Emma Wenger
16 Apr 2019
15-112 TP Spring '19

TP1: Project Proposal

Project Description

My project, Violin Hero, will be an interactive application that aids in the practice of music, specifically for the violin (and other chromatic instruments). The user will upload an image or PDF of their sheet music and then play it into their microphone. The application will let them know how they are playing, and then generate a report providing an overall grade, sections to focus on, and specific notes that were played out of tune. In addition, there is a gamified option, where the musician can learn their music in a guitar hero-like fashion. This makes sight reading enjoyable while still helping the musician learn new music.

Competitive Analysis

There are a number of existing games that have come out similar to guitar hero but that help musicians learn real instruments. However, these games are often pricey, require specific equipment, and/or don't allow the user to upload their own sheet music that they have to learn.

Somebody has already made a version of guitar hero as a 112 term project[i], so it was interesting for me to see how they implemented the graphics using tkinter. I might have a similar aesthetic in my game mode. However, my project is quite different from this as I'm detecting the audio coming in rather than key pressing.

In addition, there exists software that will tell you if you are on pitch, such as Tunable[ii], but this doesn't have all the features that I am developing, such as the gamified sight reading and the uploading of sheet music from image.

My project, therefore, is more well-suited to the already established musician who is practicing a piece that they have been assigned and are reading as sheet music. It makes the initial sight reading of the piece much more enjoyable with the game format and also helps an accomplished musician to finetune their performance with the pitch feature.

Structural Plan

The back end will include one main function broken down into several helpers. Both the pitch and game mode will rely on this main function:

1. Comparison of live audio with sheet music
    a. Conversion of jpg sheet music to a list of pitches/times
        i. Conversion of jpg sheet music to a MIDI file
        ii. Conversion of MIDI to wave
        iii. Conversion of wave to list of pitches/times
    b. Detection of pitch/time from live audio
    c. Number crunching

The gui will include:

1. Tune mode
    a. Visualization of sheet music
2. Game mode
    a. Visualization of notes in real time, guitar hero style
3. Help

a. Explanation of project

Algorithmic Plan

The trickiest part of the project is easily converting the jpg sheet music to a list of pitches and corresponding times. After extensive research, I have found SheetVision on github, which utilizes OpenCV to detect notes along with flats and sharps and converts this information to a MIDI file. From there, I can utilize another library called midi2audio, to convert the MIDI file to a wave. Finally, I can use a library called aubio to convert wave to a list of pitches and corresponding times and write this information in a txt file. I can then read this file in python. Although this approach will work, it is quite roundabout and will likely take a long time to process, meaning a long loading time. I will look into modifying the SheetVision code so that it outputs a list of pitches and corresponding lengths, rather than a MIDI. In addition, I know that I will have to implement computer vision of rests, as SheetVision does not come with this feature. This will be a challenge for me that I'm hoping I can achieve by reading and understanding the computer vision processes used for the rest of notes in SheetVision.

As for the detection of pitch and time from live audio, I will be able to use aubio, which will allow me to detect the incoming pitch as it is played through the microphone on the device being used. By comparing these incoming pitches with the expected pitches I calculated earlier in my txt file, I will be able to generate statistics on how well the musician is playing.

Timeline Plan

1. Tech demo – show that I can detect pitch
2. TP1 – able to detect pitch and read from PDF, tell if same
3. MVP – graphics and system working with game implementation
4. TP2 – project complete, extra features added as time allows
5. TP3 – project complete with full user interface, organization of code

Version Control Plan

I will be using github for version control. My repository is located at: https://github.com/ejaynew/112tp.7yn

Module List

I will be using a number of resources from github including:

1. SheetVision
2. MidiUtil
3. Aubio

In addition, I will be importing a number of python libraries including:

1. Opencv_python
2. Matplotlib
3. Numpy
4. Pdf2image
5. Midi2audio
6. Pygame

[i] https://www.youtube.com/watch?v=WBJNbtvHPTo

[ii] https://www.fastcompany.com/1672227/an-ipad-tuner-visualizes-your-pitch-in-real-time