UNIVERSITY OF SCIENCE AND TECHNOLOGY OF HANOI

Digital Signal Processing
Final Project Report 2019 - 2020

AUTOMATIC SPEAKER RECOGNITION

GROUP 2 - ICT

Members

- Nguyễn Thành Vinh - BI8 - 187
- Nguyễn Tiến Dũng - BI9 - 071
- Nguyễn Thành Gia Hiển - BI9 - 099
- Vũ Văn Huy - BI9 - 116
- Trần Thanh Long - BI9 - 157
- Trịnh Tuấn Minh - BI9 - 168
- Nguyễn Hồng Quang - BI9 - 194
- Nguyễn Phương Thảo - BI9 - 212
- Nguyễn Thị Lan Anh - BI9 - 244

Table of content

1. Introduction

Speaker recognition (also called voice recognition) is using properties of voices to identify the person who is speaking.

We can classify **speaker recognition** tasks into two categories: speaker identification and speaker verification

- In the speaker identification task, we will compare an unknown voice with dataset of valid users. The best match is used to identify the speaker.
- In the speaker verification task, the unknown speaker and the claimed sample will be used for identification. If the match is bigger than predefined threshold then the identity claimed is accepted.

The speech used for these tasks can be either text dependent or text independent: In text dependent, the system has the prior sample of text to be spoken and the user will speak the same text as the predefined text. But in text independent, there is no prior knowledge by the system of the text to be spoken.

Speech produced complex signals as a result of some alterations occurring at three levels: semantic, linguistic and acoustic. Differences in these alterations can lead to differences in the properties of signals. The characteristics of the speaker are affected by the linguistic massage, age, health, emotional state… Noise and quality of the recording device also interfere for the process.

Speaker recognition is one of the most important part of human-computer interaction. As the development of technology, Voice User Interface has been a necessary part of such computers. It will give people many useful applications.

- Some devices are really small, so they are easy to lose and be stolen. In these situations, speaker recognition can also be a combination of seamless interaction with computer and security guard when the device is lost. Hence, people can find the devices faster.
- The need of personal identity verification will become more acute in the future.
- Speaker verification may be essential in business telecommunications. Telephone banking and telephone reservation services will develop rapidly when secure means of authentication are available.
- In court cases, speaker recognition technique may bring a reliable scientific determination to confirm criminal identity.

Moreover, these techniques can be used in environments which require high security. It can be combined with other biological metrics to form a multi-modal authentication system.

In this task, we have built a proof-of-concept text-independent speaker recognition system with GUI support. It is fast and exactly based on our tests on large corpus. And the GUI program only requires very short expressions to quickly reply. The whole system is fully described in this report. The repository contains the source code, all documents, …

2. Speaker recognition system

The process of identifying the speaker is done through phases. There are two phases in our speaker recognition system.

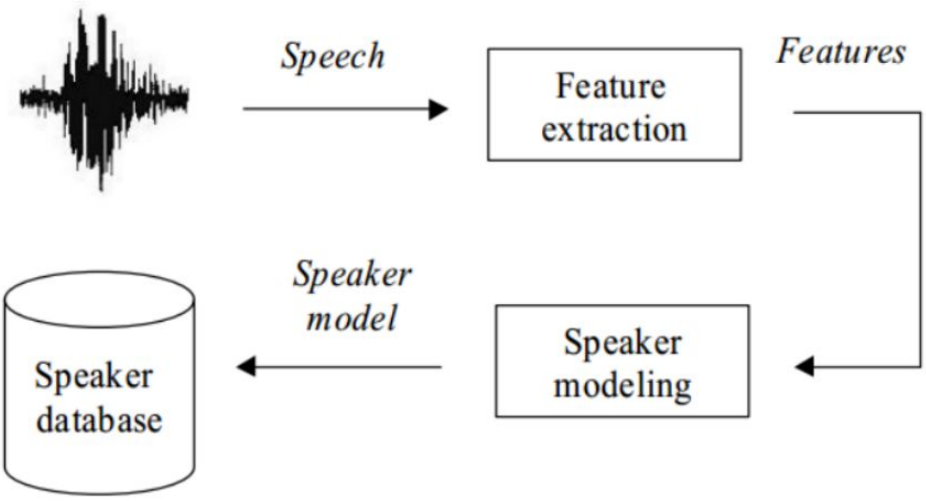

Figure 1: Diagram of Enroll Speakers phase

2.1. Enroll speakers

Voice of the speakers are collected and used to **training** model. The set of models of multiple speakers is also known as the speaker database.

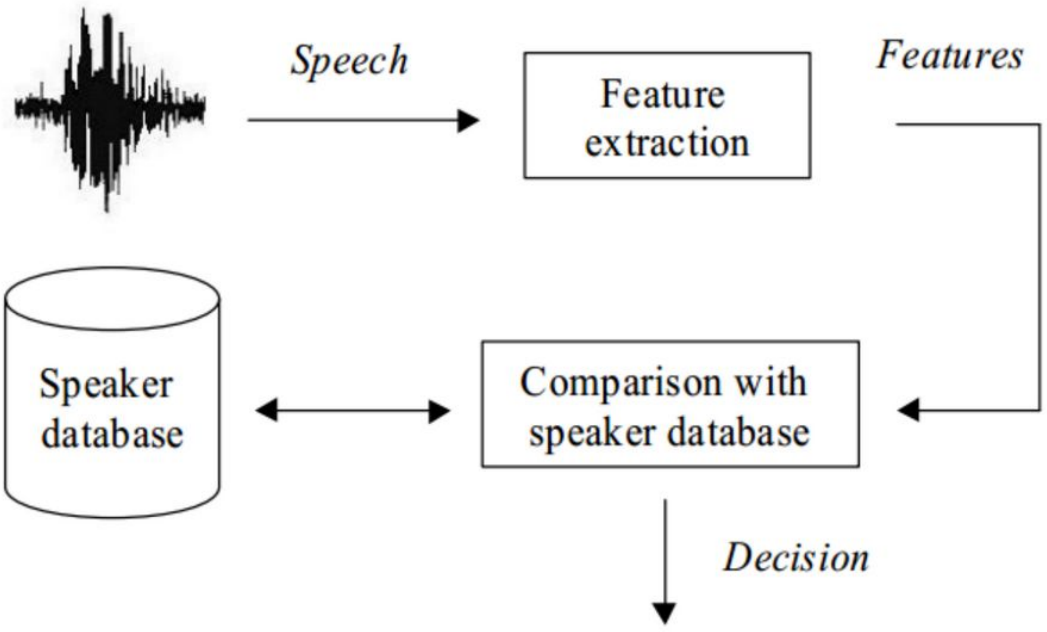

Figure 2: Diagram of Identifying Speakers phase

2.2. Identifying Speakers

Voice data of an unknown user are put into the system and matched with the models in the speaker database.

Details of two phases are as follows:

- Both phases share the same first two steps. The first step is to collect voices. Voices can be collected via microphones and converted into discrete digital signals. However, this data will usually have noises, so it needs to be filtered when entering the second step phase.
- The second step is feature extraction, aimed at reducing data size but still ensuring enough information to distinguish the speaker. In this report, we will present MFCC features.
- In the third step of the enroll phase, the speaker information that has been extracted has been modeled and stored in a database. The report will use the Gaussian Mixture Model to model speaker data to build GMM that corresponds to the MFCC characteristics passed in.
- In the third step of the identification phase, the extracted data is matched with the data in the database and makes a decision as to who that person is.

It can be seen that the two phases are implemented separately but very closely related, in which the two most difficult phases are characteristic extraction, modeling and data matching. The next section of the report will present the main ideas of the feature extraction algorithm and modeling.

3. Algorithms

3.1. Extracting voice features with Mel-Frequency Cepstral Coefficient (MFCC)

The short-term power spectrum of a sound which is based on a linear transform of a log power spectrum on a non linear mel-scale of frequency can be called **Mel-Frequency Cepstral Coefficient**. It can be used in Speaker Recognition tasks.

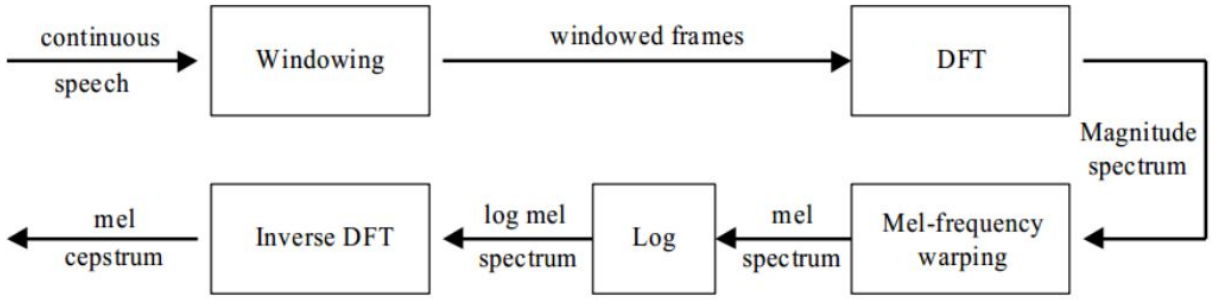

Figure 1: MFCC extraction process

First, we will separate the input voice into short-time frames of length L and other frames shall have overlap R. Figure 2 shows that they are widowed by Hamming Window.

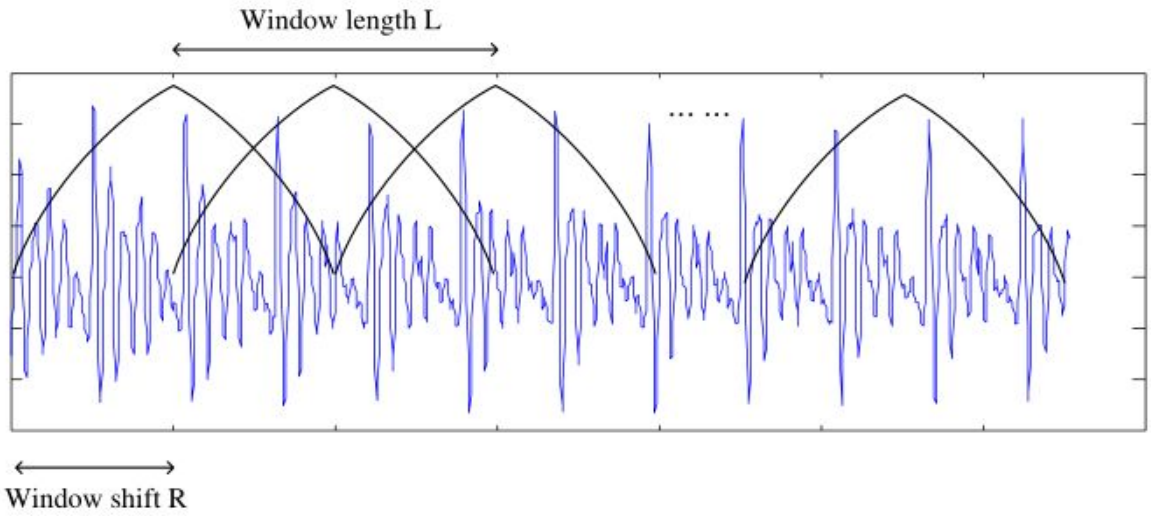

Figure 2: Framing and Windowing

Second, the windowed signal's spectrum will be computed by using the Discrete Fourier Transform (DFT). We obtain a complex number X[k] for each discrete frequency band which represents the magnitude and period of that frequency component in the original signal.

Mel-scale are pointed to scale the waveband so that humans can hear more clearly because human's ear cannot get all frequency domains. They are close to linear below 1 kHz and logarithmic above 1 kHz, as shown in Figure 3:

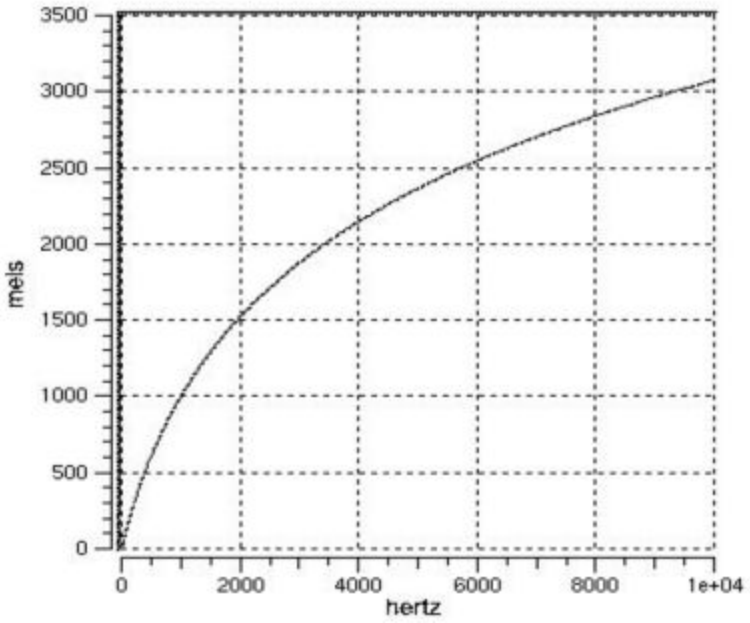

Figure 3: Mel-scale

In MFCC, Mel-scale is applied on the spectrums of the signals. The expression of Mel-scale wrapping is:

$$M(f) = 2595 log_{10}(1 + \frac{f}{700})$$

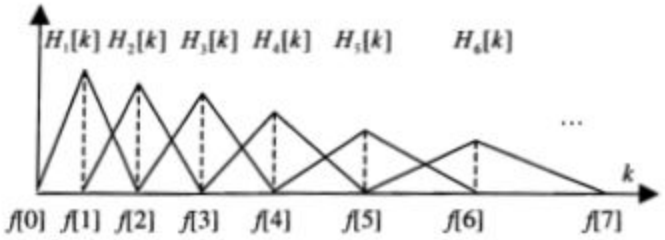

Figure 4: Filter banks

Then, we apply the bank of filters according to Mel-scale on the spectrum, compute the logarithm of energy under each bank by

$E_i[m] = log(\sum_{k=0}^{N-1} X_i[k]^2 H_m[k])$ and apply Discrete Cosine Transform (DCT) on E_i[m] (m = 1,2,3,...,M) to get an array c_i :

$$c_i = \sum_{m=0}^{M-1} E_i[m] cos(\frac{\Pi n}{M}(m - \frac{1}{2}))$$

3.2. Modelling speaker's voice using Gaussian Mixture Model (GMM) and identifying speaker

3.2.1. Modelling using GMM

A Gaussian mixture model is a probabilistic model that assumes all the data points are generated from a mixture of a finite number of Gaussian distributions with unknown parameters.

GMM is usually used in acoustic studying assignments such as speaker recognition. GMM assumes that the probability of a vector x from the model is:

$$p(x|w_i, \mu_i, \Sigma_i) = \sum_{i=1}^{K} w_i \mathcal{N}(x|\mu_i, \Sigma_i) \quad (1)$$

where

$$\mathcal{N}(x|\mu_i, \Sigma_i) = \frac{1}{(2\pi)^{\frac{d}{2}}\sqrt{|\Sigma_i|}} \exp\left(-\frac{1}{2}(x - \mu_i)^T \Sigma_i^{-1}(x - \mu_i)\right)$$

subject to

$$\sum_{i=1}^{K} w_i = 1$$

GMM can describe the distribution of feature vector with several clusters

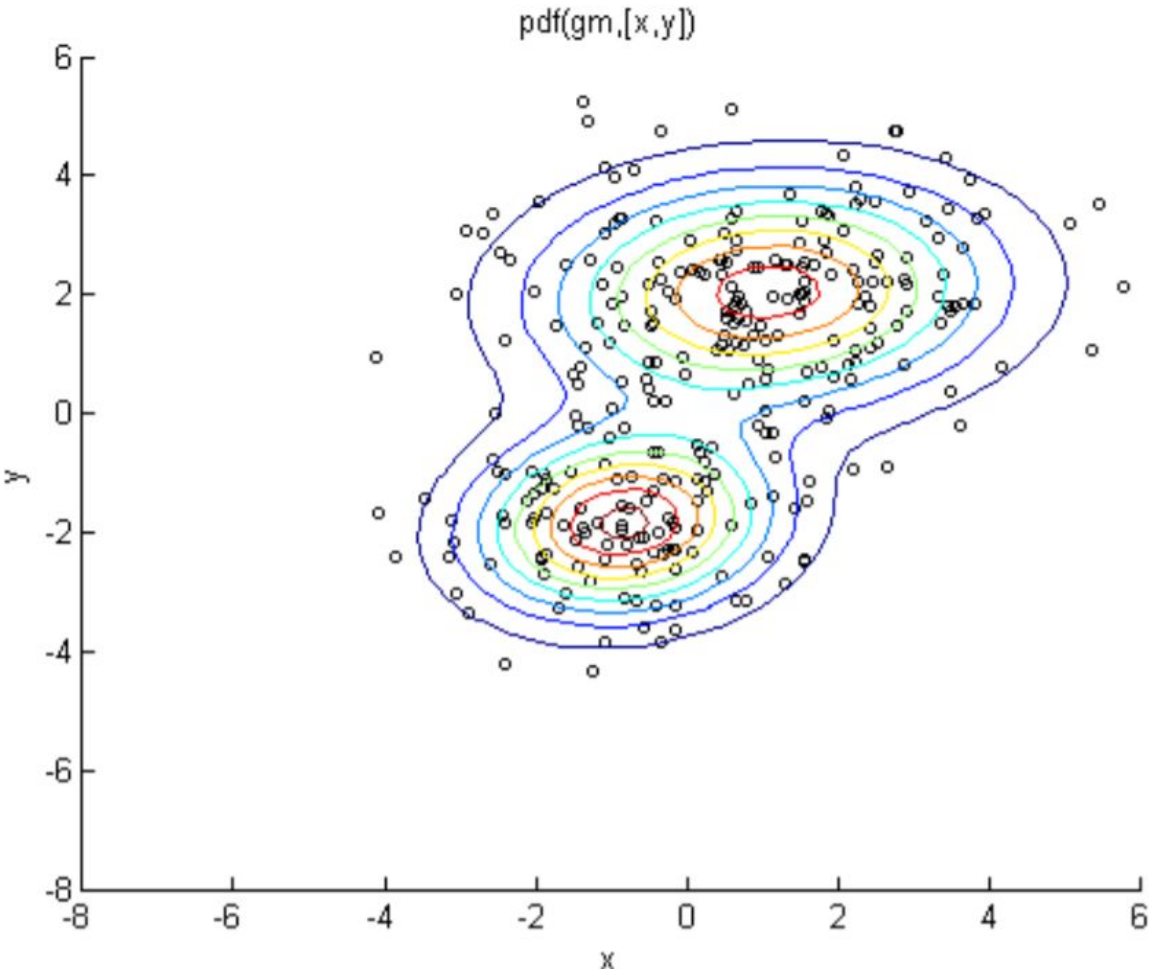

Figure 5: A two-Dimensional GMM with Two Components

Expectation-Maximization Algorithm used to maximize the possible. In one iteration of GMM training case there two separated steps:

- **E-step:** For each feature vector, estimate the probability that each Gaussian generated it which can be done with (1).
- **M-step:** Emend the parameters of GMM to maximize the prospectively of data. The hidden variable z_{ij} is introduced to show where *i-th* is generated by Gaussian *j*. It can be shown that instead of maximizing the prospectively of data, we can maximize the expectation of log prospectively of data with respect to Z

Let θ = {*w*,θ,$\sum$}, the log prospectively function is:

$$Q(\theta',\theta) = \mathbf{E}_z[\log p(X,Z)|\theta]$$

Where θ is current parameters and θ' is the estimated parameters.

Incorporating the constraint $\sum_{i=1}^{K} w_i = 1$ using Lagrange multiplier gives

$$J(\theta',\theta) = Q(\theta',\theta) - \lambda\left(\sum_{i=1}^{K} w_i - 1\right)$$

Set derivatives to zero, we can get the update equation

$$Pr(i|x_j) = \frac{w_i \mathcal{N}(x_j|\mu_i', \Sigma_i')}{\sum_{k=1}^{K} w_k \mathcal{N}(x_j|\mu_k'\Sigma_k')}$$

$$n_i = \sum_{j=1}^{N} Pr(i|x_j)$$

$$\mu_i = \frac{1}{n_i}\sum_{t=1}^{T} Pr(i|x_j)x_j$$

$$\Sigma_i = \left(\frac{1}{n_i}\sum_{t=1}^{T} Pr(i|x_j)diag(x_j x_j^T)\right) - diag(\mu_i'\mu_i'^T)$$

$$w_i = \frac{n_i}{N}$$

3.2.2. Identify the speaker

After the speaker model is available, we can identify the speaker with the new original data. The new data will be pre-processed, extracted from the MFCC feature and compare with the saved models in database.

Suppose the set of speakers includes S people represented by S GMM models $\lambda_1, \lambda_2, \ldots, \lambda_s$. Target is to find the model for the highest priori probability with a new input, in detail:

$$\hat{S} = argmax_{1 \le k \le S} \Pr(\lambda_k | X) = argmax_{1 \le k \le S} \frac{p(X|\lambda_k)\Pr(\lambda_k)}{p(X)}$$

According to the Bayes law. Assume the probability of the speakers $\Pr(\lambda_k)$ are equal, due to the probability p(X) is the same with every speaker models, the formula above can be simplified as follows:

$$\hat{S} = argmax_{1 \le k \le S} p(X|\lambda_k)$$

In fact with many MFCC feature vectors extracted from an original audio sample, the speaker recognition system performs the following calculation:

$$\hat{S} = argmax_{1 \le k \le S} \sum_{t=1}^{T} logp(x_t|\lambda_k)$$

4. Implementation

4.1. Source code

- The whole system is written in Python. The system strongly relies on the support of python_speech_features, numpy and scipy library.

```
def get_feature(fs, signal):
     mfcc_feature = mfcc(signal, fs)
# get audio file
     if len(mfcc_feature) == 0:
     print >> sys.stderr, "ERROR.. failed to extract mfcc
feature:", len(signal)
     return mfcc_feature
```

- MFCC parameters:
    - Number of cepstral coefficient: 13
    - Number of filter banks: 26
    - The length of the analysis window in seconds: 25 milliseconds (Default)
    - The step between successive windows in seconds: 10 milliseconds (Default)
- We use GMM from scikit-learn with 13 components. The covariance matrix of every Gaussian component is assumed to be diagonal, since each dimension of the feature vector is independent.

```
def enroll(self, name, fs, signal):
     feat = get_feature(fs, signal)
     self.features[name].extend(feat)

def train(self):
     self.gmmset = GMMSet() #GMM set and scores acquired
     start_time = time.time()
     #train for each person
     for name, feats in self.features.items():
          try:
               self.gmmset.fit_new(feats, name)
```

```
            except Exception as e :
                print ("%s failed"%(name))
        print (time.time() - start_time, " seconds")

#get the models and identify the speaker
def predict(self, fs, signal):
        try:
            feat = get_feature(fs, signal)
        except Exception as e:
            print (e)
        return self.gmmset.predict_one(feat)
```

4.2. Dataset

- We record our own dataset with our own voices, out of 9 people, there are 2 females and the rest are males. The transcript of the dataset has about 30 lines of quotes from books. So it is mostly recorded in Reading speaking style. The average duration for all 30 lines of quotes is 150s.
- We do not use any Voice Activity Detection algorithms so certain noises are not filtered out.
- Bit rate: 705 kbps
- Sample rate: 44100 Hz
- Encoding: WAV
- Audio channel: mono

4.3. Testing and results

- The first 20 files to train and the last 10 files to test.
- The tests were all performed on “Style-Reading” corpus with 9 speakers, each with about 90 seconds for enrollment and 40 seconds for recognition.

- No filters applied.
- The accuracy obtained is as follows:

Speaker	Correctly identified as MFCC percentage (out of 10 files)
Nguyễn Thành Vinh	50%
Nguyễn Tiến Dũng	90%
Nguyễn Thành Gia Hiển	100%
Vũ Văn Huy	80%
Trần Thanh Long	80%
Trịnh Tuấn Minh	90%
Nguyễn Hồng Quang	100%
Nguyễn Phương Thảo	80%
Nguyễn Thị Lan Anh	90%

Overall accuracy: 84.4%

5. Conclusion

Speaker recognition has many applications in real life. Speech recognition is a long-standing research problem and there are many algorithms used in the speech recognition process.

The speech recognition method using the MFCC feature and the GMM modeling give relatively stable results with high accuracy, but the accuracy is easily affected by receiver quality and noise. Therefore, filters play a very important role to the accuracy of the algorithm.