feature matching, we have applied four common ASR algorithms: Vector Quantization (VQ), Gaussian Mixture Models (GMM), Artificial Neural Networks (ANN), and Decision Trees (DT).

The speaker database used in SIS is composed of 120 speakers. The procedure followed for the proposed SIS is to train then test different sizes of speaker databases by extracting their features. First MFCCs is applied and then, in the training and testing phases, VQ, GMM, ANN, and DT algorithms are used. Our system tries to improve the identification rate by fusing the identification results of the four algorithms by majority decision method. Identification rate results show that fusion method gives better results than VQ, ANN, and DT algorithms when applied separately. Compared with other methods, GMM provides the best identification rate and the lowest rate of speaker misidentification.

When evaluating performance, the most accurate procedure for identifying true speakers and rejecting imposters is the fusion method. VQ algorithm is the second best in accuracy. Fusion method have accuracy rate of 99% for 100 speakers and 96% for 25 and 50 speakers. The proposed SIS in this study proved its ability to identify a text-independent closed set of speaker groups effectively.

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER 2: LITERATURE REVIEW

CHAPTER 3: PROPOSED METHOD

CHAPTER 4: RESULTS

CHAPTER 5: DISCUSSION

LIST OF TABLES

LIST OF SYMBOLS AND TERMINOLOGY

ANN	Artificial Neural Networks
ASR	Automatic Speaker Recognition
BP	Back propagation
CART	Classification and Regression Trees
DCT	Discrete Cosine Transform
DFT	Discrete Fourier Transform
DT	Decision Trees
DTU	Technical University of Denmark
ED	Euclidian distance
ELSDSR	English Language Speech Database for Speaker Recognition
EM	Expectation Maximization
FA	False Acceptance
FFT	Fast Fourier Transform
FN	False Negative
FP	False Positive
FR	False Rejection
GMM	Gaussian Mixture Models
GPL	General Public License
HMM	Hidden Markov Models
IMFCCs	Inverted Mel Frequency Cepstrum Coefficients
LBG	Linde, Buzo, and Gray
LPC	Linear Predictive Coding
MFCCs	Mel-Frequency Cepstral Coefficients
ML	Maximum Likelihood
MLP	Multilayer Perceptron Network
PIN	personal identification number

SI	Speaker Identification
SIS	Speaker Identification System
SR	Speaker Recognition
SV	Speaker Verification
SVM	Support Vector Machine
TI	Texas Instruments
TN	True Negative
TP	True positive
VQ	Vector Quantization

Chapter 1

Introduction

Chapter I

Introduction

1.1 Introduction

Communicating with other people around the world for different purposes is a daily necessity in our lives. Due the rich information that can provide, speech is the most powerful form of communication. The information provided by speech is vast and include gender, attitude, emotion, health situation and identity of a speaker. These aspects can be extracted using speech processing techniques. Speech processing is a diverse field with many applications and Speaker Recognition (SR) area is one of these applications that has a great research attention and rapid advancements with the increase of human-machine interactions.

Automatic Speaker Recognition (ASR) is the ability to extract, characterize and recognize the information about speaker identity [1]. Speaker recognition is usually divided into Speaker Identification (SI) and Speaker Verification (SV). Both SI and SV need a stored speaker database as a reference model for pre known speakers. In speaker verification there is an identity claim and the SV job is to verify the claimed identity through binary decision for acceptance or rejection [2] [3] [4]. The main applications of SV are in the security area. In case of speaker identification there is no identity claim, and the SI job is to identify the unknown speaker from groups of known speakers [2] [3] [4]. Speaker identification system is divided into closed-set and open-set system [2]. Closed-set identification means the unknown speaker must have pre-recorded data in the speaker database and the result should give the best match between the unknown speaker and one of enrolled speakers. Open-set identification means that unknown speaker doesn't find a match from enrolled speakers and a possible result is that the unknown speaker is an imposter and we have to add the category "unregistered" to the system [4]. According to speech modalities, SV and SI can be text-dependent or text-independent. Text-dependent means that in order to perform the identification a fixed text must be spoken whereas in text-independent, the speaker can speak freely. Although text-independent adds more flexibility to the system, it reduces the accuracy and is more vulnerable to mistakes.

Any speaker identification system should pass three steps: feature extraction, training, and testing. There are several algorithms can be used for identification procedure. Our research is proposing a closed-set speaker identification system. Text-independent is the speech modality used here.

1.2 Motivation

Suppose that there is a group of scientists from all over the world that need to exchange their last experiments in an audio conference. When a scientist starts to talk, all the listeners know his name, specialty, and a brief biography of his scientific work. Another example is if there is a group of students in a class, their teacher is in another location and he needs to take the attendance or know who is participating during the class. In this case, there is a need for a program or a special device that works as a lab assistant, analyzing the speech signal, identifying who is speaking, and producing the result to the teacher.

1.3 Problem Statement

With the great existence of educational technologies that replaces the face-to-face learning such as e-learning, several tools are needed to upgrade the educational process. Some types of e-learning methods include online tests or presentations and there is a need for an attendance system that lets the teacher know who is presenting or attending.

Our university and many Saudi Arabia universities have a separated section for female students. The teacher may present in another location and the only way for communication is the audio communication. When the student wants to participate to a set of classes, her voice is the only biometric available to the teacher. Having a program or a device that acts as a speaker identification system can assist the teacher in identify who is speaking. Taking the attendance also is a task that can be performed with a speaker identification system.

1.4 Thesis Objectives

This research aims to propose a speaker identification system that identifies a closed-set of speakers. The main goal is to use well-known speaker identification algorithms for training and identification purposes and upgrade the identification results so that the proposed system can be efficiently used for speaker identification applications such as lab assistant.

1.5 Research Methodology

To achieve thesis objectives the following steps are performed:

1. Study the general architecture for building a speaker identification system and choose the most suitable techniques that can be followed in this study.
2. Search for convenient speaker corpora that can be used in speech researches.
3. Organize the speaker audio files by separating them into train and test folders and label the speaker folder by an identification number.
4. Study Mel Frequency Cepstrum Coeffients as feature extraction method.
5. Study vector quantization, Gaussian Mixture Models, Artificial Neural Networks, and Decision Trees as four algorithms to be applied.
6. Write the feature extraction part and speaker identification algorithms part as an executable functions in Matlab.
7. Build the system by assembling the code parts needed for feature extraction, training, and testing.
8. Test the program system on the speaker databases collected for this study to see if any changes or modification needed.
9. Extract features from speaker speech signals to be ready for training.
10. Train all speakers by the four SI algorithms to build a reference model for each enrolled speaker in the system.
11. Test all speakers by the four SI algorithms to do the identification part.
12. Compare the identification results between the four SI algorithms.
13. Modify the identification results by fusion method.
14. Evaluate the SI algorithms and fusion method to measure system performance.

1.6 Thesis Organization

The thesis is organized into six chapters. Chapter 2 provides a literature review about speech processing and production, speaker recognition concepts, background of speaker recognition and its applications. In addition, chapter 2 focuses on the basic structure of speaker identification system and its general modules. Chapter 3 discusses the methodology in details followed in this study. In chapter 4, all the experiments and results findings are illustrated. A detailed discussion to analyze results in chapter 5. Chapter 6 concludes the work, with a special emphasis on results and limitations. Also, some directions for future work are suggested.

Chapter 2

Literature Review

Chapter II

Literature Review

2.1 Speech Processing

Among various ways of human communications like speech, body language, textual language, and pictorial language, speech is a powerful source for communicating due to the rich information it contains. Speech is a complicated signal produced as a result of several transformations occurring at several different levels: semantic, linguistic, articulatory, and acoustic [5]. As speech signal is a carrying message of diverse information, speech processing field has many different applications depending on the kind of information we are interested in. Speech processing is the extraction of the needed information from a speech signal to be digitalized and processed by the computer [6]. Figure 2.1 shows that speech processing is a diverse. Analysis, recognition, and coding are the main fields of research in speech processing [7].

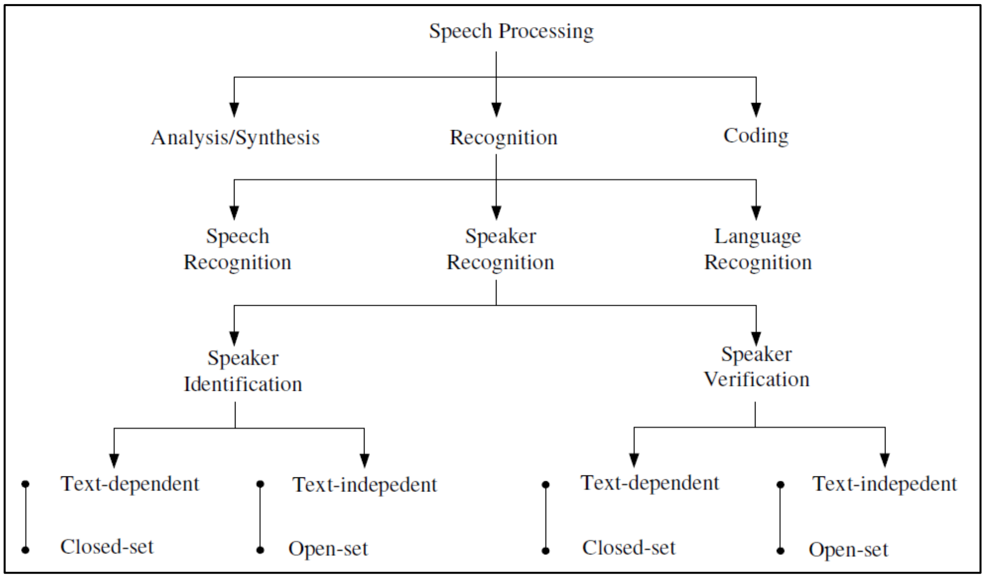


Figure 2.1 Speech Processing Taxonomy

Recognition field in particular is the most area that has been studied for several decades. From figure 2.2 as mentioned in [8], recognition system is divided into three subsystems: Speech, Speaker and Language recognition systems and their results speech text, language, and speaker identity respectively. As in this thesis, we are focusing on speaker recognition field, we will discuss in detail the speaker recognition concepts and principles.

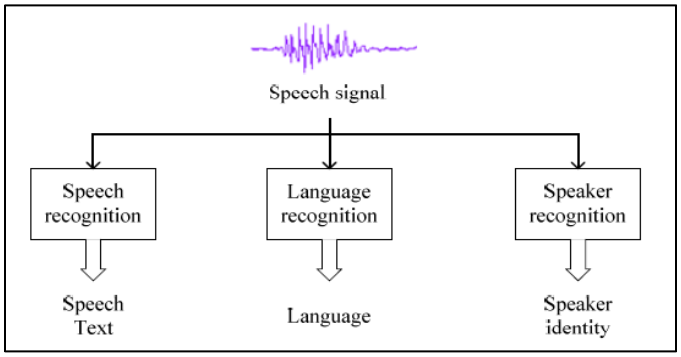


Figure 2.2 Recognition Field's Parts and Their Outputs

2.2 Speech Production

Speech has a rich dimension character because the speech signal carries a message information in its waveforms. Words, gender, emotion, health situation, attitude, and identity are various character dimensions in speech signal [7]. Therefore, speech signal is a powerful biometric that can be used as a voiceprint in certain applications. Physical anatomy and speak behavior are different between humans and no two individuals have an identical voice [2, 5]. The sound producing organs (like vocal tract length, larynx shape, and other parts) and the mannerisms of speaking (like accent, pronunciation, rhythm, and tone) make up the specific characteristics for each speaker. Speech is the result of a complex procedure happening in the speaker's respiratory system [9]. Figure 2.3 shows the elements that contribute to produce the voice [10]. The human speech production can be divided into three main groups: lungs, larynx and the vocal tract [11].

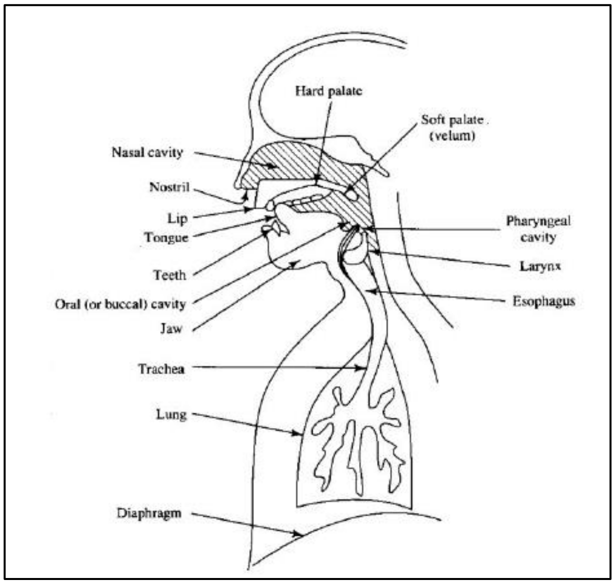


Figure 2.3 Respiratory System Involved in Speech Production

2.2.1 Lungs

From the speech production point of view, lungs are the power source that supplies energy to the rest of the respiratory system. Breathing is the process of inhalation and exhalation that are rhythmically repeated. However, when the speaker talks, the normal breathing is overridden until the end of a sentence or phrase time. The muscles around the rib cage control the little air taken and released during speaking.

2.2.2 Larynx

Larynx or “voice box” has two important parts for voice production: the vocal folds and the glottis. Vocal folds create unvoiced sounds when they are open or vibrate to produce voiced sounds. The quick open and close of the airflow exit by vocal folds is known as the Bernoulli’s Principle in the glottis, which is explained in [5].

2.2.3 Vocal Tract

The term vocal tract refers to the organs above the larynx that are responsible for voice production. They are the pharyngeal, oral, and nasal cavities and the velum. The length and shape of the vocal tract is the main source for extracting different speech features [2, 9]. From the spectral shape we can estimate the vocal tract shape because when the acoustic wave passes through the vocal tract, its spectrum is altered by the formants of the vocal tract [6].

2.2.4 The Human Vocal Mechanism

The human vocal mechanism needs to be excited by an excitation source. The excitation can be phonation, whispering, frication, compression, vibration, or a combination of these [6]. The excitation is generated when the airflow comes from the lungs and goes into the vocal cords through the trachea. The shape of the glottis is manipulated during speech production that causes an irregular airflow called the glottal source [12]. Speech sounds is classified into voiced, unvoiced or mixed excitation sounds [10]. Voiced sounds (characterized by periodicity and high energy) cause the vocal cords to vibrate. Unvoiced sounds are produced with no vibration, like white noise. Mixed sounds happen when there is a constriction in the vocal tract besides the vibration from the vocal cords.

2.3 Speech Recognition and Speaker Recognition

The goal of speech recognition system is to recognize what are the spoken words, while speaker recognition system aims to recognize who is the speaker with no need to understand what is being said [13] and this is our concern in this study.

2.4 Speaker Recognition Principles

The main goal of speaking is to deliver an understandable message via words. In addition, many characteristics can be known from part of speech, such as language, emotions, gender, and identity of speaker. [14] defines speaker recognition as the identification of a speaker from his/her speech features. To build a speaker recognition system we have to automate the speech waveform by using some measurements that are capable of identifying who is speaking. The main goal of the automatic speaker recognition is to extract, characterize and recognize the information about speaker identity [13]. Speaker recognition area itself is usually divided into speaker identification and speaker verification systems. Some researches added speaker segmentation and clustering [4]. The author of Fundamentals of Speaker Recognition book [15] divided the six speaker recognition branches into two groups: simple and compound. Speaker verification, speaker identification, and speaker classification are the simple group. The compound group is represented by speaker segmentation, speaker detection, and speaker tracking. The data needed in these systems is categorized into text-dependent and text-independent methods. The following sections will explain some of these principles in details.

2.4.1 Speaker Verification

The most popular branch in speaker recognition is speaker verification because it has a useful contribution in security and access control. Speaker verification is the process of verifying whether the person who claims to be has the same speaker identity with the one already enrolled in the system [7]. It can be considered as true/false binary decision because it is a one-to-one comparison between the claimed speaker (X) and the preregistered speaker (Sc) as depicted in figure 2.4 (a) from [5]. The decision of accepting or rejecting the claimed identity should be chosen carefully to avoid type-I and type-II errors [2]. Type-I error or False Rejection (FR) occurs when the verification system reject a speaker who is really registered in the system. Type-II error or False Acceptance (FA) occurs when the verification system accepts an imposter as a true speaker. Most of commercial speaker recognition applications rely on verification for security purposes like bank services and communication control access.

2.4.2 Speaker Identification

Speaker identification is more complicated than speaker verification due to the need of comparing the whole group of preregistered speakers' voices with the input speaker. So, speaker identification is the process of finding the identity of an unknown speaker (X) by comparing his/her voice with voices of a group of (N) registered speakers in the database (S1,S2,…,SN) as shown in figure 2.4 (b) [5] [7]. This system can be divided into closed-set and open-set system. Closed-set means that the unknown input speaker must be one of the enrolled speakers that are already stored in the database of the system. Open-set means that the unknown input speaker may or may not be from the preregistered speakers in the system and this will add the option "unknown" to the results [7] [2] [4].

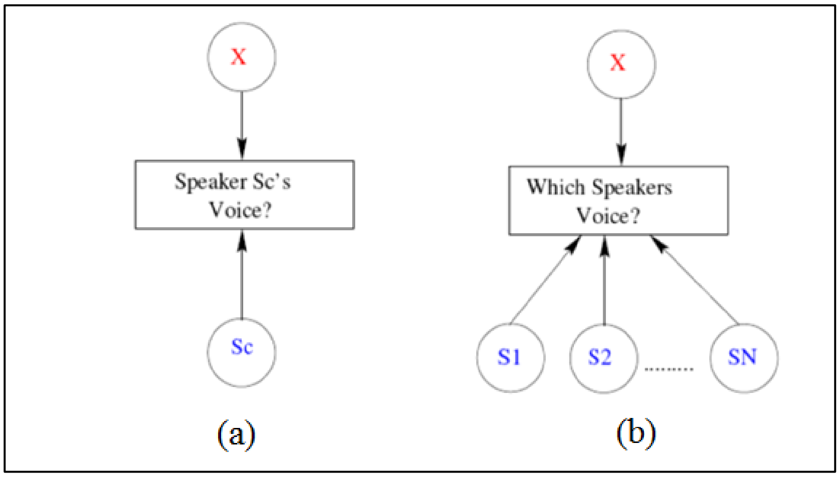


Figure 2.4 (a) Speaker Verification (b) Speaker Identification

2.4.3 Text-Dependent vs. Text-Independent Systems

Depending on the way of training/testing the spoken text of speakers, speaker recognition speech modalities are divided into text-dependent and text-independent [2] [4] [16]. In text-dependent systems, the spoken text is previously known in both training and testing phases, where in text-independent systems the speaker is allowed to speak freely. System relies on text-independent model should be intelligent to figure out the distinguishing speech characteristics of vocal sounds that belong to speakers participating in that system [2]. On the other hand, the text-dependent system needs specific phrases to be spoken (like card number, passwords etc.) [5] which leads to better speaker recognition results. Even if the text-independent system does not know the spoken words or phrases, it provides flexibility in speaking and there is no need for co-operation from speakers.

2.5 Background of Speaker Recognition

In the early 1960s, at Bell labs, Lawrence Kersta developed a spectrographic voice identification called voiceprint analysis or visible speech and this is considered a major step in automatic speaker identification [4] [17] [7]. Since the mid-1980s, there has been a great concern in speaker recognition discipline in industry, national laboratories, and universities. The famous labs and institutions that have researched and designed several generations of speaker-recognition systems are AT&T (and its derivatives); Bolt, Beranek, and Newman; the Dalle Molle Institute for Perceptual Artificial Intelligence (Switzerland); ITT; Massachusetts Institute of Technology Lincoln Labs; National Tsing Hua University (Taiwan); Nagoya University (Japan); Nippon Telegraph and Telephone (Japan); Rensselaer Polytechnic Institute; Rutgers University; and Texas Instruments (TI) [6]. In [6], the selected chronology of speaker-recognition progress has been shown in a table. At the date when the book was published, the author of Fundamentals of Speaker Recognition [15] estimated that in speaker recognition area, there are more than 3500 research papers. With the assistance of human listeners, in the 1950s, an early research had been done by analyzing speaker speech in order to find voices distinguishing personal characteristics [18] [19]. With the rising of communication networks, the need of speaker identification is an important aspect [18]. The early works in speaker recognition chose text-dependent analysis to simplify the identification [20]. In [21] and [22], an automatic statistical comparison of speakers using a text-dependent approach was done by analyzing a population of 10 speakers uttering several unique words. However, using text-dependent in speaker identification is not practical but it may be more useful in speaker verification systems.

From that time until now, speaker recognition research has developed rapidly. The need for voice biometric in commercial and industrial technology has increased for the unique characteristics of speech signal. Speaker recognition is the only biometric that can be tested as it transfers over long distances and this make it more valuable with the growing complexity of cellular telephones [15]. It is easily collected even without speaker knowledge by existing infrastructure, and can't be stolen, lost, or forgotten [5] . Several parametric and non-parametric classifier approaches are used for speaker recognition like Gaussian Mixture Models (GMM) [23], Hidden Markov Models (HMM) [24], Vector Quantization (VQ) [25], Support Vector Machine (SVM) [26], Artificial Neural Networks (ANN) [27], and Decision Trees (DT) [28]. Depending on several factors, some of pattern matching models proved high performance among others. In this study, we are choosing the most dominant approaches that have a high identification rates in text-independent speaker identification field and we perform a comparison between them.

2.6 Applications

Speaker recognition technologies are used in wide range of applications. Here we are mentioning some examples of most popular applications that use speaker recognition technology [5] [2] [4] [15]:

- Security :
    - Transaction authentication.

    - Facility or computer access control.
    - Monitoring or Surveillance.
    - Telephone voice authentication for long distance calling or banking access.
    - Credit card validation or Personal Identification Number (PIN) entry.
    - Confidential information Forensic purposes.
    - National border control to monitor the movements of individuals in and out of the country.
- Personalization:
    - Intelligent answering machines (personalized dialog systems)
    - Voice command and control.
    - Voice dialing in hands-free devices.
    - Biometric Login to telephone aided shopping systems Information and Reservation Services.
    - Telephone-Banking/Booking.
    - Information Retrieval.
    - Personalizes the content of an entertainment source (like interactive game play).
- Audio Indexing:
    - Build an indexing speaker database.
    - Automatic speaker labeling of recorded meetings for speaker-dependent audio indexing.
- Speaker Tracking:
    - Attendance systems.

- Tele-conference speaker identity especially when participants don’t know each other or there are many of them.
- Free and paid sites meetings.
- Proctorless Oral Testing.

2.7 Basic Structure of Speaker Identification System (SIS)

Any Speaker Identification System (SIS) has two basic modules: feature extraction and feature matching (pattern matching) [29] [2] [7] [4]. Feature extraction is the process of extracting set of features from a speech signal that represent a specific speaker. Feature matching is the actual procedure of recognition, which finds the best match between the extracted features of unknown input voice signal and the set of predefined speakers stored in database of the system.

The speaker identification process has two phases: enrollment phase (training) and identification phase (testing) [5] [30] [7]. In the training phase, speakers enroll into the system by collecting their voices to build a reference model for each speaker then store the models in a speaker database. In the testing phase, to make a decision of speaker identity, a test voice of unknown speaker is entered to the system and compared with all the reference models registered during training phase. The comparison can be done by probability density estimation like in GMM model or by measuring the distance like in VQ model [30] [31] [29] .

Both training and testing phases should start with feature extraction module to extract the speaker dependent features from the speech signal and that’s why in some works, the processes are divided into three processes: feature extraction, training, and testing [2]

[29]. In SIS, the main components of training and testing phases is shown in the block diagram from [2] in figure 2.5.

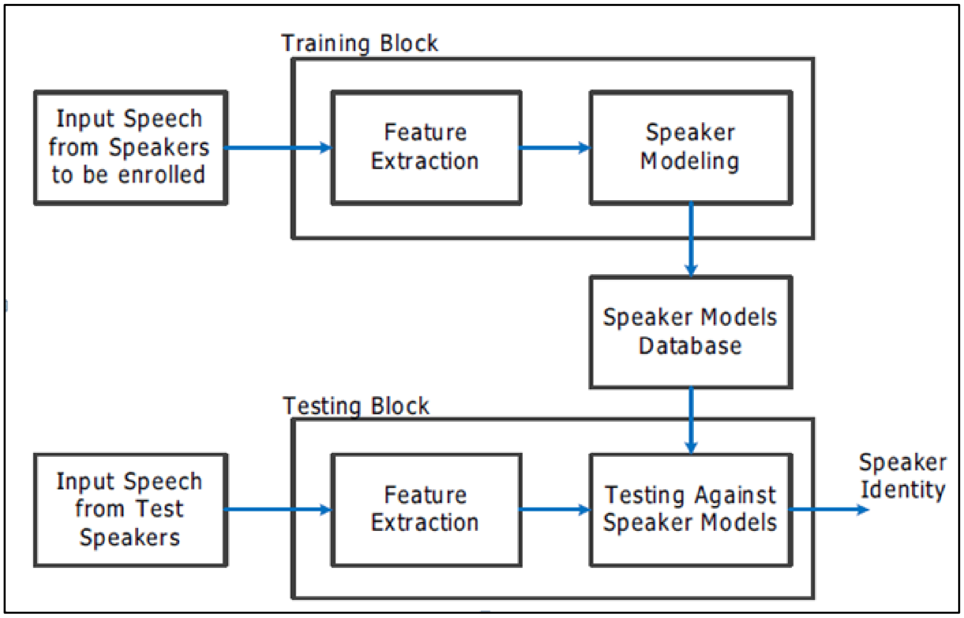


Figure 2.5 Block Diagram of a SIS

2.7.1 Feature extraction

Feature extraction or front-end processing aims to convert the speech waveform to some type of parametric representation for further analysis and processing in order to produce the speaker discriminative features.

The speech signal characteristics are stationary in short period (between 5 and 100 msec). However, these characteristics start to change to reflect speaker-specific information when there is enough period of time (1/5 sec or more) [9]. Therefore, the speech signal is a slowly time varying signal and, to characterize the speech signal we use the most common way: *short-term spectral analysis* [32]. Short-term spectral analysis carries the speech in frequency domain with short segments of speech through a sequence of analysis, as depicted in figure 2.6 [7] [9] and explained in detail in the following subsections:

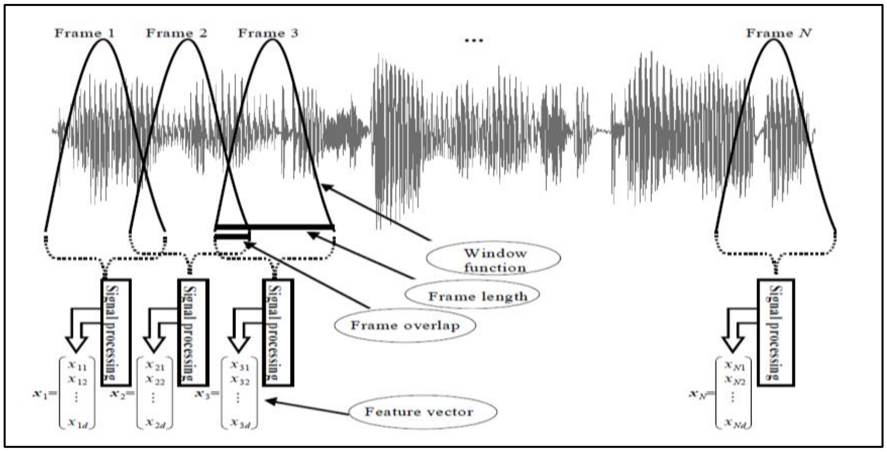

Frame 1
Frame 2
Frame 3
...
Frame N
Window function
Frame length
Frame overlap
Signal processing
Signal processing
Signal processing
Signal processing
$x_1 = [x_{11}, x_{12}, \vdots, x_{1d}]$
$x_2 = [x_{21}, x_{22}, \vdots, x_{2d}]$
$x_3 = [x_{31}, x_{32}, \vdots, x_{3d}]$
$x_N = [x_{N1}, x_{N2}, \vdots, x_{Nd}]$
Feature vector

Figure 2.6 Short Term Spectral Analysis

2.7.1.1 Signal Pre-Processing

Before extracting feature vectors from the speech signal, a pre-processing procedure should be done. When the speaker speaks by the microphone, the continuous speech signal is converted into discrete domain. Then, segmentation is done on the speech signal to get overlapped frames to obtain quasi-stationary units of speech. A pre-emphasis filter is applied to each frame to enhance the speech by lifting the high frequency spectral components of speech. Now, the speech signal is ready for the feature extraction subsystem.

2.7.1.2 Mel-Frequency Cepstral Coefficients (MFCCs)

Mel-Frequency Cepstral Coefficients (MFCCs) is the most common feature extraction technique used for speaker identification. The technique of computing MFCC is based on the short-term analysis, and from each segmented frame a MFCC vector is computed

[5]. Identification process is mainly affected by feature extraction step for effective modeling. In this study we chose MFCCs for feature extraction for the following reasons [33] [5] [34] [31]:

- Based on the known variation of the human ear's critical bandwidths with frequency because filters spaced linearly at low frequencies and logarithmically at high frequencies have been used to capture the phonetically important characteristics of speech.
- Easy to extract and measure because its features occur frequently and naturally in speech.
- It is not affected by ambient noise.
- It is used as a standard acoustic feature for speaker recognition systems.
- Has the ability to detect speech characteristics even in low frequency regions.

The MFCCs processor performs the following processes as illustrated in the block diagram of figure 2.7 from [5]:

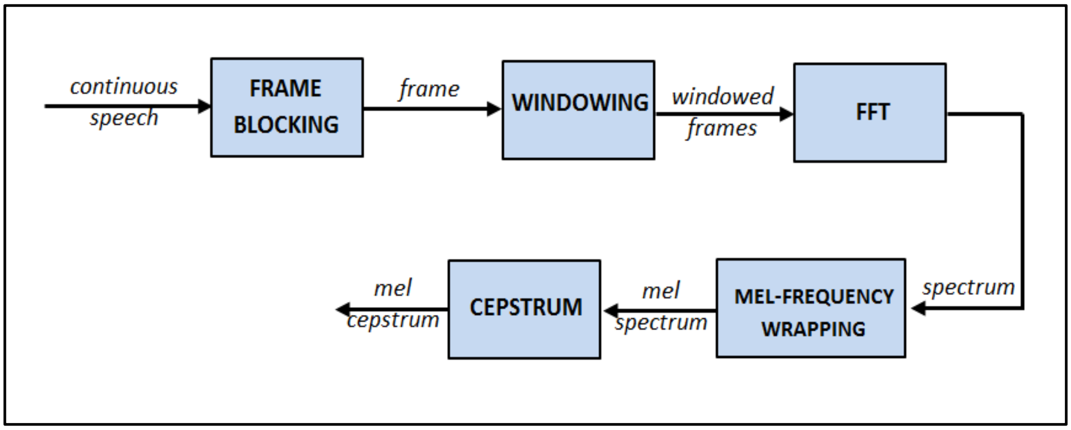


Figure 2.7 Block Diagram of MFCCs Processor

1. Framing

Framing means partitioning a continuous speech signal into adjacent number of segments or frames, as depicted in figure 2.6. Overlapping frames with 30% to 50% are done to avoid losing any information [34].

2. Windowing

To minimize the discontinuities in the signal due to framing, windowing is applied at the beginning and end of each frame [29]. This is done by multiplying each frame with a window function w(n) of length N, where N is the length of the frame [35]. Several window functions can be used. Typically, Hamming window is used because it has accurate spectral estimation for its frequency response [9]. Hamming window function offers the bell shaped weighting function with no zero at the edges of the window [36] [37]:

$$w(n) = 0.54 - 0.46\cos(2\pi n/N - 1) \quad ,where \quad 0 \leq n \leq N-1 \qquad (2.1)$$

If we define the signal before windowing as $x(n),$ the result of windowing will be the signal $y(n)$ where:

$$y(n) = x(n)w(n) \qquad ,where \quad 0 \leq n \leq N-1 \qquad (2.2)$$

3. Fast Fourier Transform (FFT)

The aim of this step is to convert each frame in time domain to frequency domain to obtain the magnitude frequency response from each frame [29]. Fast Fourier Transform (FFT) is a name given to fast algorithms to compute the Discrete Fourier Transform (DFT) [38]. The FFT algorithm reduces the time complexity when calculating DFT from $O(n^2)$ to $O(n \log n)$ [39]. FFT is defined on the set of N samples $\{x_n\}$, as follow [5]:

$$X_n = \sum_{k=0}^{N-1} x_k \; e^{-2\pi jkn/N} \qquad ,n = 0,1,2,\dots,N-1 \qquad (2.3)$$

The resulting sequence $\{x_n\}$ is interpreted as follows:

Zero frequency: $f = 0 \qquad ,when\, n = 0 \qquad (2.4)$

Positive frequency: $0 < f < F_s \qquad ,when\; 1 \le n \le N/2\, -1 \qquad (2.5)$

Negative frequency: $-F_s/2 < f < 0\; ,when\;\; N/2 + 1 \le n \le N - 1 \qquad (2.6)$

F_s denotes the sampling frequency. The result after this step will give a spectrum.

4. Mel-Frequency Wrapping

The goal of this important step is to convert the frequency spectrum to Mel spectrum. The signal now is in actual frequency, $f,$ measured in Hz. Because the human ear frequency perception is non-linear, we need to convert it into Mel-frequency by “Mel-scale” [40] . To compute the “Mel” from given frequency “f”, the following formula is used:

$$mel(f) = 2595 * \log_{10}(1 + f/700) \quad ,where\; f\; is\; frequency\; in\; Hz \quad (2.7)$$

To simulate the human perception, a filter bank is built with bandwidth given by the Mel scale and pass the magnitudes of spectra through bank filters to produce the Mel-frequency spectrum [15]. The filter banks are a set of triangular windows spaced uniformly on the Mel scale as shown as an example in figure 2.8 from [9]. This Mel-frequency wrapping step keeps only the part of useful information [29].

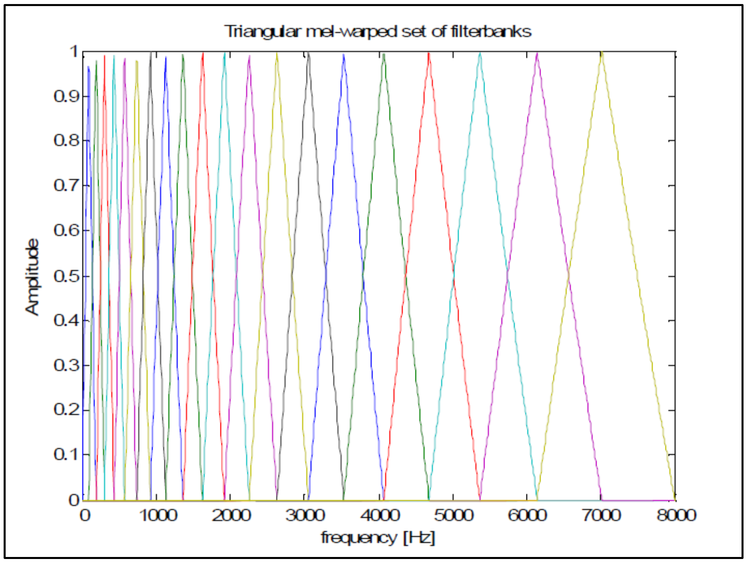


Figure 2.8 Mel-Warped Filter Bank with 20 Frequency Bands

5. Cepstrum

To get the result that is called Mel Frequency Cepstrum Coefficients (MFCC), the final step is converting the log spectrum back into time domain. To obtain the Cepstrum, first a logarithmic power spectrum is calculated to be the new analysis window, and then, an inverse FFT is performed to get a signal in time domain [5]. Therefore, cepstrum is defined as the inverse Fourier transformation of the logarithm of the magnitude of the Fourier transformation [41] and it is denoted mathematically as:

$$c(n) = ifft(\log|fft(s(n))|) \qquad (2.8)$$

Discrete Cosine Transform (DCT) is used to convert the Mel spectrum and their logarithm into time domain. The MFCCs is calculated using this equation [5]:

$$\tilde{c}_n = \sum_{k=1}^{K} (\log \tilde{s}_n) \cos\left[n\left(k - \frac{1}{2}\right)\frac{\pi}{K}\right], \mathrm{n} = 1,2, \dots, \mathrm{K} \qquad (2.9)$$

2.7.2 Pattern Matching

The pattern matching module resembles (in its role) the brain of a human when it registers the voices characteristics of each speaker. In the testing phase, the pattern matching identifies a certain speaker by comparing the unknown voice with the set of preregistered and stored voices. The identification procedure is actually done in this module, where a proper classification algorithm is used to get the likelihood score for each speaker.

Feature extraction is applied on a speaker's utterances to produce the feature vectors that are used to train the speaker model. This is called the enrollment phase. The result of this phase is creating the enrolled speaker database. Testing phase starts when an unknown speaker enters the system and needs to be identified as one of the speaker database. This is done by choosing the best matching model. There are two main classes of pattern matching algorithms: non-parametric (template) models and parametric (stochastic) models [6]. Non-parametric models, as appears from the name, has no parametric assumptions and it needs more data to figure out the distribution and get the optimal solution [42]. Some of the common non-parametric models are discussed here are (VQ), (ANN), and (DT). Parametric models need parameters to build their structure and these parameters are adjusted to fit the distributed data. These approaches have faster computation times than non-parametric models [2]. GMM is a common example for a parametric model.

2.7.2.1 Vector Quantization (VQ)

To apply an example of template or a non-parametric model, vector quantization algorithm is chosen to be tested for speaker identification. VQ is one of the most effective and widely used in pattern matching techniques for automatic person identification systems [36] [43]. VQ can be defined as a lossy data compression method that chooses (from a large vector space) a finite set of feature vectors represented in clusters. Each cluster has a center called “centroid” that is the mean value of all data belonging to the same cluster. The whole set of centroids is called “codebook”. A distortion measure is required in VQ to do the matching between the input vector and its centroid in the codebook. The index of the centroid becomes the new value of the input vector, which has the smallest distortion in the codebook [31].

2.7.2.1.1 Linde, Buzo, and Gray (LBG) algorithm

To generate the codebook used for each speaker modeling, the Linde, Buzo, and Gray (LBG) algorithm is employed as follows [44]:

1. Initialization:

   Is the design stage M=1. The N vectors available in training are calculated to produce the mean value as the initializing centroid or codevector of the training vectors $C_1(0)$ and it is given by:

$$C_1(0) = \frac{1}{N}\sum_{n=1}^{N} x_n \qquad (2.10)$$

   where x_n is the nth vector in the training.

2. Splitting:

   Now, each codevector in the codebook is split into two with a splitting parameter ϵ, where the new codebook C_{M+1} is given by:

$$C_{M+1} = (1+\epsilon)C_M(k) \tag{2.11}$$

$$C_{M+1}(2^{M-1}+k) = (1-\epsilon)C_M(k) \tag{2.12}$$

   where $k = 1,2, \dots , 2^{M+1}$, and $\epsilon < 1$. The value of the M is incremented by 1.

3. Optimization:

   The optimization process has two steps:

   - Partitioning: each codevector is assigned to $C_M(k),$ which minimizes the distortion $\|x_n - C_M(k)\|$, where $\| \,.\, \|$ is the norm.
   - Updating: each codebook entry is updated by calculating the mean of the training vectors belonging to a cluster, reducing the quantization error in each of the clusters.

The optimization process is repeated until the average distortion within the cluster is below a predefined threshold.

4. Steps 2 and 3 are repeated until the number of codevectors is converged.

To explain the LBG algorithm in another way, the following flowchart of LBG in figure 2.9 from [45] is used. The LBG algorithm goal is to cluster a set of M codevectors. In the figure 2.9 of the flowchart "Cluster vectors", the algorithm assigns each training vector to a cluster associated with the closest codeword by the nearest-neighbor search procedure. "Find centroids" updates the centroid at each iteration. "Compute D (distortion)" performs the summation for the distances of all training vectors to determine whether the procedure has converged.

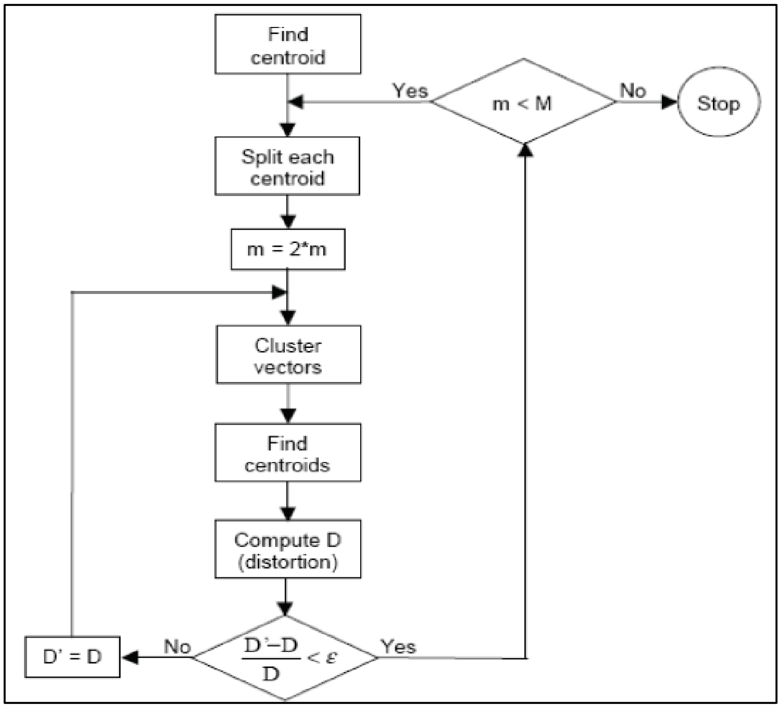


Figure 2.9 Flowchart of LBG Algorithm Implementation

2.7.2.1.2 Euclidean Distance Computation

Euclidian distance (ED) is a way to measure the VQ-distortion. The VQ-distortion is the distance from a vector to the closest codeword of a codebook [5]. ED is given as [46]:

$$ED = \sqrt{(x - x_1)^2 + (y - y_1)^2} \qquad (2.13)$$

where (x, y) represents coordinates of trained speaker, (x_1, y_1) is coordinates of unknown speaker. The ED is used during the feature matching phase in order to measure the similarity between two speakers. Vector quantization is done on the unknown speaker and the VQ distortion is computed for all the speakers stored in the codebook and for the unknown input speaker. The unknown speaker is identified with the smallest distortion.

2.7.2.2 Gaussian Mixture Model (GMM)

The GMM is a type of parametric model that, in this study, is also chosen to be tested for speaker identification. GMM uses only a unique Gaussian Distribution Matrix to represent all the speakers in the system [47]. To estimate the parameters of the GMM model, Maximum Likelihood (ML) estimation is used. Expectation Maximization Algorithm is obtained iteratively to estimate the ML parameters [29].

The probability of speech signal features for a particular speaker model λ is modeled by a Gaussian mixture density that is the weighted sum of M component densities given by the following equation [23]:

$$p(\vec{x}|\lambda) = \sum_{i=1}^{M} p_i b_i(\vec{x}) \qquad (2.14)$$

where $\vec{x}$ is a random vector of D-dimension, λ is the speaker model, p_i are the mixture weights with the constraint $\sum_{i=1}^{M} p_i = 1$, $b_i(x)$ are the density components, and each density component is a D-variate-Gaussian distribution, so the Gaussian pdf of a feature vector for ith state is given by [48]:

$$\mathrm{b_i}(\vec{\mathrm{x}}) = \frac{1}{(2\pi)^{\frac{D}{2}}|\Sigma_i|^{1/2}} \exp(-\frac{1}{2}(\vec{x} - \overrightarrow{\mu_i})' \Sigma_i^{-1}(\vec{x} - \overrightarrow{\mu_i})) \qquad (2.15)$$

As appears from (2.14) $\mathrm{b_i}(\vec{\mathrm{x}})$ is formed by the mean vector $\vec{\mu}_i$ and the covariance matrix Σ_i, D is the dimension of the vector.

The parameters needed for a complete Gaussian mixture density are the mean vectors, covariance matrices and mixture weights that are available from all component densities. Each speaker model λ has a GMM that is used for speaker identification and is represented by the notation:

$$\lambda = \{p_i, \vec{\mu}_i, \Sigma_\mathrm{i}\} \quad , i = 1,2, \ldots , M \tag{2.16}$$

Figure 2.10 from [23] shows a depiction of an M component Gaussian mixture density where the parameters are shown.

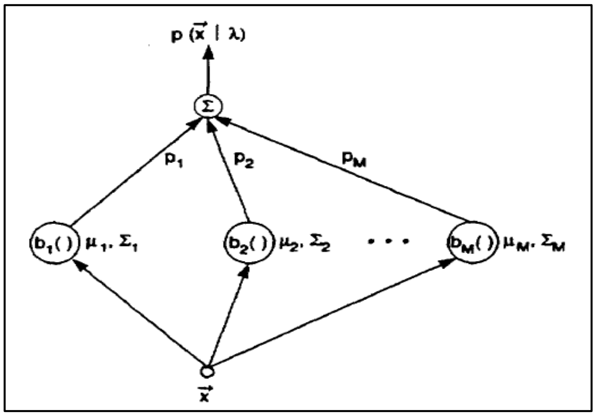


Figure 2.10 Gaussian Mixture Density of M Component

2.7.2.2.1 Maximum Likelihood (ML) Parameter Estimation

To find the GMM parameters we need a good estimation to obtain an optimum model that best match the distribution of training feature vectors representing each speaker [47] [23] [48] . ML estimation is used to find the parameters that maximize the likelihood of GMM.

For a sequence of training vectors $x = (\vec{x}_1, \vec{x}_2, \ldots , \vec{x}_t)$, the GMM likelihood can be written as:

$$p(X|\lambda) = \prod_{t=1}^{T} p(\vec{x}_t|\lambda) \tag{2.17}$$

But direct maximization likelihood of GMM is not possible because the expression of (2.17) is a nonlinear function. Expectation Maximization (EM) algorithm is the solution to estimate the parameters iteratively until convergence.

2.7.2.2.2 Expectation Maximization (EM) Algorithm

The idea of EM algorithm to estimate the parameters is shown in the following flowchart of figure 2.11. The algorithm starts with the initial model λ to estimate the new model λ_1. The new model λ_1 becomes the initial model and the process is repeated until some convergence threshold is reached. The idea is to refine the GMM parameters to monotonically increase the likelihood of the estimated model.

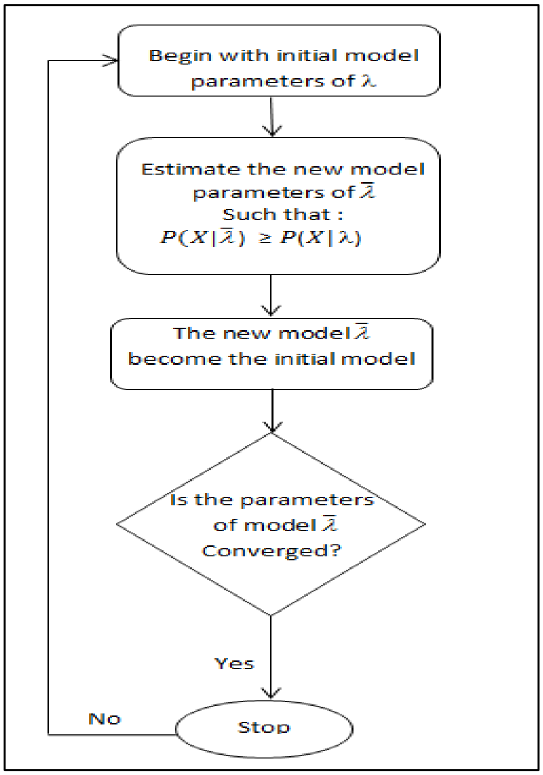

Begin with initial model parameters of λ
Estimate the new model parameters of $\bar{\lambda}$ Such that : $P(X|\bar{\lambda}) \geq P(X|\lambda)$
The new model $\bar{\lambda}$ become the initial model
Is the parameters of model $\bar{\lambda}$ Converged?
Yes
No
Stop

Figure 2.11 Flowchart of Expectation Maximization Algorithm

On each iteration, the EM algorithm computes the mean vector, weights, and variance by following re-estimation formulas [23]:

Mixture weights:

$$\vec{p}_i = \frac{1}{T}\sum_{t=1}^{T} p(i|\vec{x}_t,\lambda) \tag{2.18}$$

Means:

$$\vec{\bar{\mu}}_i = \frac{\sum_{t=1}^{T} p(i|\vec{x}_t,\lambda)\vec{x}_t}{\sum_{t=1}^{T} p(i|\vec{x}_t,\lambda)} \tag{2.19}$$

Variance:

$$\overline{\sigma}_i^2 = \frac{\sum_{t=1}^{T} p\big(i|\vec{x}_t,\lambda\big) x_t^2}{\sum_{t=1}^{T} p\big(i|\vec{x}_t,\lambda\big)} - \overline{\mu}_i^2 \qquad (2.20)$$

Selecting M of the mixture and a good parameters initialization are two main factors that affect training speaker model.

The posteriori probability for acoustic class i is given by:

$$p(i|\vec{x}_t,\lambda) = \frac{p_i b_i(\vec{x}_t)}{\sum_{k=1}^{M} p_k b_k(\vec{x}_t)} \qquad (2.21)$$

The decision is taken after all the models GMM for each speaker are estimated. If we have a group of S speakers $S = \{1,2,\dots,S\}$ and each represented by GMM $\lambda_1,\lambda_2,\dots,\lambda_S$, the goal is to find the model with maximum likelihood a posteriori for observation sequence. The estimated identity of the speaker will be in the form:

$$\hat{S} = \arg\ \max_{1\le k\le S} \sum_{t=1}^{T} \log p(\bar{x}_k|\lambda_k) \qquad (2.22)$$

where $p(\vec{x}_k|\lambda_k)$ represents the Gaussian mixture density given by the equation (2.14).

2.7.2.3 Artificial Neural Network (ANN)

ANN can be defined as a system of interconnected nodes “neurons” which exchange information between each other. Links between neurons are weighted with some numeric values to make the network able for learning. Each neuron consists of a processing element with synaptic input connections and a single output. The neuron output is given by an activation function $\mathrm{o} = \mathrm{f}(\Sigma)$. There are different kinds of activation functions and sigmoid function is one of the most used activation functions for training data. Figure 2.12 shows the neuron model structure as explained in [49].

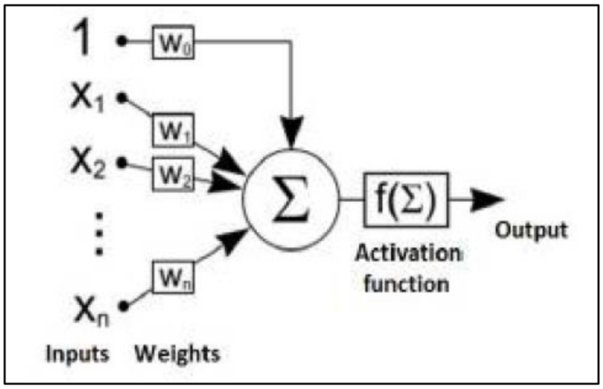


Figure 2.12 Neuron Model Structure

2.7.2.3.1 Multilayer Perceptron Network

The neural network is designed using Multilayer Perceptron Network (MLP). MLP is a feed forward artificial neural network used in supervised learning [50] . It is composed of multiple layers of nodes that are fully connected with nodes of next layer. Each link between nodes has an initial weight. MLP has the following layers: input layer, one or more hidden layer, and output layer as shown from [51] in figure 2.13. Some activation function is used by nodes in hidden layer and output layer. Back propagation (BP) algorithm is a widely used supervised learning algorithm for multilayer feed forward neural network. BP is used for training the nodes in MLP network. The BP works according to error-correction learning rule. Because BP is a supervised learning algorithm, input and target output vectors are provided for training. Then, the difference between actual outputs and target outputs are computed for more changes until reaching desired results.

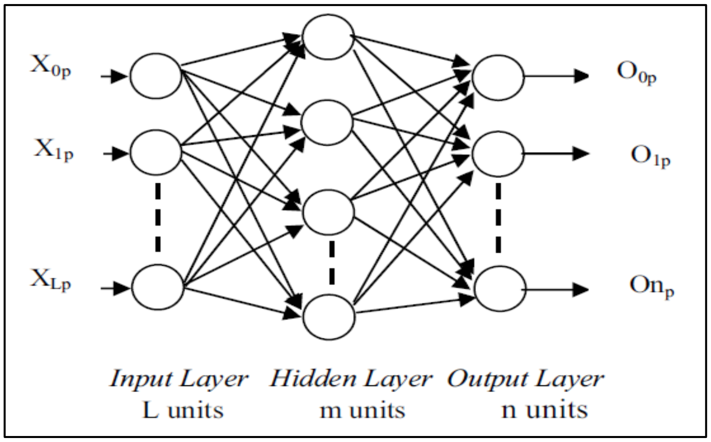


Figure 2.13 Model of Multilayer Perceptron Neural Networks

The general methodology to compute error and update the results in BP consists of two passes: forward pass and a backward pass [52]. In the forward pass, the input vector is applied to the nodes in network to be propagated layer by layer and synaptic weights are all fixed to produce set of outputs. Nodes of output layer calculate the difference between actual output results and the desired ones. The result of the difference is considered as error value. In the backward pass the error signal is propagated backward through the network and the synaptic weights in links between nodes are adjusted according to the error-correction rule [50] [52] [53]. This is done for each input sample of all speakers need to be enrolled in the system. This is implemented as follows [49]:

- Input vector X_p is applied to the input units where:

$$X_p = (x_{p1}, x_{p2}, \dots, x_{pN})^t \tag{2.23}$$

- Then the net input values is calculated to the hidden layer units:

$$net_{pj}^{h} = \sum_{i=1}^{N} w_{ji}^{h} x_{pi} + \theta_j^h \tag{2.24}$$

where:

- net_{pj}^{h} is the net input to hidden layer,
- w_{ji}^{h} is the weight in the connection from i^{th} input unit,
- θ_{j}^{h} is the bias term,
- and "h" refers to quantities on the hidden layer.

- Calculate the outputs from the hidden layer:

$$i_{pj} = f_{j}^{h}(net_{pj}^{h}) \tag{2.25}$$

where i_{pj} is the output from hidden layer and f_{j}^{h} is the activation function.

- Now move to the output layer and calculate the net-input values to each units:

$$net_{pk}^{o} = \sum_{j=1}^{L} w_{kj}^{o} \; i_{pj} + \theta_{j}^{o} \tag{2.26}$$

where:

- net_{pj}^{o} is the net input to output layer,
- w_{ji}^{o} is the weight in the connection from j^{th} hidden unit,
- θ_{j}^{o} is the bias term,
- and "o" refers to quantities on the hidden layer.

- Calculate the outputs:

$$O_{pk} = f_{k}^{o}(net_{pk}^{o}) \tag{2.27}$$

where O_{pk} is the output from output layer and f_{k}^{o} is the activation function.

- Now we can calculate the error terms for the output units:

$$\delta_{pk}^{o} = (y_{pk} - O_{pk}) f_{j}^{o\prime}(net_{pk}^{o}) \tag{2.28}$$

where $\delta^{o}_{pk}$is the error at each output unit.

$$\delta^{o}_{pk} = y_{pk} - o_{pk} \tag{2.29}$$

where y_{pk} is the desired error and o_{pk} is the actual error.

- Then we calculate the error terms for the hidden units:

$$\delta^{h}_{pj} = f^{h\prime}_{j}(net^{h}_{pj}) \sum_{k} \delta^{o}_{pk}\, w^{o}_{kj} \tag{2.30}$$

where δ^{h}_{pj} is the error at each hidden unit.

- Now we can update the weights:

To update weights on the output layer:

$$w^{o}_{kj}(t+1) = w^{o}_{kj}(t) + \eta\delta^{o}_{pk} i_{pj} \tag{2.31}$$

To update weights on the hidden layer:

$$w^{h}_{ji}(t+1) = w^{h}_{ji}(t) + \eta\delta^{h}_{pj} x_{i} \tag{2.32}$$

Where η is the learning rate parameter.

- Finally, the error term should be calculated and if it is acceptably small for each of the training-vector pairs, training can be stopped; to calculate the error term:

$$E_{p} = \frac{1}{2} \sum_{k=1}^{M} \delta^{2}_{pk} \tag{2.33}$$

Figure 2.14 illustrates the implementation of BP algorithm to train the MLP network and make it ready for speaker identification.

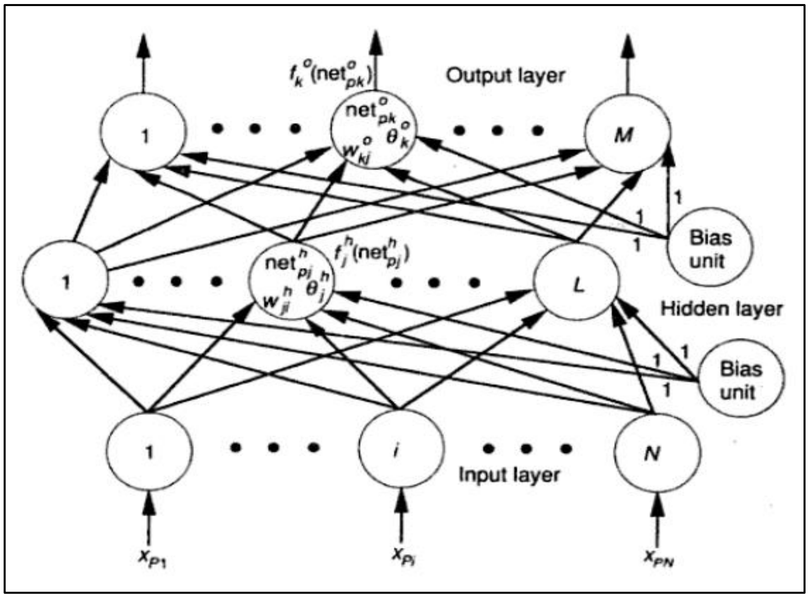


Figure 2.14 MLP Network Trained by BP Learning Algorithm

Now MLP is designed by BP to provide best matching speaker for each input vector. After training part, the stored parameters from training parts are used now for identification. This is done as following:

1. Extracted features form unknown speaker (need to be identified) are fed into the network and without target output.
2. Weights and thresholds for each trained speaker are used by the network.
3. Compare the output with some predefined output decision.
4. The network finds the closest matching output using the weights and thresholds stored before.
5. Decision is made and the speaker is identified.

2.7.2.4 Decision Trees (DT)

DT are popular approaches for representing classifiers by collect rules, which are organized in a hierarchical fashion, that implement a decision structure [54]. DT use a predictive model that predicts the value of a target based on several inputs. The representation of the model follows a tree-based that is composed of nodes and edges. DT is a directed tree that starts with a node called the “root”, which has no incoming edges. The other nodes have one incoming edge and could be an internal (test) or leaves (decision) nodes [28]. The test vector path in DT will move through the non-leaves nodes to be evaluated and then directed to one of two subsequent nodes based on a decision. The process continues until a leaf is reached where it corresponds to a specific class, which, in our situation is a speaker model.

In speaker recognition area, DT can be used by training a binary DT for each speaker. This is done by obtaining the feature vectors from the training data for all speakers. Then all the data is labeled in the following manner: for a specific speaker, all his/her feature vectors are labeled as “one” and “zero” for the feature vectors of other speakers. The leaves labeled with the speaker class “one” indicate that “this is the speaker” and “zero” that “this is not the speaker”. The leaves have also the probability measure for each speaker. In speaker identification, all the feature vectors of the test utterances are applied to each decision tree. A likelihood measure is used to identify a certain speaker by using the decision tree probabilities [54].

2.7.2.4.1 Classification and Regression Trees (CART)

Breiman’s CART is one of several popular decision trees that is selected for our study when DT is applied to do the identification part on the enrolled speakers. CART stands for Classification and Regression Trees that is presented by Breiman, Freidman, Olshen, and Stone in 1984 [55]. CART decision tree constructs binary trees, where each internal node has exactly two outgoing edges. The splits are selected using the towing criteria.

The CART constructed tree is pruned by cost–complexity Pruning [28]. Pruning is the process of removing the subtrees that have been grown excessively to classify a small portion of the training data and it is used to improve the performance on the testing data [54]. CART has the feature of generating regression trees where their leaves predict a real number and not a class, based on the weighted mean for node. Another feature in CART is the ability to handle missing values by surrogating splits.

CART algorithm follows only yes/no questions to search for all possible variables in order to find the best split with maximum homogeneity and the process is repeated for each of the resulting data fragments [56].

Building a classification tree (like in our case) where learning samples are split up to last observations is time consuming. Splitting rules are used to construct the tree by splitting learning sample to smaller parts whereas each time data have to be divided into two parts with maximum homogeneity. Figure 2.15 presents the CART splitting algorithm in general.

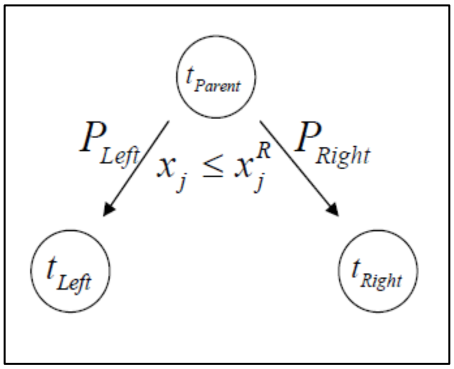


Figure 2.15 Splitting Algorithm of CART

where, $t_{Parent}, t_{Left}, and\ t_{Right}$ are parent, left and right nodes; $P_{Left}\ and\ P_{Right}$ are the probabilities of left and right nodes; x_j is variable j; x_j^R is best splitting value of variable x_j .

To partition the acoustic space according to [57] [58], let $R = \{R_k\}_{1 \le k \le K}$ be a partition of acoustic feature space B, defined as:

$$\bigcup R_k = B \qquad and \qquad R_i \cap R_j = \emptyset \tag{2.34}$$

Each partition R corresponds to terminal leaf in the tree, where each leaf can be reached by a particular path through the tree. To train a tree to perform a good separation between classes, we need to define a criterion $C(R)$ to measure the purity of the partition and define the best partition R^* with respect to criterion C as:

$$R^* = arg_R \max C(R) \tag{2.35}$$

R^* can be found by several algorithm such as CART. The criterion C in CART is expressed as the weighted average of a local criterion c_k to measure the region purity. Two common region purity measurements are used:

1. Entropy criterion:

$$c_k = \sum_{j=1}^{J} P_{jk} \log p_{jk} \tag{2.36}$$

2. Gini criterion:

$$c_k = \sum_{j=1}^{J} P_{jk}^2 - 1 \tag{2.37}$$

In (2.36), (2.37), c_k is the criterion value on region R_k, P_{jk} is the probability of the class j in region k, J is the total number of classes. The overall criterion value $C(R)$ is the summation of $c_k$over all the partitions, as:

$$C(R) = \sum_{k=1}^{K} c_k \tag{2.38}$$

Since there is no stopping rule to give an optimal tree, Breiman et al. introduced the pruning procedure after the over growing procedure of the tree. The scores in each leaf can be binary score or log probability ratio score, defined as:

- Binary score:

$$S_{R^X} = \begin{cases} +1 & , N_{R^X}(X) > N_{R^X}(\bar{X}) \\ -1 & , otherwise \end{cases} \tag{2.39}$$

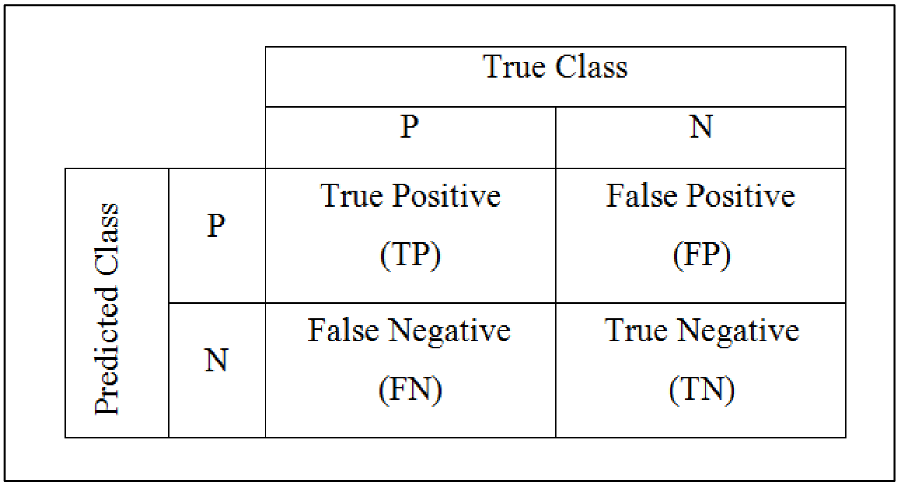
- Log probability ratio score:

$$S_{R^X} = \log P(X|R^X) - \log P(\bar{X}|R^X) \tag{2.40}$$

where X is the true speaker, $\bar{X}$ is any other speaker.

2.8 Performance Measurements

Speaker identification algorithms work as classifiers to classify speakers into their real identity. There are several evaluation parameters to measure the performance of the proposed classifiers such as classification accuracy, precision, and recall. To calculate these parameters we need first to compute the confusion matrix elements of each classifier. As shown in figure 2.16, confusion matrix is composed of four elements: True positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN) [59] [60]. If the predicted and true classes are the same, we have a true result. TP measure indicates how many true speakers are accepted into the system. TN measure indicates how many imposters are rejected by the system. Error occurred when the predicted class predicts positive or negative value, where it is the opposite value in the true class. FP measure indicates how many imposters are accepted to the system as a true speaker. FN measure indicates how many true speakers are rejected by the system as an imposter.



		True Class	
		P	N
Predicted Class	P	True Positive (TP)	False Positive (FP)
	N	False Negative (FN)	True Negative (TN)

Figure 2.16 Confusion Matrix

Precision (or positive predictive value) measures that fraction of speakers classified as positive that are truly positive. Recall (or sensitivity) measures the fraction of positive speakers that are correctly labeled. Accuracy measures all the correctly labeled speakers from all the speakers participated to the system. The following formulas represent precision, recall, and accuracy according to confusion matrix:

$$Precision = \frac{TP}{TP + FP} \qquad (2.41)$$

$$Recall = \frac{TP}{TP + FN} \qquad (2.42)$$

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \qquad (2.43)$$

Chapter 3

Proposed Methodology

Chapter III

Proposed Methodology

3.1 Introduction

In this chapter, we will propose a text-independent speaker identification system and discuss our methodology for identifying groups of speakers. In this thesis, our goal is to upgrade the identification rate by using various speaker identification algorithms separately and together. As mentioned in chapter 2, any SIS is composed of two modules: feature extraction and pattern matching. SIS is the process of finding the identity of an unknown speaker by comparing his/her voice with voices of registered speakers in the database as illustrated in figure 3.1 [7] . Extracting feature vectors is done to build a reference speaker model, as shown in figure 3.1, let “M” is the number of speakers, there are M speaker models registered in speaker database. This step is called the training phase (or enrollment phase). In the testing phase (or identification phase), unknown speaker enters the system. The feature extraction part is done on the unknown voice, a parallel comparison between all the M speaker models and the unknown speaker being performed. The most likely score is reported to make the decision of the unknown speaker identity.

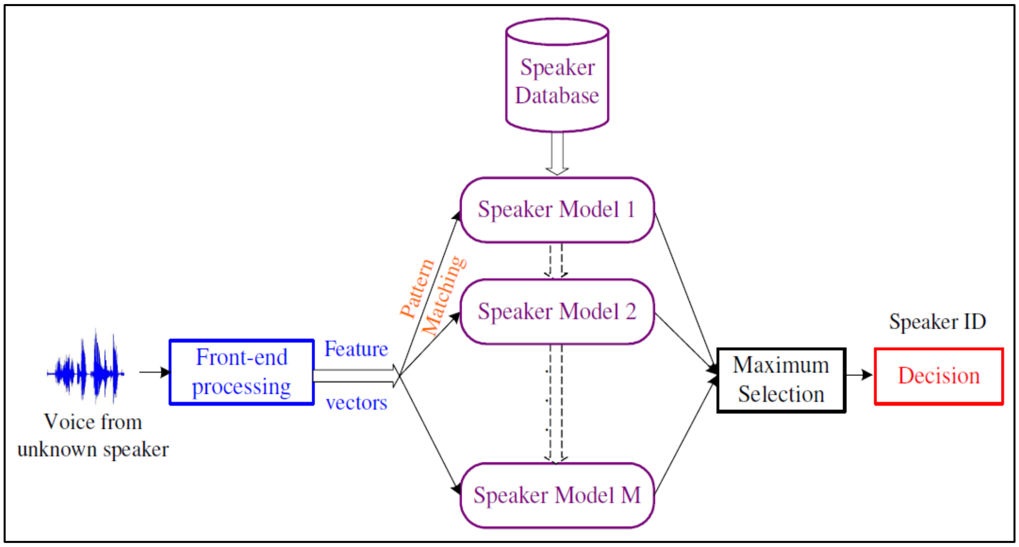


Figure 3.1 Basic Structure of SIS

In our case, we are working in a close-set identification where the result will be: i) one of the preregistered and trained speaker models or ii) "not identified" from the M speaker models.

The procedure starts with signal pre-processing, the signal being prepared for feature extraction and after that, the MFCCs technique is applied for feature extraction. Once the speakers' models are built into the system, the actual procedure of speaker identification can be applied by pattern matching models.

There are two main categories of models, unsupervised and supervised models [61]. In unsupervised training algorithms, the speaker model is built from feature vectors for that speaker only. In supervised training algorithms, to build the model, the feature vectors (associated with labels to determine a class information) are coming from numerous speakers. Unsupervised classifier has the advantage of avoiding complicated computation because there is no need to reconfigure all speaker models when a new speaker is added to the system. However, even if the extensive time consumed in supervised algorithms is not preferred, some tasks found that adding additional speakers during training enhanced performance [61] [54].

In this work, we are following a comparative study between two unsupervised (VQ and GMM) and two supervised (ANN and DT) classifiers. The scope is to analyze their identification accuracy over the same dataset of speakers. The following sections will provide some details of the entire automatic speaker identification system that is performed in this study.

3.2 Proposed Methodology

The goal of the work proposed here is to identify a closed set of speakers from different speech corpora and to prove the identification accuracy with different inputs. The speakers recorded their utterances under different conditions and have various dialects of English language. The proposed SIS is composed of two main modules: feature extraction module and feature matching (pattern classification) module. The feature extraction module is done using MFCCs. Then, feature matching module is applied with VQ, GMM, ANN, and DT algorithms to compare the identification results. Finally, the system tries to improve the identification rate by fusion the results outcome from the four classifiers by applying the majority rule.

3.2.1 Speaker Database

To test the identity of each speaker, all the simulations in this study are done using a speaker database. Our speaker database is composed of 120 English speakers (80 male and 40 female) from open source corpora available for speech researches. Speaker voices files are downloaded from English Language Speech Database for Speaker

Recognition (ELSDSR) corpus [62], VoxForge speech corpus [63], and Pacific Northwest/Northern Cities (PN/NC) corpus [64]. The speaker database in this study is organized as follows: each speaker has a folder labeled with a number (serially labeled). The speaker folder contains all the audio speech files of his/her voice. Each speaker folder has two distinct folders, one for training, and one for testing. The audio files duration are divided into 80% for training and 20% for testing. The training folder contains about one minute to two minutes duration of audio files. The testing folder contains audio files with a length varying between 15 to 30 seconds. To satisfy the required time needed for training and testing, the number of these files depends on the corpus.

In ELSDSR and PN/NC corpora all the speakers recordings speak the same sentences, where in VoxForge corpus speakers have their own sentences. ELSDSR corpus design was a joint effort of the faculty, Ph.D. and Master students from the department of Informatics and mathematical modelling, Technical University of Denmark (DTU). The speakers are talking in English and most of them are non-native speakers. The goal of this work is to provide speech data for the development and evaluation of automatic speaker recognition system.

VoxForge corpus is a collection of transcribed speech for use with free and open source speech recognition engines. The submitted audio files meet the General Public License (GPL), and they are compiled into acoustic models for use in speech and speaker recognition purposes. Most of the speakers in this work are came from this corpus and some of them are non-native English speakers.

PN/NC corpus is available from University of Washington's Phonetics Laboratory, Department of Linguistics. The lab concentrates on acoustic analysis of spoken language and speech perception. The corpus includes 3600 audio files. Files are readings of 180 sentences by different persons from each of two dialect regions of American English (the Pacific Northwest and the Northern Cities).

Our speaker database is composed of 10 female and 12 male from ELSDSR, 10 female and 10 male from PN/NC, the rest are from VoxForge corpus (58 male/ 20 female). Table 3.1 shows the audio file information used in this work.

Table 3.1 Audio File Information

Recording Attribute	Value
File type	wav
Sampling Rate (Hz)	16000 to 48000
Bit-depth (bit)	16
Number of channels	1

3.2.2 Feature extraction by MFCCs

As mentioned earlier in chapter 2 that some studies divide the processes into feature extraction, training and testing. Feature extraction module is needed in both training and testing phases to extract the speaker dependent features from the speech signal. In our work, we are using MFCCs technique for extracting the most significant features in modelling speaker, which are the first 13 MFCCs [65]. To perform feature extraction we do the following:

1. Reading the speaker audio file to convert the continuous speech signal is into discrete domain.
2. Blocking the speech signal into frames with the length of 20 to 40ms, and overlap of 50% to 75%.
3. Windowing each frame with hamming window function to avoid problem brought by truncation of the signal.
4. Applying a pre-emphasis filter to each frame to enhance the speech by lifting the high frequency spectral components of speech.
5. Spectral analyzing, frame by frame, to transfer speech signal into short-term spectrum.
6. Extracting features for converting speech into parameter representation.

3.2.3 Speaker Identification

The identification procedure is actually done in pattern match module. As discussed in chapter 2, for the identification process, the pattern matching module should pass two important phases: enrollment (training) phase and identification (testing) phase. In this module, we are applying VQ, GMM, ANN, and DT models as speaker identification algorithm. The feature vectors of speakers are trained by VQ, GMM, ANN, and DT to build reference model for each speaker. This is called the enrollment phase. The result of this phase is creating the enrolled speaker database. Testing phase starts when an unknown speaker enters the system and needs to be identified as one of the speaker database. This is done by VQ, GMM, ANN, and DT to choose the best speaker matching model. Tests are also done by fusing the results of the four algorithms and compare the results of each algorithm separately and when they are fused together. The goal is to come with an efficient SIS that identifies most of speakers, if not all of them. Now we will explain the training and testing phases during speaker identification process.

The first speaker identification algorithm used in this study is VQ algorithm. In this algorithm, the trained feature vectors are saved into codebooks for each speaker. By using LBG algorithm, codebooks are generated. When unknown speaker enters the system, the features of speech signal is extracted and ED comparison is done between feature vectors of unknown speaker and the codeword of codebooks of trained speakers to find out the best match and identify the unknown speaker. Figure 3.2 illustrates VQ model for identifying speakers.

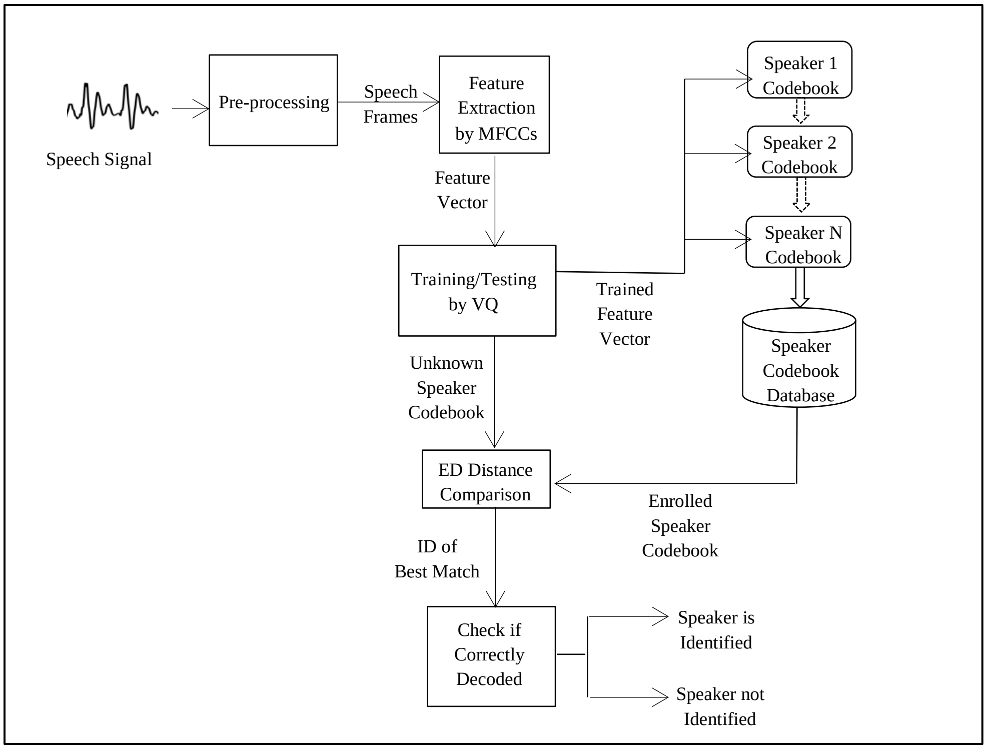


Figure 3.2 Architecture of Speaker Identification System Using VQ

The second speaker identification algorithm used in this study is GMM algorithm. After feature extraction, the feature vectors are trained to build GMM speaker models λ. Each speaker model λ is modeled by a Gaussian mixture density that is the weighted sum of M component densities. ML estimation is used to find the parameters that maximize the likelihood of GMM. EM algorithm is used to refine the GMM parameters to monotonically increase the likelihood of the estimated model. The parameters needed for a complete Gaussian mixture density are the mean vectors, covariance matrices and mixture weights that are available from all component densities. In our work we are training the GMM for all components starting from 1 up to 4 mixtures, and return the GMM model with the best fit. To identify unknown speaker we have to find the model with maximum likelihood a posteriori for observation sequence. Figure 3.3 illustrates GMM model for identifying speakers.

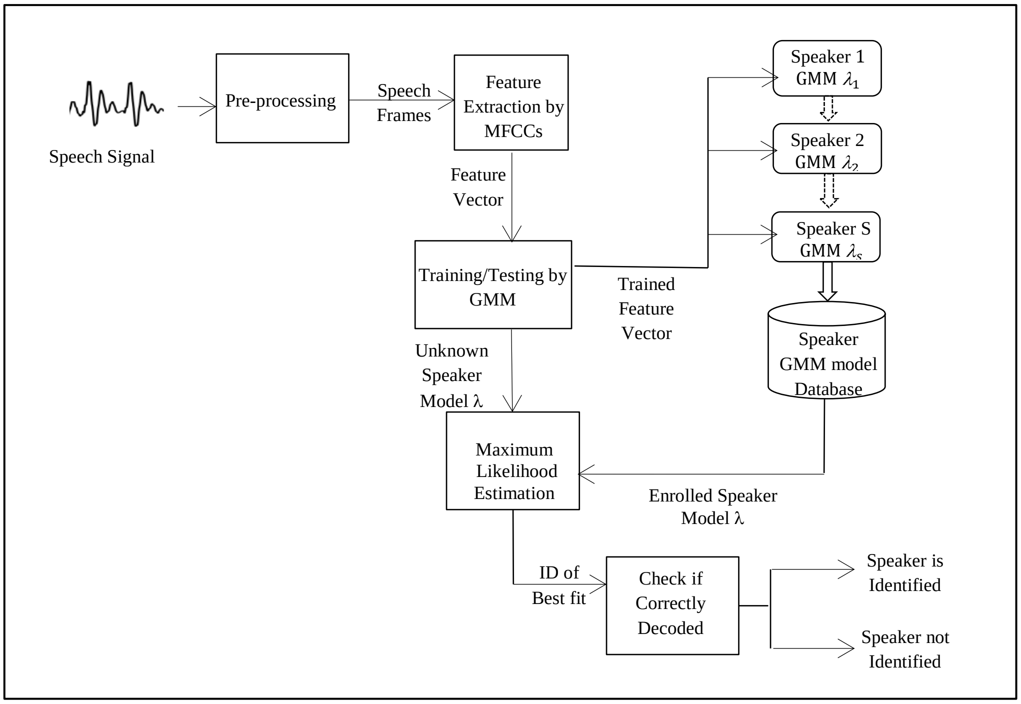


Figure 3.3 Architecture of Speaker Identification System Using GMM

The third speaker identification algorithm used in this study is MLP neural network algorithm. BP is used for training the nodes in MLP network according to error-correction learning rule. Input vectors are the 13 MFCCs and target output vectors are features of each speaker enrolled in the system. Input and target output vectors are provided for training. Then, the difference between actual outputs and target outputs are computed for more changes until reaching desired results. MLP provides best matching speaker for each input vector. After training part, the stored parameters from training parts are used now for identification. Extracted features form unknown speaker are fed into the network. Weights and thresholds for each trained speaker are used by the network. The network compares the output with some predefined output decision to find the closest matching output using the weights and thresholds stored before. Finally, Decision is made and the speaker is either correctly identified or not. Figure 3.4 illustrates MLP model for identifying speakers.

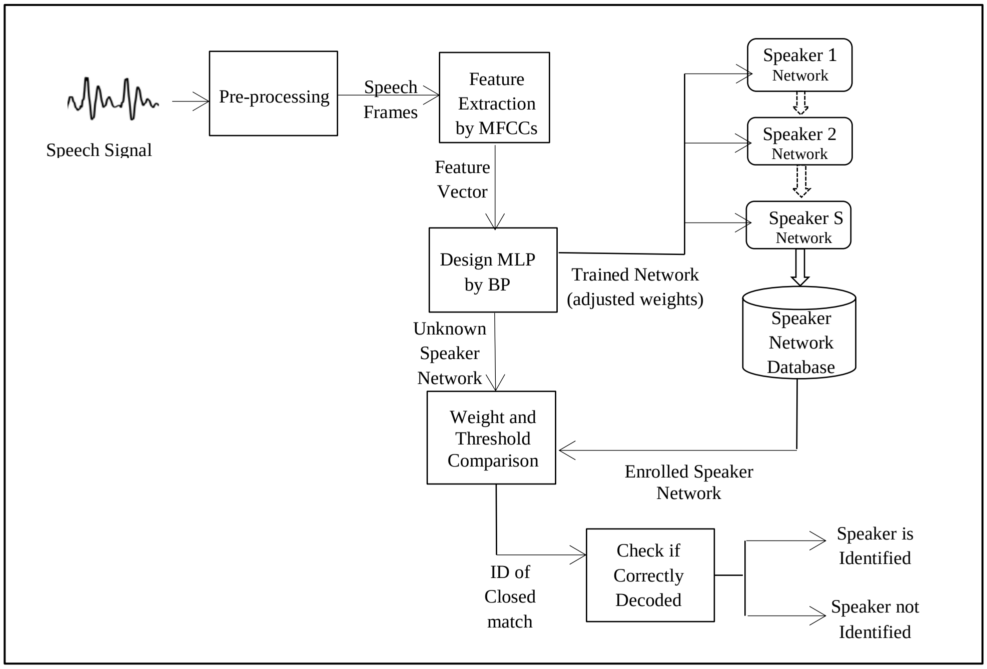


Figure 3.4 Architecture of Speaker Identification System Using MLP

The fourth speaker identification algorithm used in this study is CART algorithm. CART decision tree constructs binary trees, where each internal node has exactly two outgoing edges. For each speaker, a binary DT is trained by obtaining the feature vectors from the training data for all speakers. Then all the data is labeled in the following manner: for a specific speaker, all his/her feature vectors are labeled as “one” and “zero” for the feature vectors of other speakers. The leaves labeled with the speaker class “one” indicate that “this is the speaker” and “zero” that “this is not the speaker”. The leaves have also the probability measure for each speaker. To identify a certain speaker, all the feature vectors of the test utterances are applied to each decision tree. A likelihood measure is used to identify a certain speaker by using the decision tree probabilities. Figure 3.5 illustrates CART model for identifying speakers.

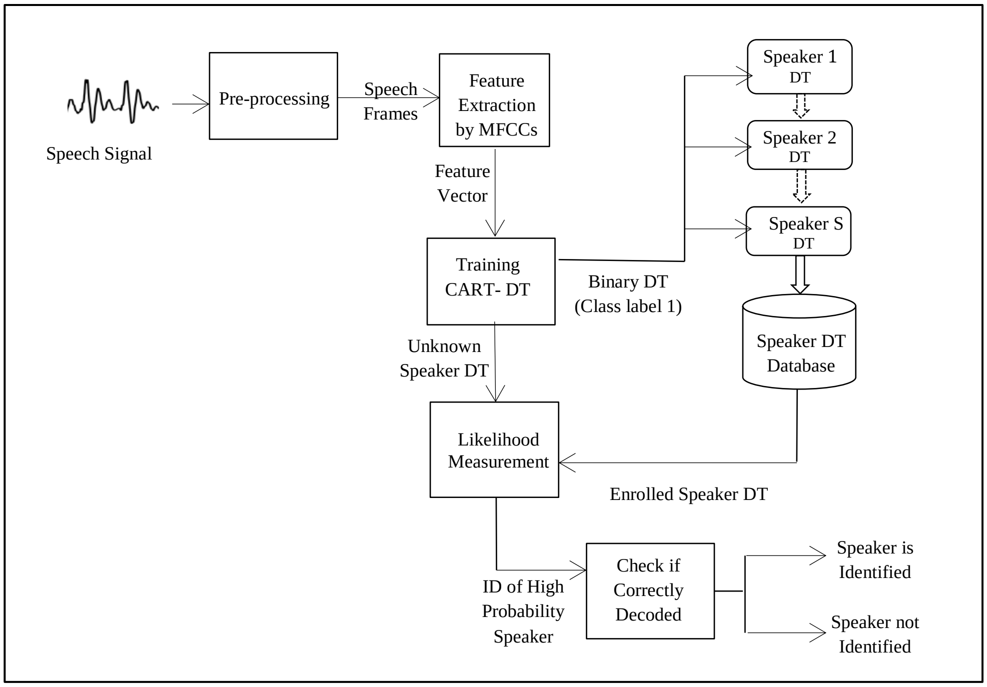


Figure 3.5 Architecture of Speaker Identification System Using CART

3.2.4 Majority Rule Fusion

In speaker recognition area, relying on a single feature extraction method or a single pattern classifier does not always come out with the desired high identification rate. To compensate for this issue, a new direction in SR research (called fusion) has been proposed [66] [67]. The idea of fusion is to get the final decision depending on multiple feature extraction methods [68] or multiple classifiers [69] [70] to improve the identification. Data fusion combines scores from these different models trained for a speaker. However, if all models agree on the same type of error, no improvement occurs from fusion [61]. So, some degree of un-correlation between models is needed to improve the identification rate.

There are different data fusion techniques that depend on the type of information to be combined. Linear or log opinion pools can be used if we are dealing with probabilities as results [61] [71]. If the results are class labels, voting can be used [72].

In our study, we will follow the voting procedure by majority rule because the outputs are speaker labels. The majority test rule is based on the voting procedure, which means that the score for success rate will be based on the majority of votes received by a testing feature vector during the recognition process. This is done as following: i) if three or all classifiers voted for a certain speaker identification number, then the speaker is identified by the majority voting, or ii) if two classifiers voted for a certain speaker identification number and the other two classifiers disagree on the speaker identification number, then the speaker is identified by the majority voting. Majority voting fail to produce correct result in two cases: i) all four classifiers agree on the same speaker identification number, or ii) two classifiers agree on certain identification number where the other two classifiers agree on another identification number. The next chapter will discuss in details the experiments of these classifiers individually and in fusion. Figure 3.6 illustrates the general structure of the text-independent speaker identification system that is proposed here.

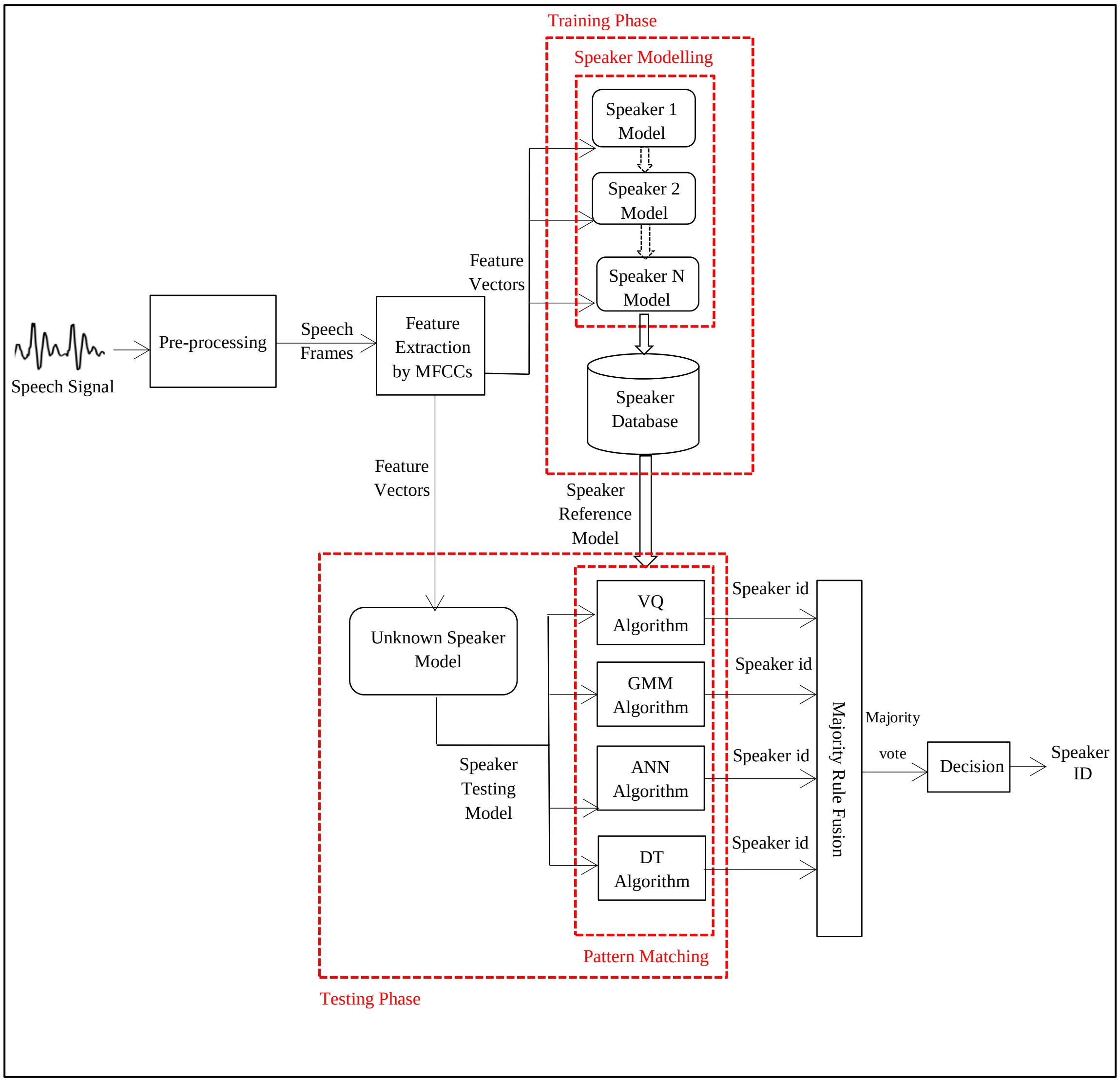


Figure 3.6 Architecture of Proposed Speaker Identification System

Chapter 4

Results

Chapter IV

Results

4.1 Introduction

In this chapter, several experiments on a closed set of speakers reading sentences (that are chosen from the corpus they belong to) are performed. We have 120 speakers from three open source corpora available to be downloaded for speech researches. Speakers are speaking English with different dialects such as British, Canadian, American, and Indian. The main purpose of these tests is to get the identification rate for each speaker identification algorithm discussed in the previous chapter. Then, by applying some strategies, their performance is upgraded.

4.2 Experiments and Results

The experiments in this study are done by using software MATLAB R2012a (win64) with operating system version: Microsoft Windows 7 Home Premium Service Pack1, with the processor: Intel(R) Core(TM) i5- 2450M CPU @ 2.50 GHz., and 4GB system memory. To perform the simulations, three methods are employed. The first one is applied on 10 to 120 speakers to test the accuracy percentage of identification. The procedure starts with MMFCs feature extraction followed by feature matching algorithm. As mentioned earlier we are applying VQ, GMM, ANN, and DT algorithms.

Then, an additional step is added to these results by fusion the four algorithms according to majority rule.

In the second approach, we are randomly choosing six groups of 25 speakers and re-apply the same procedure of the first experiments method. These groups are labeled with A, B, C, D, E, and F.

The goal of these two methods is to compute the accuracy of identification (%) applied to various speaker groups using different Speaker Identification algorithms according to the following formula:

$$\text{Identification Rate of SIS} = \left(\frac{\text{No.of Correctly Identified Samples}}{\text{Total No.of Tested Samples}}\right) * 100 \qquad (4.1)$$

The third method is applied on group of speakers containing a mixture of true speakers and imposter speakers. This is done to check the performance of proposed classifiers by using some evaluation measures. To find these parameters we need to compute the TP, TN, FP, and FN.

4.2.1 MFCCs

To capture the desired features from speech signals, MFCCs is the feature extraction step that is necessary to start any of our experiments. The time needed for extracting features are different according to the speaker group size. Table 4.1 shows the time needed for MFCCs to extract features from 10 to 120 speakers. Table 4.2 shows the time needed for MFCCs to extract features from six groups of 25 speakers.

Table 4.1 MFCCs Feature Extraction Time for 10-120 Speakers

No. of Speakers	Feature Extraction Time (sec)
10	4.961617
20	8.24186275
30	17.318721
40	27.55157367
50	37.62242733
60	45.66863467
70	51.84141475
80	64.60847
90	64.27714133
100	64.2426995
110	77.41341
120	97.694765

Table 4.2 MFCCs Feature Extraction Time for 25 Speakers in Groups

No. of Speakers	Feature Extraction Time (sec)
A	24.541644
B	28.125971
C	33.163677
D	37.39924
E	19.572934
F	21.471976

4.2.2 Vector Quantization

VQ is the first algorithm used in this study for speaker identification. Tables 4.3 and 4.4 show the time consumed in training and testing by VQ. Table 4.5 illustrates the identification rate and speakers not identified by VQ for 10 to 120 speakers. Table 4.6

illustrates the identification rate and speakers not identified by VQ for six groups of 25 randomly chosen speakers labeled with A, B, C, D, E, and F.

Table 4.3 VQ Training/Testing Time for 10-120 Speakers

No. of Speakers	Train (sec)	Test (sec)
10	54.1906315	25.5666305
20	105.815471	77.726491
30	136.665454	194.681687
40	154.478126	325.517194
50	163.664888	504.6094675
60	191.9997045	803.424704
70	208.4509125	1305.5708
80	220.2587615	1737.748911
90	232.697627	2258.182822
100	252.170425	3040.665092
110	271.7416595	3870.655405
120	410.0407425	4614.292152

Table 4.4 VQ Training/Testing Time for 25 Speakers in Groups

Group Label (25 speaker)	Train (sec)	Test (sec)
A	106.442202	160.8639
B	47.765229	221.20324
C	165.745871	189.074383
D	103.727004	190.380027
E	66.544932	162.18909
F	51.292002	269.756507

Table 4.5 VQ Identification Rate for 10-120 Speakers

No. of Speakers	Identification Rate (%)	Speaker not Identified
10	100	none
20	96.1112	none
30	90.9091	none
40	87.5862	none
50	89.6552	none
60	86.5979	none
70	77.8626	none
80	73.6634	sp61
90	68.4818	sp61, sp70
100	60.8696	sp61, sp90
110	60.7099	sp50, sp97
120	53.7981	sp66, sp71, sp97, sp105, sp118

Table 4.6 VQ Identification Rate for Groups of 25 Speakers

Group Label (25 speaker)	Identification Rate (%)	Speaker not Identified
A	97.1429	none
B	92.4812	none
C	88.9764	none
D	88.1944	none
E	89.2473	none
F	84.7953	none

4.2.3 Gaussian Mixture Models

GMM is the second algorithm used in this study for speaker identification. Tables 4.7 and 4.8 show the time consumed in training and testing by GMM. Table 4.9 illustrates the identification rate and speakers not identified by GMM for 10 to 120 speakers. Table 4.10 illustrates the identification rate and speakers not identified by GMM for six groups of 25 randomly chosen speakers labeled with A, B, C, D, E, and F.

Table 4.7 GMM Training/Testing Time for 10-120 Speakers

No. of Speakers	Train (sec)	Test (sec)
10	107.752653	1.606699667
20	198.267689	3.467813333
30	290.439653	6.704860667
40	408.462581	14.77693333
50	496.657676	15.61544933
60	572.438726	23.69149033
70	690.543315	36.08702067
80	802.272165	46.63123267
90	951.66434	62.72145367
100	1049.391832	76.089278
110	1218.654934	99.15649767
120	1270.527955	124.8263563

Table 4.8 GMM Training/Testing Time for 25 Speakers in Groups

Group Label (25 speaker)	Train (sec)	Test (sec)
A	343.332542	7.338984
B	409.913978	10.184273
C	391.25891	22.698471
D	264.914517	20.283482
E	318.923986	13.376885
F	373.99106	33.968833

Table 4.9 GMM Identification Rate for 10-120 Speakers

No. of Speakers	Identification Rate (%)	Speaker not Identified
10	100	none
20	98	none
30	96.9697	none
40	91.7241	none
50	92.1182	none
60	89.0034	none
70	84.9873	none
80	77.8218	none
90	78.5479	sp61
100	72.791	sp61, sp88
110	70.9914	sp61, sp92
120	65.7648	sp71

Table 4.10 GMM Identification Rate for Groups of 25 Speakers

Group Label (25 speaker)	Identification Rate (%)	Speaker not Identified
A	100	none
B	95.4887	none
C	93.7008	none
D	97.2222	none
E	91.3978	none
F	88.8889	none

4.2.4 Artificial Neural Networks

ANNs represent the third algorithm used in this study for speaker identification. Tables 4.11 and 4.12 show the time consumed in training and testing by ANN. Table 4.13 illustrates the identification rate and speakers not identified by ANN for 10 to 120 speakers. Table 4.14 illustrates the identification rate and speakers not identified by ANN for six groups of 25 randomly chosen speakers labeled with A, B, C, D, E, and F.

Table 4.11 ANN Training/Testing Time for 10-120 Speakers

No. of Speakers	Train (sec)	Test (sec)
10	247.699941	3.6522675
20	982.225262	11.898487
30	2887.185018	20.602352
40	5237.510393	40.548091
50	10931.01889	70.417462
60	17555.02375	125.986802
70	26296.2029	188.4948665
80	36772.80875	296.655542
90	47249.02103	331.3452505
100	59404.94375	475.37045
110	71590.79588	553.83079
120	83735.17831	781.033206

Table 4.12 ANN Training/Testing Time for 25 Speakers in Groups

Group Label (25 speaker)	Train (sec)	Test (sec)
A	2013.975236	15.959819
B	2407.481957	51.917671
C	1795.01945	35.321869
D	2076.721296	39.454763
E	2231.354544	40.233194
F	1504.557204	45.254354

Table 4.13 ANN Identification Rate for 10-120 Speakers

No. of Speakers	Identification Rate (%)	Speaker not Identified
10	93.3333	none
20	88	sp6
30	78.7879	sp26
40	77.931	sp6, sp14, sp26
50	77.3399	sp7, sp26, sp43
60	78.3505	sp26, sp37
70	70.9924	sp6, sp7, sp26, sp61, sp66, sp67, sp70
80	62.9703	sp14, sp26, sp40, sp61, sp62, sp67, sp70,sp74, sp76, sp78
90	65.3465	sp7, sp14, sp26, sp61, sp62, sp66, sp67, sp70, sp71, sp74, sp76, sp78, sp83
100	64.5161	sp7, sp11, sp26, sp31, sp61, sp62, sp67, sp70, sp71, sp74, sp76, sp78, sp83, sp94
110	61.6891	sp6, sp7, sp26, sp40, sp53, sp55, sp61, sp67, sp70, sp74, sp76, sp78, sp83, sp94, sp105, sp110
120	52.2373	sp6, sp7, sp14, sp26, sp50, sp61, sp66, sp67, sp70, sp71, sp74, sp76, sp78, sp83, sp94, sp105, (sp110-sp120)

Table 4.14 ANN Identification Rate for Groups of 25 Speakers

Group Label (25 speaker)	Identification Rate (%)	Speaker not Identified
A	90	sp14
B	93.985	none
C	74.8031	sp11
D	84.0278	sp25
E	84.9462	none
F	85.3801	sp14

4.2.5 Decision Trees

DT is the fourth algorithm used in this study for speaker identification. Tables 4.15 and 4.16 show the time consumed in training and testing by DT. Table 4.17 illustrates the identification rate and speakers not identified by DT for 10 to 120 speakers. Table 4.18 illustrates the identification rate and speakers not identified by DT for six groups of 25 randomly chosen speakers labeled with A, B, C, D, E, and F.

Table 4.15 DT Training/Testing Time for 10-120 Speakers

No. of Speakers	Train (sec)	Test (sec)
10	193.984666	25.242898
20	260.919403	39.303117
30	351.886925	64.2044215
40	487.625768	139.0263385
50	641.225143	181.775019
60	911.575004	175.653267
70	1061.486839	239.9670805
80	1931.438243	488.5699825
90	193.984666	25.242898
100	260.919403	39.303117

110	351.886925	64.2044215
120	487.625768	139.0263385

Table 4.16 DT Training/Testing Time for 25 Speakers in Groups

Group Label (25 speaker)	Train (sec)	Test (sec)
A	49.894788	13.761732
B	51.133344	16.268922
C	49.202585	19.890045
D	43.061313	15.117821
E	38.246927	10.22457
F	50.780036	10.804681

Table 4.17 DT Identification Rate for 10-120 Speaker

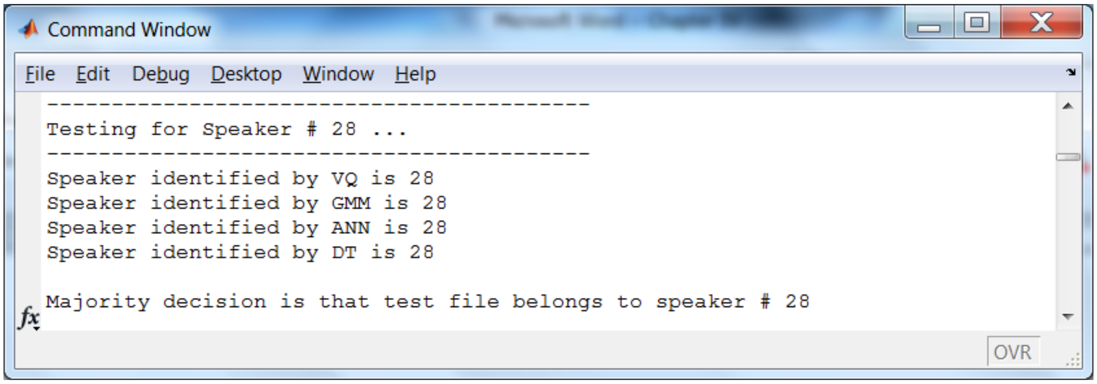
No. of Speakers	Identification Rate (%)	Speaker not Identified
10	100	none
20	94	sp6
30	86.8687	sp17
40	82.069	sp6, sp8, sp14, sp17
50	80.2956	none
60	71.134	sp7, sp10, sp40, sp43
70	63.3588	sp4, sp7, sp11, sp31, sp61, sp62
80	54.0594	sp4, sp7, sp18, sp40, sp45, sp46, sp55, sp61, sp62
90	52.6403	sp7, sp46, sp60, sp61, sp62
100	47.5456	sp4, sp33, sp40, sp58, sp61, sp71, sp90, sp98
110	45.6548	sp4, sp7, sp11, sp17, sp20, sp23, sp40, sp43, sp53, sp61, sp71
120	41.3111	sp1, sp6, sp11, sp40, sp46, sp61, sp89

Table 4.18 DT Identification Rate for Groups of 25 Speakers

Group Label (25 speaker)	Identification Rate (%)	Speaker not Identified
A	92.8571	none
B	90.2256	none
C	78.7402	sp6
D	84.7222	none
E	83.871	none
F	75.4386	none

4.2.6 Fusion

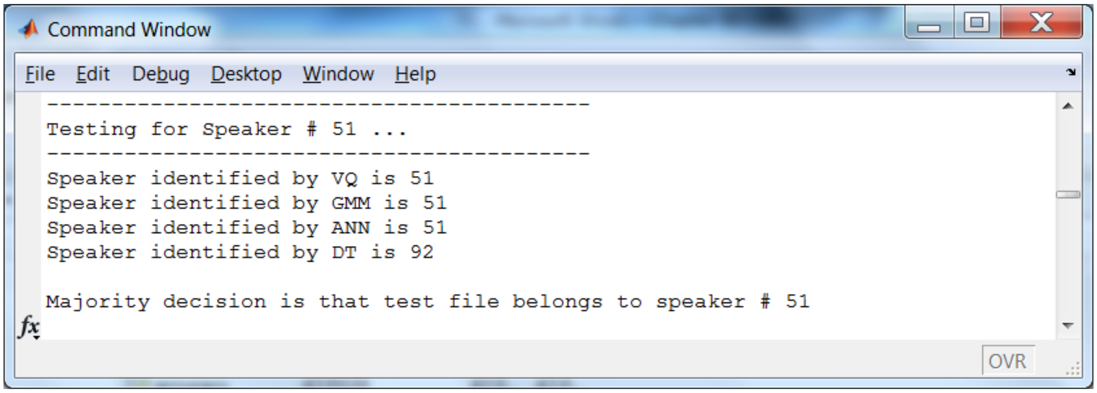
The Majority Rule is applied to the results of the four previous algorithms as a fusion method. The solutions of VQ, GMM, ANN, and DT are compared and the final result is represented by the majority of solutions. Because four results are obtained, two or more equal results will be chosen as a fusion result. Figure 4.1 shows an example of case 1 where all the four algorithms agree to identify the same speaker. Figure 4.2 shows an example of case 2 where three from four algorithms agree to identify the same speaker. Figure 4.3 shows an example of case 3 where two algorithms agree to identify the same speaker and the other two algorithms did not agree on a result. Cases 1, 2, and 3 show when fusion is applied successfully but if results are equally produced, then fusion cannot be done. Table 4.19 illustrates the identification rate and speakers not identified by fusion for 10 to 120 speakers. Table 4.20 illustrates the identification rate and speakers not identified by fusion for six groups of 25 randomly chosen speakers labeled with A, B, C, D, E, and F.



```
------------------------------------------
Testing for Speaker # 28 ...
------------------------------------------
Speaker identified by VQ is 28
Speaker identified by GMM is 28
Speaker identified by ANN is 28
Speaker identified by DT is 28

Majority decision is that test file belongs to speaker # 28
```

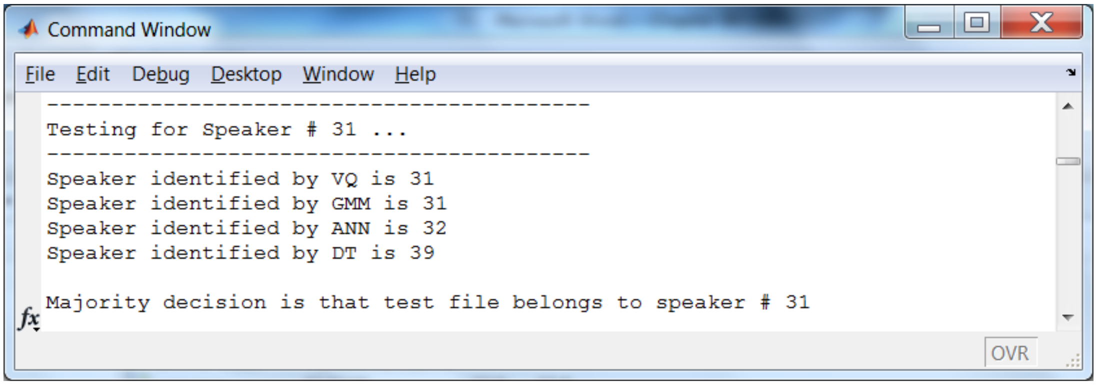
Figure 4.1 Majority Decision Fusion Example of Case 1



```
------------------------------------------
Testing for Speaker # 51 ...
------------------------------------------
Speaker identified by VQ is 51
Speaker identified by GMM is 51
Speaker identified by ANN is 51
Speaker identified by DT is 92

Majority decision is that test file belongs to speaker # 51
```

Figure 4.2 Majority Decision Fusion Example of Case 2



```
------------------------------------------
Testing for Speaker # 31 ...
------------------------------------------
Speaker identified by VQ is 31
Speaker identified by GMM is 31
Speaker identified by ANN is 32
Speaker identified by DT is 39

Majority decision is that test file belongs to speaker # 31
```

Figure 4.3 Majority Decision Fusion Example of Case 3

Table 4.19 Identification Rate for 10-120 Speakers by Fusion

No. of Speakers	Identification Rate (%)	Speaker not Identified
10	100	none
20	98	none
30	87.8788	none
40	88.9655	none
50	90.1478	none
60	88.6598	none
70	77.6081	sp7, sp61, sp70
80	75.0495	sp7, sp40, sp61, sp71
90	73.9274	sp61
100	72.0898	sp61, sp71
110	70.3794	sp71
120	62.3309	sp61, sp71, sp115, sp118

Table 4.20 Identification Rate for Groups of 25 Speakers by Fusion

Group Label (25 speaker)	Identification Rate (%)	Speaker not Identified
A	98.5714	none
B	95.4887	none
C	86.6142	none
D	95.1389	none
E	89.2473	sp21
F	90.6433	none

4.2.7 Evaluation

The third set of experiments aims to evaluate the performance of the four classifiers plus the fusion. Tests are applied on group of speakers contain both true and imposter speakers. The database is divided into 60% for true speakers and 40% for imposters. We took the 25 speakers of groups A, B, C, D, E, and F and replace 40% of them with other speakers (as imposters) to get 10 imposters and 15 true speakers. We also took the database of 50 and 100 speakers and divided the speakers into 60% true speakers and 40% imposters, resulting in 30 true speakers plus 20 imposters for the 50-speaker database and 60 true speakers plus 40 imposters for the 100-speaker database. Precision, Recall, and accuracy are shown for VQ, GMM, ANN, DT, and fusion for performance evaluation. Tables 4.21, 4.22, 4.23, 4.24, and 4.25 illustrate the performance measures when applied to groups of 25 speakers by VQ, GMM, ANN, DT, and fusion respectively. Tables 4.26 and 4.27 illustrate performance measures of VQ, GMM, ANN, DT, and fusion when applied on 50 and 100 speakers respectively.

Table 4.21 VQ Performance Measures on Groups of 25 Speakers

Group Label (25 speaker)	TP	TN	FP	FN	Precision	Recall	Accuracy(%)
A	15	9	1	0	0.9375	1	96
B	15	10	0	0	1	1	100
C	15	7	3	0	0.8333	1	88
D	15	10	0	0	1	1	100
E	15	8	2	0	0.8824	1	92
F	15	9	1	0	0.9375	1	96

Table 4.22 GMM Performance Measures on Groups of 25 Speakers

Group Label (25 speaker)	TP	TN	FP	FN	Precision	Recall	Accuracy(%)
A	15	10	0	0	1	1	100
B	15	10	0	0	1	1	100
C	15	6	4	0	0.7895	1	84
D	15	9	1	0	0.9375	1	96
E	15	7	3	0	0.8333	1	88
F	15	10	0	0	1	1	100

Table 4.23 ANN Performance Measures on Groups of 25 Speakers

Group Label (25 speaker)	TP	TN	FP	FN	Precision	Recall	Accuracy(%)
A	14	10	0	1	1	0.9333	96
B	15	10	0	0	1	1	100
C	14	9	1	1	0.9333	0.9333	92
D	15	9	1	0	0.9375	1	96
E	15	8	2	0	0.8824	1	92
F	14	10	0	1	1	0.9333	96

Table 4.24 DT Performance Measures on Groups of 25 Speakers

Group Label (25 speaker)	TP	TN	FP	FN	Precision	Recall	Accuracy(%)
A	15	9	1	0	0.9375	1	96
B	15	10	0	0	1	1	100
C	14	6	4	1	0.7778	0.9333	80
D	15	6	4	5	0.7895	1	84
E	15	7	3	0	0.8333	1	88
F	15	9	1	0	0.9375	1	96

Table 4.25 Performance Measures on Groups of 25 Speakers by Fusion

Group Label (25 speaker)	TP	TN	FP	FN	Precision	Recall	Accuracy(%)
A	15	10	0	0	1	1	100
B	15	10	0	0	1	1	100
C	15	8	2	0	0.8824	1	92
D	15	10	0	0	1	1	100
E	15	7	3	0	0.8333	1	88
F	15	10	0	0	1	1	100

Table 4.26 Performance Measures on 50 Speakers by Different Methods

SI Method	TP	TN	FP	FN	Precision	Recall	Accuracy(%)
VQ	30	17	3	0	0.90909	1	94
GMM	30	17	3	0	0.90909	1	94
ANN	25	17	3	5	0.89285	0.83333	84
DT	30	17	3	0	0.90909	1	94
Fusion	30	18	2	0	0.9375	1	96

Table 4.27 Performance Measures on 100 Speakers by Different Methods

SI Method	TP	TN	FP	FN	Precision	Recall	Accuracy(%)
VQ	60	39	1	0	0.9836	1	99
GMM	60	37	3	0	0.9524	1	97
ANN	56	39	1	4	0.9825	0.9333	95
DT	56	36	4	4	0.9333	0.9333	92
Fusion	60	39	1	0	0.9836	1	99

Chapter 5

Discussion

Chapter V

Discussion

5.1 Introduction

This chapter focuses on discussing results of all the experiments presented in previous chapters. As shown in chapter 4, the identification rates of VQ, GMM, ANN, and DT are presented for 10 to 120 speaker databases. In addition, the identification rate by the same four algorithms is performed on groups of 25 speakers labeled with A, B, C, D, E, and F. For any speaker database, majority decision is applied to get the final result of identification rate. As illustrated before, some approaches did not identify correctly all the speakers. So, evaluation measurements are taken into account to measure the performance by certain metrics such as precision, recall, and accuracy. These performance measures need TP, TN, FP, and FN to be computed for each algorithm applied to any speaker database.

5.2 Execution Time

All the experiments in this study are time calculated in seconds to figure out the time consumed to execute certain procedure. There are two main modules: feature extraction by MFCCs and feature matching by VQ, GMM, ANN, DT, and fusion.

5.2.1 Feature Extraction

Feature extraction module is needed before training or testing speaker databases. As shown in figure 5.1 as long as the size increased in speaker database from 10 to 120 speaker, more time needed for MFCCs to extract features from speech signal with a notice that time is almost the same for 80, 90, and 100 speakers. The case of 25 speakers randomly chosen and grouped in A, B, C, D, E, and F, the time for MFCCs is varied from about 19 to 37 seconds as shown in figure 5.2.

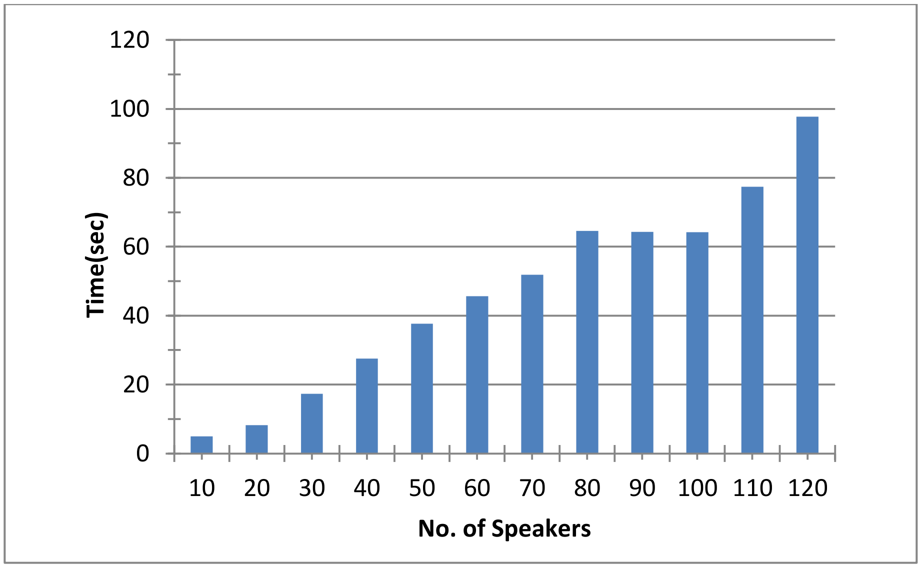


Figure 5.1 MFCCs Feature Extraction Time for 10 to 120 Speakers

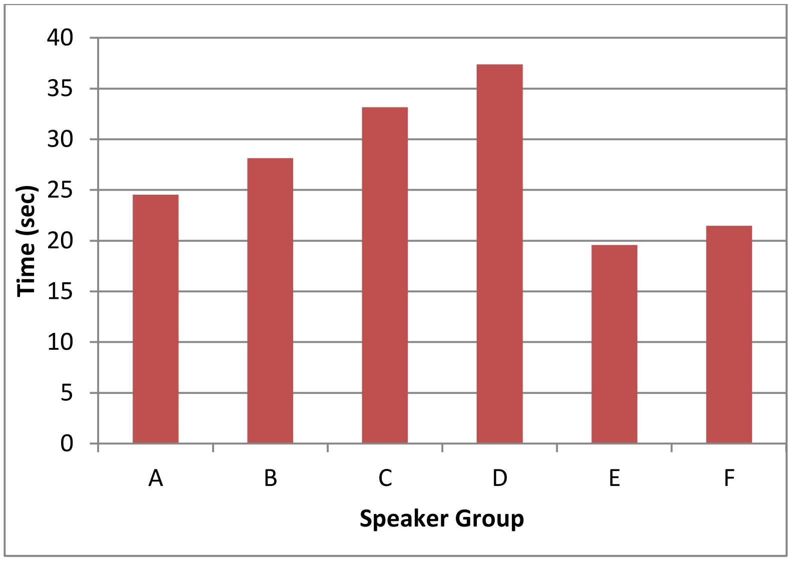


Figure 5.2 MFCCs Feature Extraction Time for Speaker Groups

5.2.2 Pattern Matching

After feature extraction by MFCCs, the pattern matching module is applied. The speaker database is represented by either 10 to 120 speakers or groups of 25 speakers. The main observation about VQ is that the time needed for training is small compared to testing. For GMM, ANN, and DT the results show that we need more time for training compared to testing. Figure 5.3 shows the time difference in training by VQ, GMM, and DT algorithms. VQ is the fastest training algorithm for 10 to 120 speaker databases, then GMM, after that DT. The ANN results are omitted from comparison because it took very long time for training in comparison to the other three algorithms as shown in figure 5.4. This is obvious with groups A, B, C, D, E, and F of 25 speakers where ANN takes the longest time in training as shown in figure 5.5.

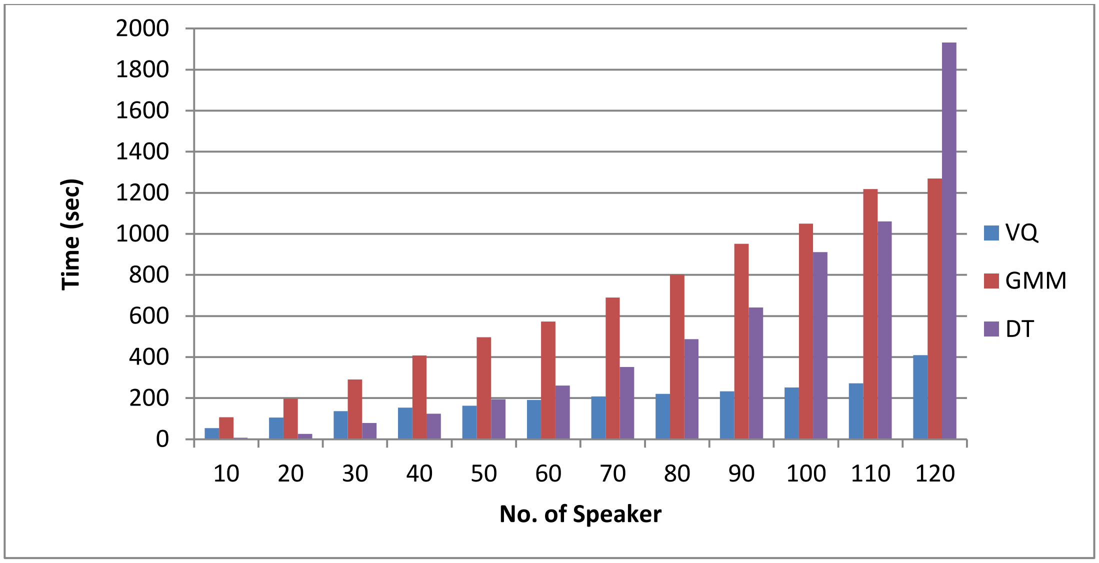


Figure 5.3 Training Time by VQ, GMM, and DT for 10 to 120 Speakers

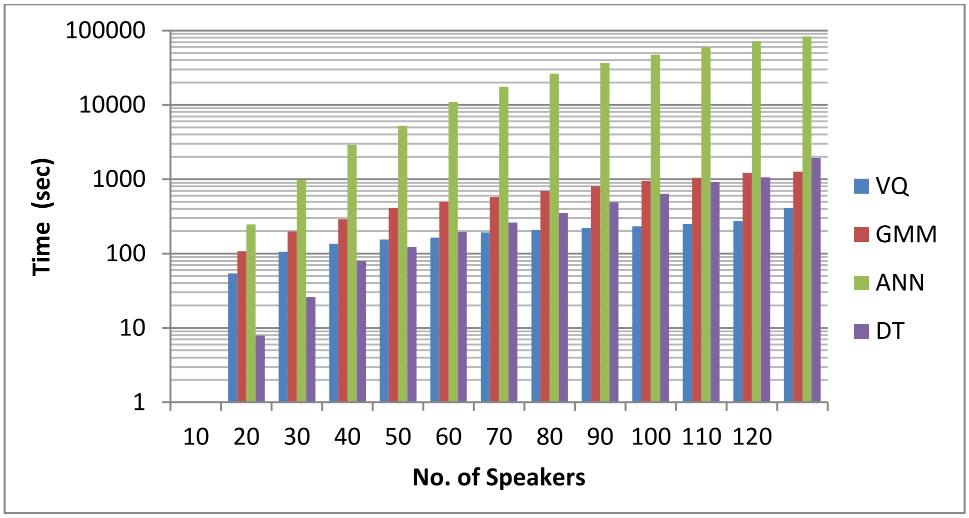


Figure 5.4 Training Time by VQ, GMM, ANN, and DT for 10 to 120 Speakers

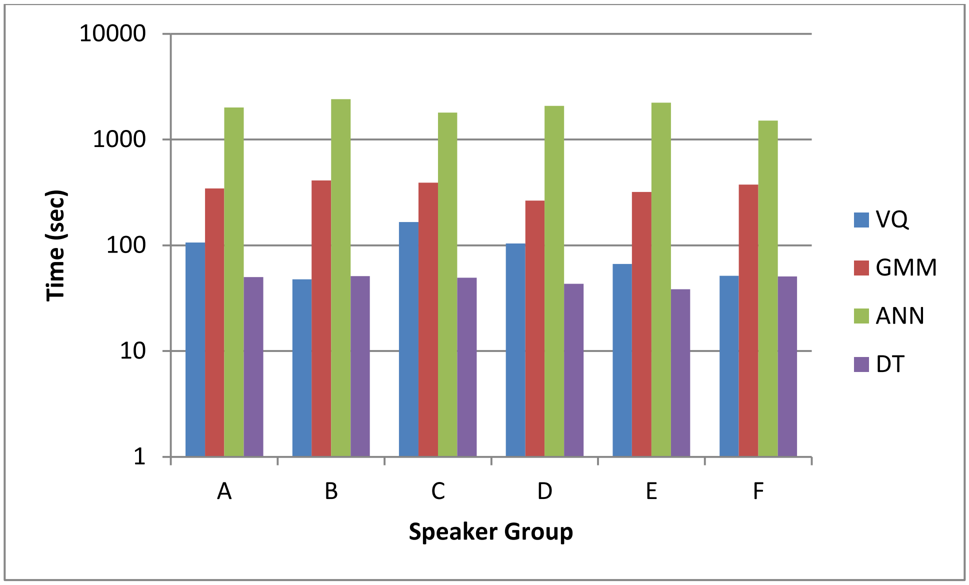


Figure 5.5 Training Time by VQ, GMM, ANN, and DT for Speakers Groups

Testing time as illustrated in figure 5.6 shows that GMM and DT takes the least time and VQ takes the longest for 10 to 120 speakers and this is true for groups of 25 speakers as shown in figure 5.7.

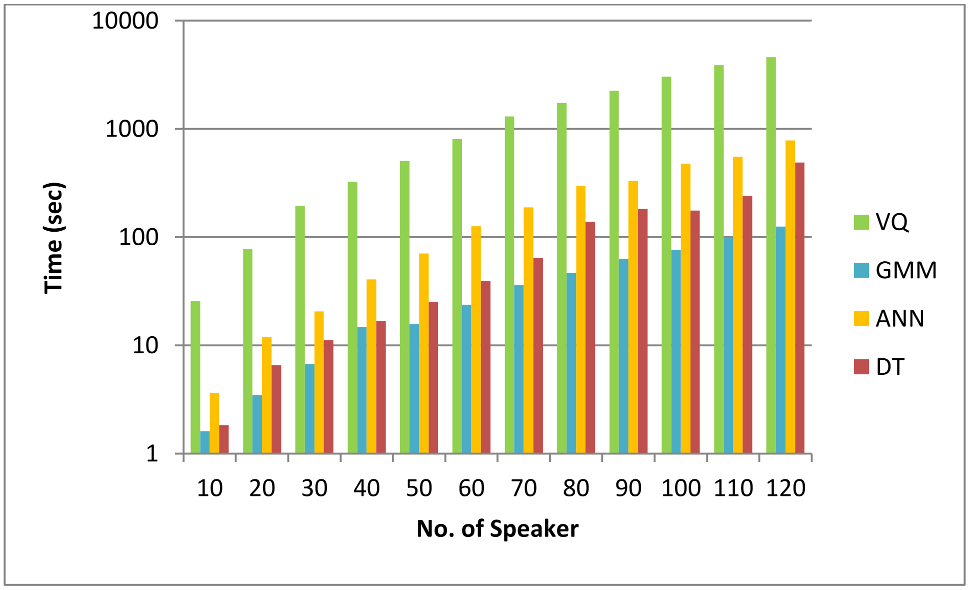


Figure 5.6 Test Time by VQ, GMM, ANN, and DT for 10 to 120 Speakers

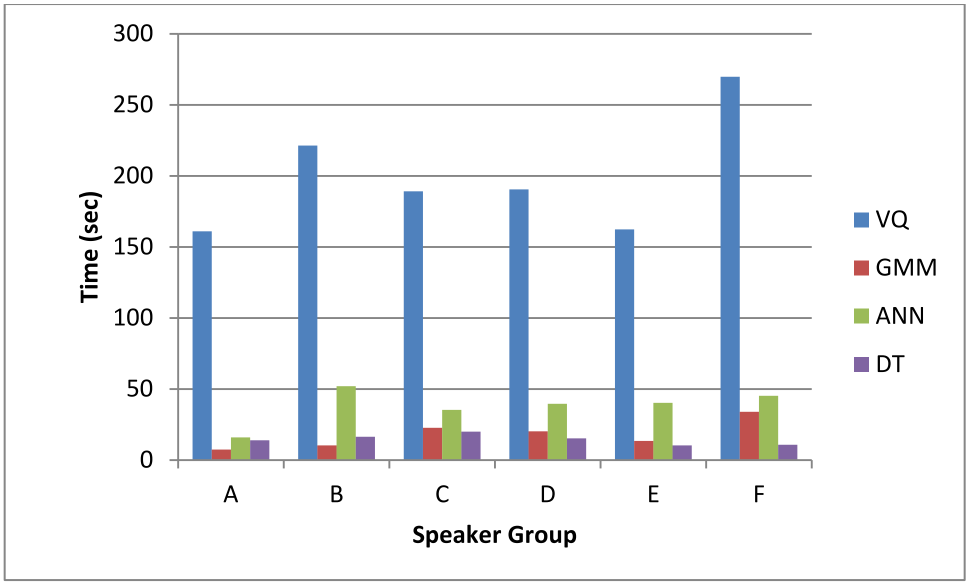


Figure 5.7 Test Time by VQ, GMM, ANN, and DT for Speakers Groups

As shown in previous chapter, the fusion method is combining the testing results of VQ, GMM, ANN, and DT algorithms, so the testing time is the time needed for testing the four algorithms together and decide the majority result.

5.3 Identification Rate

The main goal of this study is to reach to a system that effectively identifies a person from his/her speech. To do that, for training and testing chosen speakers, we depend on common and well known speaker identification algorithms. These algorithms are evaluated based on number of correctly identified test samples divided by the total number of test samples and then multiplied by 100. To measure the accuracy of our experiments, the identification rate is computed. Figure 5.8 shows the different of identification rate between VQ, GMM, ANN, DT, and majority decision fusion. It is obvious from the figure 5.8 that identification rate decreases with the increasing size of speaker database.

From results for 10 to 120 speakers when applying VQ, GMM, ANN, DT, and majority decision fusion method, we got the following observations:

- The best identification rate we got is from GMM then fusion method because both of them maintain identification rate above 90% for 50 speakers and above 60% for 120 speakers.
- VQ results are near from fusion results until 80 speakers then from 90 to 120 speakers results are worse than fusion.
- ANN has stable results of identification rate from 30 to 60 speakers and from 80 to 110 speakers.
- Although DT scores better than ANN from 10 to 50 speakers, it has the worst identification rate results from 60 to 120 speakers.

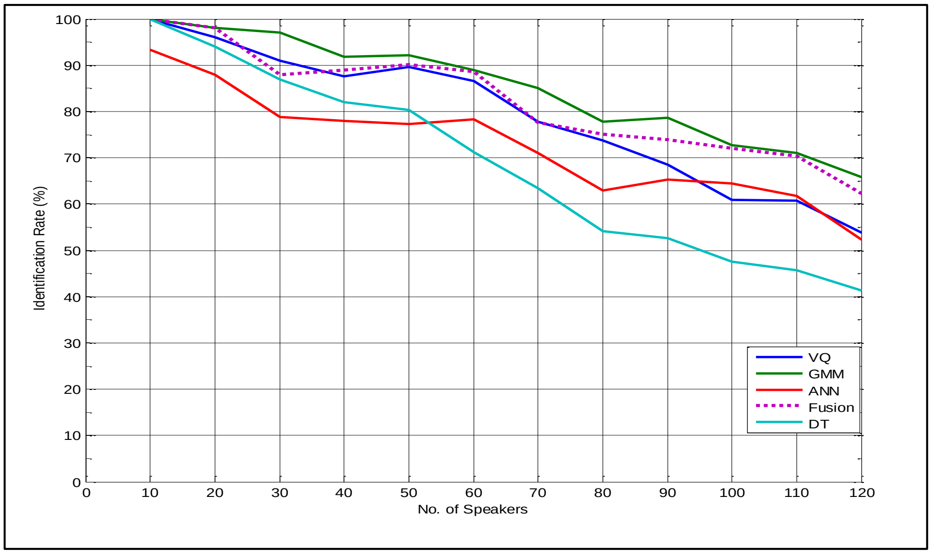


Figure 5.8 Identification Rate for 10 to 120 Speakers by Proposed SIS

It is clear from the previous observations that identification rate scores a high rate (between 85% to 100%) when the number of trained speakers is small. Identification rate scores become worse with large number of speakers.

In this study, we divide the large database to groups and analyze the difference in identification rates. As shown in chapter 4, the second experiments method was to take groups of 25 speakers randomly from the same speaker database of first method experiments. We labeled six groups with A, B, C, D, E, and F and applied the same four algorithms plus the fusion method. Figure 5.9 shows the identification rate results of the six groups. From results of groups of speakers when applying VQ, GMM, ANN, DT, and majority decision fusion method, we got the following observations:

- In general, the best identification rate results come from GMM then fusion method.
- All the four algorithms plus fusion method give the best identification rate results with group A and B of speakers (between 90% and 100%).
- ANN gives the worst identification rate results with group C.
- DT gives the worst identification rate results with group F.
- Since the number of speakers is the same for all groups, the identification rate may differs from group to another according to the speakers' environment, noise, and training time.
- By taking the average identification rate from all groups A, B, C, D, E, and F for four algorithms plus fusion method, GMM gives the best identification rate, then fusion, then VQ, after that ANN, finally DT as depicted in figure 5.10.

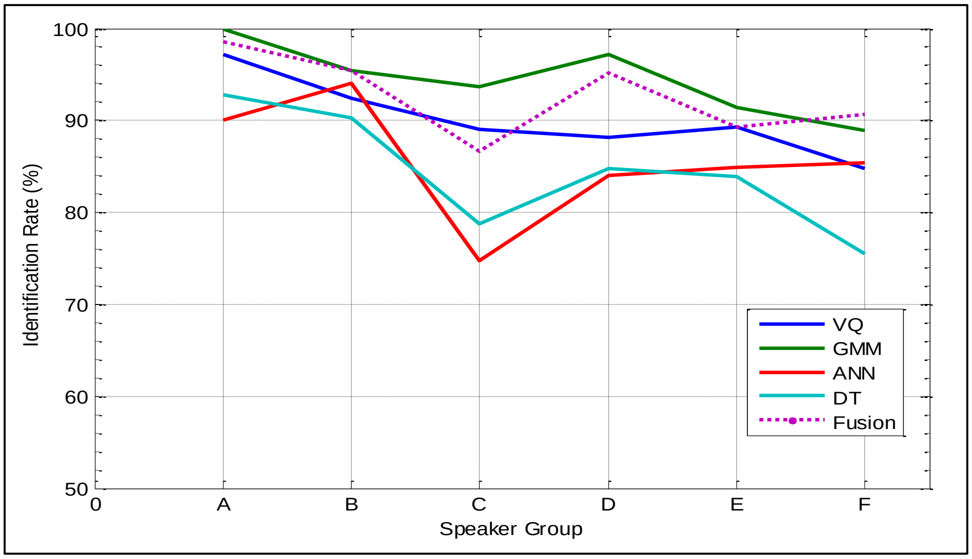


Figure 5.9 Identification Rate for Speakers Groups by Proposed SIS

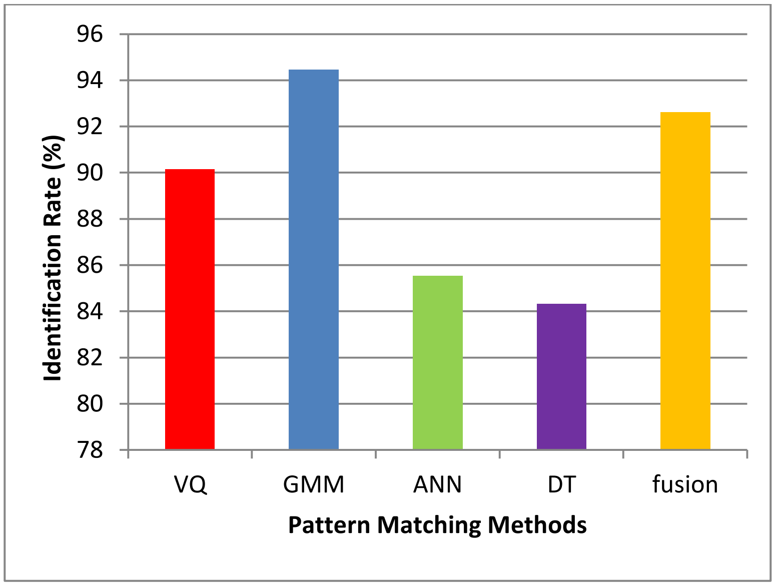


Figure 5.10 Average Identification Rate for Proposed SIS over 25 Speakers

5.4 Speaker Misidentification

Different identification rates indicate that there are some speakers had not been identified. As shown in chapter 4, as long as the speaker database size increases the number of misidentified speakers increases and this is varies from one algorithm method to another. Figure 5.11 illustrates the following observations:

- GMM has the best identification number of speakers because the worst case is 2 misidentification.
- VQ has almost the same of GMM results but when speaker database becomes 120 speakers, we got 5 misidentification of speakers.
- Fusion results give us the best identification number of results with 10 to 60 speakers, then with 70 or more speakers: 1 to 4 misidentification starts to appear.
- When the size of speaker database is 10 speakers, all the four algorithms with the fusion procedure identify all the 10.
- Start to see 1 misidentification speaker by ANN and DT algorithms when speaker database between 20 and 30 speakers.
- Rapid increase in misidentification appears clearly by ANN algorithm with 70 to 120 speakers.
- Generally, misidentification results start to appear with 80 or more speakers for all the four algorithms plus fusion procedure.
- The worst result is 26 misidentification with 120 speakers by ANN algorithm.

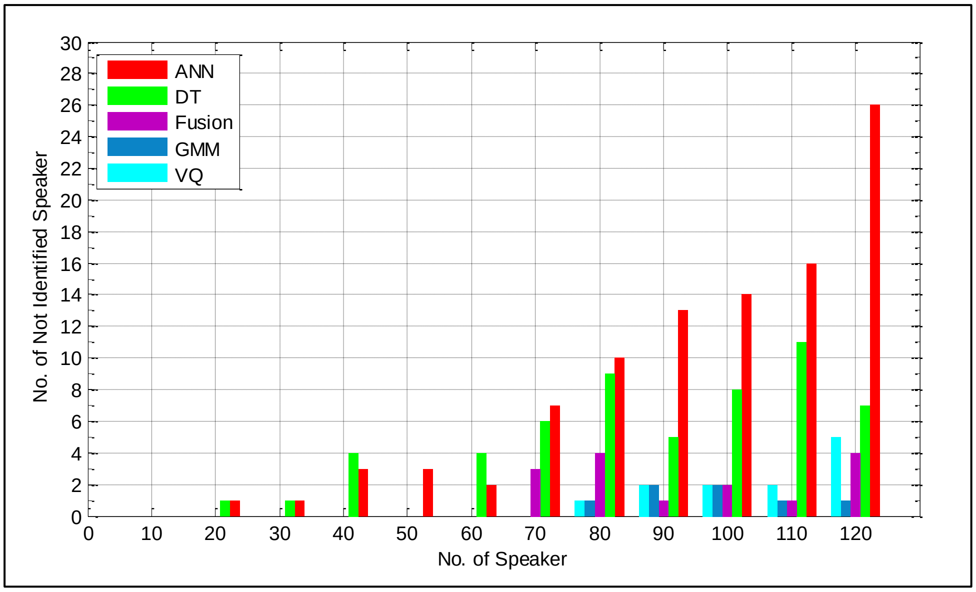


Figure 5.11 No. of Misidentification for 10 to 120 Speakers by Proposed SIS

As results start to be not effective when number of speakers is large, we tested another method of experiments by choosing randomly 25 speakers and put them in groups. As shown in figure 5.12, misidentification number of six groups labeled with A, B, C, D, E, and F and applied the same previous algorithms to these groups. The worst case is one speaker has not been identified by ANN, DT in group C. Beside group C, groups A, D, and F has one misidentification when ANN algorithm applied. Fusion method has only one misidentification in group E.

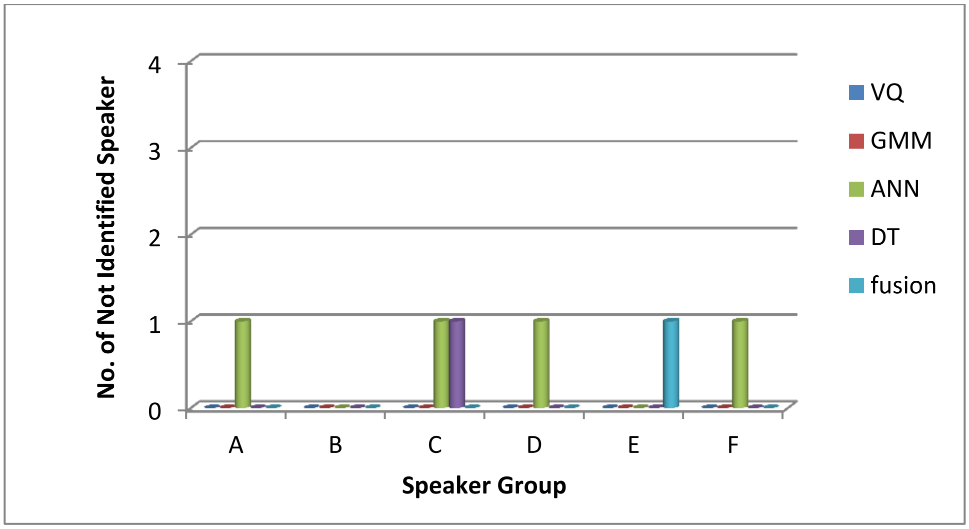


Figure 5.12 No. of Misidentification for Speakers Groups by Proposed SIS

5.5 Performance Evaluation Measurements

Precision, recall, and accuracy are the performance metrics used in this study to evaluate classification algorithms. To calculate these metrics we need confusion matrix elements: TP, TN, FP, and FN. We are testing the following speaker databases:

- Groups of 25 speakers labeled with A, B, C, D, E, and F. Each group contains 15 true speakers and 10 imposters.
- Speaker database of 50 speakers divided into 30 true speakers and 20 imposters.
- Speaker database of 100 speakers divided into 60 true speakers and 40 imposters.

5.5.1 VQ Performance Measurements

Figure 5.13 shows FP and FN of VQ algorithm with speaker groups. As seen from the figure there is no reject for true speaker (false negative). On the other hand, there is one false acceptance for imposters (false positive) in groups A and F, two FP in group E, and three FP in group C. There is no FP in groups B and D.

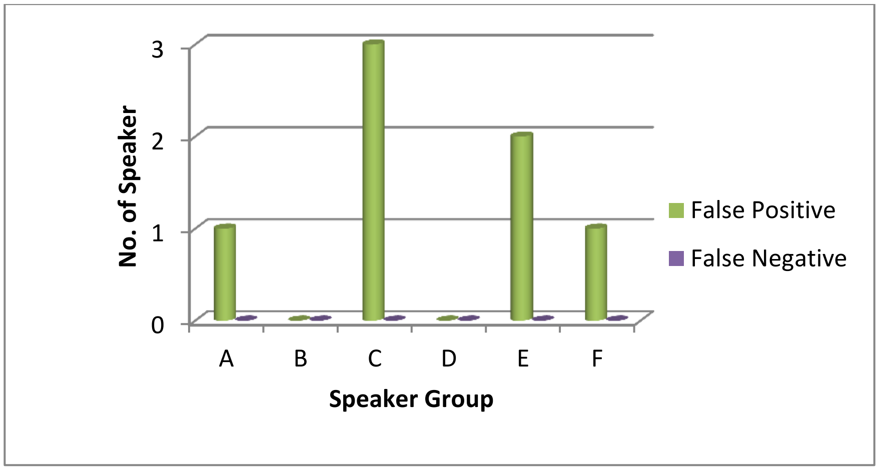


Figure 5.13 False Positive/False Negative for Speakers Groups by VQ

From confusion matrix of speaker groups when VQ algorithm is applied we had calculated precision and recall as illustrated in figure 5.14. Because of the FN=0, Recall=1 in all groups which is the optimal value, where is precision has the optimal value in groups B and D. Group C has the worst precision value since FP=3 in this group, then group E with FP=2. Both groups A and F have the same precision value which is about 0.9 for FP=1.

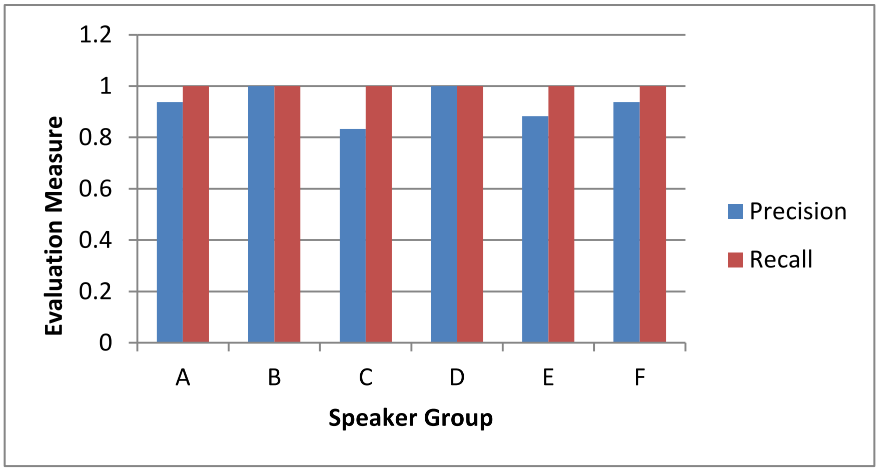


Figure 5.14 Precision and Recall for Speakers Groups by VQ

Overall accuracy is calculated from the confusion matrix for each speaker group when VQ algorithm is applied, as shown in figure 5.15. The worst accuracy is with group C with 88% and the best one is achieved in groups B and D with 100%. The other groups are between 92% and 96%. Since all the groups have the same number of speakers, the average accuracy of VQ is 95.3333% with 25 speakers groups.

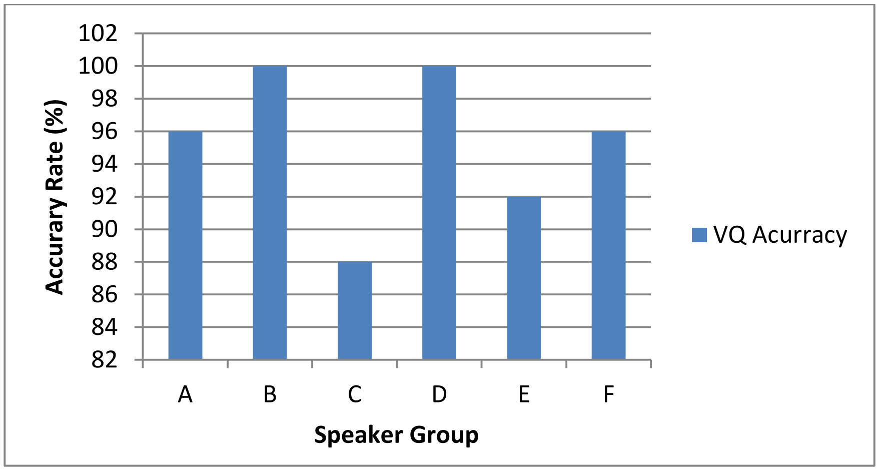


Figure 5.15 Accuracy Rate (%) for Speakers Groups by VQ

The same tests are done for FP and FN (Figure 5.16), precision and recall (Figure 5.17) and overall accuracy (Figure 5.18) in cases of groups of 50 and 100 speakers. The database of 50 speakers has FP=3 which give precision= 0.9091 and FN=0 which give the optimal recall=1. The database of 100 speakers has FP=1 which give precision= 0.9836 and FN=0 which give the optimal recall=1. Accuracy results of VQ for 50 and 100 speakers are 94% and 99% respectively.

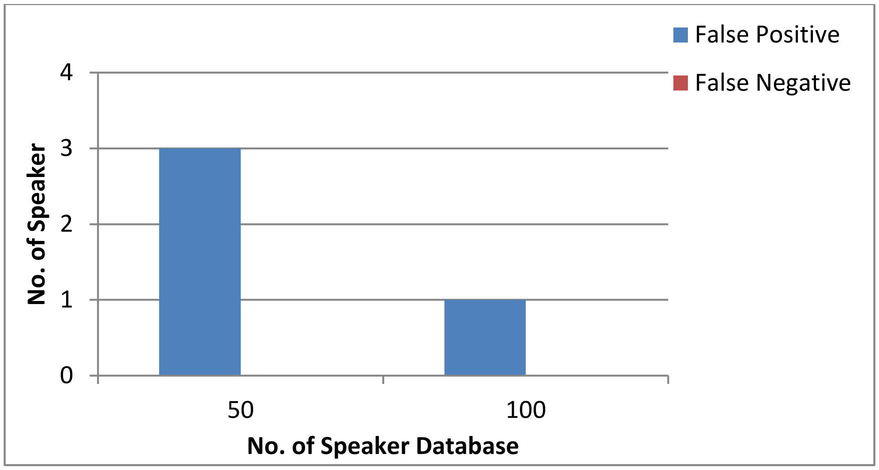


Figure 5.16 False Positive/False Negative for 50 and 100 speakers by VQ

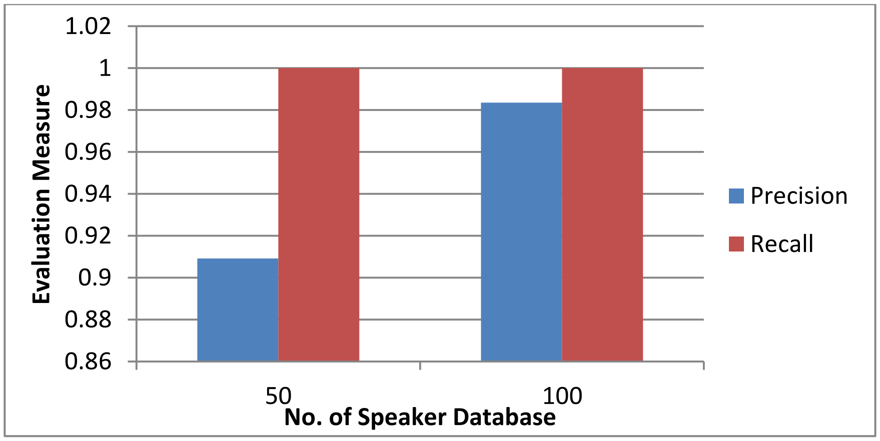


Figure 5.17 Precision and Recall for 50 and 100 speakers by VQ

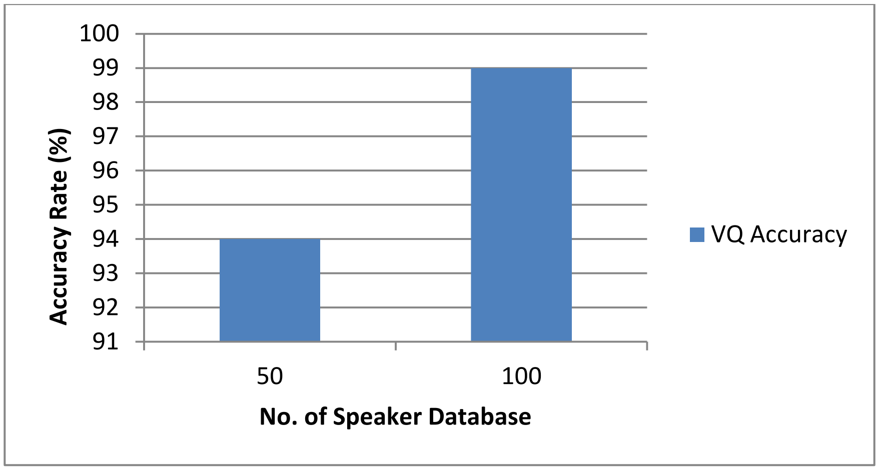


Figure 5.18 Accuracy Rate (%) for 50 and 100 speakers by VQ

5.5.2 GMM Performance Measurements

Figure 5.19 shows FP and FN of GMM algorithm with speaker groups. As it can be observed, there is no reject for true speaker (false negative) as VQ. On the other hand, there is one false acceptance for imposters (false positive) in group D, three FP in group E, and four FP in group C. There is no FP in groups A, B, and F.

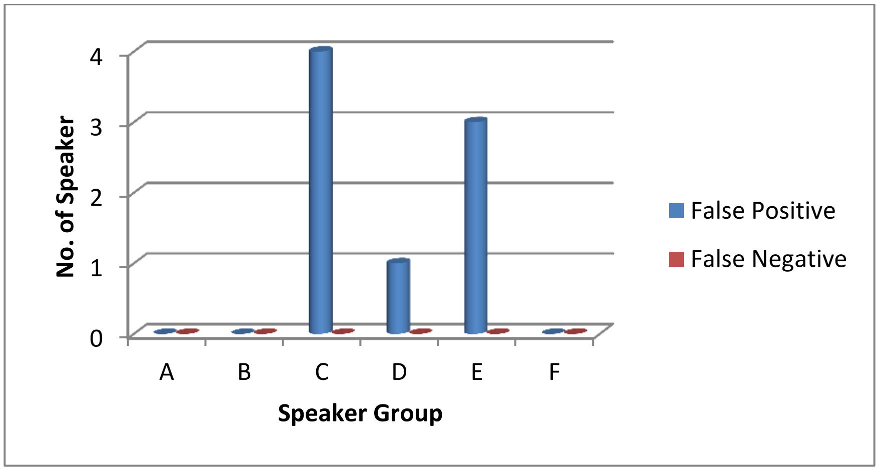


Figure 5.19 False Positive/False Negative for Speakers Groups by GMM

From confusion matrix of speaker groups when GMM algorithm is applied we had calculated precision and recall as illustrated in figure 5.20. Because of the FN=0, Recall=1 in all groups which is the optimal value, where precision has the optimal value in groups A, B, and F. Under optimal, group D has a precision of 0.9375. Group C has the worst precision value since FP=4 in this group, then group E with FP=3.

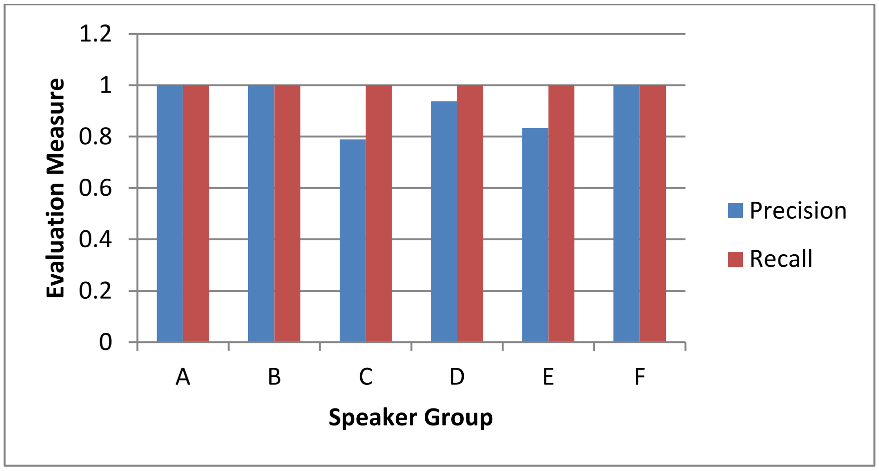


Figure 5.20 Precision and Recall for Speakers Groups by GMM

Overall accuracy is calculated from confusion matrix for each speaker group when GMM algorithm is applied to give results as shown in figure 5.21. As it can be observed, the worst accuracy is with group C with 84% then group E with 88% and the best one is achieved in groups A, B, and F with 100%. Group D has an accuracy of 96%. Since all the groups are with the same number of speakers, the average accuracy of GMM is 94.6667% with 25 speakers groups.

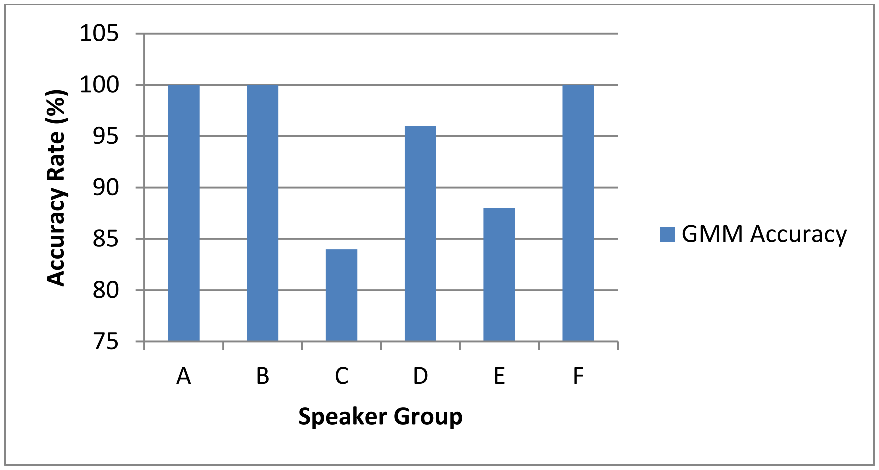


Figure 5.21 Accuracy Rate (%) for Speakers Groups by GMM

The same tests are done for FP and FN (Figure 5.22), precision and recall (Figure 5.23) and overall accuracy (Figure 5.24) in cases of groups of 50 and 100 speakers. The database of 50 and 100 speakers has FP=3 which give precision= 0.9091 for 50 speakers and precision= 0.9524 for 100 speakers. Since FN=0, in both 50 and 100 speaker databases, the recall result gives the optimal value which is recall=1. Accuracy results of GMM for 50 and 100 speakers are 94% and 97% respectively.

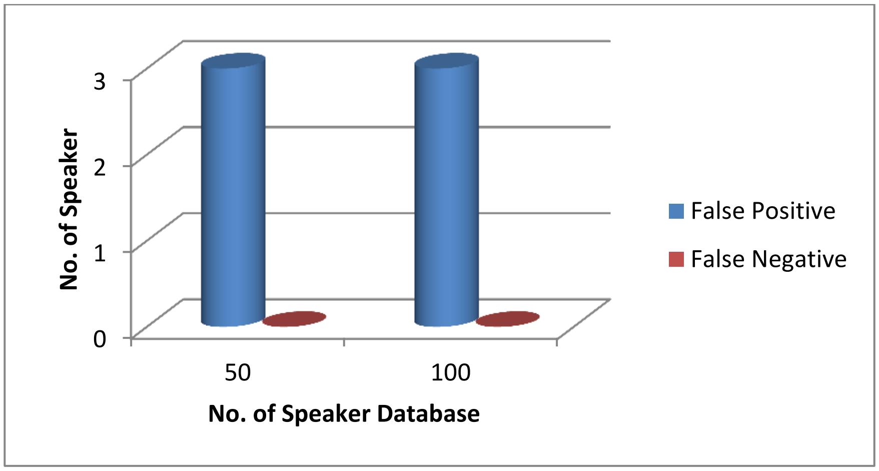


Figure 5.22 False Positive/False Negative for 50 and 100 speakers by GMM

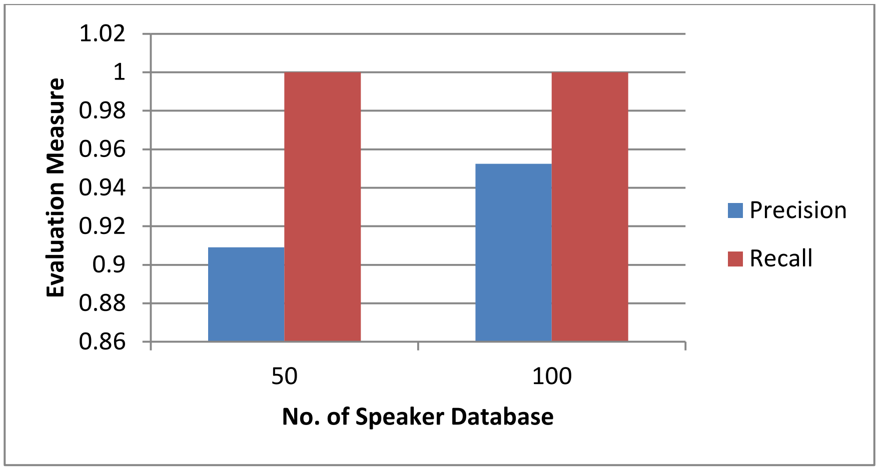


Figure 5.23 Precision and Recall for 50 and 100 speakers by GMM

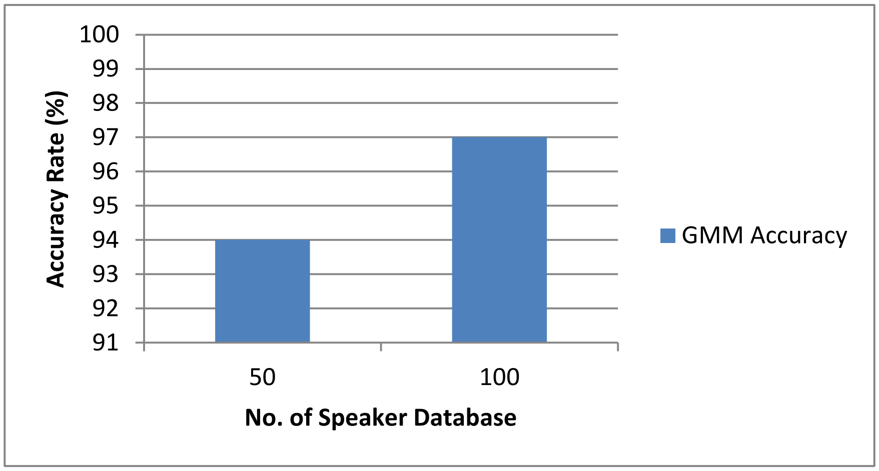


Figure 5.24 Accuracy Rate (%) for 50 and 100 speakers by GMM

5.5.3 ANN Performance Measurements

Figure 5.25 shows FP and FN of ANN algorithm with speaker groups. There is one reject for true speaker (false negative) in groups A, C, and F, but no false reject in groups B, D, and E. In addition, there is one false acceptance for imposters (false positive) in groups C and D, and two FP in group E. There is no FP in groups A, B, and F.

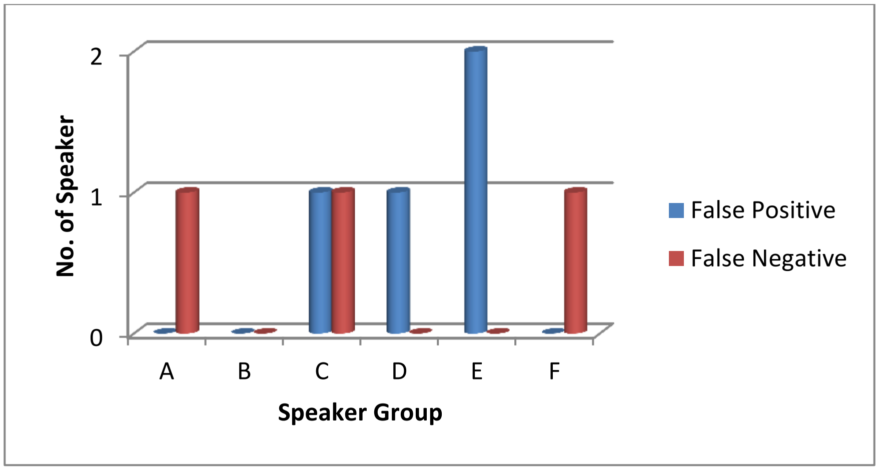


Figure 5.25 False Positive/False Negative for Speakers Groups by ANN

From the confusion matrix of speaker groups when ANN algorithm is applied we had calculated precision and recall as illustrated in figure 5.26. FN=0 in groups B, D, and E, so recall=1 which is the optimal value, but FN=1 in groups A, C, and F, so recall= 0.9333. Precision has the optimal value in groups A, B and F. Group E has the worst precision value since FP=2 in this group, then groups C and D with FP=1.

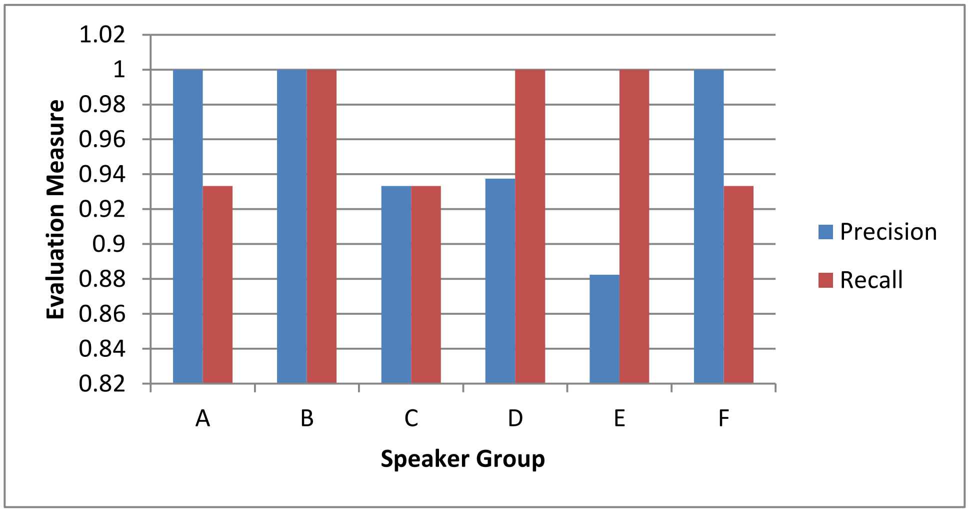


Figure 5.26 Precision and Recall for Speakers Groups by ANN

The overall accuracy is calculated from confusion matrix for each speaker group when ANN algorithm is applied to give results as shown in figure 5.27. As can be observed, the worst accuracy is with groups C and E with 92%. Groups A, D, and F achieved the same accuracy rate, which is 96%, and the best one is achieved by group B with 100%. Since all the groups are with the same number of speakers, the average accuracy of ANN is 95.3333% with 25 speakers groups.

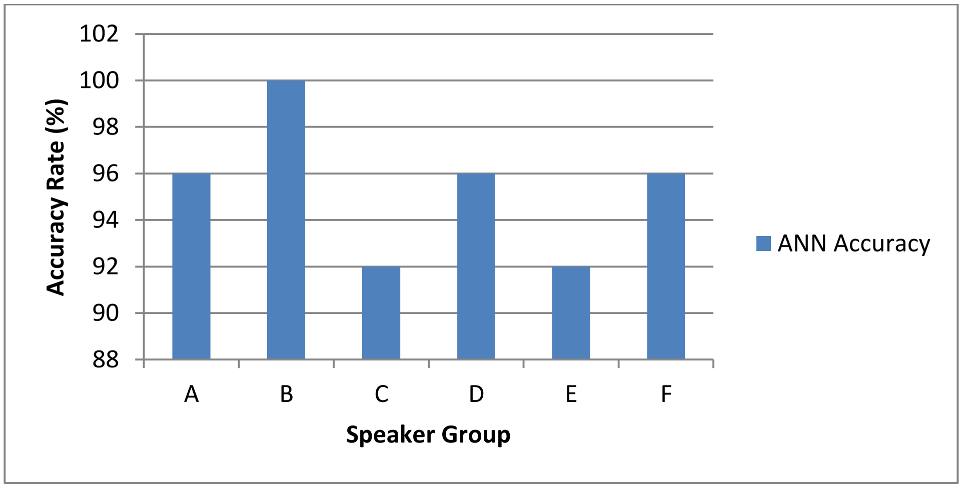


Figure 5.27 Accuracy Rate (%) for Speakers Groups by ANN

The same tests are done for FP and FN (Figure 5.28), precision and recall (Figure 5.29) and overall accuracy (Figure 5.30) in cases of groups of 50 and 100 speakers. The database of 50 speakers has FP=3 and FN=5 which give precision= 0.89285 and recall= 0.83333. Speaker database of 100 has FP=1 and FN=4 which give precision= 0.9825 and recall= 0.9333. Accuracy results of ANN for 50 and 100 speakers are 84% and 95% respectively.

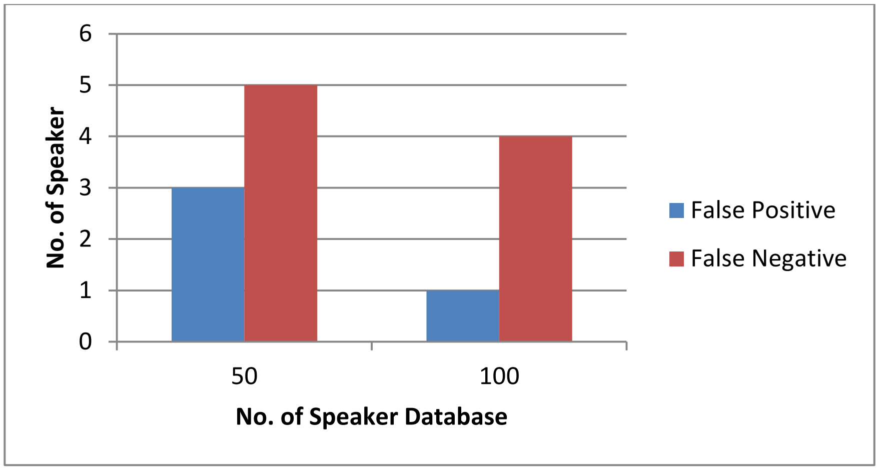


Figure 5.28 False Positive/False Negative for 50 and 100 speakers by ANN

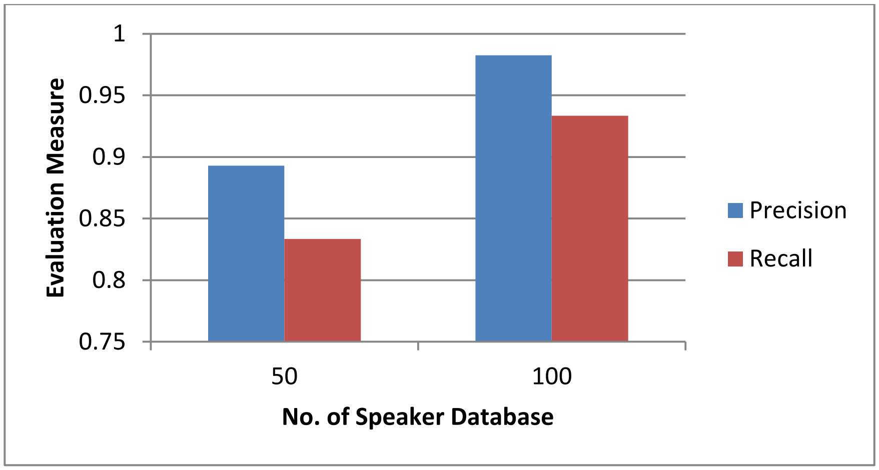


Figure 5.29 Precision and Recall for 50 and 100 speakers by ANN

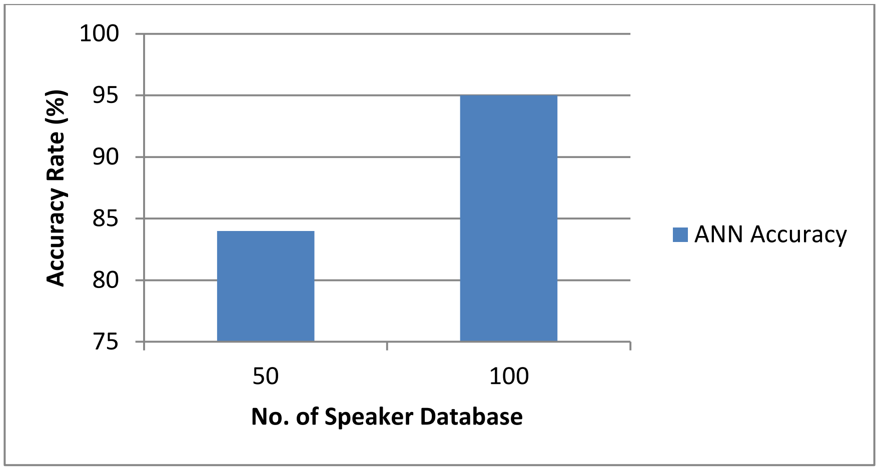


Figure 5.30 Accuracy Rate (%) for 50 and 100 speakers by ANN

5.5.4 DT Performance Measurements

Figure 5.31 shows FP and FN of DT algorithm with speaker groups. As seen from the figure there is only one reject for true speaker (false negative) in group C and no false reject in other groups. Group B is the only one with no false acceptance of imposters (false positive), while groups A and F have one false acceptance, group E with three false acceptances, and groups C and D with four false acceptances.

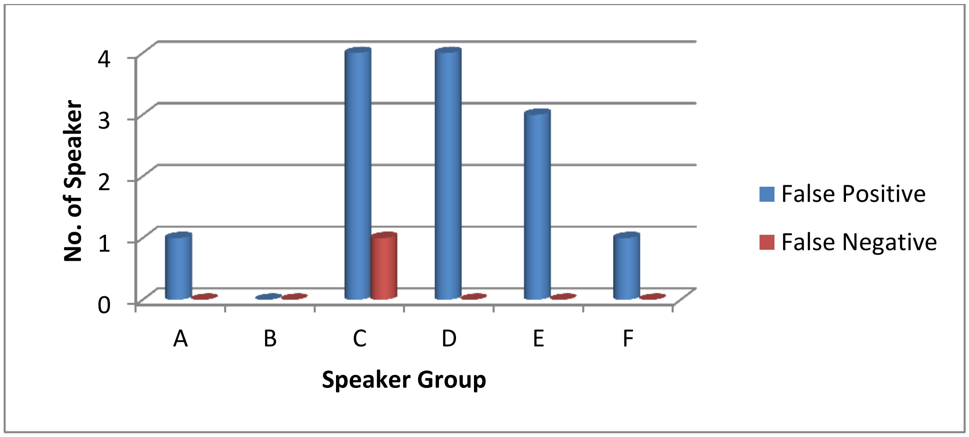


Figure 5.31 False Positive/False Negative for Speakers Groups by DT

From the confusion matrix of speaker groups when DT algorithm is applied we had calculated precision and recall as illustrated in figure 5.32. FN=0 so recall =1 in all groups except group C has FN=1 so recall= 0.9333. Precision has the optimal value in group B only. Groups A and F have precision= 0.9375 since FP=1. Group E has FP=3 so precision= 0.8333. With FP=4 in groups C and D we got the worst precision values 0.7778 and 0.7895 respectively. Notice that the slight difference between C and D precision values with the same FP is according to the TP value.

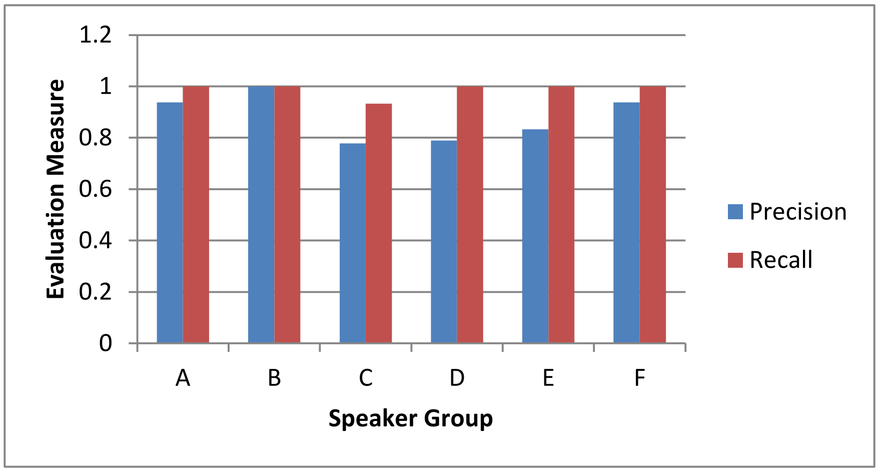


Figure 5.32 Precision and Recall for Speakers Groups by DT

The overall accuracy is calculated from confusion matrix for each speaker group when DT algorithm is applied to give results as shown in figure 5.33. As can be observed, the worst accuracy rate is in group C with 80%, then group D with84%, then group E with 88%. Groups A and F achieved both accuracy rate of 96%. The optimal rate is achieved by group B with 100%. Since all the groups are with the same number of speakers, the average accuracy of DT is 90.6667% with 25 speakers groups.

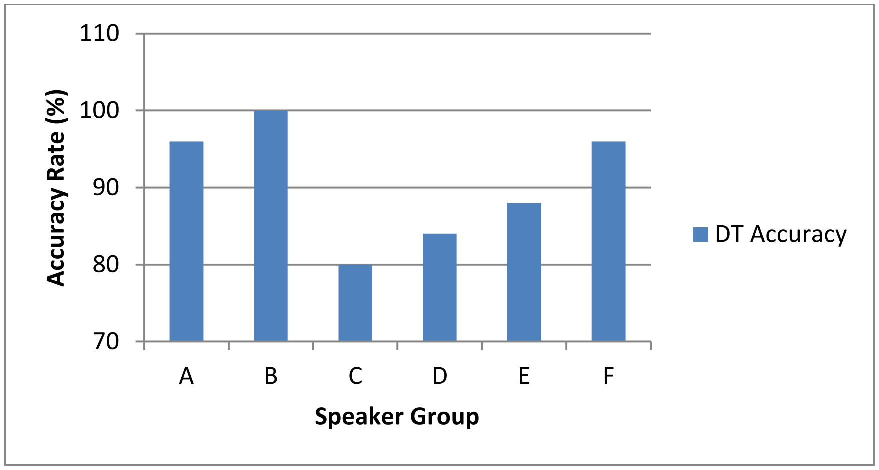


Figure 5.33 Accuracy Rate (%) for Speakers Groups by DT

The same tests are done for FP and FN (Figure 5.34), precision and recall (Figure 5.35) and overall accuracy (Figure 5.36) in cases of groups of 50 and 100 speakers. The database of 50 speakers has FP=3 and FN=0 which give precision= 0.90909 and recall=1. Speaker database of 100 has FP=4 and FN=4 which give precision= 0.9333 and recall= 0.9333. Accuracy results of DT for 50 and 100 speakers are 94% and 92% respectively.

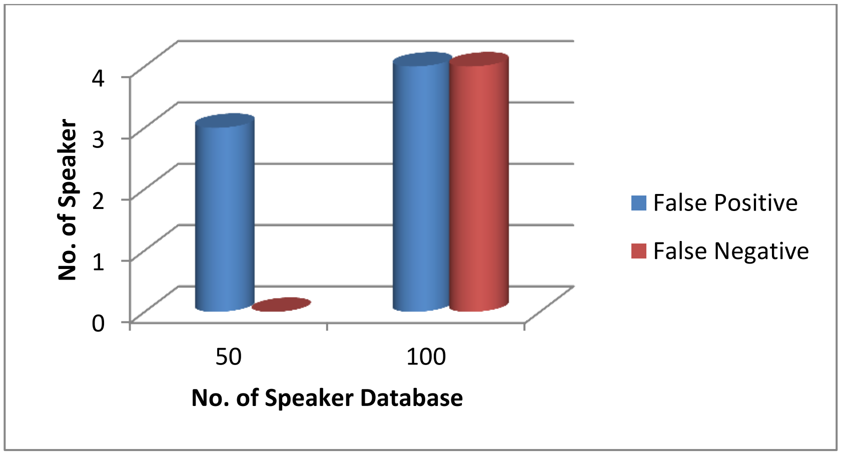


Figure 5.34 False Positive/False Negative for 50 and 100 speakers by DT

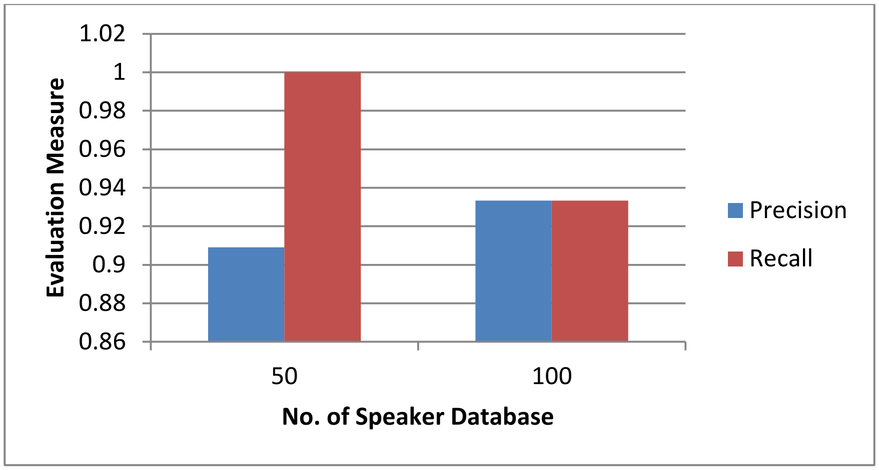


Figure 5.35 Precision and Recall for 50 and 100 speakers by DT

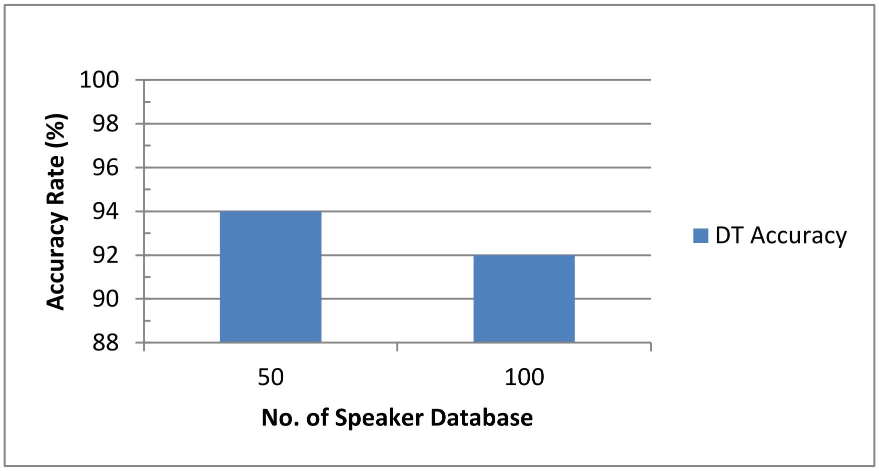


Figure 5.36 Accuracy Rate (%) for 50 and 100 speakers by DT

5.5.5 Fusion Performance Measurements

Figure 5.37 shows FP and FN of fusion method with speaker groups. As seen from the figure there is no reject for true speaker (false negative) in all groups. However, group C has two false acceptance of imposters (false positive) and group D has three false positive of imposters.

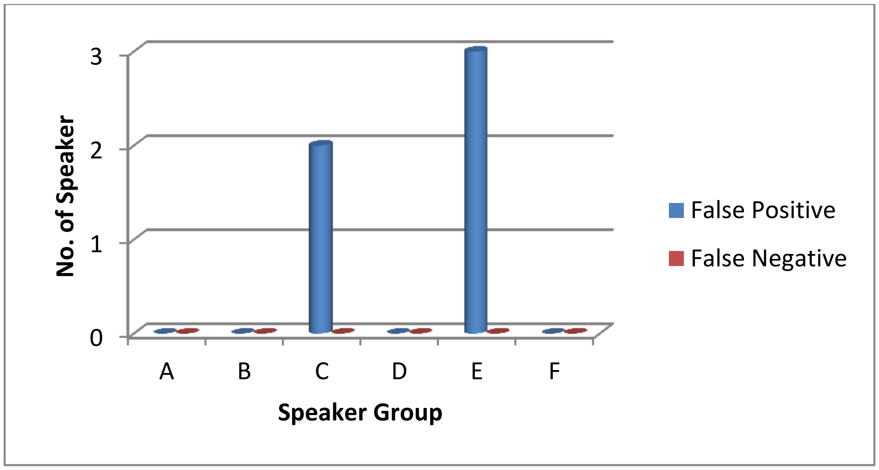


Figure 5.37 False Positive/False Negative for Speakers Groups by Fusion

From the confusion matrix of speaker groups, when fusion method is applied, we had calculated precision and recall as illustrated in figure 5.38. Recall=1 in all groups because FN=0. Precision=1 in all groups except group C and E. Groups C and D have precision= 0.8824 and 0.8333 respectively.

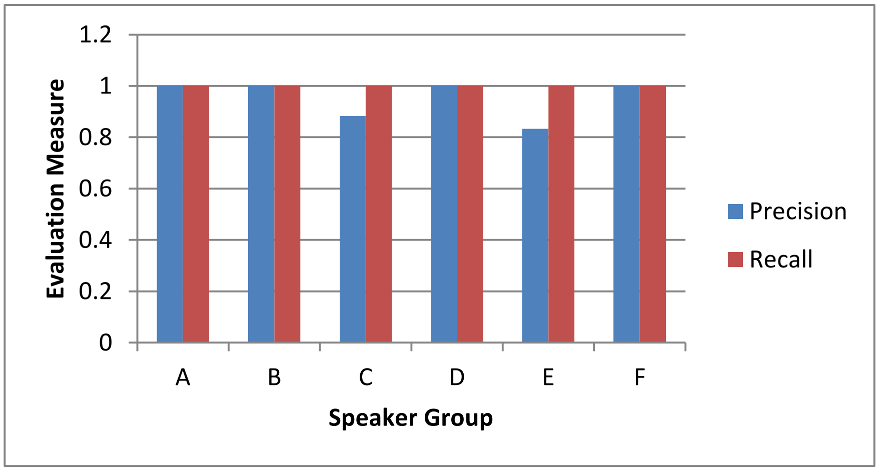


Figure 5.38 Precision and Recall for Speakers Groups by Fusion

The overall accuracy is calculated from confusion matrix for each speaker group when fusion method is applied to give results as shown in figure 5.39. As seen in the figure, the best accuracy rate is achieved by most of the groups in fusion method. Groups A, B, D, and F have accuracy rate of 100%. On other hand, group C has 92% and group E has 88% of accuracy. Since all the groups are with the same number of speakers, the average accuracy of fusion method is 96.6667% with 25 speakers groups.

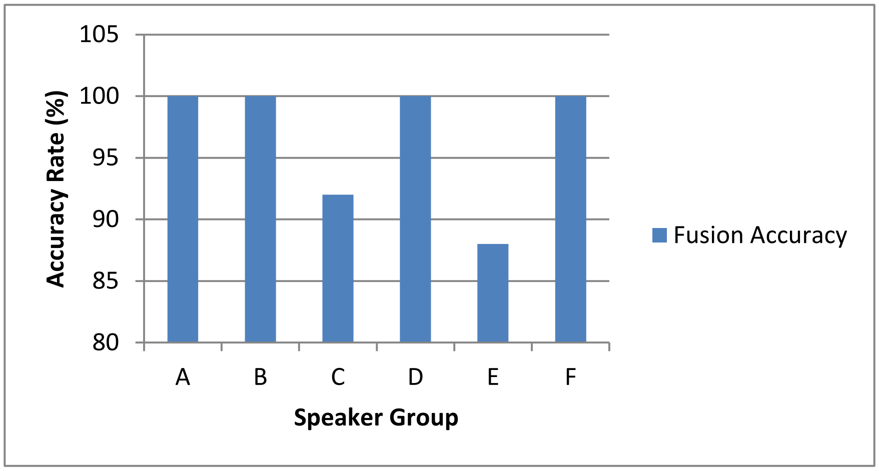


Figure 5.39 Accuracy Rate (%) for Speakers Groups by Fusion

The same tests are done for FP and FN (Figure 5.40), precision and recall (Figure 5.41) and overall accuracy (Figure 5.42) in cases of groups of 50 and 100 speakers. The databases of 50 and 100 speakers has FN=0 which give recall=1. The database of 50 speakers has FP=2 which give precision= 0.9375. Speaker database of 100 has FP=1 which give precision= 0.9836. Accuracy results of fusion for 50 and 100 speakers are 96% and 99% respectively.

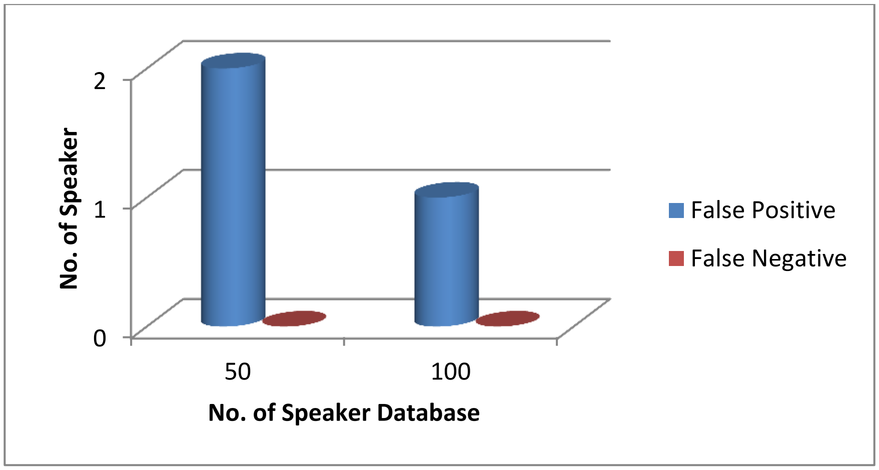


Figure 5.40 False Positive/False Negative for 50 and 100 speakers by Fusion

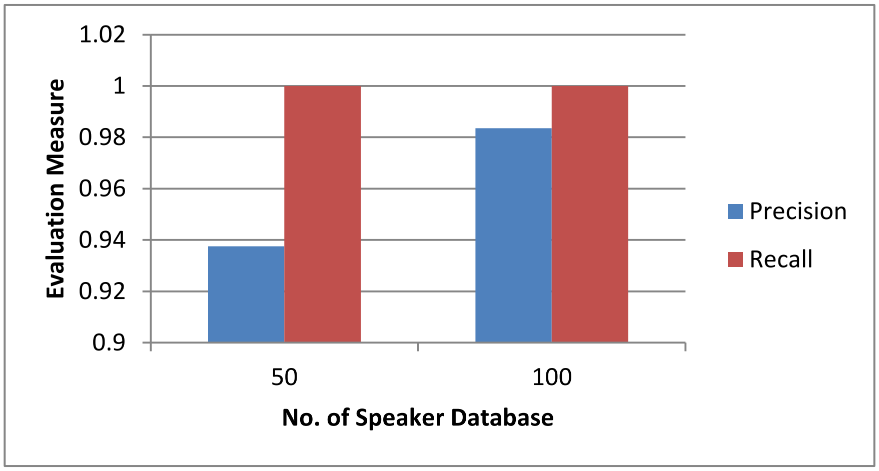


Figure 5.41 Precision and Recall for 50 and 100 speakers by Fusion

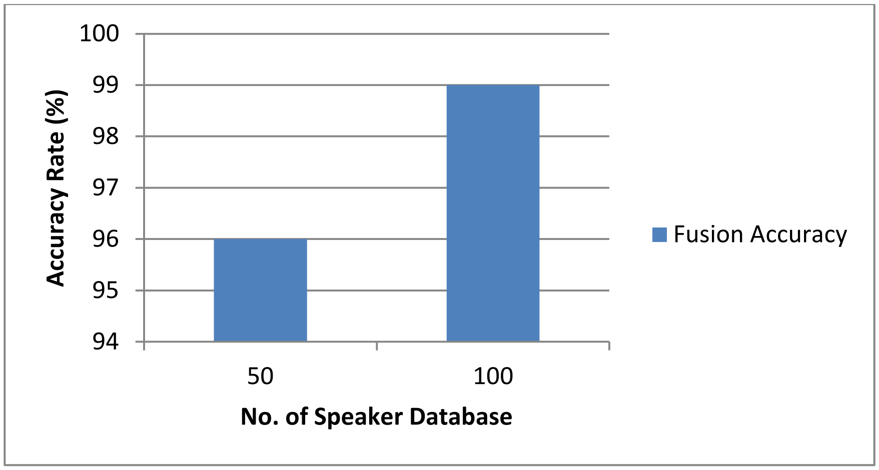


Figure 5.42 Accuracy Rate (%) for 50 and 100 speakers by Fusion

5.6 Accuracy Rate for Proposed SIS

From confusion matrix, we had calculated TP, TN, FP, and FN for VQ, GMM, ANN, and DT algorithms then for fusion method. We used these results for computing accuracy rate for 25, 50, and 100 speakers as shown before and illustrated in figures 5.43, 5.44, and 5.45 respectively. As shown clearly in figure 5.43 that fusion method achieved the best accuracy rate with 25 speakers averaged from six groups, about 96%, then VQ and ANN both have about 95%, then GMM with about 94%. The worst accuracy rate is achieved by DT with about 90%. In figure 5.44, fusion has the best accuracy rate with 96% for 50 speakers, where VQ, GMM, and DT all have 94% and 84% for ANN as the worst accuracy rate. In figure 5.45, for 100 speakers fusion also has the best accuracy rate but here also VQ achieved the same rate, which is 99%, then GMM with 97%, then ANN with 95%, and the worst result is DT with 92%.

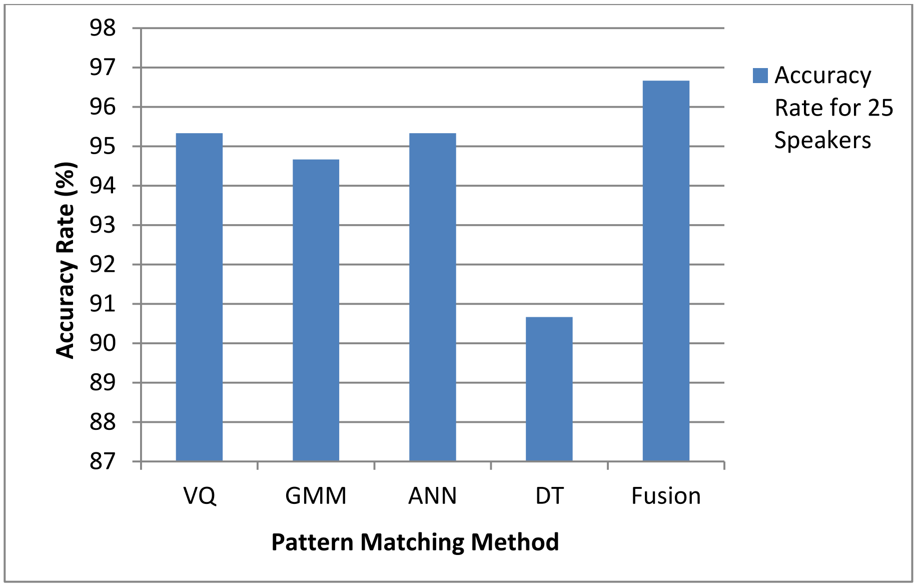


Figure 5.43 Accuracy Rate(%) for 25 Speakers by Proposed SIS

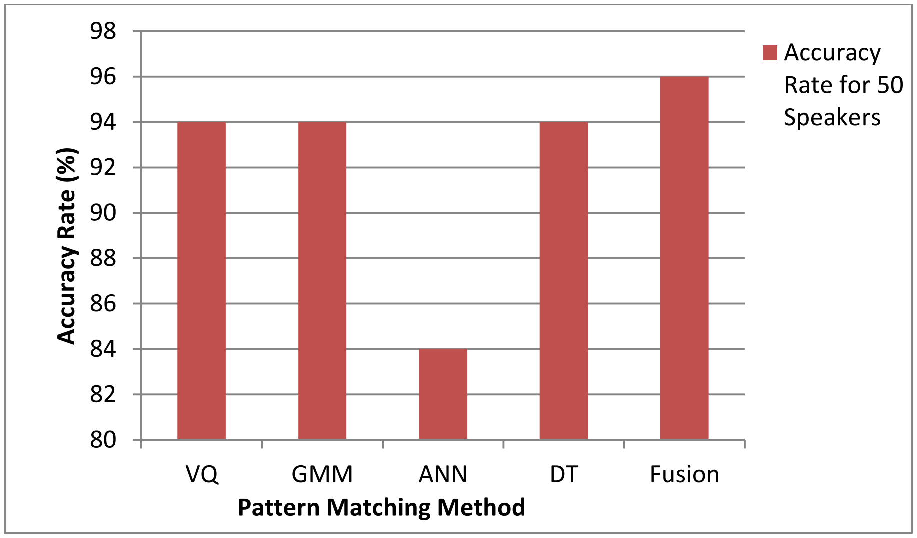


Figure 5.44 Accuracy Rate(%) for 50 Speakers by Proposed SIS

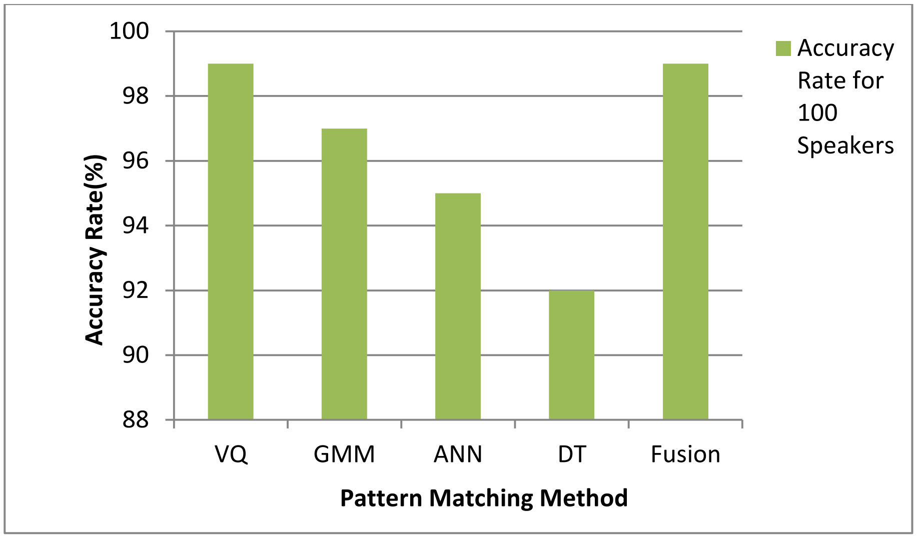


Figure 5.45 Accuracy Rate(%) for 100 Speakers by Proposed SIS

As a conclusion, fusion method that used in this study has the best accuracy rate in speaker identification when the speaker database is composed of 60% true speakers and 40% imposters. In 25, 50, and 100 speaker databases VQ and GMM give better results than ANN and DT. ANN gives the best accuracy rate result with 25 speakers. DT gives the best accuracy rate result with 50 speakers.

We have to notice that there is a difference between the identification rate results and accuracy rate results. As explained before, identification rate is focusing on number of correctly identified audio samples over the total number of tested audio samples. Accuracy rate is a general performance metric and is computed from confusion matrix which take the entire true positive and true negative speakers over the total number of speakers entered the system (true speaker and imposter).

Chapter 6

Conclusion and Future Work

Chapter VI

Conclusion and Future Work

6.1 Research Summary

Speaker Recognition discipline is a rich area for researches since the mid-1980s. From that time until now, rapid development and new technologies have emerged to contribute in different fields. Speaker identification technology in particular becomes a necessity in smart environments like distance learning, teleconferences, attendance systems, personalized dialog systems, dialing machines, police investigations, etc. In this study, we are proposing a SIS that uses different techniques to identify any speaker effectively. This system can be a lab assistant that help the lecturer or any user to identify who are participating in a class or a lab in a separated location.

The goal of this study is to propose a speaker identification system for an effective identification of all registered persons based on their speech. For extracting features from the speech signal MFCC’s was used. After that, VQ, GMM, ANN, and DT algorithms were applied for feature matching. Then, in order to take advantage of the various speaker identification algorithms techniques, the results were fused through majority decision method.

As explained previously, SIS is composed of feature extraction and feature matching modules. These two modules participate in the whole process of SIS, which is composed of training and testing phases. Training phase is responsible for building the speaker database by training all the speakers participating to the system. In our case, we have different organizations of speaker databases: the increasing size of speakers from 10 to 120 speakers, and six groups having the same size of 25 speakers chosen randomly. The testing phase represents the actual work for the system used for speaker identification.

We noticed that identification rate results are affected by increasing the size of speakers databases used. In addition, it is affected by the environment of recorded speech as shown from different identification results from groups A, B, C, D, E, and F of 25 speakers. Overall training and testing speakers in groups with small size achieve better identification rate and less misidentification of speakers.

With the speaker database comprised of 10 to 120 speakers, GMM gives the best identification rate results. Fusion method gives better results than VQ, ANN, and DT when they are applied separately. From 60 to 120 speaker database, DT algorithm provides the worst identification rate.

In the speaker database of groups A, B, C, D, E, and F with 25 speakers in each group, GMM gives the best identification rate results when applied to all groups. Fusion method gives better results than VQ, ANN, and DT when they are applied separately. By taking the average identification rate from all groups, GMM scores the best and DT scores the worst identification rate results.

The number of misidentification increases with the increasing size of speakers. This is evident when comparing misidentification number between 25 speaker groups and 40 to 120 speaker databases. ANN algorithm scores the highest misidentification number with 120 speakers.

Changing speakers groups of 25, 50, and 100 speaker databases with 40% of untrained speakers (imposters) is done for performance evaluation. In this study, we have computed TP, TN, FP, and FN to measure precision, recall, and accuracy for VQ, GMM, ANN, DT, and fusion. The optimal results of recall were obtained with VQ, GMM, and fusion method. The lowest recall value is computed from 50 speaker database with ANN algorithm which is recall=0.8333. We have obtained the optimal result of precision in group B only with all algorithms methods. The lowest precision value is computed from group C with DT algorithm which is precision=0.7778.

Accuracy rate results show that fusion method is the most accurate procedure for identifying true speakers and rejecting imposters. VQ algorithm is the second best in accuracy rate results.

6.2 Limitations

As shown from simulations, when we depend on a single speaker identification algorithm such as VQ or ANN or DT, we are facing lower identification rate results and more misidentification number of speakers than fusing the results. To compensate this issue we used four different speaker identification techniques to create un-correlations between results and get the most benefit from each algorithm by majority rule decision.

We suggest that the main reason for the decreasing identification results are the increasing number of similar features from different speakers. This is lead to make the identification process more difficult and identifies speakers mistakenly. One choice is to increase the training time for each speaker to get more unique features. Another way is to use more than one feature extraction technique to produce diverse features.

As we are aiming to build a real time system as a lab assistant, it is better to parallelize running the four speaker identification algorithms to produce real time results for fusion. This will be more efficient than applying each algorithm separately. However, in this study we did not use the parallelism method. This can be done in future work.

6.3 Conclusion

The identification of good approaches for the improvement of speech recognition system is a difficult task, the results obtained showing that by adding majority decision to fuse VQ, GMM, ANN, and DT algorithms the accuracy rate level is improved up to 96% for 25 and 50 speakers and up to 99% for 100 speakers.

Identification rate results show that, compared to the other approaches, GMM has the best performance. Also dividing a large number of registered speakers into groups for training gives more stable and higher identification results. Based on the data obtained, the proposed SIS can be used to facilitate lab assistant that is specialized in speaker identification.

6.4 Recommendations for Future Work

As this study relied on an English language speaker database, the performance of the proposed system can be tested on another language such as Arabic. In this phase of work, the SIS goal was to identify speakers by their identification number. We can add the gender, age, emotional state and other factors for identification. Speaker identification is a part of pattern recognition so we can upgrade the performance by adding another biometric for identification such as face, or fingerprint.

Also using another feature extraction method such as Linear Predictive Coding (LPC), Inverted Mel Frequency Cepstrum Coefficients (IMFCCs), etc., can produce different results.

Fusion method can be done depending on various theories and can be applied to two or more feature extraction techniques or pattern matching techniques. Speaker corpora can be changed or add some factors as clean and noise environment. In addition, another improvement can be represented by the identification of a speaker when multiple speakers speak simultaneously. Parallelism technique can be applied on several pattern matching models to upgrade the performance in real time systems.

LIST OF REFERENCES

[1] D. A. Reynolds, "An overview of automatic speaker recognition," in *Proceedings of the International Conference on Acoustics, Speech and Signal Processing (ICASSP)(S. 4072-4075)*, 2002.

[2] Z. Haider, "Robust speaker identification against computer aided voice impersonation," U580437 Ph.D., University of Surrey (United Kingdom), Ann Arbor, 2011.

[3] L. P. Cordella, P. Foggia, C. Sansone, and M. Vento, "A real-time text-independent speaker identification system," in *Image Analysis and Processing, 2003. Proceedings. 12th International Conference on*, 2003, pp. 632-637.

[4] Q. Jin, "Robust speaker recognition," Carnegie Mellon University, 2007.

[5] A. Rajsekhar. G, "Real time speaker recognition using MFCC and VQ " Master of Technology In Telematics and Signal Processing Department of Electronics & Communication Engineering, National Institute of Technology, Rourkela, 2008.

[6] J. P. Campbell Jr, "Speaker recognition: a tutorial," *Proceedings of the IEEE,* vol. 85, pp. 1437-1462, 1997.

[7] L. Feng, "Speaker recognition," Technical University of Denmark, DTU, DK-2800 Kgs. Lyngby, Denmark, 2004.

[8] D. A. Reynolds and L. P. Heck, "Automatic speaker recognition," in *AAAS 2000 Meeting, Humans, Computers and Speech Symposium*, 2000, pp. 101-104.

[9] D. S. Rodríguez, "Text-Independent Speaker Identification," Master Science Thesis, Faculty of Electrical Engineering, Automatics, Computer Scienceand Electronics, AGH UNIVERSITY OF SCIENCE AND TECHNOLOGY KRAKOW, Kraków, 2008.

[10] A. M. Kondoz, *Digital Speech: Coding for Low Bit Rate Communication Systems*, 2nd, edition ed.: John Wiley & Sons Ltd, 2004.

[11] T. F. Quatieri, *Discrete-time Speech Signal Processing: Principles and Practice*: Prentice Hall PTR, 2002.

[12] K. Fukunaga, *Introduction to statistical pattern recognition*: Academic press, 2013.

[13] D. A. Reynolds, "An overview of automatic speaker recognition technology," in *Acoustics, Speech, and Signal Processing (ICASSP), 2002 IEEE International Conference on*, 2002, pp. IV-4072-IV-4075.

[14] B. Squires and C. Sammut, "Automatic speaker recognition: An application of machine learning," in *ICML*, 1995, pp. 515-521.

[15] H. Beigi, *Fundamentals of Speaker Recognition*, 1 ed.: Springer US, 2011.

[16] S. Furui, "Recent advances in speaker recognition," in *Audio-and Video-based Biometric Person Authentication*, 1997, pp. 235-252.

[17] L. G. Kersta, "Voiceprint Identification," *Nature,* vol. 196, pp. 1253-1257, 12/29/print 1962.

[18] I. Pollack, J. M. Pickett, and W. H. Sumby, "On the Identification of Speakers by Voice," *The Journal of the Acoustical Society of America,* vol. 26, pp. 403-406, 1954.

[19] J. Shearme and J. Holmes, "An experiment concerning the recognition of voices," *Language and Speech,* vol. 2, pp. 123-131, 1959.

[20] H. Beigi, *Speaker Recognition: Advancements and Challenges*, 2012.

[21] S. Pruzansky, "Pattern-Matching Procedure for Automatic Talker Recognition," *The Journal of the Acoustical Society of America,* vol. 35, pp. 354-358, 1963.

[22] S. Pruzansky and M. V. Mathews, "Talker-Recognition Procedure Based on Analysis of Variance," *The Journal of the Acoustical Society of America,* vol. 36, pp. 2041-2047, 1964.

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

[24] L. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition," *Proceedings of the IEEE,* vol. 77, pp. 257-286, 1989.

[25] F. Soong, A. Rosenberg, L. Rabiner, and B. H. Juang, "A vector quantization approach to speaker recognition," in *Acoustics, Speech, and Signal Processing, IEEE International Conference on ICASSP '85.*, 1985, pp. 387-390.

[26] H. Fenglei and W. Bingxi, "Text-independent speaker recognition using support vector machine," in *Info-tech and Info-net, 2001. Proceedings. ICII 2001-Beijing. 2001 International Conferences on*, 2001, pp. 402-407.

[27] H. Hattori, "Text-independent speaker recognition using neural networks," *IEICE TRANSACTIONS on Information and Systems,* vol. 76, pp. 345-351, 1993.

[28] L. Rokach and O. Maimon, "Decision trees," in *Data Mining and Knowledge Discovery Handbook*, ed: Springer, 2005, pp. 165-192.

[29] T. Mahboob, M. Khanum, M. S. H. Khiyal, and R. Bibi, "Speaker Identification Using GMM with MFCC," *International Journal of Computer Science Issues (IJCSI)*, vol. 12, p. 126, 2015.

[30] D. H. B. Kekre and V. Kulkarni, "Performance Comparison of Speaker Recognition using Vector Quantization by LBG and KFCG," *International Journal of Computer Applications*, vol. 3, p. 32, 2010.

[31] A. Abushariah, T. Gunawan, J. Chebil, and M. Abushariah, "Voice based automatic person identification system using Vector Quantization," in *Computer and Communication Engineering (ICCCE), 2012 International Conference on*, 2012, pp. 549-554.

[32] M. R. Hasan, M. Jamil, M. Rabbani, and M. Rahman, "Speaker identification using Mel frequency cepstral coefficients," *variations*, vol. 1, p. 4, 2004.

[33] O. P. Prabhakar and N. K. Sahu, "Speaker Identification system using Mel Frequency Cepstral Coefficient and GMM technique," *IOSR Journal of Electrical and Electronics Engineering (IOSR-JEEE)*, pp. 51-56, 2014.

[34] M. K. Gill, R. Kaur, and J. Kaur, "Vector quantization based speaker identification," *International Journal of Computer Applications*, vol. 4, pp. 1-4, 2010.

[35] J. W. Picone, "Signal modeling techniques in speech recognition," *Proceedings of the IEEE*, vol. 81, pp. 1215-1247, 1993.

[36] A. Zulfiqar, A. Muhammad, and A. M. Enriquez, "A speaker identification system using MFCC features with VQ technique," in *Intelligent Information Technology Application, 2009. IITA 2009. Third International Symposium on*, 2009, pp. 115-118.

[37] F. Zhuo, S. Choi San, C. Chen, Y. He, and Z. Wang, "Use Hamming window for detection the harmonic current based on instantaneous reactive power theory," in *Power Electronics and Motion Control Conference, 2004. IPEMC 2004. The 4th International*, 2004, pp. 456-461 Vol.2.

[38] R. Mukherjee, "Speaker Recognition Using Shifted MFCC," UNIVERSITY OF SOUTH FLORIDA, 2012.

[39] C. Becchetti and L. P. Ricotti, *Speech recognition: theory and C++ implementation*: John Wiley \& Sons, Inc., 1999.

[40] S. J. Abdallah, I. M. Osman, and M. E. Mustafa, "Text-independent speaker identification using hidden Markov model," *World Comput. Sci. Inf. Technol. J*, vol. 2, pp. 1-6, 2012.

[41] L. R. Rabiner and R. W. Schafer, *Theory and Applications of Digital Speech Processing*: Pearson, 2011.

[42] H. Gish and M. Schmidt, "Text-independent speaker identification," *Signal Processing Magazine, IEEE,* vol. 11, pp. 18-32, 1994.

[43] S. Singh and E. Rajan, "MFCC VQ based Speaker Recognition and its Accuracy Affecting Factors," *International Journal of Computer Applications,* vol. 21, pp. 1-6, 2011.

[44] Y. Linde, A. Buzo, and R. M. Gray, "An algorithm for vector quantizer design," *Communications, IEEE Transactions on,* vol. 28, pp. 84-95, 1980.

[45] L. R. Rabiner and B.-H. Juang, *Fundamentals of speech recognition* vol. 14: PTR Prentice Hall Englewood Cliffs, 1993.

[46] M. G. Sumithra and A. K. Devika, "Performance Analysis of Speaker Identification System Using GMM with VQ " *International Journal of Computer Network and Security(IJCNS),* vol. 4, 2012.

[47] E. Simancas-Acevedo, A. Kurematsu, M. Nakano Miyatake, and H. Perez-Meana, "Speaker Recognition Using Gaussian Mixtures Models," in *Bio-Inspired Applications of Connectionism*. vol. 2085, J. Mira and A. Prieto, Eds., ed: Springer Berlin Heidelberg, 2001, pp. 287-294.

[48] R. S. S. Kumari and S. S. Nidhyananthan, "Fused Mel Feature sets based Text-Independent Speaker Identification using Gaussian Mixture Model," *Procedia Engineering,* vol. 30, pp. 319-326, 2012.

[49] P. Sibi, S. A. Jones, and P. Siddarth, "Analysis of different activation functions using back propagation neural networks," *Journal of Theoretical and Applied Information Technology,* vol. 47, pp. 1264-1268, 2013.

[50] V. Srinivas, C. Santhi rani, and T. Madhu, "Neural Network based Classification for Speaker Identification," *International Journal of Signal Processing, Image Processing and Pattern Recognition,* vol. 7, pp. 109-120, 2014.

[51] M. Hossain, B. Ahmed, and M. Asrafi, "A real time speaker identification using artificial neural network," in *Computer and information technology, 2007. iccit 2007. 10th international conference on*, 2007, pp. 1-5.

[52] N. Khaled and S. N. Al Saad, "Neural Network Based Speaker Identification System Using Features Selection," *Journal of Convergence Information Technology,* vol. 9, p. 9, 2014.

[53] S. Pinjare and A. Kumar, "Implementation of neural network back propagation training algorithm on FPGA," *International Journal of Computer Applications,* vol. 52, pp. 1-7, 2012.

[54] K. R. Farrell, R. J. Mammone, and K. T. Assaleh, "Speaker recognition using neural networks and conventional classifiers," *Speech and Audio Processing, IEEE Transactions on,* vol. 2, pp. 194-205, 1994.

[55] L. Breiman, J. Friedman, C. J. Stone, and R. A. Olshen, *Classification and Regression Trees*: Taylor & Francis, 1984.

[56] T. Roman, "Classification and Regression Trees(CART)Theory and Applications," Master of Art, CASE - Center of Applied Statistics and Economics, Humboldt University, Berlin, 2004.

[57] R. Blouet and F. Bimbot, "A tree-based approach for score computation in speaker verification," in *2001: A Speaker Odyssey-The Speaker Recognition Workshop*, 2001.

[58] G. Gonon, F. Bimbot, and R. Gribonval, "Probabilistic scoring using decision trees for fast and scalable speaker recognition," *Speech Communication*, vol. 51, pp. 1065-1081, 2009.

[59] T. Fawcett, "An introduction to ROC analysis," *Pattern recognition letters*, vol. 27, pp. 861-874, 2006.

[60] V. Srinivas and T. Madhu, "Neural Network based Classification for Speaker Identification," *International Journal of Signal Processing, Image Processing and Pattern Recognition*, vol. 7, pp. 109-120, 2014.

[61] R. P. Ramachandran, K. R. Farrell, R. Ramachandran, and R. J. Mammone, "Speaker recognition—general classifier approaches and data fusion methods," *Pattern Recognition*, vol. 35, pp. 2801-2821, 2002.

[62] (2004). *ELSDSR English language speech database for speaker recognition*. Available: http://www2.imm.dtu.dk/~lfen/elsdsr/index.php?page=index

[63] (2006-2015). *VoxForge Speech Corpus*. Available: http://www.voxforge.org/home/downloads

[64] D. R. McCloy, P. E. Souza, R. A. Wright, J. Haywood, N. Gehani, and S. Rudolph. (2013). *The PN/NC corpus*. Available: http://depts.washington.edu/phonlab/resources/pnnc/

[65] N. Praveen and T. Thomas, "Text Dependent Speaker Recognition using MFCC features and BPANN," *International Journal of Computer Applications*, vol. 74, pp. 31-39, 2013.

[66] L. I. Kuncheva, *Combining Pattern Classifiers: Methods and Algorithms*: Wiley, 2004.

[67] H. A. B. Patil, T. K., "A New Data Fusion Technique and Performance Measure for Identification of Twins in Marathi," presented at the International Symposium on Chinese Spoken Language Processing (ISCSLP), Kent Ridge, Singapore, 2006.

[68] L. Mary, K. S. R. Murty, S. M. Prasanna, and B. Yegnanarayana, "Features for speaker and language identification," in *ODYSSEY04-The Speaker and Language Recognition Workshop*, 2004.

[69] K. Chen, L. Wang, and H. Chi, "Methods of combining multiple classifiers with different features and their applications to text-independent speaker identification," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 11, pp. 417-445, 1997.

[70] S. Z. Boujelbene, D. Ben Ayed Mezghani, and N. Ellouze, "Application of combining classifiers for text-independent speaker identification," in *Electronics, Circuits, and Systems, 2009. ICECS 2009. 16th IEEE International Conference on*, 2009, pp. 723-726.

[71] J. A. Benediktsson and P. H. Swain, "Consensus theoretic classification methods," *Systems, Man and Cybernetics, IEEE Transactions on*, vol. 22, pp. 688-704, 1992.

[72] L. Xu, A. Krzyzak, and C. Y. Suen, "Methods of combining multiple classifiers and their applications to handwriting recognition," *Systems, Man and Cybernetics, IEEE Transactions on*, vol. 22, pp. 418-435, 1992.

خوارزمية تحديد المتحدث التي تستخدم للمساعدة في المعامل والتعليم عن بعد

بلسم زكريا إسحاق خوجة

المستخلص

إن الحاجة المتزايدة لاتصال الإنسان بالآلة في حياتنا اليومية أدى إلى تنمية وتطور متسارع . المقياس الحيوي للصوت يعتبر تقنية غنية بالمعلومات، فكما نعلم أن المستمع إلى صوت المتحدث لا يفهم الكلمات فحسب بل قد يستطيع أن يتعرف على جنس المتحدث، وحالته الصحية، ووضعه الانفعالي، كما أنه قد يتمكن من معرفة هوية المتحدث إذا سبق له معرفته. إن مجال معالجة الكلام يهتم بفهم هذه الأنواع من المعلومات من خلال صوت المتحدث وذلك حسب الغرض من التطبيق المستخدم. التعرف الآلي للمتحدث (ASR) على وجه الخصوص هو عبارة عن قدرة البرنامج أو الآلة التعرف على هوية المتحدث من خلال صوته عندما يتحدث، وينقسم ال ASR عادةً إلى قسمين رئيسيين هما: التحقق من المتحدث (SV) والتعرف على المتحدث (SI). تطبيقات الSV تتعلق عادة بالمجال الأمني لأن وظيفتها التحقق من هوية المدعي ليتم قبوله أو رفضه من النظام. أم تطبيقات الSI تعتمد على عملية التعرف على المتحدث عن طريق مقارنته بجميع الأشخاص المسجلين مسبقاً في النظام، لذلك نجد هذا النوع في المؤتمرات الصوتية وما يشابهها من تطبيقات. هذه الدراسة تقترح نظام للتعرف على المتحدث (SIS) هدفه التعرف بكفاءة على جميع المتحدثين المسجلين لدى النظام من خلال أصواتهم عند تحدثهم. SIS يتألف من مكونين أساسيين هما: استخراج الخصائص من الإشارات الصوتية ومطابقة تلك الخصائص. في هذه الدراسة MFCCs تستخدم لغرض استخراج الخصائص الصوتية المميزة للمتحدث، أما لمطابقة هذه الخصائص فإننا نستخدم أربعة من أكثر الخوارزميات فيASR شيوعاً واستخداماً وهي:VQ وGMM وANN وDT. أما بالنسبة لقاعدة بيانات المتحدثين المستخدمة في هذه الدراسة تتألف من 120 متحدث. العملية المتبعة في نظام SIS المقترح هي تدريب ثم اختبار مجموعات مختلفة العدد من المتحدثين وذلك باستخراج خصائصهم الصوتية المميزة بواسطة MFCCs، بعد ذلك ندخل مرحلة التدريب بواسطة الخوارزميات الأربعة المذكورة آنفاً. أما مرحلة الاختبار فهي تختبر التعرف على هوية المتحدث بواسطة هذه الخوارزميات. هذا النظام المقترح يهدف إلى رفع معدل التعرف على المتحدثين عن طريق دمج نتائج الخوارزميات بواسطة طريقة قرار الأغلبية (majority decision method). أثبتت النتائج أن طريقة الfusion تعطي نتائج أفضل في التعرف على المتحدث من الخوارزميات VQ وANN وDT كلاً على حدة، لكننا وجدنا أن الخوارزمية GMM تعطي أفضل النتائج مع أقل عدد من المتحدثين الغير متعرف عليهم. عندما طبقنا مقاييس الأداء على الطرق الأربعة مع طريقة الfusion وجدنا أن الfusion سجل أعلى معدل دقة في التعرف على المتحدثين المسجلين مسبقاً في نظام SIS واستبعاد المتحدثين الذين لا ينتمون للنظام. نتائج طريقة الfusion كانت بمعدل دقة 99% ل 100 متحدث و 96% ل 25 و 50 متحدث على التوالي. النظام المقترح SIS في هذه الدراسة أثبت قدرته على التعرف بكفاءة على مجموعة مغلقة من المتحدثين والذين يتحدثون بعبارات غير معرفة مسبقاً لدى النظام.