

Team 22

Project Title: Rabbit

Date: 10/28/2021

### Member role this week:

- Samuel Fuller – OpenCV and lane detection
- Ben Dunkerton- Getting the Jetson Nano running lane detection
- Marshall Boser- Make a test application for iPhone
- Lars Lofquist- Gearing the car to go the required speed
- Darron - Work with the Jetson Nano and computer vision
- Kevin Scanlon - OpenCV for computer vision
- Darrshen - Lane detection filtering

### What we've accomplished in the past week/what we've been researching

- Samuel Fuller – researched lane detection algorithms
- Ben Dunkerton- set up the Jetson Nano
- Marshall - worked with car control and the App
- Lars - Feather microcontroller is able to control both car steering & driving
- Darron - set up the Jetson Nano
- Kevin Scanlon - Researched computer vision
- Darrshen - researched filtering techniques for lane detection

### What we're planning to do in the coming week

- Samuel Fuller – further research and meet to implement line detection
- Ben Dunkerton- Meet and start implementing lane detection on the Nano
- Marshall - Create a test App that can connect to the car with Bluetooth
- Lars - Research and order parts for gearing the car correctly
- Darron - Meet and create a lane detection program
- Kevin Scanlon - Meet and start implementation of CV
- Darrshen - Meet and work on line detection

## Issues we had in the previous week

- Samuel Fuller – Simple lane detection for straightish lines might be a simple problem because of the generally controlled scope of our project (being on a track). The curves of the track may or may not be a larger problem, needs to be tested.
- Ben Dunkerton- Powering the nano on the car needs to be looked into.
- Marshall - An apple laptop is needed to flash our React App to an iPhone
- Lars - I can run faster than the car at max speed.
- Darron - How do we use the data after “lane detection” to control the car?
- Kevin Scanlon- How do we port the camera data into our program so it's usable.
- Darrshen - How can we filter out “noise” from our camera