Mid-project Check-in

We are going with the updated MVP for MP4 – making a piano that would interact with the user through computer vision. The computer will read the keys that are being touched by the user on a printed version of a piano and play the note as an output.

The current status of the program is that we have separate files that do each of the processes that are necessary for the piano, but we are yet to integrate it to see if these pieces of code work together – we expect it to need a bit of debugging and moving around.

For the music section, we currently have simple text input in lieu of user interaction input. We have one octave available to play (C4 through C5), and sharp and flat options. The Music21 package makes producing music through Python simple, but it is necessary to download a MIDI player. We recommend using Timidity. Keeping the code to one octave will make interactivity simpler for users.

The implementation plan for the rest of the project is as follows:

-Finish polishing the music program. Currently it works, but can be improved.

Estimated time: To be finished after user interactivity. Due: Monday

-Focus on user interactivity. Currently the code recognizes user movement and color, but not if a user pushes on the ‘keys.’ We need to focus on this aspect.

Estimated time: Due by this Friday.

The classes that are present in the current program are:

1. Piano (Note Range) – all the notes present on a standard grand piano
2. Notes – Combination of notes and registers that make up the whole range
3. Keyboard Input – Reading the user input through the keyboard for writing sheet music
4. CV - Reading the user input through the camera for writing sheet music

The libraries that we are currently using:

1. OpenCV
2. PyGame
3. Music21

The packages that we are currently using:

1. Timidity
2. Scipy
3. Numpy
4. Sys
5. Warnings

For this week, we want to work on the integration of the modules and the OpenCV input (it still contains some bugs – there is still some confusion about the user input section of the code)

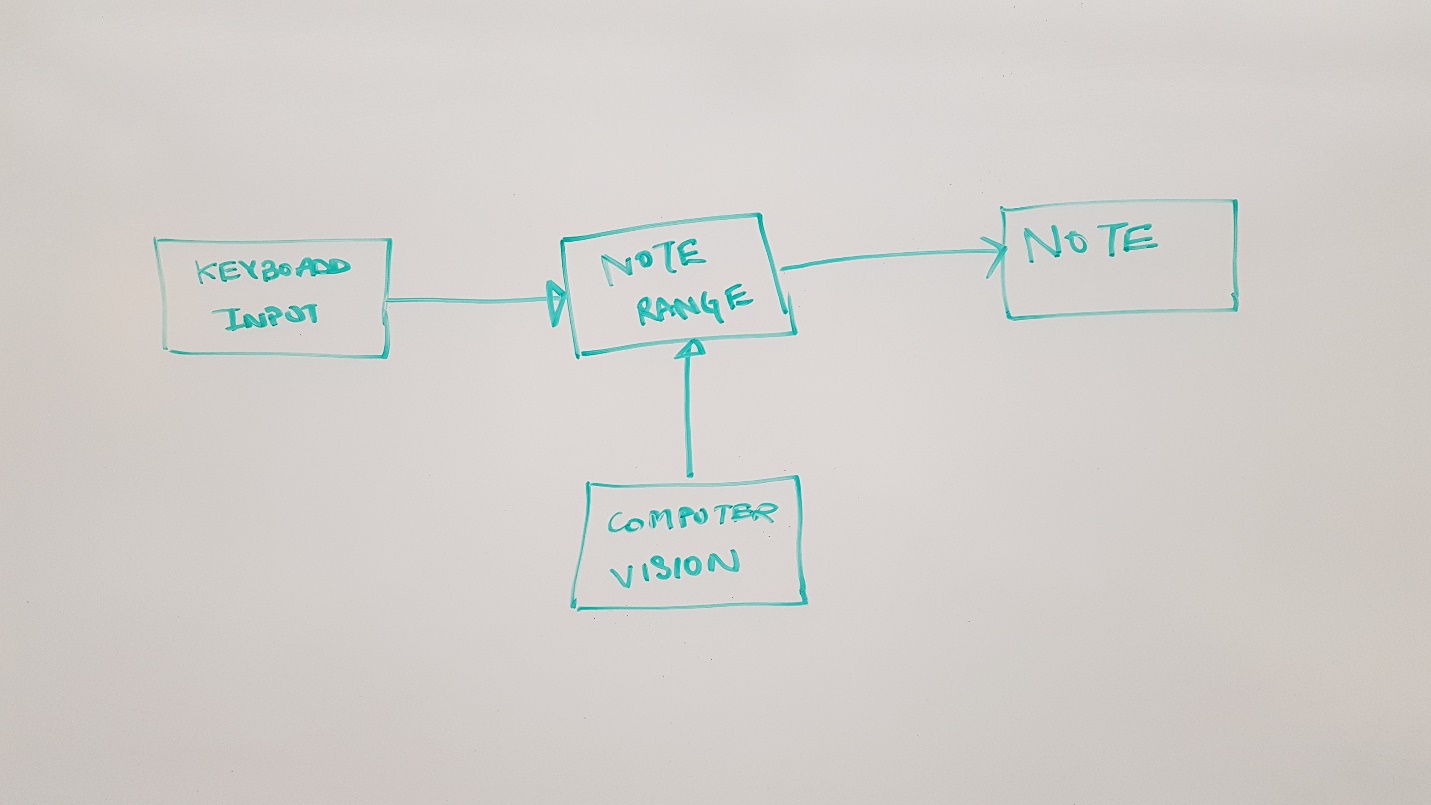


Figure 1: UML Diagram for the program