HOSEI UNIVERSITY

Orange2023

Design Report

Kengo Hozumi kengo.hozumi.5n@stu.hosei.ac.jp

Natsumi Matsumo natsumi.matsumoto.6j@stu.hosei.ac.jp

Shun Saito shun.saito.6t@stu.hosei.ac.jp

Yuhei Takamori yuhei.takamori.3i@stu.hosei.ac.jp

Yutaro Kato yutaro.kato.7n@stu.hosei.ac.jp

Faculty Advisor Statement

I hereby certify that the engineering design on Orange2023 was done by the current student team, and it is significant and equivalent to what might be awarded credit in a senior design course.

Signed Kazuyuki Kobayashi Date May 12, 2023

Prof. Kazuyuki Kobayashi May 12, 2023

Professor Kazuyuki Kobayashi

Faculty of Science and Engineering, Hosei University
3-7-2 Kajinocho Koganei, Tokyo 184-8584, Japan
E-mail; ikko@hosei.ac.jp

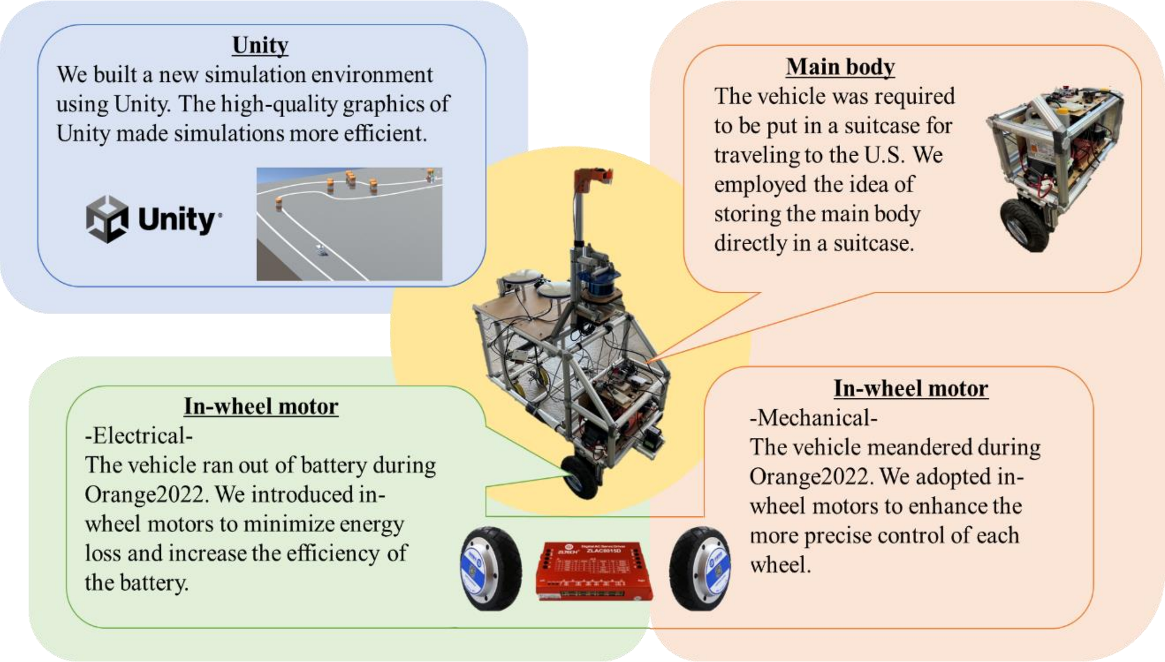
1. Introduction

The Hosei University Autonomous Robotics Laboratory (ARL) team is proud to present a newly redesigned vehicle, Orange2023. This vehicle has been improved based on the reflections from Orange2022 used for participation in the previous year. Through team discussions, we identified problems in the mechanical, electrical, and software development areas. To solve these problems, we identified the aspects to be solved during the development of Orange2023. Table 1 summarizes the problems that occurred during Orange2022 and the requirements for solving them.

Table 1. Problems that occurred during Orange2022 and the requirements to solve them.

	Problem	Requirement
Mechanical	Carrying problem to bring the vehicle from Japan to the U.S.	The vehicle must be put in a suitcase without removing the tires
	It took time to attach and remove the tires	
	The vehicle meanders while running	Enhance more precise control of wheels
Electrical	The vehicle ran out of battery during Orange2022	Minimize energy loss and increase efficiency for a battery
Software	The discrepancy between the simulation environment and the real world.	Use a more realistic simulation environment than Gazebo simulator

After identifying the problems and requirements, we considered the innovations that could be made to create a vehicle that would satisfy these requirements. Figure 1 shows the innovations incorporated into Orange2023.



Unity

We built a new simulation environment using Unity. The high-quality graphics of Unity made simulations more efficient.

Main body

The vehicle was required to be put in a suitcase for traveling to the U.S. We employed the idea of storing the main body directly in a suitcase.

In-wheel motor

-Electrical-

The vehicle ran out of battery during Orange2022. We introduced in-wheel motors to minimize energy loss and increase the efficiency of the battery.

In-wheel motor

-Mechanical-

The vehicle meandered during Orange2022. We adopted in-wheel motors to enhance the more precise control of each wheel.

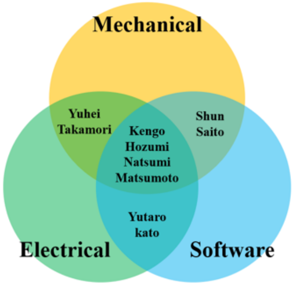
Figure 1. Innovations incorporated into Orange2023.

1.1. Team organization

This year's ARL team comprises two graduate students and three undergraduate students. To improve the development efficiency, we divided the members into mechanical, electrical, and software sub-teams based on their design skills. Table 2 summarizes the roles assigned and time spent by each member.

Table 2. Roles of individuals and hours spent by them on the project.

Areas of Concentration				
Team Member	**Mechanical**	**Electrical**	**Software**	**Hours**
Natsumi Matsumoto	○	○	○	700
Kengo Hozumi	○	○	○	700
Yuhei Takamori	○	○		200
Shun Saito	○		○	200
Yutaro Kato		○	○	200



1.2. Design Process

Figure 2 shows the design process for Orange2023 using an agile development model. In the previous year, a V-shaped waterfall model was used as the vehicle design for Orange2022. Because most members were not experienced in vehicle development, this year, the agile development model was adopted to monitor progress rather than emphasize planning.

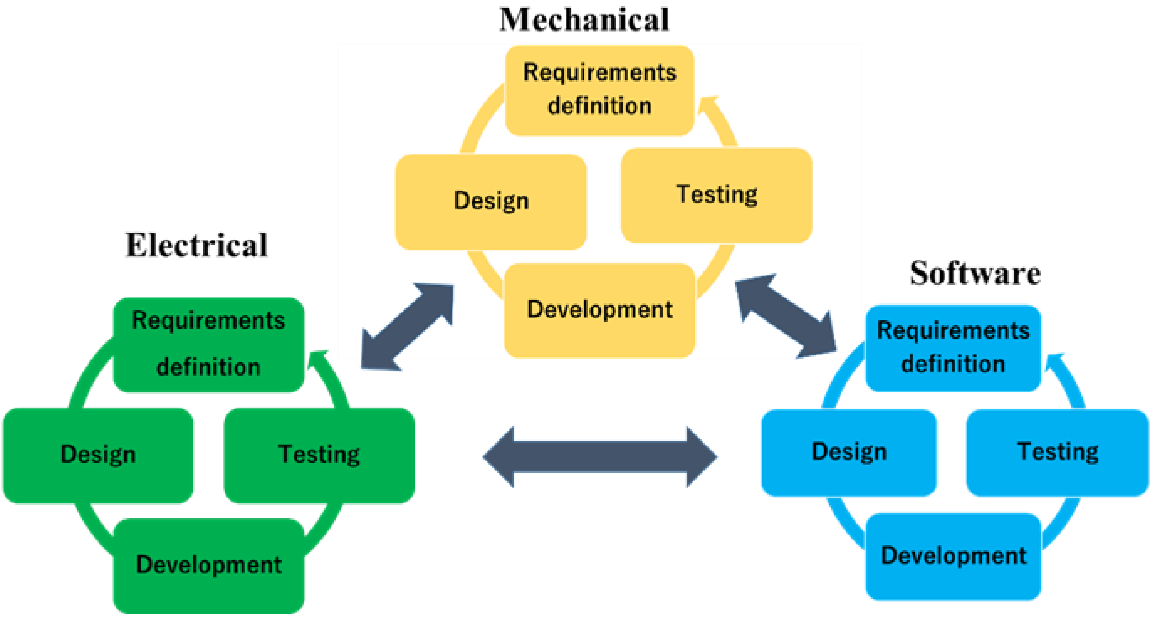


Figure 2. Our development approach based on the agile development model.

With this development approach, we achieved steady progress by immediately testing any corrections made in response to the detected defects during the design phase. We have revised our explanation of the development model and what we accomplished to provide accurate information.

2. Innovations

This section summarizes various innovations in vehicle design, including mechanical, electrical, and software innovations. Table 3 summarizes the design problems and innovative improvements.

Table 3. Mechanical, electrical, and software problems and improvements.

	Problem	Improvement
Mechanical	Carrying problem to bring the vehicle from Japan to the U.S.	Design the vehicle that can be put in a suitcase without removing the tires
	It took time to attach and remove the tires	
	The vehicle meanders while running	Introduction of in-wheel motors to enhance more precise control of each wheel
Electrical	The vehicle ran out of battery during IGVC2022	Introduction of in-wheel motors to minimize energy loss and increase efficiency for a battery
Software	Time-consuming problem to understand discrepant between simulation environment and real world	Build the new simulation environment on Unity

2.1. Redesign the main body to solve the carrying problem. 🟧Mechanical

To reduce the time required to disassemble and assemble the vehicle body, we employed the idea of storing the main body directly in a suitcase for a new vehicle design without disassembling the entire vehicle body. As shown in Figure 3, we divided the vehicle into three bodies: the main, upper, and lower bodies. The main body is composed of several modules, which are explained in Section 3. This plan simplified storage by eliminating the need to remove each module, particularly the motor driver and DC/DC converter. It also eliminated the need to reassemble the wiring, preventing wiring connection errors and reproducing the same vehicle configuration as during the test run, thereby improving performance stability.

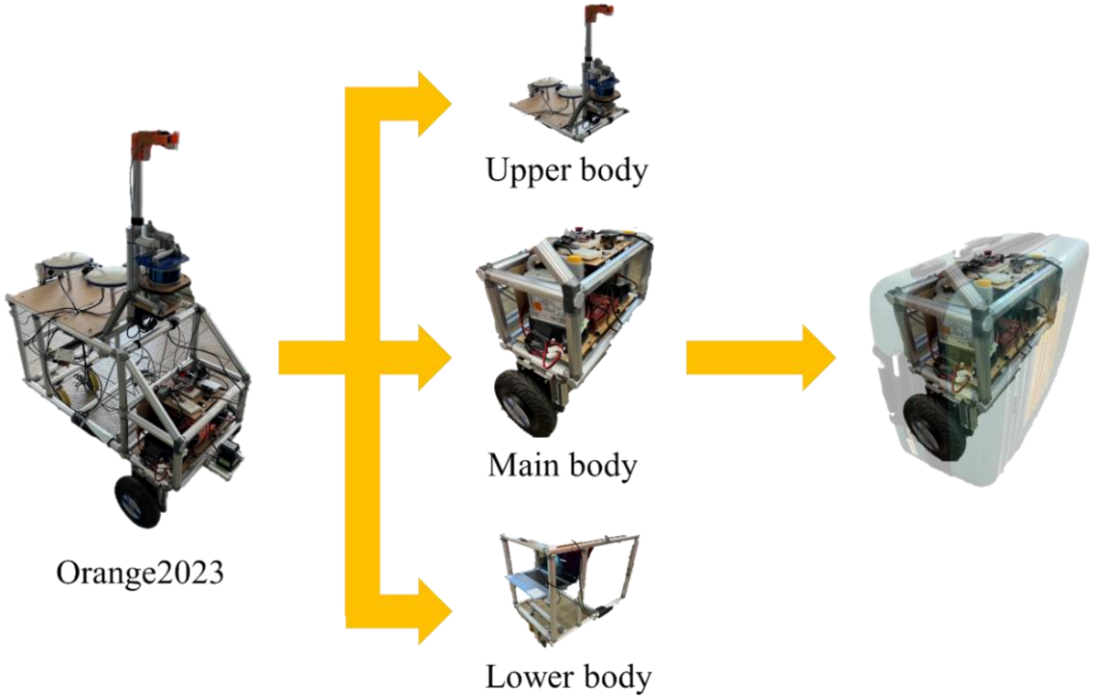


Figure 3. Vehicle body composition.

2.2. Adoption of in-wheel motors. 🟧Mechanical 🟩Electrical

As shown in Figure 4, based on our reflections of the vehicle design of the previous year, we decided to use a new in-wheel motor. Table 4 summarizes the requirements for satisfying our problems and the advantages of in-wheel motors in solving these requirements.

Figure 4. Adopted in-wheel motors for Orange2023.

Table 4. Requirements of our vehicle and the advantages of in-wheel motors.

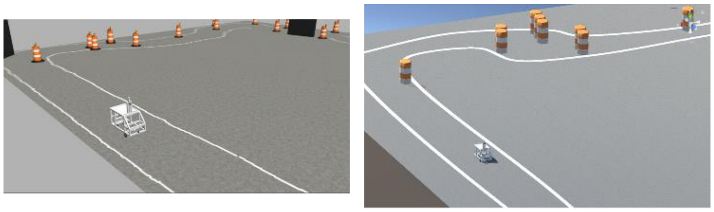
	Requirements	Advantages
Mechanical	Enhance more precise control of wheels to prevent meandering while driving	Fast torque application makes it easy to adjust vehicle speed
		It is simple to control the vehicle attitude due to Independent driving force control
Electrical	Increase battery efficiency to reduce the number of recharges and to allow the battery to run even if it fails	Transmitting rotational energy directly to the wheels enables minimizing energy loss

In-wheel motors have rapid torque application, which makes adjusting the vehicle speed easy. In addition, the motor drivers employed were designed for two-wheeled mobile robot control such that each wheel can be controlled independently to improve the turning capability of the vehicle. This reduces the meandering of the vehicle when moving, which was a problem with Orange2022.

These motors transmit rotational energy directly to the wheels, eliminating the necessity for transmission or other components. This minimizes the energy loss and eliminates battery shortage, which was a problem with Orange2022. Consequently, the number of battery recharges is reduced. In addition, by placing two batteries in the vehicle, if one battery fails, the other can continue operating the vehicle.

2.3. Introduction of a new simulation environment 🟦Software

As shown in Figure 5, in this year's IGVC, we have applied a new realistic simulation environment using Unity instead of the ROS standard Gazebo. Unity is a game engine used primarily for game development, and it simulates obstacles more precisely than Gazebo. This eliminates the differences between the simulation and test runs on an actual vehicle and accelerates the cycle between each process. Therefore, the use of Unity can increase simulation efficiency.



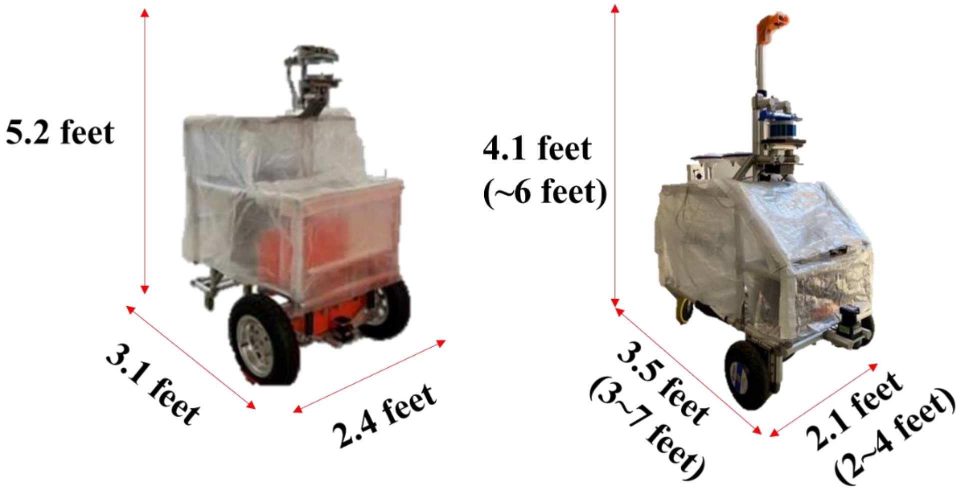
(a)Gazebo (b)Unity

Figure 5. Simulation environment of (a) Gazebo and (b)Unity.

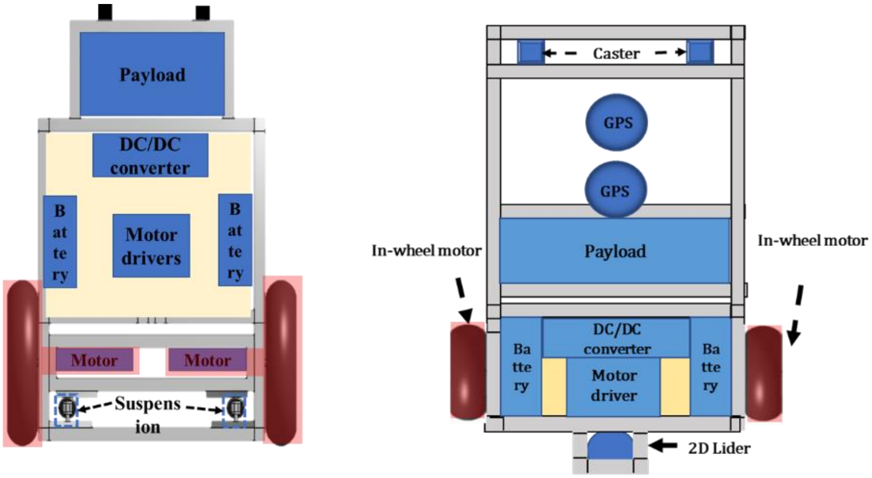
3. Mechanical Design

3.1. Design Overview

Figure 6 shows a comparison between Orange2022 and Orange2023.



(a)Orange2022 (b)Orange2023



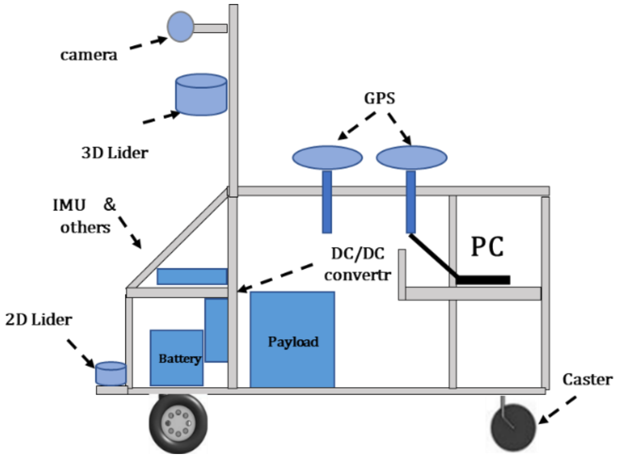
(a)Orange2022 (b)Orange2023

Figure 6. Top view of Orange2022 and Orange2023.

The dimensions of the Orange2023 body are 2.1 feet × 3.5 feet × 4.1 feet, satisfying the IGVC 2023 rule entry requirements. As Figure 6 shows, the new in-wheel motor enables the motor used in Orange2022 to be positioned where it can be used as a battery, making it 0.3 ft narrower and improving mobility. The exterior was designed to be lightweight and durable under inclement weather conditions, utilizing a waterproof cover similar to that used in Orange2022. In addition, the payload of Orange2022 was placed in the center, closer to the wheels than in the tail section, thereby minimizing its impact on driving.

3.2. Vehicle Configuration

Figure 7 shows a side view of the module configuration, and the list describes the modules located at each position.



Part	Module
Main	In-wheel motor ×2
	Motor driver
	DC/DC converter
	Battery ×2
	IMU sensor
	2D-Lider
Upper	3D-Lider
	GPS ×2
	camera
Lower	PC

Figure 7. Side view of Orange2023 and list of modules.

Figure 7 shows the locations of the modules. Most of the modules necessary for driving are in the main body, minimizing the rewiring of electrical harnesses during disassembly and simplifying assembly in the US.

4. Electrical Design

Orange2022 had only one battery, which provided a battery run time of approximately four hours. To solve the problem of a vehicle's short runtime on the competition day, we decided to employ two batteries. Figure 8 shows a photograph of the front of the vehicle.

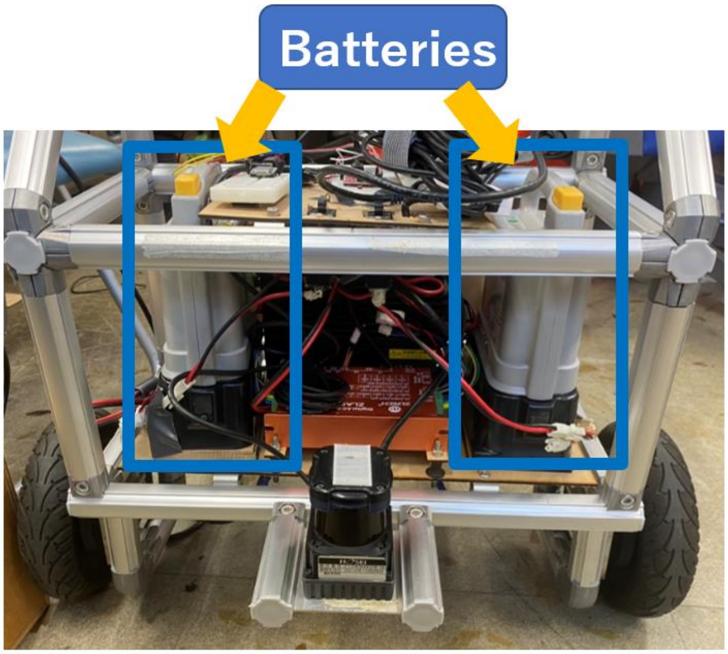


Figure 8. Photograph of the front of Orange2023.

4.1 Power and signal distribution system

Figure 9 shows the overall power and signal distribution of Orange2023. The red and blue lines represent power and signal flow, respectively.

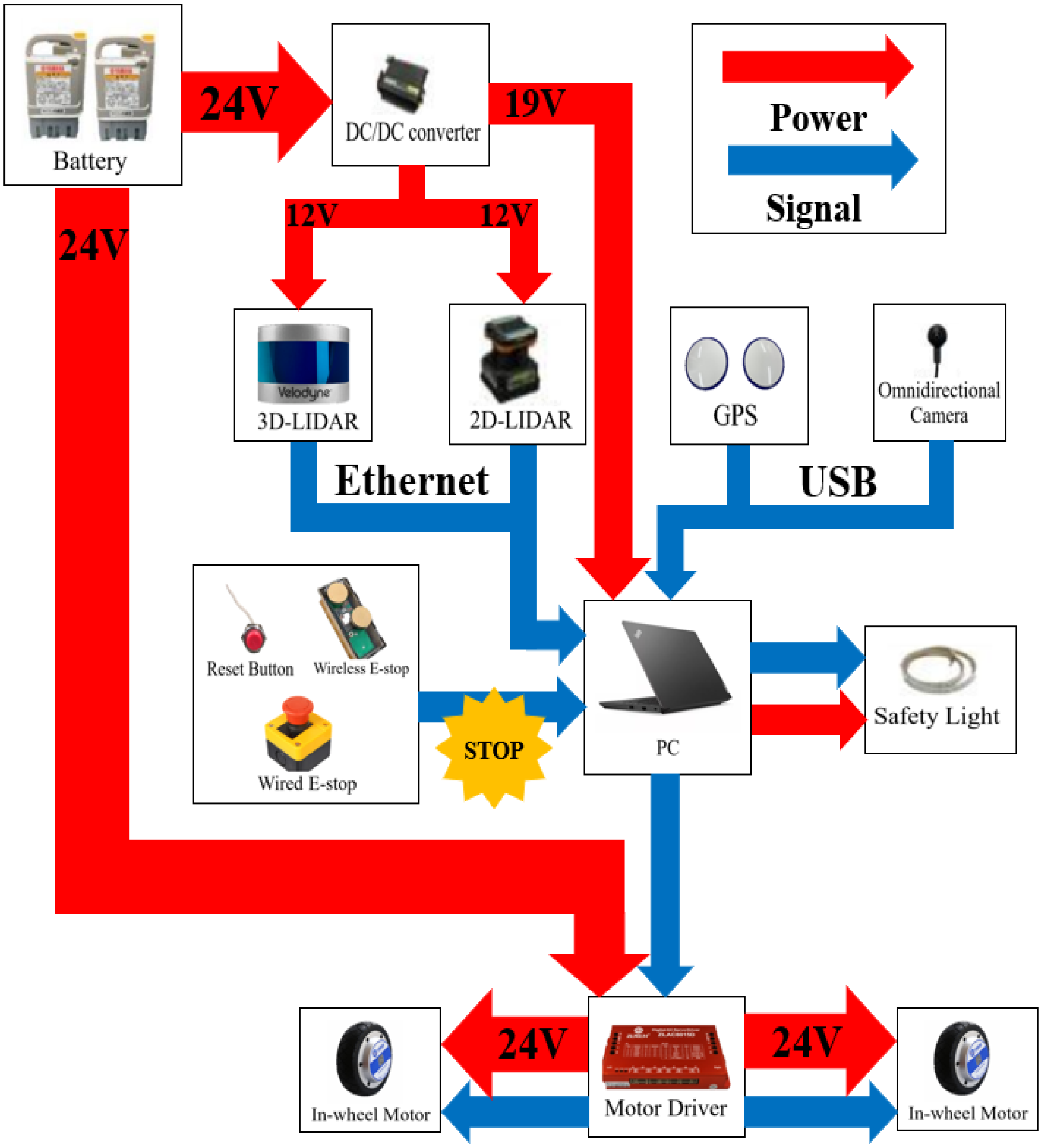
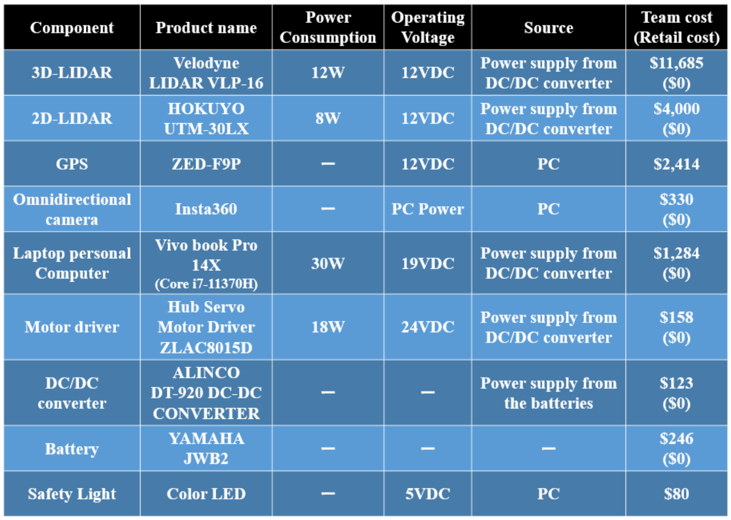


Figure 9. Power and signal flow of Orange2023

- **Power distribution system:** The 24 V NiMH batteries supply power to the motor driver and DC/DC converter. The motor drivers supply power to the left and right in-wheel motors, whereas the DC/DC converter adjusts the input voltage to supply 19 V to the PC and 12 V to the 3D-LIDAR and 2D-LIDAR.

- **Signal distribution system:** The PC is the central component of the signal distribution system: it receives data from sensors such as the 3D-LIDAR, 2D-LIDAR, and omnidirectional cameras. In addition, it transmits commands to the safety lights and motor drivers. However, when the E-stop is pressed, signals are transmitted to the motor driver to deactivate the power supply for the in-wheel motor without any PC operation.
Table 5 summarizes the electrical components of Orange2023.

Table 5. Electrical components of Orange2023



Component	Product name	Power Consumption	Operating Voltage	Source	Team cost (Retail cost)
3D-LIDAR	Velodyne LIDAR VLP-16	12W	12VDC	Power supply from DC/DC converter	$11,685 ($0)
2D-LIDAR	HOKUYO UTM-30LX	8W	12VDC	Power supply from DC/DC converter	$4,000 ($0)
GPS	ZED-F9P	—	12VDC	PC	$2,414
Omnidirectional camera	Insta360	—	PC Power	PC	$330 ($0)
Laptop personal Computer	Vivo book Pro 14X (Core i7-11370H)	30W	19VDC	Power supply from DC/DC converter	$1,284 ($0)
Motor driver	Hub Servo Motor Driver ZLAC8015D	18W	24VDC	Power supply from DC/DC converter	$158 ($0)
DC/DC converter	ALINCO DT-920 DC-DC CONVERTER	—	—	Power supply from the batteries	$123 ($0)
Battery	YAMAHA JWB2	—	—	—	$246 ($0)
Safety Light	Color LED	—	5VDC	PC	$80

- **3D-LIDAR:** The 3D-LIDAR builds a 3D map of the surrounding environment by transmitting laser beams in 360 degrees from 0.7 to 328 feet with an accuracy of ±0.1 feet and measures the time it requires for the beams to bounce off objects.
- **2D-LIDAR:** The 2D-LIDAR provides information on the distance and reflectivity of objects in a single plane by measuring the surrounding environment at 180 degrees from 0.3 to 98.4 feet ahead with an accuracy of ±0.1 feet.
- **GPS:** Two GPS receivers (u-blox F9P) and RTK GPS antennas (BT-200) enable accurate heading-direction estimation, even when the vehicle is stationary.
- **Omnidirectional Camera:** This camera comprises upper- and lower-side lenses and captures a full 360-degree field of view in all directions. The lower-side lens is used to detect the lane, whereas the upper-side lens is used to detect the ambient light intensity, through which the lane detection threshold is determined.
- **Motor Driver:** The motor driver controls the speed, direction, and rotation of the in-wheel tires on both sides of the vehicle.
- **DC/DC Converter:** The DC/DC converter converts 24 DC voltage into 12 DC voltage to supply the optimum voltage required by the installed sensors, 3D-LIDAR, and 2D-LIDAR.

4.2 Safety devices and system integration

Figure 10 shows the wired and wireless E-stops. The wireless E-stop operates with ZigBee, which is a low-power wireless communication protocol that uses the 2.4 GHz band. When the E-stop is pressed, the power supply from the motor driver to the in-wheel tires is instantly shut down without any intervention from the laptop computer.

(a) Wired E-stop (b) Wireless E-stop

Figure 10. (a) Wired E-stop, (b) wireless E-stop

As shown in Figure 11, the LED lights were designed to display red blinking in warning mode as a safety light every time the vehicle is operating autonomously. When an unexpected event occurs and the E-stop is pressed, the LED lights display blue blinking as a safety mode.

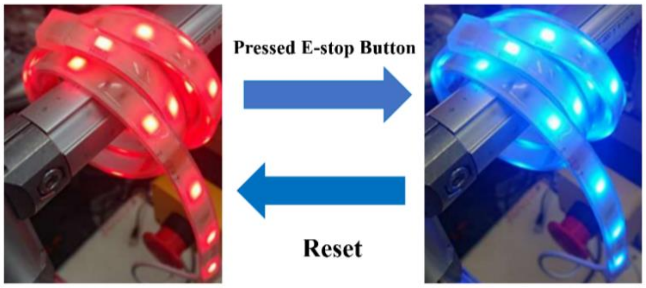


Figure 11. Mechanism of the E-stop and safety light

5. Software Design

We used the game engine Unity for software development. Using ROS and Unity, we built a simulation environment for IGVC. Figure 12 shows how ROS and Unity communicate with each other.

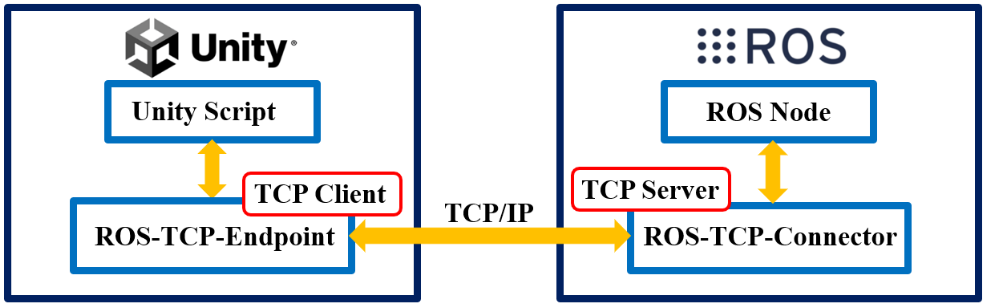


Figure 12. Communication between ROS and Unity.

ROS-TCP-Connector and ROS-TCP-Endpoint are ROS packages that enable communication between ROS and non-ROS systems over TCP/IP connections. ROS-TCP-Connector provides a TCP server that listens for incoming connections from Unity, whereas ROS-TCP-Endpoint provides a TCP client that connects to the ROS-TCP-Connector server.

Connecting ROS-TCP-Connector and ROS-TCP-Endpoint enables Unity to send and receive messages with ROS nodes from the script in Unity. In addition, the ROS node can communicate using Unity-based simulations. This provides a method of controlling vehicles, simulating their behavior, and visualizing sensor data from ROS in Unity.

Table 6 summarizes the details of the operating system software used in the development of Orange2023.

Table 6. Overview of software used during development.

Software	Details
Ubuntu 22.04	Ubuntu is an open-source operating system for general-purpose computers and servers.
ROS noetic	ROS is an open-source framework for building robot systems. It provides libraries and tools necessary for developing robot applications.

5.1 Obstacle detection and avoidance and lane detection

Table 7 lists the sensors used for obstacle detection and their roles.

Table 7. Sensors used to detect obstacles and their use applications.

Sensors	Use Applications
3D-LIDAR	Used to generate a global SLAM map
2D-LIDAR	Used to detect and avoid obstacles and build a local map
Omnidirectional Camera	Used for map generation and local map generation

As shown in Figure 13, the omnidirectional camera was installed such that the two lenses face upward and downward. The lens facing downward captures white lines on the ground, whereas that facing upward measures the ambient light intensity to determine the lane-detection threshold.

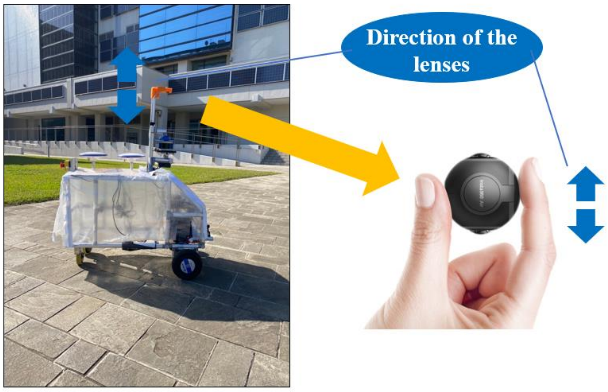


Figure 13. Installed location of the omnidirectional camera

Orange2023 uses simultaneous localization and mapping (SLAM), a technology that enables a moving robot to build a map of its environment while estimating its own location by collecting information from the 3D-LIDAR, 2D-LIDAR, and omnidirectional cameras. Based on the data collected, global and local maps are generated for obstacle detection and avoidance. Table 10 summarizes the explanations of the global and local maps. Figure 14 shows the generation of global and local maps using SLAM.

Table 8. Explanations of the global and local maps

	Details
Global Maps	A global map provides an overview of the environment that the autonomous robot has explored.
Local Maps	A local map provides detailed information on the immediate surroundings of the autonomous robot in real-time.

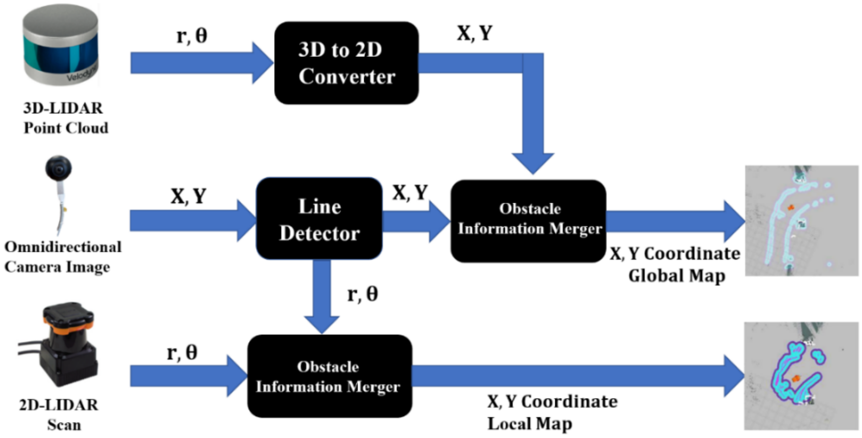


Figure 14. Global and local maps generated using SLAM.

5.2 Lane-recognition procedure

Figure 15 shows how lane recognition operates using data from the omnidirectional camera and 2D-LIDAR. At the end of the process, the image captured by the omnidirectional camera is combined with the data obtained using the 2D-LIDAR. Subsequently, the white lines are treated as obstacles for navigation.

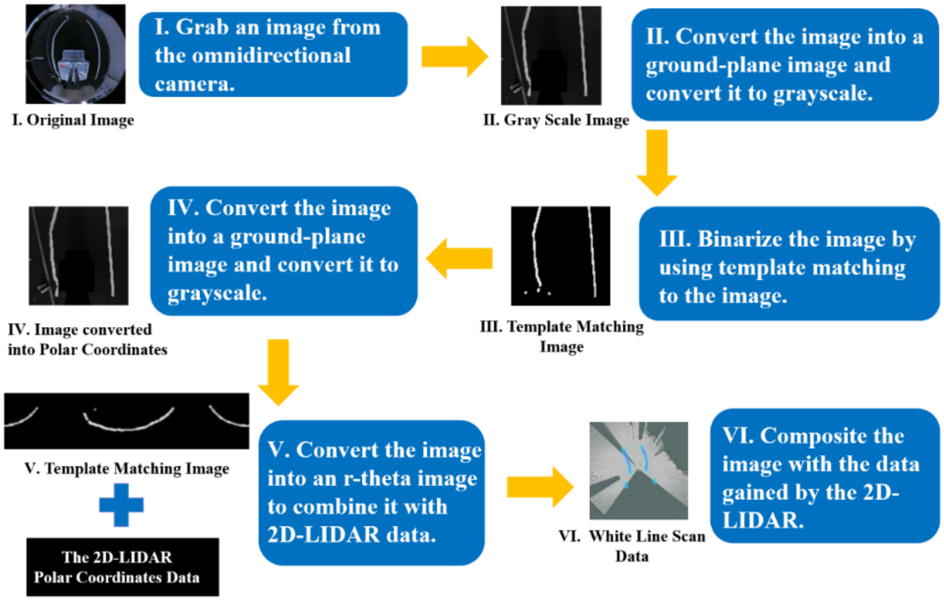


Figure 15. Lane-recognition process

5.3 Self-localization and path planning

Orange2023 uses two different methods to estimate self-position: GPS or the local and global maps generated using SLAM.

Global path planning is performed by combining the generated global map and estimated self-position. The path to the next waypoint is then determined by applying the A-star algorithm to the local map obtained using the 3D-LIDAR and omnidirectional camera. Furthermore, the 2D-LIDAR has been used to detect obstacles during path planning for safe navigation.

5.4 Map generation by 3D-LIDAR

Figure 16 shows the construction of the global map using the 3D-LIDAR.

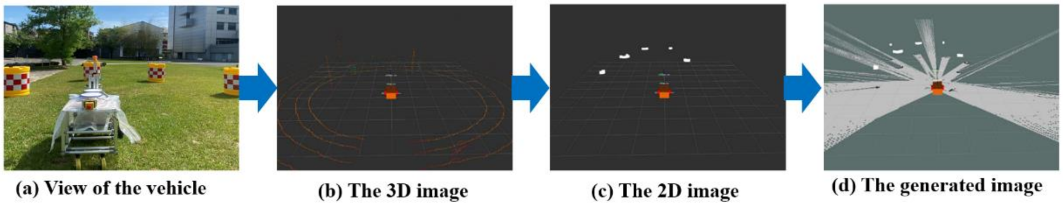


Figure 16. Example of a generated global map

Figure 16 (a) shows the surrounding view of Orange2023. Figure 16 (b) shows the 3D view is detected by the 3D-LIDAR. Subsequently, the point-cloud image is converted into a 2D image, as shown in Figure 16 (c). The white dots represent obstacles discovered by the 3D-LIDAR. Figure 16(d) shows the generated global map. The light gray areas on the image are areas in which the vehicle may move, whereas the dark gray areas are areas that have not yet been detected by the 3D-LIDAR.

5.5 Path planning to the destination

The direction in which the vehicle travels is determined using a cost map and path planning. The cost map represents the risk of collisions with obstacles to reach the desired destination in the most efficient manner possible.

Path planning is performed using the cost map to obtain the shortest path to the goal. There are two types of path planning: global and local. Global path planning uses data from a global cost map, whereas local path planning uses information from a local cost map.

Figure 17 shows an example of how a path is planned. The pink, pale blue, and red colors indicate the location of the object, areas of possible collision with obstacles, and boundary areas that the robot should avoid, respectively. The green line in the center of the figure represents the path obtained using global path planning, whereas the blue arrow is the path obtained using local path planning.

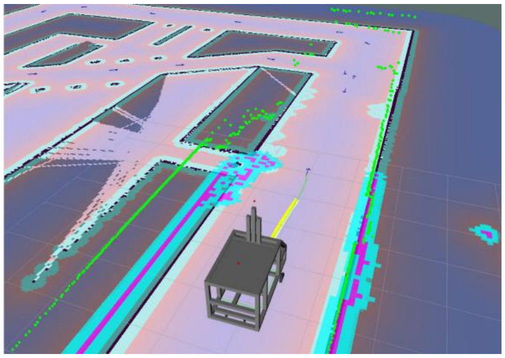
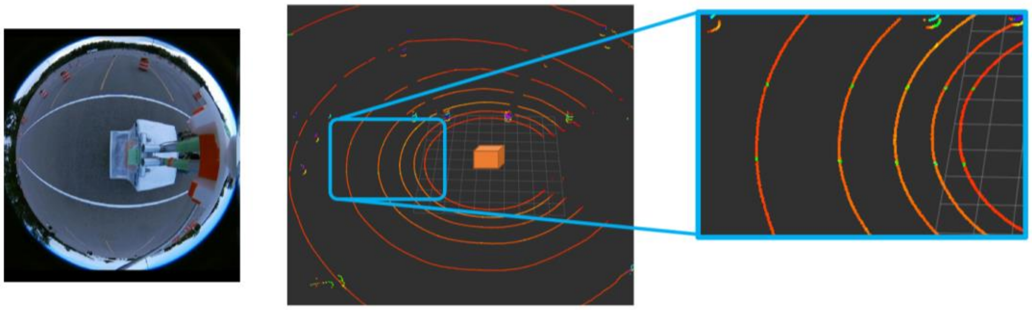


Figure 17. Path planning

5.6 Additional innovative white-line detection

As shown in Figure 18, based on the observation of the white lines at IGVC 2022, we observed that the white lines could be detected not only by the omnidirectional camera but also by the reflection intensity of the 3D-LIDAR. In the enlarged image in Figure 18(b), the green dots indicate the white lines detected by the difference in reflection intensity between the asphalt and white lines. Using this difference in the reflection intensity between the white line and asphalt, we developed an algorithm to recognize the white lines.

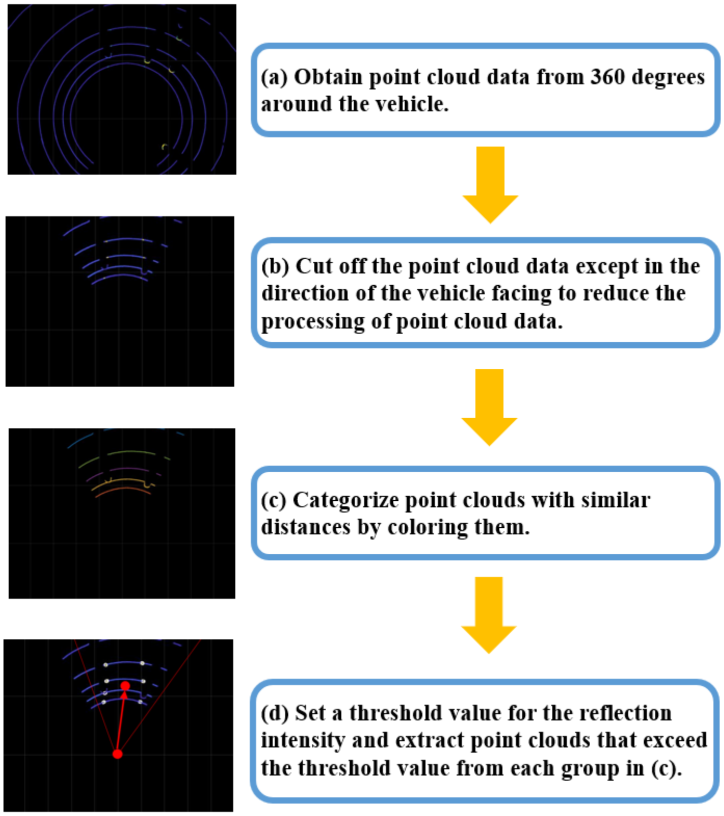


(a) Image from the Omnidirectional Camera.

(b) Point Clouds Obtained from 3D-LIDAR.

Figure 18(a) Image from the omnidirectional camera; (b) Point cloud data obtained from the 3D-LIDAR.

Figure 19 shows the algorithm for detecting white lanes from the obtained point cloud data. This enables the vehicle to recognize the white lines and operate safely even if the omnidirectional camera breaks or fails while the vehicle is moving. The combination of the detected lanes for both the omnidirectional camera and 3D-LIDAR achieves robust and stable detection.



(a) Obtain point cloud data from 360 degrees around the vehicle.

(b) Cut off the point cloud data except in the direction of the vehicle facing to reduce the processing of point cloud data.

(c) Categorize point clouds with similar distances by coloring them.

(d) Set a threshold value for the reflection intensity and extract point clouds that exceed the threshold value from each group in (c).

Figure 19. Algorithm for detecting white lines from the obtained point cloud data.

The white dots in Figure 19(d) indicate the point clouds that exceed the reflection threshold and are recognized as white lines. This algorithm enables the white lines to be recognized and the vehicle to move in the direction of the red arrow (Figure 19(d)).

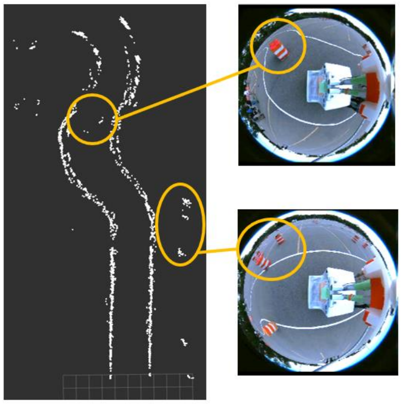


Figure 20. Map with white-line recognition based on point cloud data.

As shown in Figure 20, the white lines are successfully recognized by the point-cloud data obtained from 3D-LIDAR. Moreover, by incorporating obstacle data obtained from the 3D-LIDAR, both white lines and obstacles can be detected.

6. Failure Points

Table 9 lists the expected vehicle problems and their solutions for Orange2023.

Table 9. Vehicle failure points and resolutions

Problem	Solution
Flooding from holes for GPS, etc.	Resolve flooded areas by applying absorbent material such as sponges
Dead battery problem	Solved by installing two batteries
Thermal problem with navigation in sunny weather	Resolved by installing dual fans in the front of the car, as all electrical equipment is located in front of the car body

7. Simulation

Figure 21 shows the simulation environment in Unity. This simulation environment was built based on the IGVC2023 rules and was designed to be as similar to the actual environment as possible. This enabled us to ensure that the vehicle can navigate within the white lines or avoid obstacles before travelling in the actual environment. Moreover, it provided opportunities for understanding the causes of errors that occur while the vehicle is test-running in the simulation environment. Therefore, trial and error in the simulation environment enabled us to improve the software algorithm for navigation.

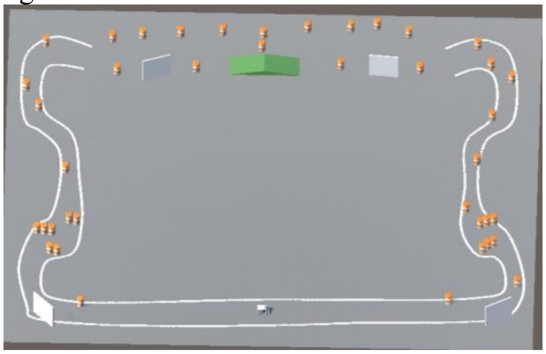


Figure 21. Simulation environment in Unity.

8. Evaluation of Initial Performance

Table 10 compares the expected performance of Orange2023 and the performance result of Orange2022.

Table 10. Comparison between performance prediction and result of Orange2023 and performance result of Orange2022.

Measurement	Performance Prediction (Orange2023)	Performance Result (Orange2023)	Performance Result (Orange2022)
Speed	6.1km/h(3.8mph)	6.3km/h(3.9mph)	6.1km/h(3.8mph)
Ramp climbing ability	16.0%incline	16.0%incline	15.5%incline
Reaction time	0.23s	0.16s	0.23s
Battery life	8h	8h	4.5h
Obstacle detection distance	0-10m(0-33ft)	0-10m(0-33ft)	0-10m(0-33ft)
Waypoint navigation	±0.08m(±0.27ft)	±0.08m(±0.27ft)	±0.08m(±0.27ft)

As shown in Table 10, the most significant differences between Orange2023 and Orange2022 are in the reaction time and battery life. The reaction time of Orange2023 is shorter than that of Orange2022 because in-wheel motors have been introduced to Orange2023 with a newly rewritten motor driver code for the ROS system. Therefore, we concluded that the reaction time of Orange2023 is shorter than that of Orange2022.

Orange2023 has two installed batteries. In addition, the in-wheel motors transmit rotational energy directly to the wheels. This eliminates the necessity for transmission or other transmission components and minimizes energy loss. Therefore, we conclude that the battery life of Orange2023 is longer than that of Orange2022.

9. Conclusion

Based on the reflections discovered in Orange2022, we redesigned Orange2023 to achieve a more stable driving performance. The specific improvements are as follows:

-Mechanical-

To fit in suitcases for carrying from Japan, we redesigned the vehicle body by decomposing it into three bodies and storing one of the vehicle bodies directly in a suitcase.

We employed in-wheel motors to adjust the vehicle speed easily and enhance precise control.

-Electrical-

The employment of new in-wheel motors enabled the vehicle to transmit rotational energy directly to the wheels, thereby eliminating the necessity for transmission or other transmission components. This minimized the energy loss and eliminated battery shortage, which was a problem with Orange2022.

-Software-

The adoption of Unity enabled us to build a realistic simulation environment with high-quality graphics and reduce the differences between simulation and test runs on an actual vehicle compared with Gazebo.

With these improvements, we are confident that Orange2023 will provide the best results.