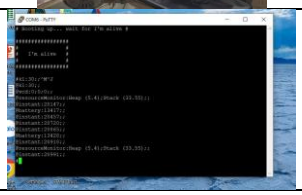
1 Introduction

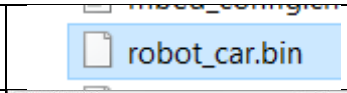
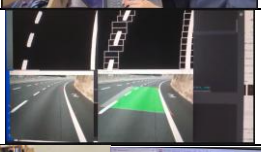
Throughout **Project Status 1**, our team maintained a strong spirit of collaboration and resilience. Even when we faced tight deadlines, unexpected bugs, and repeated iterations in testing, we stayed aligned on a common goal and supported each other consistently. We worked with an open mindset, sharing ideas, reviewing each other's code, and dividing tasks efficiently based on individual strengths - so that every improvement was the result of collective effort rather than individual work.

2 Planned activities

Receiving the car model from Bosch Vietnam		9/12	All
Checking the hardware system, chassis	Research	10/12	Trung Anh
Running Putty through PC to Nucleo	EP	11/12	Hung
Modifying file robot_car.bin	Update	13/12	Dat
Setting up an RPI embedded computer	EP	15/12	Phat
Connecting to Dashboard server	EP	18/12	All
Being able to control the car remotely with the dashboard	Update	18/12	All
Building Lane Detection test algorithm	Development	15/12	Hung

3 Status of planned activities

Activities	Results	Status	Solutions
Receiving the car model from Bosch Vietnam		Done	
Checking the hardware system, chassis		Done	Viewing the wheels, chassis, motor, servo, rpi, nucleo.
Running Putty through PC to Nucleo		Done	Changing serial port, baudrate,..

Modifying file robot_car.bin		Done	Compile speeding.cpp, steering.cpp instead.
Setting up an RPI embedded computer		Done	Installing the rpi-os (Bfmc).
Connecting to Dashboard server		Done	After 10s, connecting rpi_hot_spot
Being able to control the car remotely with the dashboard		Done	Change mode into KL30, and Manual
Building Lane Detection test algorithm		Done	Conventional Lane Detection, Hough transforms.
Calibration on the 192.168.50.1:4200 server		Ongoing	Backend lost connection after calibration (run 5 dgr)

4 General status of the project

By the end of **Project Status 1**, the car model has successfully achieved the key baseline capabilities for autonomous driving. It can move straight at a constant velocity and perform steering control using the output angle generated by the Lane Detection algorithm. At this stage, the complete perception-to-control pipeline has been validated in an offline demo setting using prerecorded video materials, allowing us to verify lane recognition stability and ensure the steering response generally follows the detected lane geometry under basic scenarios.

However, although the vehicle is able to detect lane markings, the steering angle is not yet sufficiently accurate and stable for precise lane keeping. With more development time, we would further refine this part by improving the control strategy and tuning parameters, so the car can output a more reliable steering angle and achieve smoother, more consistent driving performance

5 Upcoming activities

Working Weeks	1	2	3	4	5	6	7		8	9
Start of the week	25-nov.	2-dec.	9-dec.	16-dec.	23-dec.	30-dec.	6-ian.	13-ian.	20-ian.	27-ian.
Sensing and input working package	Documentation on the given guides and projects. Chose main languages and technologies Create/adapt project plan Members tasks asignation	Camera handling, preprocessing, noise cancelling, ROIs definition								Define other
							Define use-case and test given			
Perception and scene understanding working package		Lane detection								
Behaviour and motion plan working package		Define project architecture and communication between packages								
Vehicle control working packages		Lane following and speed control								
Final result & Demo	Team can control the phisical car remotely and the virtual car on the simulator.	Robot can keep a lane, can make a curve								
		Team defines and creates its own physical testing environment				Team defines a way of parallel developing and testing				
		Team installs the virtual testing environment								
Deadlines				22-dec.						6-feb.