Technical Report No.1

1. Introduction

This report summarizes the technical progress of the Smart Car team in the Bosch Future Mobility Challenge 2025, including task assignments for each member, implementation methods, and challenges encountered in some activities. The report provides a brief description of activities based on the plan during the period from November 18, 2024, to December 16, 2024, along with our current status and upcoming project plans.

2. Planned Activities

The work plan for team members during the first progress report period includes the following activities:

- Perception
  - Lane detection
  @Cao The Vinh and Truong Duy Thanh Nhan
  - Traffic sign detection and traffic light detection
  @Nguyen Trung Nguyen
  - Intersection detection
  @Cao The Vinh
  - Obstacle Detection
  @Truong Duy Thanh Nhan
- Control
  - Lane keeping, navigating, and speed control upon detecting traffic signals.
    @Truong Duy Thanh Nhan and Cao The Vinh
  - Obstacle detection and avoidance.
  @Truong Duy Thanh Nhan and Nguyen Thanh De
- Hardware, tool working and map
  - Building the hardware system and connecting the electrical system.
  @Nguyen Thanh De
  - Creating maps, traffic signs, and object detection.
  @Lanh Cung Thien Y and Nguyen Thanh De

3. Status of planned activities

3.1 Lane Detection

3.1.1 Implementation

For the lane detection algorithm, we utilized OpenCV to detect lanes, calculate, and predict the vehicle's steering angle.

Here are the details:

Images captured by the camera are sent to the Jetson Nano as RGB images. To enhance processing speed, we converted the images into the HSV color space for edge detection. Additionally, we applied Gaussian Blur to smooth the images and remove small details.

Next, we used the Canny edge detection algorithm. To optimize lane detection, we created a mask to focus on the specific region of interest for lane detection. The Hough Transform algorithm was then applied to detect edges, and the detected lanes were averaged to determine whether the lane is on the left or right side, allowing us to calculate the appropriate steering angle.

3.1.2 Issues

Since our testing area is not made of anti-glare materials, there are sections where the vehicle's lane detection fails, leading to deviations in its movement.

3.2 Traffic light detection

3.2.1 Implementation

To detect traffic light signals, we use OpenCV and classify the traffic light colors using the HSV color space.

3.2.2 Issues

Brightness and saturation in the working environment can reduce accuracy. Therefore, we adjust the brightness and saturation to appropriate levels. Additionally, we have trained the traffic light detection using YOLOv5 to improve accuracy under challenging conditions.

3.3 Intersection, traffic sign, and object detection

3.3.1 Implementation

We have collected image data of intersections, traffic signs, and objects (pedestrians, cyclists, roadblocks, cars) using a camera. The data is then labeled for each class. Subsequently, we use the dataset, including images and labels, to train the model.

3.3.2 Issues

We initially trained the model for 30 epochs using 1,000 images. The initial results were not satisfactory, so we decided to continue training, which led to improved outcomes. However, due to resource limitations, using the trained model on our primary setup was not feasible. As a solution, we converted the PyTorch model to ONNX format, allowing it to run on a Raspberry Pi 4. The Raspberry Pi communicates with the Jetson Nano via UART to send detection signals efficiently.

4. General status of the project

In general, we successfully completed vehicle navigation based on lanes, traffic signs, and objects according to the map we designed. Despite space limitations, we were still able to fulfill the tasks and meet the requirements set by the competition.

5. Upcoming activities

At this stage, we have completed the simulation of navigating according to the competition's map. Our next goal is to integrate the simulation with the vehicle running in a real-world environment. We aim to program vehicle control in conjunction with the provided interface to meet the competition requirements. Additionally, we plan to use UWB for vehicle positioning and implement the A* algorithm for path planning and finding the shortest route.