EEC 201 Final Project Report

Andrew Fu

March 2025

1 Introduction

Speaker recognition is the process of automatically recognizing who is speaking on the basis of individual information included in speech waves. This technique makes it possible to use the speaker's voice to verify their identity for authentication of access.

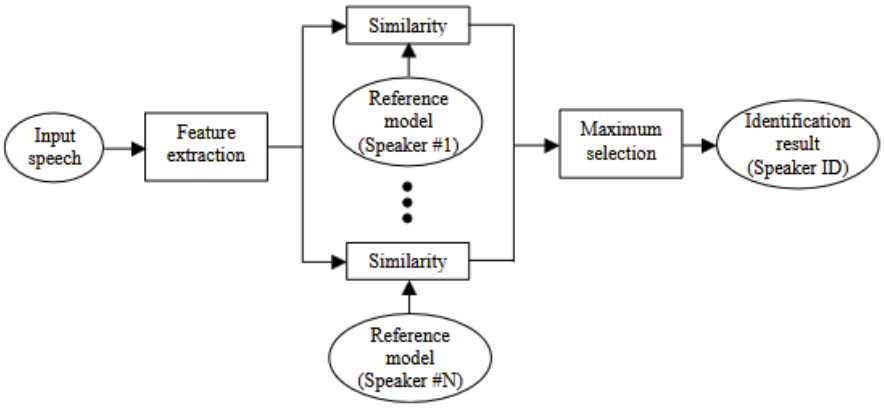

Figure 1: Basic structures of speaker identification systems

This project builds a simple and representative automatic speaker recognition system. This project report details the tests performed on all components of this speaker identification system during its design and for the system as a whole.

2 Speech Data Files

The project uses four separate speech datasets. The "original set" consists of eleven training and eight testing speech files for the word "zero." Two other datasets are recordings from EEC 201 students, one from 2024 and the other from 2025. The 2024 recordings comprise 18 pairs of training and testing files,

the two words “zero” and “twelve.” The 2025 recordings are similar in format to the previous year’s dataset, consisting of 23 pairs of the words “five” and “eleven.” The final dataset contains a distorted variation of the original dataset, with its recordings’ most prominent frequency bands cut off by notch filters. It also contains five pairs of “Additional Voices” files used in extension to the original set. The entire system test uses parts of the whole of a dataset or a combination of files across multiple datasets.

All of the component tests were conducted using the “original set.” My personal recognition rate for this set is 100%.

3 Speech Preprocessing

The first processing block of this speaker recognition system is interfaced directly with a digital continuous speech signal. An example of such a signal is shown below.

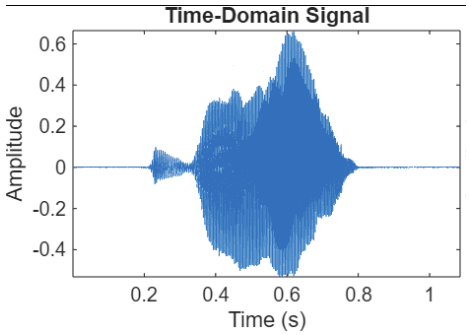

Figure 2: Speaker 2’s Time-Domain Signal

The system aims to perform feature extraction by first processing the frequency components of the entire signal in “frames.” Different frame sizes of a signal’s periodogram are shown below. A typical implementation uses blocks of 256 samples overlapped by 156 samples.

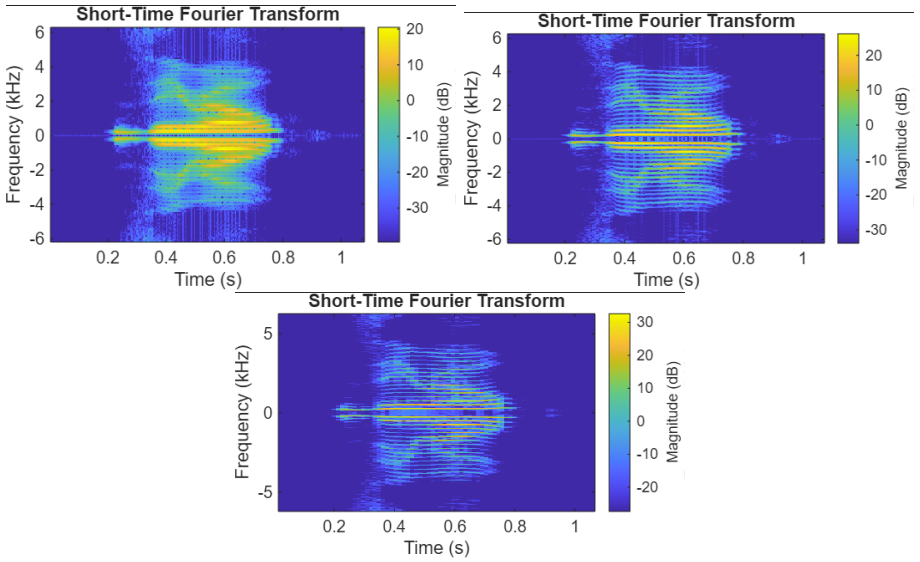

Figure 3: Speaker 2's Periodogram; N = 128, 256, 512; M = N/3

The system doesn't directly perform feature extraction with a signal's blocked frequency components. Instead, the input signal is converted into Mel-frequency cepstrum coefficients (MFCCs).

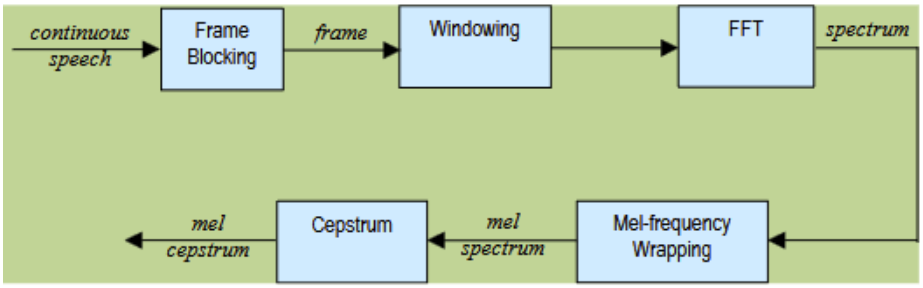

Figure 4: Block diagram of the MFCC processor

Mel-frequency wrapping is achieved by filtering the input signal by a Mel-spaced filterbank. A typical implementation uses 20 filters.

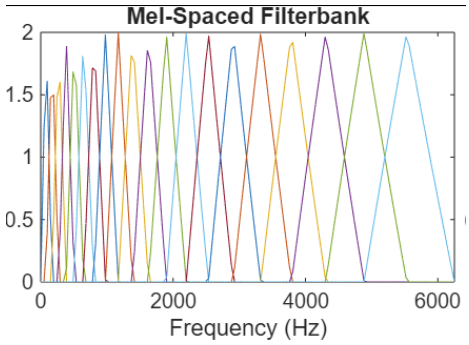

Figure 5: Mel-spaced Filterbank; K = 20

The limited resolution of the filterbank introduces minor differences between the simulated and theoretical responses. The frequency bin containing the maximum amplitude of each bandpass filter lies always lies between two of the closest bins in the simulated response, thus an amplitude of two is never achieved, and the filter isn't perfectly triangular.

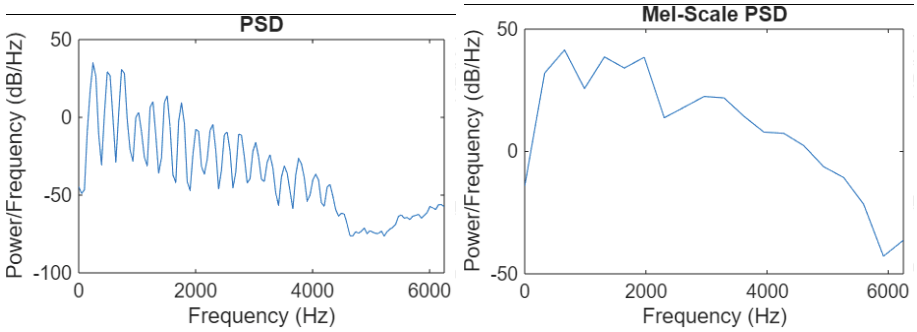

Figure 6: Speaker 2's Spectrum Before and After Frequency-Wrapping

When operating on the power spectrum of an input signal, the Mel-spaced filterbank warps the frequency response by overrepresenting low-frequency components. It also compresses the frequency response from 129 points to 20.

The collection of all resulting MFCC coefficients for each frame in the input signal represents the speaker's unique speech features. The next step of the speaker recognition system is to perform feature matching using MFCC vectorized data.

4 Vector Quantization

Feature matching operates under its best conditions when each set of vectors is spatially distinctive from another set.

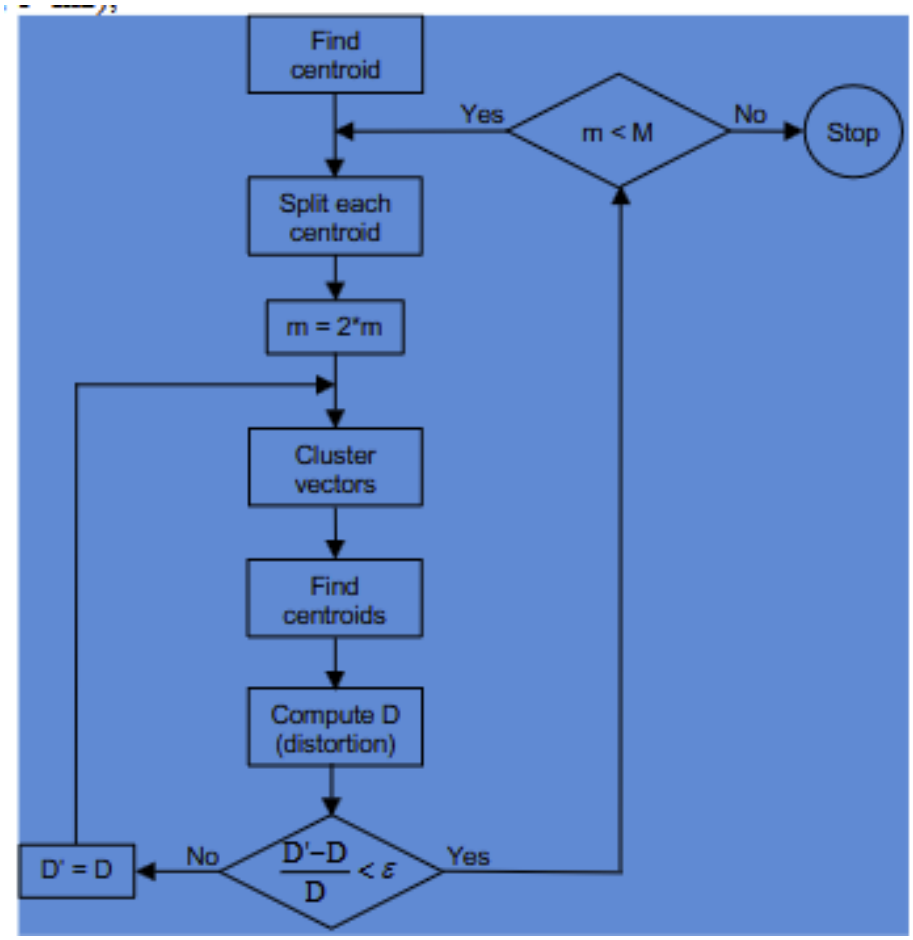

Figure 7: Flow diagram of LBG algorithm (Adopted from Rabiner and Juang, 1993)

While performing feature matching directly with MFCC vectorized data is possible, it is undesirable because of large computational cost. The data can be compressed by performing clustering, and this system uses the LBG algorithm to achieve this.

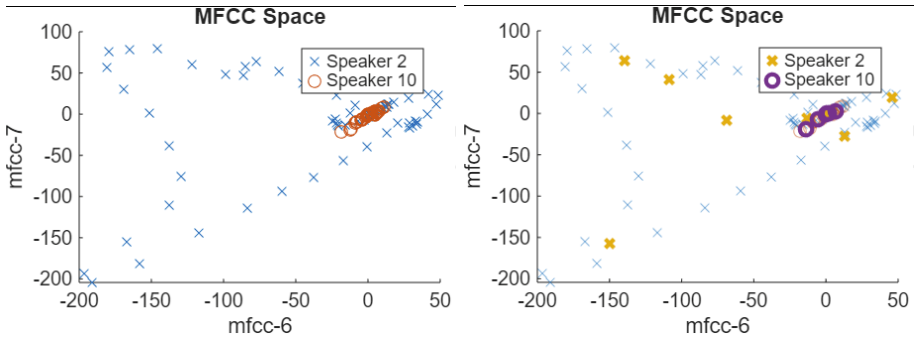

Figure 8: MFCC Space for Speaker 2 and 10 in MFCC-6 and MFCC-7; M = 8

In some dimensions, the vectorized data forms clusters. In Figure 7, speaker 10's MFCCs form clusters in MFCC-6/7, causing its centroids to group closely together, whereas speaker 2's MFCCs are spread out very far.

The system detects for similarity between the training and testing signals by identifying the codebook ID with the minimal VQ-distortion.

5 Full Test and Demonstration

The MFCC processor uses windows with a frame size of 256 and an overlap of 156 samples, as well as 20 filters in its Mel-Spaced filterbank. The LBG algorithm makes a codebook of size 128 and epsilon set to 0.01. The full test of this speaker identification system is shown below.

Data Set	Accuracy (%)
Original Set	87.5
Additional Voices	NaN
Notch Filters	62.5
Augmented Set	77.778
2024 0 Set	55.556
2024 12 Set	72.222
2024 Full Set	61.111
2025 5 Set	78.261
2025 11 Set	78.261

The system is somewhat intolerant to noise/distortion, as we observe a 25% drop in accuracy when the original set of testing signals were passed through a notch filter.

The accuracy in identifying "zeros" in the 2024 set is particularly low (55.6%). This causes accuracy to drop by 9.7% when it is added to the original set.

The accuracy in identifying "twelves" in the 2024 set is 16.7% higher than "zeros." Due to this, the system is able to more accurately distinguish a speaker saying twelve in the full 2024 set.

The system is more accurate when it uses "five" or "eleven" instead of "0" (+22.7%) or "twelve" (+6%).

6 Conclusion

To identify the speaker, this speaker recognition system uses various DSP tools to extract a set of features for analysis. Across all data sets tested, the system achieves an overall accuracy of 71.6%.