Speaker Identification Using Mel Frequency Cepstrum Coefficients

Implemented using a RaspberryPi platform

Iby and Aladar Fleischman Faculty of Engineering Tel Aviv University

הפקולטה להנדסה ע"ש איבי ואלדר פליישמן אוניברסיטת תל-אביב

Project Number: 17-2-1-1474

By: Guy Eliyahu, Matanel Hamiel Advisor: Roi Raich

Project Carried Out at Tel Aviv University

BACKGROUND

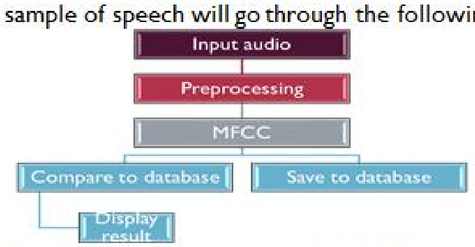
- Voice or speaker recognition is the ability of a machine or program to receive and interpret dictation or to understand and carry out spoken commands.
- Voice recognition has gained prominence and use with the rise of AI and intelligent assistants, such as Amazon's Alexa and Apple's Siri
- Speaker identification is a sub-field of voice recognition and refers to identifying the speaker, rather than what they are saying.
- Cepstrum is a simplified method for speech processing and described as performing an inverse Fourier transform on logarithm of the absolute value of the sample spectrum:
- Mel Frequency Cepstral Coefficients, or MFCC, is an improved method of speech processing, considers the un-linear Human perception of speech. In this method, the speech spectrum is filtered by Mel filter and the frequency axis is scaled into Mel scale.

PROJECT PURPOSE

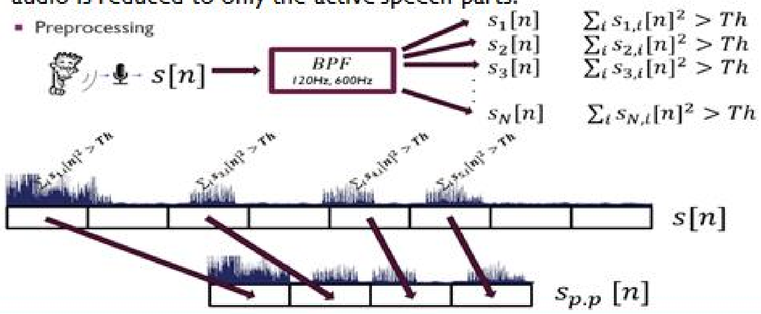
- ✓ Writing a MFCC based algorithm for speaker identification in MATLAB.
- ✓ Implementing the algorithm on hardware creating a stand alone system for speaker recognition.

ALGORITHM

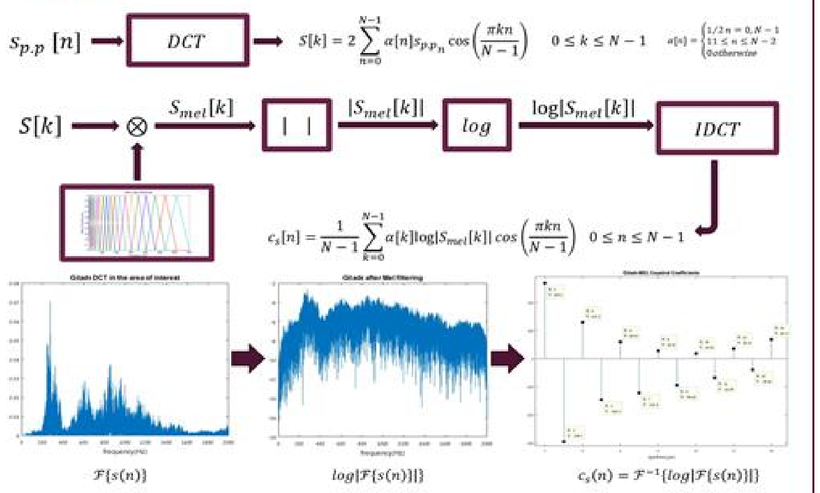
- A inputted sample of speech will go through the following process:



- The inputted audio goes through the **preprocessing** stage, where the audio is reduced to only the active speech parts.



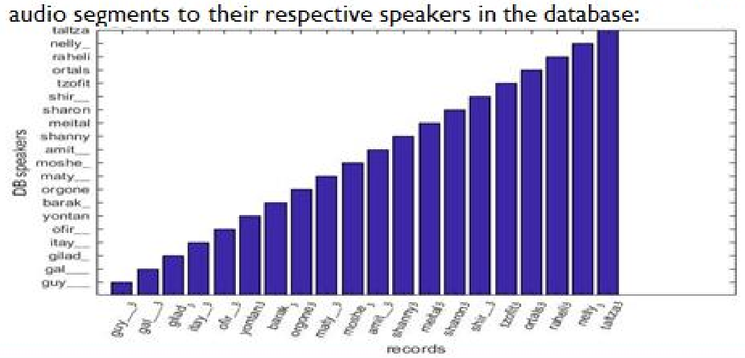
- Then, the preprocessed audio goes through the stage of computing it's **MFCC**:



- After preprocessing and computing each of the frames' MFCC, an audio **inserted** to database will go through the left branch process, while an audio **compared** to database will go through the right branch process:

RESULTS

- Our MATLAB program was 100% successful for matching 20 inputted audio segments to their respective speakers in the database:



HARDWARE IMPLEMENTATION

- The algorithm was re-written in Python to work on a Raspberry Pi platform as a speaker identification application.
- Using Python's "TKinter" library, a user-friendly GUI was designed:

- A size adjustable GUI was designed to fit a 3.5" screen, creating a stand alone system:

CONCLUSIONS & FUTURE WORK

- The algorithm works for basic scenarios.
- Future projects may contain noise reduction and runtime reduction making the application more efficient.
- Future projects may contain other voice-processing algorithms such as Linear Prediction Cepstral Coefficients and Machine Learning for performance comparing.
- Future projects may contain work on an embedded system such as STMicroelectronics's STM32.

SOURCES

- Joseph P.Campbell "Speaker Recognition:A Toutorial", Invited Paper.
- Ahmad Al Marashli and Dr Oumaima Al Dakkak, "Automatic, text_independet, Speaker Identification and Verification System Using Mel Cepstrum and GMM".
- Ladde L Molgaard and Kasper W Jorgensen "Spekaer Recognition", special course.