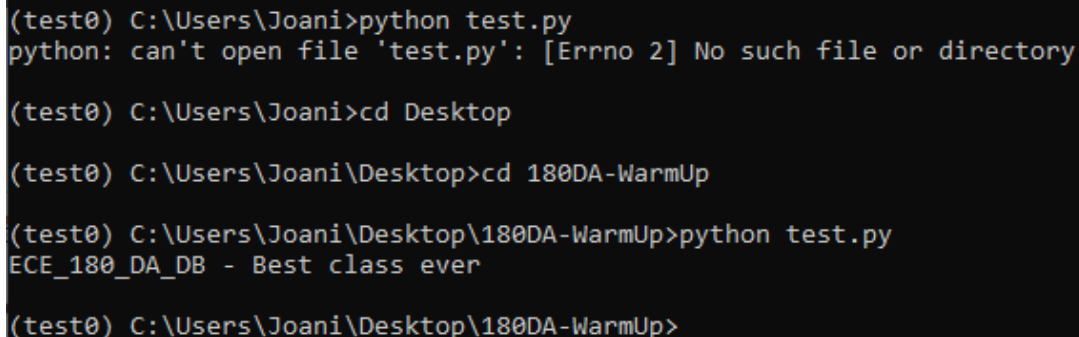


Week 0 – Lab Report

1. What did you plan to do last week?
 - Setup a Github Repo for this class
 - Install/Update Anaconda
 - Create a virtual environment
 - Install OpenCV
 - Figure out how to use the camera in a python script
 - Track an object based on its color with OpenCV
 - Find the dominant color in a set rectangle
 - Brainstorm ideas for project
2. What did you end up doing last week?
 - **Task 1:** I setup the Github repo on both my Mac and Windows, using Github Desktop, so it will be very easy to pull and pull from the repo.
 - I already had Anaconda downloaded so I only needed to follow the tutorial on updating it using Terminal
 - **Task 2:** I created a virtual environment using the instructions in the tutorial and added the test.txt file to it, then committed it.
 - **Task 3:** I installed OpenCV using the instructions in the tutorial. I then copied the given code in test.py and ran it. Screenshot shown below:



```
(test0) C:\Users\Joani>python test.py
python: can't open file 'test.py': [Errno 2] No such file or directory

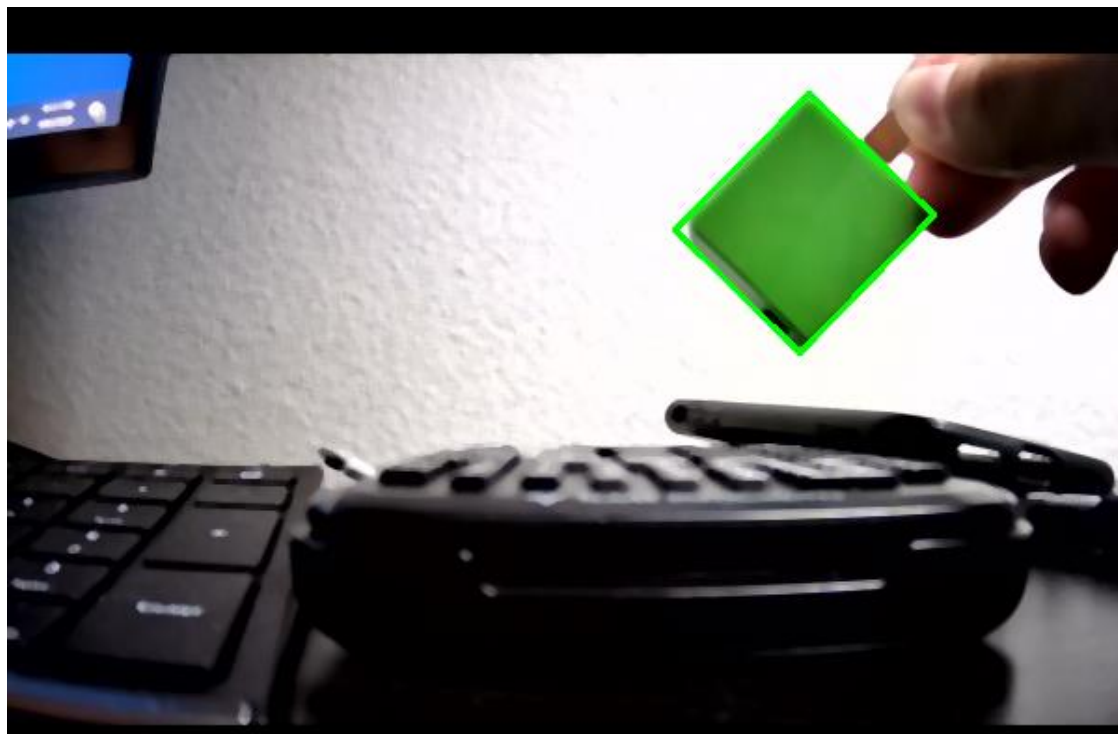
(test0) C:\Users\Joani>cd Desktop

(test0) C:\Users\Joani\Desktop>cd 180DA-WarmUp

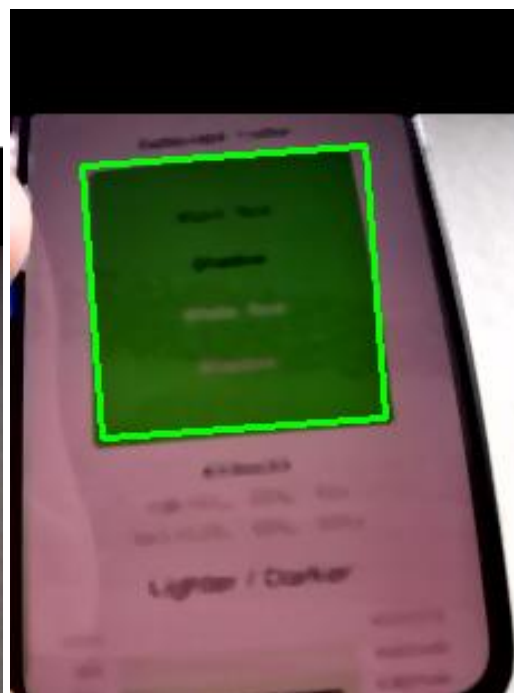
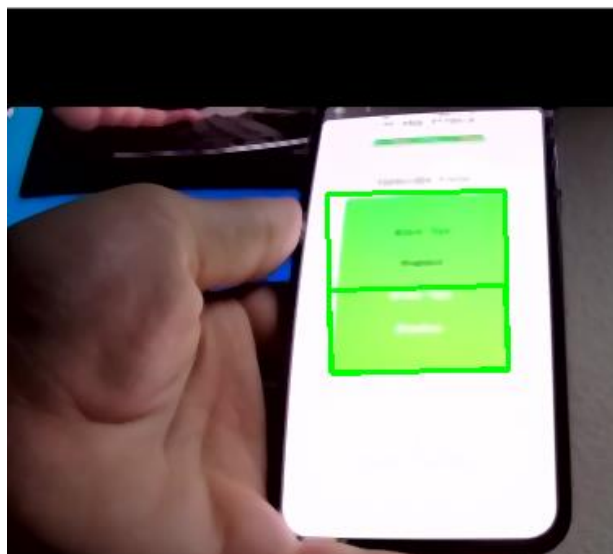
(test0) C:\Users\Joani\Desktop\180DA-WarmUp>python test.py
ECE_180_DA_DB - Best class ever

(test0) C:\Users\Joani\Desktop\180DA-WarmUp>
```

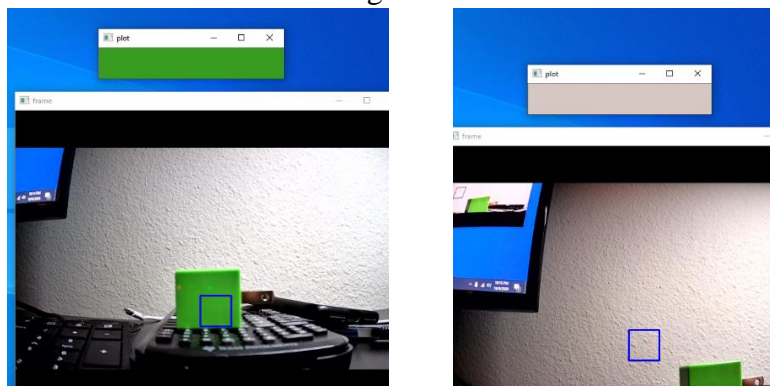
- Getting the camera to work was a bit of a challenge, but I was able to have a good setup for my code where it analyzed each frame as an image using a while loop.
- I followed the different tutorials on converting color schemes, thresholding, edge detection, template matching, and bounding box.
- **Task 4:**
 - *Part 1:* I chose a bright green power brick used to charge phones as my object, as it provided a unique color compared to my surroundings and was a square shape so it would be a clean object to track. I created a video stream where a green object is tracked when it is in the frame by displaying a box around the object. I used both HSV and RGB, but HSV was much easier to use as for RGB you must worry about both R and B values, so it becomes really complicated to figure out. I used an HSV chart online and saw the range for green for be from 40 to 70, but after some trials, I narrowed my shade of green to 50 to 70. The other two parameters weren't that important as the first one for HSV.



- *Part 2:* I changed the lighting condition by turning off my desk lamp, which was located near my laptop and only kept my bed-side lamps on. There was a major difference as it was now much darker, but because there was not a bright light right in front, my objects color remained the same, so it still tracked it well. Previously, the light would reflect off my power brick as it was shiny so it would look white on camera at certain angles.
- *Part 3:* I found a random color picker on my phone and pulled up the color green. If my brightness was high, the code would have trouble drawing an accurate rectangle around the box. When I put the brightness down, the code drew a proper box, so the lowering brightness helped the detection.



- *Part 4:* The dominant color code was fairly simple. I followed the tutorial and then just changed the plot function to determine the dominant color by going through the np array of colors and seeing which had the largest length on the histogram. I would then display that color on the entire histogram.



3. This week I plan to keep brainstorming ideas for the project as well as completing the new tasks in the lab section. Brainstorming, however, is the most important part because some of my partners are in very different time zones so it is a little difficult to coordinate.