Scott Dudley

**Sprint 5 Research/Report:**

***Connecting to Our Router (4 hours)***

We ran into several problems during Sprint 5 moving forward towards the autonomous programming. First of all, our movement testing from last sprint was done only on one robot. Also, the second robot had not yet been connected to the router that we had been using for the first robot. To connect to the router, I had to see what was on the raspberry pi. Unfortunately, the HDMI port on the pi was blocked by the servo for turning the wheels on the robot, so I had to take apart the front half of the robot to configure the connection to the router. I connected the robot to our router through Raspian once I could get to the HDMI port. Afterwards, I saved the IP address so that I may connect to the pi via VNC rather than HDMI when on the router. Before I put the rest of the kit back together, I made sure to change the raspberry pi’s network protocol so that it prioritized connecting to our router when starting up. Now the pi will connect to the router first when the both the system and the router are turned on, and I can connect to the pi via VNC once it has turned on and connected to the router. The sources for the research of changing certain raspberry pi files to configure its network are provided below:

Sources for Solving Network Issues:

<https://www.raspberrypi.org/forums/viewtopic.php?t=121934>

<https://raspberrypi.stackexchange.com/questions/58304/how-to-set-wifi-network-priority>

***General Code Debugging (5 hours)***

Now that I could connect to the router, it was time to test if the raspberry pi could follow the same code as the other one to test if something was wrong with this pi. The server would run normally, and the client app started off running normally. Unfortunately, trying to select the robot after connecting to it crashed the application. It took awhile to figure out the problem, but it ended up being a combination of not having the latest code on GitHub and needing to update the NuGet packages in Visual Studio on the client-side. There were also a few problems with getting the movement of the second robot to work with the code. It turned out that trying to turn left or right ended up turning the camera. I realized that this was just a simple issue of the servos’ order of wiring being reversed. After switching the wiring around, the turning worked perfectly from the program. However, trying to move forward and backwards crashed the program. After a bit of debugging, I realized that I had an outdated version of the driver we were using for the car. This fixed forward motion but not backwards. The issue with backwards turned out to be a simple syntax error where the client was trying to execute a call to ‘moveBackward’, while the actual name of the method was ‘moveBackwards’. A simple change of the syntax, and the robot was fully functional. Later on in the sprint there was a problem with the program crashing, but it was due to the camera trying to call both image recognition and video streaming code at the same time. I helped try to solve a few of the issues by finding where the errors were occurring, but the rest of the group did most of the work fixing the issues, since they were the ones working on the desktop app, image recognition, and video streaming. We eventually got everything meshed together correctly to demo image recognition and video streaming in our Sprint 5 presentation.

***OpenCV Installation / Configuration (4 hours)***

I installed and configured OpenCV on the second Raspberry Pi so we could start testing the Image Recognition code. The lengthy but straightforward guide for this is provided below:

Source for Installing OpenCV on the Raspberry Pi:

<https://www.pyimagesearch.com/2017/09/04/raspbian-stretch-install-opencv-3-python-on-your-raspberry-pi/>

***Continued Work on Final Documentation: (5 hours)***

I continued to work on the Final Documentation by completing the first section, most of the second section, and most of the third section. This work is shown in pages 1-10 of the Final Documentation.docx file provided in my submission.

***Total Work for Sprint 5: 18 hours***