Week 1 Lab Reflection

This week in lab, we focused on learning how to use GitHub, Anaconda, and OpenCV.

The GitHub portion of the lab wasn’t too difficult for me as I have some prior experience pushing and committing commits on a shared repo as part of other classes and clubs. The hardest part was setting up the remote on the local computer and making sure GitHub authenticated correctly, as I set up local repositories very infrequently. A common issue I get is my computer trying to push to “master”, which raises errors as GitHub has migrated to using “main” instead. Luckily, the lab introduced SourceTree to me, which made setting up the local remote very easy. SourceTree, which provided a GUI for GitHub, made setting up the remote easy as all I needed to do was specify the local path where I wanted the remote to be and the repository link. SourceTree handled the rest of the linking and cloning for me which was very nice. I created commits and pushed code through the command line as that’s what I’m familiar with.

Setting up Anaconda was a little more difficult. Anaconda is a package manager that allows you to create virtual environments to easily manage multiple projects with potentially conflicting packages. I have used Anaconda in personal projects before, so setup wasn’t completely foreign. However, the “update conda” command was tricky as it look ~15 minutes to complete without many visual updates on progress. In the end, I was able to successfully install Anaconda itself along with a virtual environment with the numpy, matplotlib, and pandas Python packages.

The OpenCV portion of the lab was pretty challenging as I don’t have any experience with OpenCV. A big part of the learning curve was learning how to correctly utilize HSV values, as I have only worked with RGB before. With the color tracking assignment, I heavily utilized the OpenCV website, where I learned to establish a link with my webcam, threshold with the inRange function, and finding contours. I looped through the found contours and only plotted the one with the largest area. Additionally, I only plotted it if it was greater than a specified area, in my case 1000 pixels. I found that thresholding with HSV worked far better than BGR. For the K-means assignment, I heavily utilized the K-means tutorial from likeagirl.io. I added in a bounding central rectangle with my knowledge from part 1 and cropped the sample area to that bounding central rectangle. A challenge was getting the figure to update, which I achieved by creating a matplotlib figure object and calling plt.ion() to make it interactive. I found that my phone was much more resistant to lighting changes for the K-means tracking than my non-phone object, which in my case was my blue AirPods case.

Overall, I feel confident in my ability to use Git and Anaconda and can definitely learn more about OpenCV.