1 Introduction

During this period, we focused on several key aspects to advance the project effectively. After assessing the capabilities of the Raspberry Pi, we determined that it lacked the necessary performance to handle mission-critical tasks efficiently. Therefore, additional equipment, specifically the Jetson Orin Nano, was selected and purchased to meet the project's technical requirements.

On the algorithmic side, we successfully configured the simulation environment provided by the BFMC GitHub repository. Simultaneously, we initiated the development of a lane detection algorithm to achieve precise path recognition, which is essential for autonomous navigation. In parallel, the A* algorithm was developed to improve path planning efficiency, ensuring the generation of optimal and effective routes.

2 Planned activities

(1) Geunwoo Kim
- a. Select and purchase additional equipment
- b. Implement embedded code and brain code

(2) Hyoseok Bang
- a. Configure the BFMC simulation environment

(3) Minhyek Choi
- a. Develop a lane detection algorithm
- b. Initial hardware setup (Jetson Orin Nano)

(4) Myungeun Cho
- a. Develop the A* algorithm

3 Status of planned activities

(1) Hardware

a. Select and purchase additional equipment (completed)

We decided to use the Jetson Orin Nano for better performance and more intuitive mission execution. The mainboard was upgraded from the Raspberry Pi, and the purchase was completed. Additionally, an SSD card was installed to facilitate smoother progress.

b. Implement embedded code and brain code (delayed)

Before replacing the mainboard, it was necessary to ensure that the BFMC-provided brain and embedded codes functioned correctly on the existing hardware. The code was successfully uploaded to the Nucleo board, and SSH connections between the Raspberry Pi and the computer were established. However, an issue with the Brain code's Dashboard was identified, which needs to be resolved in the future.

c. Initial hardware setup (Jetson Orin Nano) (completed)

Before switching to the new mainboard, we completed the initial setup of the Jetson Orin Nano, including the installation of Ubuntu 20.04 and ROS Noetic. Once the Brain code's proper functioning is confirmed on the Raspberry Pi, we plan to switch to the Jetson.

(2) Algorithms

a. lane detection algorithm (completed)

The lane detection algorithm was completed and verified to work well with the existing video data. First, we perform a preprocessing process to distinguish lanes on the image. The preprocessing process includes ROI, saturation & brightness adjusting, and image smoothing. After extracting the lane from the image through color masking, the outline of the lane is obtained and used to find the center of the lane.

b. A* algorithm (completed)

The A* algorithm was successfully developed and confirmed to perform well in the F1tenth simulation environment. Although the Map topic used in ROS is discrete data, it must be converted into an occupied grid map to apply Astar algorithm. In addition, rather than simply the shortest distance, the minimum distance to be maintained from walls or obstacles was used as a parameter to prevent collisions. When the starting point and the arrival point were input, the Astar algorithm was automatically used to find the route.

c. Configure the BFMC simulation environment (completed)

The BFMC simulation environment was set up and verified to function correctly. The next step is to integrate the algorithms developed earlier into this environment.

4 General status of the project

(1) Developed Algorithms and Simulation Integration (100%)

- The lane detection algorithm and A* algorithm have been successfully developed and tested. The lane detection algorithm demonstrated accurate performance using video data, and the A* algorithm generated optimal paths in the F1tenth simulation, effectively responding to static obstacle scenarios.
- The BFMC simulation environment has been configured and is fully operational. The next step is to integrate the developed algorithms into the simulation environment to test their combined performance and refine their functionality further.

(2) Hardware Setup (80%)

- The Jetson Orin Nano has been configured with Ubuntu 20.04 and ROS Noetic, completing its initial setup. Currently, the Raspberry Pi is being used to verify the Brain code and embedded functionalities. Once confirmed, the transition to the Jetson platform will proceed for further development and testing.

5 Upcoming activities

(1) Replace the mainboard

- Once the Brain code and embedded code are confirmed to work correctly on the Raspberry Pi, we plan to replace the mainboard with the Jetson Orin Nano. This upgrade will enhance performance and support more advanced functionalities.

(2) Apply algorithms to the BFMC simulation

- The completed lane detection and A* algorithms will be integrated into the BFMC simulation environment. During this process, necessary adjustments will be made to adapt the algorithms to the simulation's specific requirements.

(3) Develop a localization algorithm

- To enable effective path planning, a precise localization algorithm will be developed. This will involve using GPS and IMU data, with plans to implement either an Extended Kalman Filter (EKF) or a Particle Filter to ensure accurate positioning.

(4) Add traffic light and stop line recognition

- To support mission execution, algorithms for recognizing traffic lights and stop lines will be developed. These features will play a crucial role in enabling the system to respond accurately to real-world traffic scenarios.