Paul Globisch

pglob2@pdx.edu

Detecting Voices Using Phoneme Features

Project Description

For this project, I will be implementing a voice detector that takes a stream of microphone input and writes to a file when voices are present. My main idea is to try to detect phonemes present in the input and use that to decide if voice is present.

In using phonemes, I think my implementation will be more selective and have fewer false positives. I recognize that phoneme detection gets complicated quickly, so I will be focusing on just vowels, plosives, and fricatives. I chose these because I believe they will be easier to detect.

The detector will do the following:

1. Select a relatively large window to search for phonemes.
2. Using a bandpass filter, select frequencies in 50Hz-3,000Hz. The first three formants of vowels tend to fall in this range, which should help with detection. This range is subject to change of course.
3. Normalize the input amplitude. Since I'm using phonemes, I probably won't be able to run the program in real time, so might as well normalize.
4. Run input through an FFT to get the frequencies present. Using this, the detector will check for a fundamental frequency in the human vocal range and formants to find vowels.
5. In the time domain, check for brief periods of silence followed by an impulse to try and find plosives.
6. Check for short segments of high-frequency noise to find fricatives.
7. Using potential phonemes, the detector will try to match them to English speech patterns.

8. Write the input to a file if the phonemes line up well.

This project will require a great deal of tweaking different parameters to detect voices, which is why I have not provided specific numbers. Trying to find specific numbers through research also proved very difficult, probably because of how variable the human voice can be. If more accuracy is needed, I can also incorporate zero crossing rate.

For testing, I was considering finding a database of tagged audio samples and running the detector on those. Since audio samples in a database are probably clear and distinct, I would like to semi-randomly mix some of them and add various noise patterns.

Discussion of Issues

Since I am using phonemes, the detector may have trouble detecting voices if there is a lot of background sound. I am not sure how distinct they will be from the rest of a messy waveform. If needed, I will do more research on removing some unwanted sounds.

In addition, I am focusing on vowels, plosives, and fricatives, so some words (like “man” with 2 nasals and a vowel) may not trigger the detector. Extending that, non-English languages that have different phonemes likely will not work as well. I am hoping that the vowel detection also picks up on some of these phonemes, although that may be too permissive with non-voice input.

Formants are needed to detect vowels, meaning I need a relatively wide bandpass filter. This will also make the implementation harder since I cannot filter out as much. I am hoping the additional check for fundamental frequencies in the typical human range will assist with this.

As stated earlier, I likely won’t be able to run the detector in real-time. This will provide some leniency in window sizes for checking phoneme patterns

Github URL

https://github.com/pglob/CS-410_Audio_Project/