Journal Report 3
9/16/19-9/20/19
Ajit Kadaveru
Computer Systems Research Lab
Period 4, White

Daily Log

Monday September 16

Researched more on how to plot the cube. Found a github that had 3-d matplot interactive display of cubing. Tried downloading it, but couldn't get it to work on my laptop. It also had a lot of code, which would require me to understand all of it and tweak it for my purposes. Because of this, I decided to use other graphics methods. I stumbled upon PyGame, which seemed good for drawing.

Tuesday September 17

Planned out how the GUI would look. Drew it out on paper with the coordinates of each important point (corners of squares) to refer to when coding each filled square. Installed PyGame, and coded the coordinates for each of the centers, and the rest of the white side. After testing it, I realized it worked but I couldn't do live display with PyGame. So did more research, and found Tkinter. I remembered Tkinter from AI, and remembered that it worked for live display and decided to go with that.

Thursday September 19

I finished calculating the coordinates for all the corners (around 54 of them). Then I refreshed my memory with tkinter, on how it can live update, and how it works in general. I coded the display for the white side, so that whenever I input moves, it updates the white side in the GUI. This involved calculating the coordinates of two of the corners for each square, and inputting them as parameters into Tkinter's create rectangle method. I'll finish the updating code for the other 5 sides next class.

Timeline

Date	Goal	Met
Today minus 2 weeks	Organize the cube data structure and program all the possible moves	No, I organized the data structure and only programmed R and R' methods. Didn't expect the indexing to be as complicated as it was since things aren't symmetric when unfolding 3-D into 2-D
Today minus 1 week	Finish programming all the moves and figure out a way of displaying things live in python	Yes, finished programming all the moves, and started to do research about displaying cube live in python.
Today	Make the T-display of the cube with colors and change live as you enter move	No, encountered some problems with live display, as PyGame didn't work. But now it's all good since Tkinter seems to work well. I coded the white side to work (see picture below)
Today plus 1 week	Finish updating the GUI for the cube moves so that the live T-display works for the whole cube. Review edge detection in opencv, and upload a sample image of me holding a cube onto the python program to test the edge detection.	
Today plus 2 weeks	Begin coding edge detection to eventually determine which frames of the video the cube are in the shape of a cube (not in the middle of turning),	

Reflection

Before this week, I didn't expect transferring the output to a GUI display to take that long. However, I realized that I'd have to refer each of the squares to specific coordinates on the GUI. I also encountered trouble with the live display. For the purposes of this project, the display needs to be live, rather than closing and restarting the program for a new GUI every time the cube moves once. I couldn't get PyGame to work with this, but Tkinter seems to work well. Below is an image of the GUI display working for the white side.

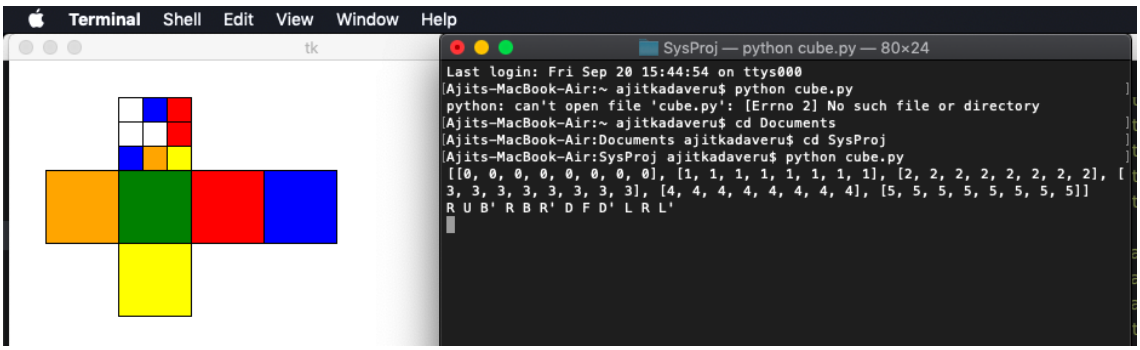

Figure 1: An image of sample scramble of cube