

1 Introduction

This report will outline the activities carried out in the first month of the competition. A remote connection with the car was established. We managed to control the vehicle remotely and stream camera output on our laptop. On the other hand, we still haven't started the simulation since we don't have a computer with enough RAM, but we are working on getting one. We started working on lane detection and successfully detected lanes on an example video on PC.

Team spirit is generally good; we overcame several obstacles by working together.

2 Planned activities

- 1) Control the physical car remotely and stream camera output on a remote device (laptop)
- 2) Control the virtual car on the simulator
- 3) Members' tasks assignation
- 4) Lane detection
- 5) Creating/defining physical testing environment

3 Status of planned activities

3.1 Control the physical car remotely and stream camera output on a remote device (laptop)

Status: completed

Implementation: Flashed Nucleo and installed Raspberry Pi 4 OS. Uploaded Brain project code on the Raspberry Pi and connected with Raspberry Pi remotely.

Difficulties: problems with connecting to Raspberry Pi remotely and obtaining a monitor to access Raspberry Pi UI

Ongoing potential problem: we are hearing a pitching noise coming out of the car when it's moving, and we at first thought that the problem was the battery wasn't charged enough, but now we think it's something else

3.2 Control the virtual car on the simulator

Status: Ongoing 25%

Implementation/difficulties: we ran the simulator on a laptop with 8GB and 12GB of RAM, but the textures weren't loading, so we are trying to get another laptop with much more RAM

3.3 Members' tasks assignation

Status: completed

Members were assigned preliminary roles.

- Aleksa Madzarevic – physical environment/lane following and speed control
- Jovan Markovic – software architecture and communication between packages
- Ognjen Aleksic, Tanja Mikovic – lane detection, camera handling, ROI definition
- Anja Vujacic - installing the virtual test environment, lane detection

3.4 Lane detection

Status: Ongoing 80%

Implementation: defined ROI; used canny algorithm for getting gradient picture; used Hough transform algorithm for detecting lines and averaging them to get two lines representing lines on the road.

Difficulties: Camera problem with Raspberry Pi

3.5 Creating/defining physical testing environment

Status: Ongoing 30%

Implementation: Traffic signs and the ramp will be printed using a 3D printer and assembled. We bought traffic lights online, and they will arrive in one month. As for the map itself, that's the most significant issue, and we are still working out how to make it exactly. Some options include printing the map at a business specializing in large-scale printing and creating a basic smaller map using some paper and basic office supplies.

Difficulties: cost and finding the place to put a 6x6 meter or a 15x15 meter map

4 General status of the project

The car can be controlled remotely. It can also send the camera stream to the laptop. We have only done lane detection on the computer, and we still must upload the code on the vehicle and apply it in practice.

5 Upcoming activities

- 1) Completely installing the physical testing environment with all its parts
- 2) Finishing lane detection as well as intersection detection
- 3) Lane following and speed control
- 4) ROIs definition and camera pre-processing
- 5) Defining project architecture and communication between packages