東海大學資訊工程研究所

碩士論文

指導教授：呂芳懌 博士

一個基於MFCC的語者識別系統

An MFCC-based Speaker Identification System

研究生：林冠良

中華民國 一〇六 年 一 月 二〇 日

東海大學碩士學位論文考試審定書

東海大學資訊工程學系 研究所

研究生 林 冠 良 所提之論文

一個基於MFCC的語者識別系統

經本委員會審查，符合碩士學位論文標準。

學位考試委員會
召 集 人 陳金鈴 簽章

委 員 余心淳

刘榮春

指 導 教 授 簽章

中華民國 106 年 1 月 5 日

中文摘要

現今的環境中，語音辨識已經有許多生活上的實際應用，諸如 iphone 的語音助理 SIRI、Google 的語音輸入辨識系統與語音的手機操作等，語者辨識則相對地還未成熟。因此，本研究著重在語者辨識的研究，方法是將某人，例如，小明，的原始語音訊號，經傅立葉轉換，而從時域轉換到頻域上。其次，經由人耳聽覺模型過濾及分解出各個頻率的能量大小，並轉換成該語音的特徵數據。而後再利用高斯混合模型的機率密度函數，來概括描述這一些特徵數據的分布，進而成為小明的聲學模型。當系統接收到某一未知人物之語音數據時，亦以相同方法處理，並與資料庫中所蒐集之人物(包括小明)的聲學模型進行語音相似度比對，以辨識該未知人物可能是誰。

關鍵詞:

語者辨識、傅立葉轉換、梅爾頻率倒譜係數、高斯混合模型、聲學模型

Abstract

Nowadays, speech recognition has many practical applications which are currently used by people in the world. Typical examples are the SIRI of iPhone, Google speech recognition system, and mobile phones operated by voice, etc. On the contrary, speaker identification in its current stage is relatively immature. Therefore, in this paper, we study a speaker identification technique which first takes the original voice signals of a person, e.g., Bob. After that, the voice signals is converted from time domain to frequency domain by employing the Fourier transformation approach. A MFCC-based human auditory filtering model is then utilized to adjust the energy levels of different frequencies as the quantified characteristics of Bob's voice. Next, the energies are normalized to the scales of logarithm as the feature of the voice signals. Further, the probability density function of Gaussian mixture model is employed to represent the distribution of the logarithmic characteristics as Bob's specific acoustic model. When receiving an unknown person, e.g., x's voice, the system processes the voice with the same procedure, and compares the processing result, which is x's acoustic model, with known-people's acoustic models collected in an acoustic-model database beforehand to identify who the most possible speaker is.

Keywords: speaker identification, Fourier transformation, Mel-frequency cepstral coefficients, Gaussian mixture model, acoustic model

致謝

此篇論文的完成，首先要感謝呂芳懌教授，在我碩士生涯這幾年的以來的指導與提攜，除了在研究期間不斷地指引論文方向，在我遭遇瓶頸的時候給予我最大的鼓勵與支持，最後的最後也仍然鼓勵我不要放棄。也感謝我的家人在這段期間的支持,讓我能在沒有經濟壓力的環境下,專心的進行論文撰寫及研究。最後，感謝這些年來,陪我度過這段緊張刺激的研究生生活的朋友們,張弘毅、林奕學、張家豪、吳天富、莉姐。在東海念書期間，要感謝的人事物還有很多很多，沒有寫在這裡的，也都在此再次感謝。

List of Contents

List of Figures

List of Tables

1. Introduction

In this information era, many high-tech products gradually enter our everyday lives, and significantly change our living habits and patterns. The biometrics identification technology which provides us with easier and more convenient methods to identify specific people has gradually replaced some existing authentication techniques, that need to be learned before people can operate them properly. The face recognition systems used at airport halls [1] and the voice assistant SIRI of iPhone [2] are two examples of the biometric identification systems.

On the one hand, voice has been the most direct method for us to express ideas, communicate with others and do something for interaction. People invented telephones, which started from home phones, then evolving to the next generation, called functional phone, and at last the current smart phone. No matter how their functions and shapes are changed, the fact that people use voice to deliver information and communicate with others has not been changed. In fact, voice is the easiest and most convenient way for people to transmit their messages. Therefore, identifying people’s identities from user’s dialogue voice and dialogue contents, and then providing the corresponding services should be a better method to practically improve and convene our everyday lives. Up to present, voice recognition technology has been well developed, and the speech recognition technology [3] is relatively matured and has been applied to our living activities. But speaker identification technology [4] is still far away from its practical applications. The reasons are that 1) there are too many parameters needed to be processed for speaker identification; 2) it is hard to collect voice features completely; 3) the identification process is complicated and takes a long time for calculation. Thus, it is difficult to be applied to those applications which need immediate response. Furthermore, the studies of speaker identification nowadays are partial, rather than a whole. For example, Hidden Markov Model Toolkit (HTK) [5], Kaldi Speech Recognition Toolkit [6], and so on, individually focus on different portions of speech recognition. HTK is developed for statement recognition without having acoustic model matching function, and Kaldi is used to recognize speech. lacking feature extraction. Therefore, in this study, we implement a practical system, which integrates several existing partial techniques and subsystems and improve their interfaces/functions to make them as a whole so as to practically bring more convenience to people's lives.

The rest of this paper is organized as follows. Sections 2 and 3 introduce related work and background of this study, respectively. Section 4 presents our system architecture, System implementation and evaluation are described and discussed in Section 5. Section 6 concludes this paper and outlines our future studies.

2. Related Work

Speech is one of human's most convenient biological communication tools due to its features of universe, convenience and uniqueness. In recent years, people start using biometrics to validate one's identity, e.g., using voice to recognize who the speaker is. This issue is called speaker identification. Compared with other biometrics technologies, the advantages of speaker recognition are as follows.

(1) Voice is easy to acquire, and users are also relatively easier to accept and use voice to identify a person when the person's face cannot be seen currently, only voice.

(2) Voice access costs are low, and voice is easily available and able to be simply used.

2.1 Voice Recognition System

Voice recognition technology can be roughly divided into two sub-areas: speech recognition and speaker recognition [7]. The former is to analyze the content of the words/speech spoken by a speaker, whereas the latter is to identify who the speaker is. Some applications of speech recognition can be found in the market. But as mentioned above the speaker recognition is far away from mature. This study focuses on the latter, which can be roughly divided into two parts: Speaker Identification and Speaker Verification [8]. The speaker verification is a process used to determine the probability that a speaker, e.g. u, is really the speaker x_i in a set of known speakers $S = \{x_1, x_2, x_3 \dots x_n\}$ as a verification. On the other hand, when receiving voice signals of a speaker u, a speaker recognition system will find out the most likely and possible speakers from S by comparing the similarities between u and each $x_i,\ x_i \in S, 1 \leq i \leq n$.

The speaker identification systems can be divided into two types, including text-dependent [9] and text-independent [10], according to the words pronounced or speech given, i.e., the context of the voice. The type of text-dependent is the case in which the context is fixed to specified words or a specific speech/article. All speakers read the same words or sentences, then the identification system records the voice signals, and extract their features with which to perform its identification. The design of such a system is relatively simple. The type of text-independent is the case in which a speaker can say anything that he/she likes, without any limitation. This type of system extracts speaker's pronunciation features for modeling and compares the similarity between u and $x_i,\ x_i \in S$, so as to identify who the speaker is. Because the scope of words/sentences involved is wider, the design of such a system is relatively complex, and the implementation is difficult. But the system flexibility is high. It is useful to the real world, and the space of its future development is wide. In fact, this type of system is more helpful to people in our everyday activities than the type of text-dependent systems.

2.2 The Environmental Noise

Voice recognition has been applied to a variety of domains [11][12][13]. But voice recognition systems are very susceptible to noise, often resulting in poor recognition rate. In the real world, different environments will generate different types of noise of different features. Generally, the sounds collected, no matter whether outdoors or indoors, usually have a certain degree of environmental noise. The problem is that when the same noise Y appears in both the training phase and test phase, the noise will be a part of the acoustic models of x_i and u. Consequently, even λ_{x_i} and λ_u after removing Y are quite different, λ_{x_i} and λ_u with Y will be something similar. On the other hand, if the environmental noise individually in λ_{x_i} and λ_u are the different, even x_i and u are the same person, something different will be found between λ_{x_i} and λ_u. Both cages will affect the comparison result of similarity. Researchers are considering how to reduce the effects due to environmental noise, i.e., how to increase the degree of anti-interference, so as to correctly recognize voice signals. This is also one of the key topics in the research of voice recognition. One of the methods is Spectral Subtraction (SS) [14], which superimpositions a small background sound as noises over the original voice signals. In the original voice signals, those voice components the same as those of the noises will be hiddened, so as to achieve the purpose of noise reduction. However, based on the unrecoverability of the voice signal superimposition, spectral subtraction may also destroy some spectral details in the original signal [15], leading to the loss of some useful information. In order to improve this deficiency, the Support Vector Machine(SVM) [16] classifies voice features into different classes, aiming to reduce the difference among voice features of the same class to improve recognition accuracy. But this method often requires a lot of training voice, and is not conducive to a timely response system.

2.3 Operational Efficiency

A voice recognition system installed in a mobile device has its market advantages compared to those voice recognition systems installed in a PC. If a voice recognition system can be one of the applications of a mobile device, smart phones and other 3C portable products, it can then process people's voice, and understand user's commands. With the system, some valuable operations, functions and algorithms can be implemented in them, to not only prevent these devices from unauthorized use, i.e., voice recognition can be employed as one of security tools, but also make these systems more convenient to be utilized and more human-oriented, i.e., they can make system more friendly, thus greatly enhancing these products' market competitiveness.

Generally, PC's computing power and capability exceeds mobile devices'. So the voice recognition algorithms run on a PC can be more complex than those on a mobile phone. When wanting to port these algorithms from a PC to a mobile device, we need to consider the processing speed and capability of the mobile device, its operational efficiency, the amount of resources it consumes and the amount of calculation required, which are the new challenges of the voice recognition system in the future.

3. Background of this study

Generally, sound is an analog signal which must be converted to a digital signal before it can be processed by a computer. Often, the processed voice data is very large. So from processing efficiency viewpoint, we must extract the most representative features from the data. This is so-called feature extraction.

The process of speaker identification consists of two phases, i.e., training and test. The training phase is to extract voice features of a speaker *u* by using a feature extraction technique and then establish an acoustic model for *u*. The test phase is to calculate the similarity between the acoustic model created for the voice of an unknown speaker *x* and the acoustic models of *u*, and then judge whether or not *x* is *u*.

3.1 Feature Extraction

Voice signals in time domain change very fast and sharply. But if we transform the voice signals from time domain to frequency domain, the corresponding spectrum can be clearly shown. The spectrum is the connotative characteristics of the voice signals. Also, the voice signals have a characteristic of short time stationary [17], meaning it is stable in a short time period without changing significantly. Therefore, we can also observe the instantaneous frequency [18] of the signals from the spectrum.

To extract features from voice signals, people often divide voice signals into units, each of which consists of continuous signals. A unit comprises signals in a very short time period, e.g., *T*. Often *T* is fixed. Generally, the signals in a unit is called a frame, from which we can extract the voice features. In this study, the feature extraction technique employed is Mel-Frequency Cepstral Coefficients (MFCC) [19]. The MFCC is designed based on the characteristics of human ears which have different acoustical sensitivities to the sounds of different frequencies. Mel scale, as shown in Figure 1, is a non-linear scale on frequencies following the sensitivities of human ear when hearing sounds. It is proposed by Stevens et al. [20] in 1937. As shown, the human ears are not sensitive on high-frequency sound, but are relatively sensitive on low frequency. The equation which converts frequency f to Mel scales $f_{mel}(f)$ is as follows [20].

$$f_{mel}(f) = 2595 \times log\left(1 + \frac{f}{700}\right) \qquad (1)$$

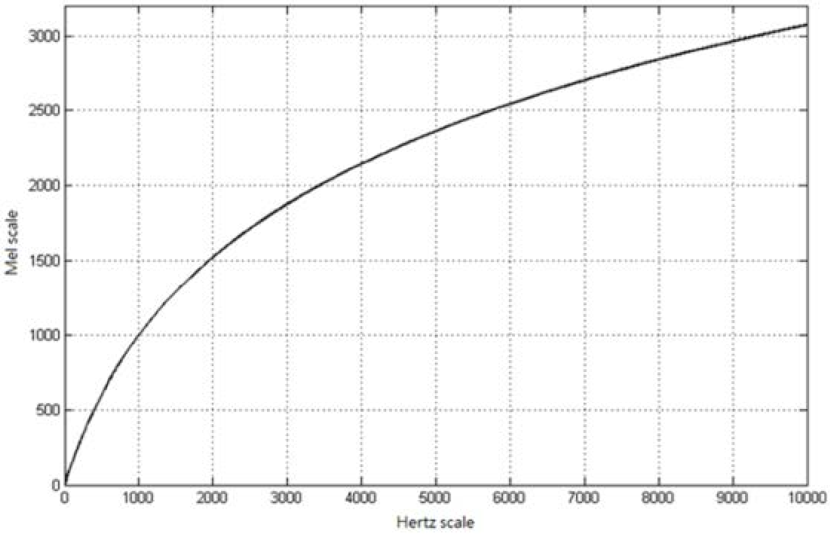

Figure 1. The Mel scale [20].

In this study, we design a set of triangular bandpass filters based on the Mel scales of different frequencies to filter input signals as the simulation of Human-ear experience when hearing sound comprising different frequencies. We first filter the energies of the frequencies in the processed signal spectrum with the triangular bandpass filters. The energies of high-frequency signals will be reduced largely due to less sensitive (i.e., attenuation is larger), whereas those of low-frequency signals will be reduced relatively lower. After that, the remaining energies of the signals will be converted and quantized into the scales of logarithm as the features of the voice signals.

3.2 Building Speaker Model

After the feature extraction, the voice signals in fact are converted to a large number of feature parameters. Then, we need to find a suitable statistical model to describe the distribution of these parameters. With this model, we can compare the similarities of the voice features among different persons.

In recent years, studies indicate that the energy distribution of human voice signals follows a Gaussian Model [21]. Therefore, this study chooses the Gaussian Mixture Model (GMM) [22] as the statistics model of the energy distribution when are text-independent speaker identification system is developed. In other words, GMM is utilized to build the feature model as the acoustic model of the speaker.

3.2.1 Gaussian Mixture Model

In the real world, many data distributions show the characteristic of the Gaussian model, also known as a normal distribution. This model has been widely used in identifying and/or classifying the distribution of something. The GMM is a combination of multiple Gaussian models, and each Gaussian model is given a weight to express the distribution among these Gaussian models. Figure 2 shows an example, in which there are a total of three Gaussian models.

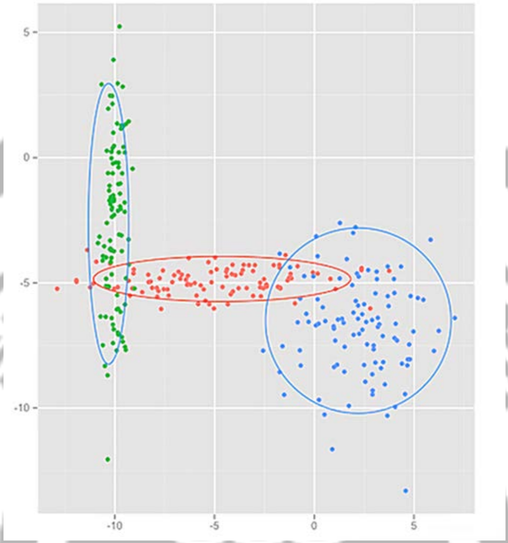

Figure 2. A GMM which consists of three Gaussian models [23].

A GMM formula can be expressed as follows.

$$p(\vec{x}|\lambda) = \sum_{i=1}^{M} w_i b_i(\vec{x}) \qquad (2)$$

in which M is the number of Gaussian models in the GMM, $\vec{x}$ is the feature vector of dimension D where D varies depending on the accuracy that a feature extraction method we would like to have, $b_i(\vec{x})$ is the $\vec{x}_i$'s Gaussian model, w_i is the weight of Gaussian model i , where $\sum_{i=1}^{M} w_i = 1$.

3.2.2 Training Phase

In order to identify the distribution probability of the voice features of a speaker, denoted by λ, we use the Maximum Likelihood Estimation (MLE) [24], which is the basic theory of Expectation Maximization (EM), to identify the most suitable parameters of the corresponding GMM, so that the conditional parameter-distribution probability of the GMM, denoted by P($X|\lambda$), is the most similar to that of the speaker's features. In the training phase, the MLE algorithm iteratively calculates the expected value of P($X|\lambda$) by using its previous features, i.e., the result of the last iteration, and initial parameters to find out the maximized λ. The detailed steps will be described in Chapter 4.

3.3 The Speaker Identification Method

When the system receives an unknown speaker, e.g., *x*'s voice signals, the identification system performs the aforementioned procedure to process the voice signals, and compares the similarity between *x*'s voice features and each of the registered users' GMMs collected in the acoustic-model database (maybe creating indexes to reduce the searching space of the objects). The purpose is to find out the registered user, e.g., *u*, whose voice features are the most similar to *x*'s. In the signal feature space, the similarity between *x* and *u* is reflected on the distance between *x*'s and *u*'s acoustic models, denoted by λ_x and λ_u, respectively. In other words, in an ideal case, if *x* and *u* are different individuals, the distance between λ_x and λ_u in the signal feature space (such as the aforementioned *D* dimensions) will be longer than the distance between the two acoustic models established for the same speaker, i.e., $|\lambda_x - \lambda_x'| \leq |\lambda_x - \lambda_u|$, where λ_x' is an acoustical model pre-established in the acoustic-model database for x. Theoretically, $|\lambda_x - \lambda_x{}'| = \min_{1\leq j\leq M}\{|\lambda_x - \lambda_j|\}$ where M is the number of trained users, i.e., the number of acoustic models collected in the acoustic-model database.

There are at least two distances that are the most widely used to express $|\lambda_x - \lambda_u|$, i.e., Euclidean distance and Bhattacharyya distance [25]. The Euclidean distance takes into account the average distance between the distributions of two arbitrary models, whereas the Bhattacharyya distance, which deals with the overall similarity of the two models, is popularly used to measure the similarity between two discrete probability distributions. However, its calculation is relatively complex and is not conducive to the real-time identification.

4. The System Architecture

Because a speaker identification system contains wide areas of techniques, it is not easy to integrate them together. The identification procedure generally comprises three steps. 1) Feature extraction; 2) Acoustic model establishment; 3) Acoustic model matching. However, the steps may be slightly changed depending on the environments in which the identification system works on. For example, if voice is transmitted through telephones or collected in a noisy place, the noise needs to be reduced first. Even this, basically they still follow the three main steps.

4.1 MFCC Process

The process flow of MFCC is shown in Figure 3. After signals are received, the system partitions the signals into frames, invokes a window function to increase the continuity of voice signals in a frame, utilizes the fast Fourier transform to convert the digital signals into spectrum data, and employs the Triangular bandpass filter designed in this study to simulate the spectral data process of our ears. Finally, the Discrete Cosine Transform (DCT) [26] is used to convert the spectral energy data into MFCC.

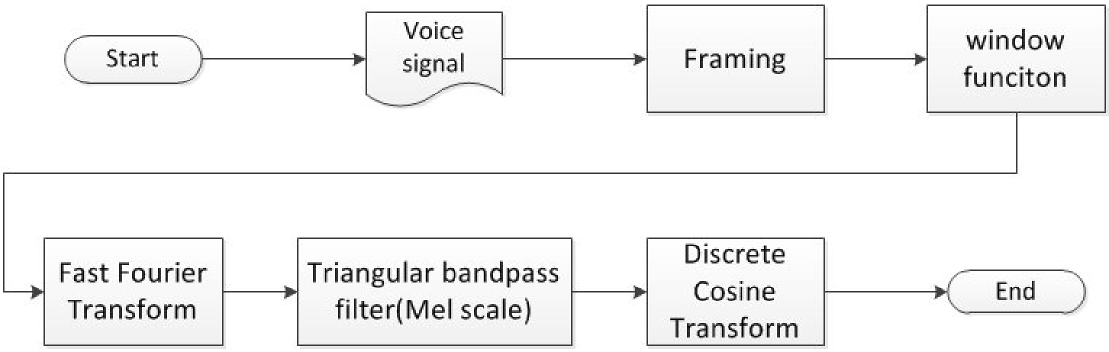

Figure 3. MFCC process

(1) Framing

Framing is a voice slicing method that divides a chosen voice file *F* info frames of fixed time period, e.g., *T*, since *F* is often long. But *T* is short, in general ranging between 20-30ms, in which the voice signals usually are regular and continuous, i.e., the main purpose of framing is to reduce discontinuity of the signals in a frame, because signal discontinuity may lead to extracting incorrect parameter values during analysis. In this study, *T* = 26ms. Also, to avoid discontinuity between two adjacent frames, every two consecutive frames as shown in Figure 4 are mutually overlapped 13ms. That is, the signals in the second half of frame *i* is also the signals of the first half of frame *i*+1, for all *i*s, $i = 1, 2, \ldots, n-1$ where *n* is the number of frames partitioned from *F*. In other words, the first half of frame 1 and the second half of frame *n* of *F* are processed only once by our scheme. The rest of the signals is processed twice.

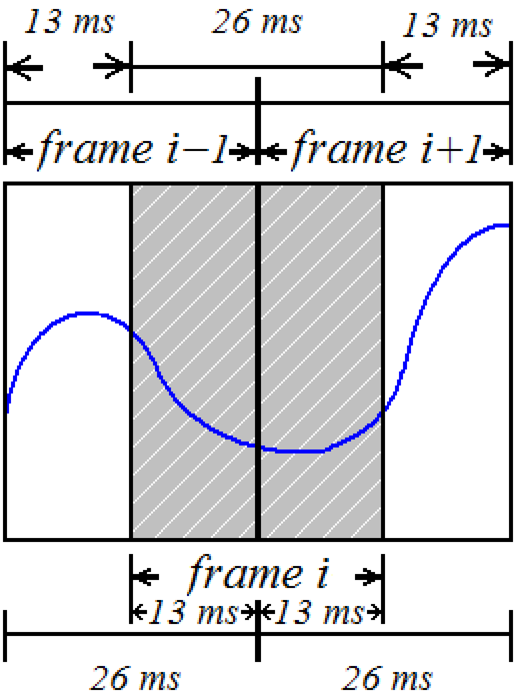

Figure 4. After Framing, each pair of consecutive frames will overlap 13ms.

(2) Window function

In this study, we use the Hamming window [27] to process a frame. The Hamming window is a window function able to change the phases of voice signals to a designated range to make the voice signals between two consecutive frames more continuous. Given a frame $S(n)$, $n = 0, 1, \ldots, N-1$, assume that $S'(n)$ is the processing result of $S(n)$ by invoking the Hamming window, where N is the total number of frames obtained by dividing a voice file, then [28]

$$S'(n) = S(n) * W(n, a) \qquad (3)$$

where $W(n, a) = (1 - a) - a \times cos\frac{2\pi n}{N-1}, 0 \le n \le N-1$, is the Hamming window function. From $W(n, a)$, we can see that different positions of the signal waveforms of $S(n)$ are modified differently, i.e., different degrees of amplitude reduction. The amplitude reduction in the head and tail areas of a frame, e.g., $S(n)$, will be greater than that in the middle area. The purpose is to increase the overall signal continuity between $S(n)$ and its direct neighbor frames. A larger a value will cause strong signal connectivity, meaning waveforms are sharply shaped. Of course, the smaller the value of a, the weaker the continuity of the signals in a frame, but more signal details are still retained.

(3) Triangular bandpass filter and Discrete cosine transform

This study uses a set of Q triangular bandpass filters to filter voice signals after the signals are transformed into frequency-domain signals. The purpose is to make the signals follow the attenuation characteristics of the Mel scale (see Figure 1). In Figure 5, the frequency band is between 0 and 8000 (Hz). A total of 10 triangular band-pass filters is given. The low-frequency part has a more dense band-pass filters, meaning after the filter, the attenuated energies are low since human ears are sensitive on low frequencies. The density of the high-frequency part is relatively lower.

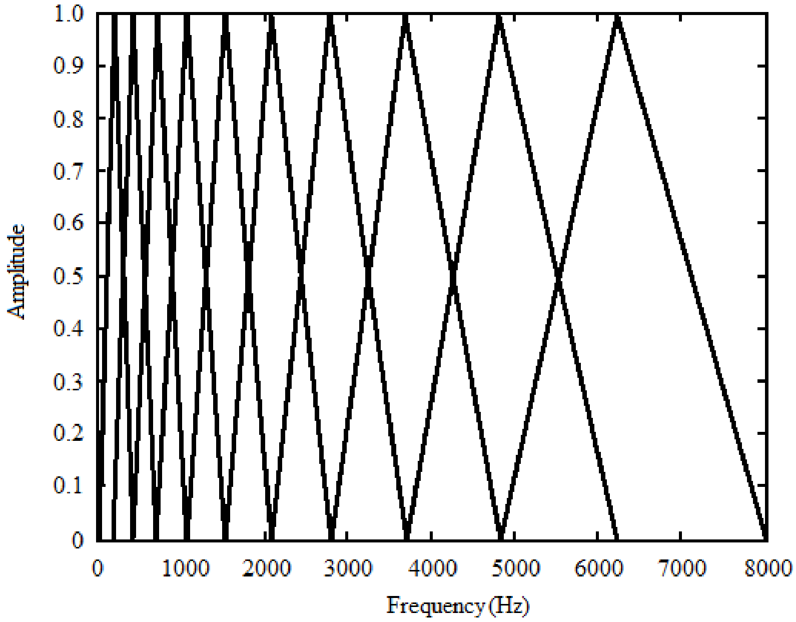

Figure 5. 10 triangular bandpass filters between 0 and 8000 (Hz) bands.

Use the Triangular bandpass filter, each filter shows its energy levels of frequency distribution with logarithm scales, and a total of Q logarithmic energies E_ks in a frame are transformed into C_ms by using the discrete cosine function [26].

$$C_m = \sum_{k=1}^{Q} E_k \times cos\left[m\left(k-\frac{1}{2}\right)\frac{\pi}{Q}\right], m = 1, \dots, L \qquad (4)$$

where E_k is the spectral energy value produced by the kth triangular filter in the previous step, and Q is the number of triangular filters. In general, Q=22 [26], even in Figures only shows 10 triangular filters. Discrete cosine transform (DCT) [26] is a transformation associated with Fourier transform. It is similar to discrete Fourier transform, but adopts only real numbers. In this study, L = 12, because a

12-dimensional feature parameter is sufficient to represent the voice feature of a frame [19].

4.2 Establishment of Gaussian Mixture Model

In this study, the time consumed to establish the corresponding Gaussian mixture model given training data in the training phase is much longer than that of the test phase. To speed up the process of acquiring the best parameters for Gaussian mixture model, it is necessary to estimate the initial parameters precisely. This can shorten the processing time for optimizing the parameters. In this study, we use the K-means clustering [7] to cluster the original feature vectors due to its fast and simple characteristics in clustering data. The detailed steps are as follows.

4.2.2 K-means clustering

(1) initialization

Given a voice file F which is partitioned into N frames, a total of N vectors, denoted by, $V_{ec} = \{y_1, y_2, \ldots, y_n\}$, will be obtained where y_i is a (L+1)-dimensional feature, since besides the L-dimensional feature, energy is also considered as a feature, $1 \leq i \leq n$. In this study, be clustered into K groups, e.g., $G = \{group(1), group(2), \ldots, group(K)\}, K \leq \text{n}$. The initial steps is randomly selecting *K* vectors, e.g., $C = \{v_1, v_2, \ldots, v_K\}$, from *Vec*, as the center vectors of *G*, where v_i is the center vector of $group(i), v_i \in C, group(i) \in G$. In this study, *K*=128. In other words, we cluster these voice vectors into 128 groups.

(2) Clustering

For each vector in the remaining vectors $R = v_{ec} - C = \{x_1, x_2, \dots, x_{n-128}\}$, e.g., x_i ，$1 \le i \le n-128$, we calculate the distances between x_i and the 128 current center vectors as $D_{x_i} = \left\{d_{x_{i,1}}, d_{x_{i,2}}, \dots, d_{x_{i,K}}\right\}$, in which $d_{x_{i,s}}$ is the distance between x_i and $v_s, 1 \le s \le K, v_s \in C$, and v_s is the center vector of $group(s)$. If $d_{x_{i,j}} = min_{1 \le s \le K}\left\{d_{x_{i,s}} | d_{x_{i,s}} \in D_{x_i},\ 1 \le s \le K\right\}$ and $x_i \notin group(j)$, then $group(j) = group(j) \cup \{x_i\}$. If originally $x_i \notin group(l)$ and $l \ne j, group(l) = group(l) - \{x_i\}$.

(3) Updating the cluster centers

For each vector group, e.g., $group(j) = \left\{x_{j1}, x_{j2}, \dots, x_{jp}\right\}$ ，$|group(j)| = p_j$, we calculate the accumulative distances between a vector, e.g., x_{ji}, and all other $p_j - 1$ elements in $group(j) - \left\{x_{ji}\right\}$, resulting in $TD_{x_{ji}} = \sum_{q=1, q \ne i}^{p} d_{x_{ji,jq}}$ where $d_{x_{ji,jq}}$ is the distance between x_{ji} and $x_{jq}, x_{ji}, x_{jq} \in group(j)$, for all is, $i = 1,2, \dots, p_j$. So, a total of p_j accumulative distances, denoted by $ATD_{x_j} = \left\{TD_{x_{j1}}, TD_{x_{j2}}, \dots, TD_{x_{jp_j}}\right\}$, can be obtained. If $TD_{x_{jm}} = min_{1 \le i \le p_j}\left\{TD_{x_{ji}}\right\}$, then x_{jm} is the new center vector of the $group(j)$, denoted by v'_j, for all js, $1 \le j \le K$. Let the K new center vectors be $\mathrm{C}' = \{v'_1, v'_2, \dots, v'_K\}$.

(4) Convergence

If no element of $group(j)$, e.g., x_{ji}, moves to $group(h), j \ne h, 1 \le i \le |group(j)|$, for all $j_s, 1 \le j \le K$, it means that the process of this algorithm is convergent, and its execution is then terminated. Otherwise, Let $v_i = v_i{}'$, for all $i_s, 1 \leq i \leq K$, i.e., $\mathrm{C} = \mathrm{C}'$ and go to Step 2.

4.2.3 EM algorithm [29]

Here, we compute a weight for each group and combine these 128 Gaussian models as a GMM. The vector distribution of these 128 groups is the initial distribution parameter distribution of the GMM. In order to obtain the best GMM parameter λ, that is, the distribution of the concerned voice features has the greatest similarity to the distribution of the model parameter λ. To achieve this, it is necessary to estimate the most suitable model parameter λ, the probability density of which can be expressed as follows [29].

$$\mathrm{P}(X|\lambda) = \prod_{i=1}^{K} P(x_i|\lambda) \qquad (5)$$

in which x_i is one of the 128 Gaussian models after clustering, and X is feature vectors of GMM which is deterministic. In order to find the model parameter λ' which can maximize the likelihood function value of the GMM, as mentioned above, we use the Estimation Maximization algorithm (EM) [29] to interactively find the Gaussian models of the GMM. In the first iteration, the EM algorithm re-estimates the new model parameter λ' by using the initialization parameter λ obtained by invoking the *K*-means clustering method so that $P(X|\lambda') \geq P(X|\lambda)$. Let $\lambda = \lambda'$, continue to iteratively update the new λ by invoking the EM algorithm until $P(X|\lambda)$ converges or reaches the upper limit of the number of iterations set by the system.

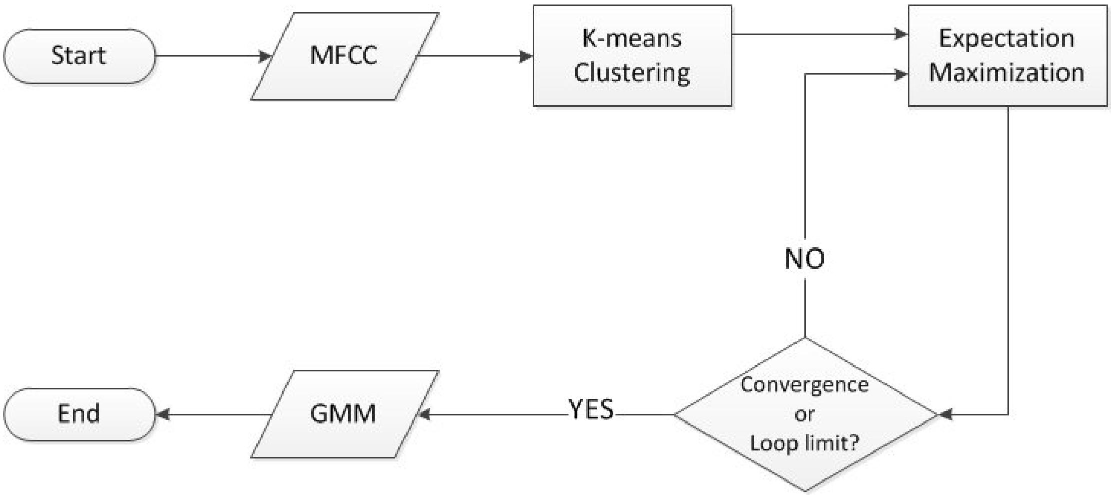

Figure 6. The establishment process of the GMM.

4.3 Bhattacharyya Distance [25]

So far, we have converted the voice signals into the GMM of the eigenvector distribution, i.e., λ, as an acoustic model of a person which is established as follows. Assuming a total of N user acoustic models has been collected, i.e., $U = \{\lambda_1, \lambda_1, \dots, \lambda_N\}$, in the acoustic-model database. The Bhattacharyya distance between two acoustic models λ_u and λ_i, denoted by $d_{BD}(\lambda_u, \lambda_i), \lambda_i \in U, i = 1, 2, \dots, N$ and λ_u is an unknown user's acoustic model. If $d_{BD}(\lambda_u, \lambda_n) = \min_{1 \le j \le N}\{d_{BD}(\lambda_u, \lambda_j)\}$, the probability that *u* is *n* will be the largest. But based on the conversion error and the impact of environmental noise, it is hard to ensure that *u* is exactly *n*. In this study, we take *m* acoustic models with the smallest $d_{BD}(\lambda_u, \lambda_r), r = 1, 2, \dots, m, m \le N$, and sort these $d_{BD}()$s in an ascending order where

$$d_{BD}(\lambda_u, \lambda_i) = \frac{1}{8}(\mu_u - \mu_i)^T \left(\frac{\Sigma_u + \Sigma_i}{2}\right)^{-1}(\mu_u - \mu_i) + \frac{1}{2} ln \frac{\left|\frac{1}{2}(\Sigma_u + \Sigma_i)\right|}{\sqrt{|\Sigma_u| + |\Sigma_i|}} \qquad (6)$$

in which $\mu_u(\mu_i)$ is the average vector of $\lambda_u(\lambda_i)$, and $\Sigma_u(\Sigma_i)$ is the covariance matrix of $\lambda_u(\lambda_i)$. We hope *u* will be one of the *m* users with the smallest $d_{BA}()$s. The similarity between λ_u and λ_i is defined as follows.

$$\text{Similarity (\%)} = \left(1 - \frac{X}{Y}\right) \times 100 \qquad (7)$$

in which X represents the distance between $\lambda_u \, and \, \lambda_i$ and Y is the maximum distance between arbitrary two acoustic models collected in the acoustic-model database. The larger the similarity, the higher the probability that the person *u* is i. Also, when the similarity between λ_u and λ_i is lower than the predefined threshold, e.g., TH_{low}, we conclude that the person of λ_i is not any one in *U*.

5. System Implementation and Evaluation

Many tools are currently available for speech feature extraction or speech modeling. But most of them focus on sentence recognition, rather than speaker identification. The famous HTK [5] as a statement recognition tool is developed based on fixed sentence-voice recognition. Kaldi [6] is a speech identification tool which was designed, based on C++ development tools, to build GMMs. But it lacks feature extraction and a variety of visualization capabilities. Also, the outputs of different tools are often of different formats. For example, when HTK is employed, the results produced by the MFCC procedure are in HTK format. The output format of the file produced by Kaldi is binary. Consequently, the format of an output file can only be read by the respective tools. In other words, the outputs of these tools, e.g., T1s, very often cannot directly be imported into existing tools, e.g., T2s, that are invoked in the next stage, particularly when the companies that release T1s and T2s are different.

In this study, our speaker identification system is implemented using the Python programming language. As mentioned earlier, many existing tools or software focuses on sentence recognition, or is just implemented for some key components of speech recognition. The reason why we choose Python programming language is that it can invoke many mathematical equations and provide many scalable libraries, e.g., SymPy [30] for algebra. In other words, this programming language has a certain degree of cross-platform characteristic [31]. SciPy [32] as another open-source Python algorithm library offers a mathematical toolkit, with which complex mathematical operations, such as linear algebra, fast Fourier transform, etc., can be invoked. In addition, due to feature extraction, a large number of parameters are generated. Scipy is also helpful. Further, the NumPy expansion library, as a high-level Python tool that supports a large number of dimensional array and matrix operations, is utilized to store the large volumes of output parameters and data, including observed scientist data, library, files, and our acoustic models. The format employed is HDF5 [33]. Therefore, the output formats of different steps of our system can be unified. Consequently, the characteristic parameter data generated at each step can be smoothly received by the follow-up procedures.

This also reduces the corresponding efforts when developers would like to carry out their research and development projects by invoking our system or components, i.e., it is easier to hugely increase the possibility of scaling up or improving the functions of a speaker identification system. Our system was tested on PC, the specifications of which are listed in Table 1.

Table 1. Tested hardware and software specifications.

Item	**Description**
CPU	Intel Core I5
OS	Ubuntu 14.04.5 LTS
Memory	4GB
Hard disk	120GB
Microphone	INTOPIC JAZZ-010
Sampling frequency	44.1 KHz
File format	16-bit linear PCM
Recording software	Audacity

A total of five experiments, denoted by Experiment 1 – Experiment 5, were performed in this study. In Experiment 1, the words involved in the training and test phases are the same. In Experiment 2, words utilized in the two phases are different. In the third and fourth experiments, we, respectively, redid Experiments 1 and 2 with the cases that Chinese sentences and characters, rather than English words, are used. Experiment 5 compares our system with the MFCC [34] and MFCC plus delta [35].

5.1 Experiment 1

The first experiment was performed in a quiet environment. Fifteen students were invited to read the Pronunciation Guide of the oxford learner's dictionaries [36] as the training voice so as to establish the fifteen students' acoustic models. The pronunciation guide contains all the English words' pronunciation. Table 2 lists the words for training.

Table 2. Training-word list used in Experiment 1.

pen	bad	tea	did	cat	get	chain	jam
fall	van	thin	this	see	zoo	shoe	vision
hat	man	now	sing	leg	red	yes	wet
happy	sit	ten	father	got	saw	put	actual
too	cup	fur	about	say	go	my	boy
near	hair	pure					

During the test phase, the fifteen students read the training words a total of 43 times. The similarities between two speakers x_i and $x_j, 1 \le i, j \le 5$, and the identification accuracies of these testers are shown in Table 3.

Table 3. The similarities and identification accuracies of Experiment 1 performed on the words listed in Table 2 as the training and test data (%).

Acoustic model Similarity(%) Tester	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	Accuracy
1	88.22	38.15	47.05	45.81	24.96	70.44	55.42	79.43	64.76	44.28	36.97	28.21	41.52	25.01	34.40	90.65
2	38.70	93.76	69.65	11.23	33.82	28.27	52.06	48.63	44.54	58.29	34.41	48.43	70.48	27.72	39.96	87.37
3	44.20	64.68	94.36	78.19	44.27	7.11	45.86	58.18	65.25	48.94	48.48	40.40	50.91	23.67	18.86	86.87
4	46.15	6.50	77.02	91.79	39.05	11.97	25.46	26.98	18.03	37.52	75.26	53.07	77.86	72.43	57.84	87.05
5	20.70	30.55	46.74	41.37	87.76	36.72	54.32	69.04	11.28	36.82	8.75	16.01	14.19	65.66	43.33	89.67
6	65.66	24.76	5.72	8.67	35.36	92.78	27.90	25.38	44.10	62.04	11.07	4.13	44.08	12.71	52.06	93.52
7	54.87	56.46	45.10	29.28	51.37	32.23	92.90	28.02	17.70	63.43	15.44	38.37	36.20	20.46	52.30	91.77
8	75.37	51.73	58.26	30.02	70.96	23.52	31.65	93.27	8.20	42.50	41.84	60.48	40.24	16.64	38.04	87.87
9	61.66	39.86	64.15	19.29	12.92	43.73	15.05	5.04	91.74	62.42	11.80	16.06	62.82	23.17	58.52	91.40
10	49.25	57.72	45.24	37.36	36.76	58.00	59.22	43.03	60.37	95.35	81.11	72.76	31.03	77.73	57.35	87.09
11	36.25	35.20	49.29	78.44	6.43	10.74	17.74	42.97	10.96	80.47	91.48	7.28	12.47	30.18	63.76	89.22
12	30.43	48.57	37.87	52.08	17.52	7.99	43.19	57.09	14.08	73.55	11.85	86.71	24.20	42.09	64.66	91.08
13	38.28	70.69	49.93	81.66	9.67	46.55	32.18	36.65	62.21	31.83	16.01	28.30	95.19	61.08	49.54	89.48
14	26.49	28.10	27.80	77.04	65.16	12.07	25.27	13.26	26.86	80.61	28.52	44.01	58.68	93.11	16.82	90.27
15	37.65	37.62	15.03	59.57	40.81	49.77	55.84	34.32	57.13	55.26	61.32	63.56	52.72	17.60	95.51	89.45

Because the words used in the test and training phases are of the same, the intonation and moods of the same tester on the same words are almost the same to those established in his/her training stage, meaning the acoustic model contains phonological features. The acoustic models of the same speaker are similar enough so that the system can identify the tester more accurately. Although the similarity is relatively high, it cannot reach 100% because even the same words pronounced by the same tester, the pronunciation (sometimes noise) on different times may vary. The accuracies of the identification are then reduced. But the similarity of the same tester is higher than those between two different testers.

5.2 Experiment 2

In this experiment, we used 40 training words, which as listed in Table 4 are different from those shown in Table 2, to test the system with the acoustic models established in Experiment 1. Table 5 illustrates the test results.

Table 4. 40 test words used in Experiment 2. They are different from those listed in Table 2.

able	advice	beauty	boot	careful	chapter	convention	cousin
credit	cushion	date	develop	discuss	down	environment	evidence
export	fear	force	grand	highlight	idea	increase	instruction
itself	lamb	lively	margin	milk	must	not	often
out	pay	pink	positive	roof	send	solve	swap

Table 5. The average similarities and average accuracies of Experiment 2 (%).

Acoustic model Similarity(%) Tester	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	Accuracy
1	84.58	47.44	48.80	41.40	30.57	60.95	43.05	75.34	58.27	49.23	34.02	30.82	41.40	34.70	34.06	87.21
2	43.53	89.51	70.22	4.80	31.89	20.51	55.71	53.28	30.97	58.47	36.11	52.83	70.49	23.12	35.95	83.25
3	47.92	68.45	90.11	74.00	37.88	9.40	49.62	61.21	54.97	47.22	49.83	40.84	52.70	22.86	12.46	82.02
4	40.63	8.91	72.03	87.50	41.74	8.10	33.32	35.00	20.84	38.37	78.35	54.30	79.40	84.61	56.58	84.93
5	25.95	29.20	42.67	45.75	84.18	37.77	51.34	70.27	16.63	37.67	16.15	22.92	3.51	69.85	43.01	86.00
6	62.35	24.80	12.26	8.59	36.33	87.00	38.40	25.09	42.62	64.69	13.26	3.67	41.89	5.97	47.91	90.00
7	47.94	51.94	51.77	35.88	55.41	36.04	87.86	24.81	11.37	53.07	13.01	45.52	32.89	24.86	64.95	88.44
8	75.84	55.83	63.66	35.36	70.20	22.04	29.72	86.28	8.61	53.86	32.42	55.88	33.30	17.35	35.69	86.73
9	60.31	35.63	58.41	18.54	15.72	37.83	15.17	10.35	86.61	65.47	9.20	16.15	62.88	27.89	57.77	86.38
10	46.21	53.76	51.88	35.05	36.56	59.92	56.25	50.00	63.67	91.94	80.65	79.68	23.35	79.28	58.67	83.26
11	35.72	32.22	48.50	80.62	11.94	9.17	13.14	36.83	13.83	80.50	87.86	12.81	10.43	28.96	61.76	83.24
12	33.36	52.79	43.90	55.19	18.69	1.69	44.30	51.51	18.21	76.19	8.79	84.36	19.72	51.21	63.90	85.22
13	44.63	66.21	50.22	76.64	6.26	44.36	30.84	36.92	62.75	26.12	10.66	22.93	88.28	55.84	62.52	87.55
14	33.19	22.73	24.38	82.81	69.32	5.66	25.89	20.18	29.29	77.28	30.90	46.43	59.64	86.28	19.16	88.22
15	35.18	37.22	15.36	57.61	47.16	50.40	60.64	36.25	58.40	62.23	64.66	59.57	58.27	21.15	89.60	86.39

As shown, accuracies illustrated in Table 5 are slightly lower than those shown in Table 3. This is because of using different test and training words. The first difference comes from the fact that even the same words pronounced by the same speaker, the pronunciations at different time points may slightly change. The second difference is due to different words of different pronunciations, resulting in a little lower accuracies, even when the speaker is the same. However, the distribution of speech features in the feature space is actually resulted from different frequency distributions in human voice signals. So when some words are not included in the training phase, our identification system can still identify the speaker among the established acoustic models.

5.3 Experiment 3

In the third experiment, English training words and test words are substituted by a Chinese article [37] which as shown in Table 6 has 22 sentences in turn consisting of 208 Chinese characters as the training data to establish testers' acoustic models during the training phase. Ten sentence as the test data are selected from the article. Table 7 shows the experimental results in which the similarities and identification accuracies, respectively, are individually lower than those listed in Table 3. The reason is that voice signals collected in the test phase are a subset of the sentence-voice set of the article. Theoretically, the voice features collected in the test phase are a part of those collected during the training phase.

Table 6. Chinese sentences as the training data to establish the testers' acoustic models in Experiment 3.

有一個人作了一個夢，夢中他來到一間二層樓的屋子。進到第一層樓時，發現了一張長長的大桌子，桌旁都坐著人，而桌子上擺滿了豐盛的佳餚，可是沒有一個人能吃得到，因為大家的手臂受到魔法師詛咒，全都變成直的，手肘不能彎曲，而桌上的美食，夾不到口中，所以個個愁容滿面。但是他聽到樓上卻充滿了愉快的笑聲，他好奇的上樓一看，同樣的也有一群人，手肘也是不能彎曲，但是大家卻吃得興高采烈。原來每個人的手臂雖然不能彎曲，但是因為對面的人彼此協助，互相幫助夾菜餵食，結果大家吃得很盡興。

Table 7. The similarities and accuracies of Experiments 3 (%).

Acoustic model Similarity(%) Tester	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	Accuracy
1	84.58	34.93	43.10	43.04	21.80	68.66	53.74	74.96	63.44	41.83	33.40	23.50	36.42	19.11	28.54	86.91
2	32.97	91.81	67.08	6.72	29.39	23.86	50.05	44.48	41.34	53.96	30.44	43.44	66.39	22.04	35.93	84.56
3	43.00	61.78	89.55	72.67	41.68	6.10	42.82	52.70	62.79	45.54	45.60	39.38	46.54	21.98	17.37	83.73
4	44.20	1.74	73.01	90.30	34.08	10.81	22.66	25.14	15.51	33.61	72.67	50.84	73.55	68.05	52.33	84.31
5	19.17	29.36	44.45	40.35	82.52	33.14	50.39	65.09	10.18	34.99	5.90	14.07	11.58	59.69	38.75	87.71
6	63.09	20.69	2.60	3.47	31.29	90.30	23.68	22.48	40.74	59.16	6.21	0.04	40.97	6.82	51.00	89.54
7	51.06	53.00	39.52	27.02	45.43	26.93	87.08	26.64	14.60	60.82	12.39	33.17	31.71	14.94	47.62	88.35
8	74.15	49.75	54.56	25.07	67.83	19.25	27.33	90.45	6.42	39.11	38.33	58.93	38.58	12.21	36.51	84.66
9	56.42	37.32	61.50	16.02	9.49	38.80	10.74	1.44	87.49	59.80	10.37	11.68	60.61	18.90	54.13	87.97
10	45.48	53.51	41.50	34.87	33.50	52.02	54.55	39.75	56.48	90.10	79.47	71.20	26.43	72.07	54.60	84.88
11	33.51	32.18	44.97	72.64	5.36	9.10	16.40	37.22	7.47	77.52	87.48	4.54	10.97	26.69	58.02	87.23
12	28.74	46.96	33.07	47.58	13.45	3.85	37.78	52.23	9.91	69.08	7.24	83.42	22.64	37.28	59.12	89.70
13	33.58	64.78	47.32	79.84	7.74	44.08	26.51	31.92	57.48	28.60	14.08	26.41	92.59	59.94	48.26	88.23
14	22.05	23.28	24.34	74.17	63.08	7.79	23.73	11.62	22.01	78.56	23.55	38.12	53.62	88.13	11.71	86.53
15	33.75	35.39	9.04	57.96	39.00	45.00	53.29	32.02	53.07	49.84	59.99	60.20	50.84	13.87	91.40	87.19

5.4 Experiment 4

In Experiment 4, we redid Experiment 2, but the trained acoustic models reuse the ones established in Experiment 3. One test sentence (杯弓蛇影) which exclude the 22 sentences shown in Table 6, is given. The similarities and identification accuracies as illustrated in Table 8 are lower than those illustrated in Table 3, 5 and 7. The reasons are the same as those mentioned above.

Table 8. The average similarities and average accuracies of Experiment 4 (%).

Acoustic model Similarity(%) Tester	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	Accuracy
1	72.26	24.86	33.91	22.93	9.07	42.03	35.53	67.12	53.20	24.06	21.59	17.80	40.50	23.07	17.15	54.14
2	24.47	61.24	48.80	0.72	14.45	7.87	41.83	49.54	23.09	39.25	14.35	38.72	52.58	8.64	20.26	63.21
3	34.51	52.71	69.50	55.30	35.14	4.10	32.87	53.97	48.29	23.48	43.42	25.09	37.81	10.23	3.09	50.10
4	25.08	5.51	52.65	69.46	25.74	8.91	26.57	23.62	1.62	25.15	64.40	42.31	57.24	68.52	41.65	61.60
5	12.43	17.88	33.97	30.52	67.45	17.30	33.68	54.88	1.93	17.64	0.88	7.91	6.47	51.13	35.50	59.74
6	45.95	12.68	2.86	4.99	18.69	74.27	18.60	3.86	27.66	41.74	8.33	11.62	22.52	7.91	27.44	54.12
7	38.83	40.99	30.50	24.02	32.44	14.49	71.34	7.22	1.36	40.08	1.37	33.20	22.36	9.42	34.92	51.31
8	68.03	46.37	57.21	25.96	53.72	6.63	10.14	68.57	3.98	36.61	22.40	38.67	24.47	2.11	18.66	55.34
9	50.83	25.66	43.72	0.38	3.94	29.61	0.88	5.10	60.90	50.99	0.77	0.15	48.82	16.80	44.66	54.52
10	28.38	39.99	27.65	20.90	16.60	43.66	39.61	37.20	48.78	80.01	64.31	54.50	6.52	61.29	41.84	52.22
11	20.44	13.93	41.40	68.93	0.35	9.62	2.07	25.97	3.31	64.40	74.11	12.57	11.00	11.14	42.35	61.28
12	17.27	36.21	27.40	42.98	6.34	15.05	35.36	43.31	4.42	52.95	9.48	62.34	6.56	33.68	53.11	61.89
13	37.86	50.07	38.48	59.38	6.51	25.18	18.38	29.10	47.63	11.23	7.64	6.62	63.24	45.86	43.40	63.40
14	22.45	9.88	6.24	67.03	52.85	8.98	11.38	0.02	18.13	59.11	12.82	32.69	48.19	70.80	7.57	51.55
15	17.12	16.14	1.05	39.71	36.21	27.25	39.81	23.13	44.69	43.82	47.16	51.62	43.62	5.28	64.98	66.34

In Chinese, there are more than 1100 different pronunciations. But Table 6 only contains 208 characters of 109 different pronunciations (In Chinese, many different characters are often of the same pronunciation). Even the similarities and identification accuracies shown in Table 8 are lower than those of previous experiments, each of them still has achieved a considerable level, showing that our system is feasible.

5.5 Experiment 5

In Experiment 5, we compare our system with the MFCC system [34] and MFCC+delta [35] by redoing the Experiments 1 - 5. Table 9 shows the results when the training words and test words are in English and the training words are the same as the test words. Table 10 shows the experiment results when the English words are used and the test words are not included in the training words. Table 11 illustrates the results when the sentences are all in Chinese and the test sentences are a subset of the training sentences. Table 12 lists the experiment results when the Chinese test sentences are not included in the Chinese training sentences.

Table 9. The identification accuracies of Experiment 5 when the training words and test words are in English and the training words are the same as the test words (%).

system tester	Ours	MFCC	MFCC+delta
1	90.65	82.82	84.16
2	87.37	81.03	82.08
3	86.87	80.47	80.75
4	87.05	78.73	80.72
5	89.67	80.92	82.29
6	93.52	87.71	88.02
7	91.77	82.02	84.81
8	87.87	82.17	83.35
9	91.40	85.37	87.60
10	87.09	81.75	82.11
11	89.22	80.99	81.51
12	91.08	85.29	87.45
13	89.48	84.45	86.31
14	90.27	81.96	82.19
15	89.45	82.43	83.57

Table 10. The accuracies of Experiment 5 when the words used are all in English and the test words are different from those training words (%).

system tester	Ours	MFCC	MFCC+delta
1	87.21	46.06	47.77
2	83.25	57.35	58.81
3	82.02	44.88	44.89
4	84.93	54.79	56.72
5	86.00	54.43	54.48
6	90.00	47.95	49.62
7	88.44	41.56	42.00
8	86.73	46.20	47.54
9	86.38	44.67	46.50
10	83.26	46.98	48.87
11	83.24	52.49	53.44
12	85.22	55.49	56.29
13	87.55	53.70	54.79
14	88.22	43.87	45.15
15	86.39	56.72	57.35

Comparing Tables 9 and 10, it is clear that the identification accuracies of the MFCC and MFCC+delta systems are lower than those of our system, no matter whether the test-word set is included in the training-word set or not. Basically, the MFCC does not establish acoustic models for all speakers, meaning the MFCC alone does not identity the distribution of voice features. The other reason is that it is difficult to avoid the change of sound volume (i.e., frequency energies), moods, and intonation during the test phase. The accuracies of MFCC+delta is higher than those of MFCC alone, because it contains dynamic characteristics of voice. But the overall identification accuracies are still lower than those of our system, indicating that the Gaussian mixture model is helpful in identifying speakers.

Table 11. The identification accuracies of Experiment 5 when the training sentences and test sentences are in Chinese and the test sentences are a proper subset of the training sentences (%).

system tester	Ours	MFCC	MFCC+delta
1	86.91	81.81	82.52
2	84.56	79.10	80.71
3	83.73	78.07	80.94
4	84.31	77.58	78.72
5	87.71	80.31	82.35
6	89.54	81.74	83.36
7	88.35	80.18	81.33
8	84.66	75.30	77.08
9	87.97	81.20	83.15
10	84.88	79.17	80.33
11	87.23	77.94	79.12
12	89.70	79.82	81.53
13	88.23	79.21	81.81
14	86.53	79.85	82.00
15	87.19	81.42	83.22

Table 12. The accuracies of Experiment 5 when the sentences used are in Chinese and the test sentences are a proper subset of the training sentences (%).

system / tester	Ours	MFCC	MFCC+delta
1	54.14	32.85	32.85
2	63.21	45.37	45.67
3	50.10	28.57	28.41
4	61.60	38.44	37.67
5	59.74	30.80	30.84
6	54.12	29.98	30.43
7	51.31	34.77	35.46
8	55.34	30.11	29.62
9	54.52	32.65	32.97
10	52.22	23.78	23.68
11	61.28	45.59	45.50
12	61.89	37.67	38.19
13	63.40	47.84	48.61
14	51.55	21.90	22.64
15	66.34	47.34	47.19

It can be seen that in Tables 11 and 12, the accuracies of both MFCC and MFCC+delta are lower than those of our scheme since they do not establishment acoustic models for all testers. Also, if comparing Tables 9 and 11 (Tables 10 and 12), we can also see that the accuracies when Chinese is used are lower than those when English words are utilized. This is because in the training phase, not all Chinese pronunciations are collected.

6. Conclusion and Future studies

The purpose of this study is to establish an acoustic model, i.e., a phonetic model, of a speaker, and to provide a set of practical system processes which can be applied to identify an unknown speaker. We summarize the operation flow of the whole identification system and implement a speaker identification system with Python. The reason for choosing Python is that it provides many basic math-related extensions, which are extensible, user-friendly, and easy to implement on a variety of platforms. From the test results of our system, we can find that the recognition accuracies of the proposed system is higher when the training voice content of the system covers the test voice content. When the test materials are not included in the training materials, different words will produce different vocal patterns, and will affect the overall system identification rate. Three conclusions can be extracted from this study.

(1) The use of Python can convene the integration and expansion of a system, because the output formats of different processing steps are unified in this study. Thus, new functions or function modifications can be conveniently and easily developed and performed, respectively.

(2) To accurately identify a speaker, during the training phase, a large number of voice training is required, causing a long training time.

(3) The number of current vocabularies is large, it is not easy to identify all pronunciations during the test phase. Also, in different situations, like after singing, after a long speech, having a cold, etc., with different environmental noises, a person’s voice may change. This will greatly affect the identification

accuracies of our system. Therefore, it is hard to produce 100% of identification rate.

Voice recognition has gradually enters human lives. It is helpful in convening people’s everyday lives, such as launching commands to a personal computer by using human voice, accessing data by inputting human voice and playing games by submitting human-voice commands. Other applications in our daily lives include automatic ticket purchasing on a vending machine, and online shopping with voice services via a mobile phone. We believe that in the near future, these applications will soon significantly join our everyday living activities. In the future, we would like to find a method to shorten the training time by identify essential words or pronunciations for English and Chinese. We will also derive the behavior model and reliability model for our system so that users can predict the behaviors and reliability of this system before using it. These will constitute our future studies.

References

[1] C. Zhan, W. Li and P. Ogunbona, “Face Recognition from Single Sample based on Human Face Perception,” *International Conference Image and Vision Computing New Zealand*, pp. 56-61, 2009.

[2] http://www.apple.com/tw/ios/siri/

[3] https://cloud.google.com/speech/

[4] D. A. Reynolds, “An overview of Automatic Speaker Recognition Technology,” *IEEE International Conference on Acoustics, Speech, and Signal Processing*, Vol. 4, pp. 4072-4075, 2002.

[5] http://htk.eng.cam.ac.uk/

[6] http://kaldi-asr.org/doc/about.html

[7] L. Rabiner and B. H. Juang, “Fundamentals of Speech Recognition,” *in Prentice Hall*, 1993.

[8] https://en.wikipedia.org/wiki/Speaker_recognition

[9] T. Stafylakis, M. J. Alam and P. Kenny, “Text-Dependent Speaker Recognition With Random Digit Strings,” *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, Vol. 24, Issue. 7, pp. 1194-1203, July 2016.

[10] D. A. Reynolds and R. C. Rose, “Robust Text-Independent Speaker Identification Using Gaussian Mixture Speaker Models,” *IEEE Transactions on Speech and Audio Processing*, Vol. 3, Issue. 1, pp. 72-83, Jan 1995.

[11] http://www.d-ear.com/

[12] http://www.playrobot.com/speech-recognition/88-arduino-chinese-voice-recognition-module.html

[13] http://www.garmin.com.tw/m/buzz/tw/minisite/nuvi3790T/feature_02.htm

[14] J. S. Lim and A. V. Oppenheim, “Enhancement and Bandwidth Compression of Noisy Speech,” *Proceedings of the IEEE*, Vol. 67, Issue. 12, pp. 1586-1604, Dec. 1979.

[15] W. Zunjin and C. Zhigang, “Improved MFCC-based feature for robust speaker identification,” *Tsinghua Science and Technology*, Vol. 10, Issue. 2, pp. 158-161, April 2005.

[16] N. Cristianini and J. Shawe-Taylor, “Support Vector Machines,” *in Cambridge University Press*, 2000.

[17] B. H. Juang and T. Chen, “The Past, Present, and Future of Speech Processing,” *IEEE Signal Processing Magazine*, Vol. 15, Issue. 3, pp. 23-48, May 1998.

[18] N. E. Huang et al, “On Instantaneous Frequency,” *in World Scientific Publishing Company*, pp. 177-229, 2009.

[19] R. Vergin, D. O'Shaughnessy and A. Farhat, “Generalized Mel Frequency Coefficients for Large-Vocabulary Speaker-Independent Continuous-Speech Recognition,” *IEEE Transactions on Speech and Audio Processing*, Vol. 7, Issue. 5, pp. 525-532, Sep 1999.

[20] D. O'Shaughnessy, “Speech Communications: Human and Machine,” *Wiley-IEEE Press*, 1999.

[21] T. T. Soong, “Fundamentals of Probability and Statistics for Engineers,” *Wiley*, 2004.

[22] X. Peng, X. Wang and B. Wang, “Speaker Clustering via Novel seudo-Divergence of Gaussian Mixture Models,” *International Conference on Natural Language Processing and Knowledge Engineering*, pp. 111-114, 2005.

[23] http://www.datasciencelab.cn/clustering/gmm

[24] L. Rabiner and B. H. Juang, “Fundamentals of Speech Recognition,” *in Prentice Hall*, pp. 215-219, 1993.

[25] A. Bhattacharyya, “On a Measure of Divergence between Two Statistical Populations,” *in Springer on behalf of the Indian Statistical Institute*, pp. 99–109, 1943.

[26] K. Rao and P. Yip, “Discrete Cosine Transform: Algorithms, Advantages, Applications,” *in Academic Press*, 1990.

[27] A. Goel and A. Gupta, “Design of Satellite Payload Filter Emulator Using Hamming Window,” *International Conference on Medical Imaging, m-Health and Emerging Communication Systems (MedCom)*, pp. 202-205, 2014.

[28] J. O. Smith III, “Spectral Audio Signal Processing,” *W3K Publishing*, 2011.

[29] H. C. Ravichandar and A. P. Dani, “Human Intention Inference Using Expectation-Maximization Algorithm With Online Model Learning,” *IEEE Transactions on Automation Science and Engineering*, Vol. PP, Issue. 99, December 2016.

[30] http://www.sympy.org/en/index.html

[31] https://www.python.org/

[32] https://www.scipy.org/

[33] https://www.hdfgroup.org/

[34] J. P. Openshaw, Z. P. Sun and J. S. Mason, “A Comparison of Composite Features under Degraded Speech in Speaker Recognition,” *IEEE International Conference on Acoustics, Speech, and Signal Processing*, Vol. 2, pp. 371-374, 1993.

[35] A. Chaudhari, A. Rahulkar and S. B. Dhonde, “Combining dynamic features with MFCC for text-independent speaker identification,” *International Conference on Information Processing (ICIP)*, pp. 160-164, 2015.

[36] http://www.oxfordlearnersdictionaries.com/us/about/pronunciation_english

[37] http://isrc.ccs.asia.edu.tw/www/essay/essay7/essay7-008.htm