| Full Name | Student ID |
|---|---|
| Ngo Minh Ngoc | V202200669 |
| To Dong Anh Khoa | V202200770 |
| Vu Ai Thanh | V202200674 |
| Le Phuong Linh | V202200771 |
| Thai Minh Dung | V202200665 |

# Final Project Report

**CECS1010 Intro to Engineering and Computer Science** **Spring 2023**

# Contents

# 1 Introduction

## 1.1 Project Background

In the modern fast-moving world, technology exists in all aspects of life, completely transforming the means of transportation. Along with the introduction of driver assistance technologies, investment, and development in automated driving systems have risen to unprecedented levels. Competitive advantages such as safety, mobility, efficiency, and convenience, as well as economic and environmental benefits, promise to lead self-driving vehicles to a potential and sustainable future.

In the face of escalating traffic issues, mankind longs for easier ways to navigate their automobiles, one of which is to incorporate Artificial Intelligence (AI) in the driving process. It is necessary to take part in understanding and advancing automated vehicles to solve our daily riddles of improving traffic conditions and increasing accessibility for people who lack the sufficient means to drive. This constantly growing demand has given several automotive enterprises great opportunities and power.

**Tesla**

Tesla upholds the world's transition to sustainable energy through manufacturing electric cars. It is the only automaker that has fully embraced the cameras-and-sensors approach, produces its own cars, and is able to pursue a unique, integrated solution. Recognized as the most valuable automotive brand worldwide in 2022 [9], the company has introduced four car models so far, with major packages being Autopilot and Full Self-Driving Capability. [2]


[6]

Autopilot

- Traffic-Aware Cruise Control: Aligns the speed of your car to that of the surrounding traffic
- Autosteer: Assists the driver in steering within the marked lane

<u>Enhanced Autopilot</u>

- Auto Lane Change: Assists in moving to an adjacent lane when indicated by the driver
- Navigate on Autopilot (Beta): Improves Auto Lane Change by providing guidance on lane changes and interchanges
- Autopark: Assists in parallel or perpendicular parking with a simple touch
- Summon: Maneuvers the car into and out of a small space using the mobile app
- Smart Summon: Navigates the car through complex surroundings and parking lots, dodging obstacles to locate the driver

<u>Full Self-Driving Capability</u>

- Basic Autopilot and Advanced Autopilot's whole feature set
- Traffic and Stop Sign Control (Beta): Identifies traffic lights and stop signs and slows the car to a stop under the driver's active supervision
- Autosteer on city streets (Upcoming)

**Waymo**


[14]

Waymo is formerly known as the Google Self-Driving Car Project, explaining why it has the most miles driven on roads and in simulation. Yet, Waymo needs to partner with automakers to successfully bring its product into the market. Defining its mission as "make it safe and easy for people and things to move around," the company presents two services: Waymo One for moving people, and Waymo Via for transporting commercial goods. Their most prominent technology is called Waymo Driver, the embodiment of fully autonomous technology that maintains control throughout the entire journey, requiring no driving experience from passengers.
Waymo Driver[13]

- A combination of software, sensors, and lidar that can be installed on any vehicle
- Maps the territory meticulously and matches with real-time sensor data to identify the car's precise road location at all times (no longer relying on GPS due to signal strength uncertainty)
- Perception system: Uses machine learning to interpret complex data acquired from its advanced set of sensors; differentiates objects and responds accordingly to traffic lights, signs, and signals
- Predicts several paths based on its understanding of a car's movement and anticipates other road users' potential actions to come up with the best trajectory to take
- Lidar: Provides a 3D picture of the surrounding
- Cameras: Provides a simultaneous 360-degree view around the vehicle with high dynamic range and thermal stability to operate both day and night
- Radar: Provides details regarding any object's distance and speed, working in rain, fog, and snow
- Computation: Combines the latest server-grade CPUs and GPUs to plan a secure path to the destination in real-time

**Zoox**

[17]

Claimed to bring about a whole new form of transportation, this subsidiary of Amazon develops autonomous vehicles that provide Mobility-as-a-Service. Zoox is the first company to gain approval for providing automated ride-hailing services to the public in the State of California [11] and also the first purpose-built, fully autonomous vehicle certified to the Federal Motor Vehicle Safety Standards (FMVSS) [10]. Zoox's robotaxis incorporate a simplistic design with no steering wheel or pedal.
Zoox Robotaxi [4]

- Calibration: Determines the exact positioning of all sensors on the vehicle
- Mapping: Draws geometric and semantic maps of predefined areas to record the high definition representation of the surroundings, speed limit, traffic lights location, etc.
- Localization: Matches the real-time sensor data to the drawn map
- Creates a 360-degree overlapping field of view extending over 150 meters around the vehicle, enabling perception of all directions equally well
- Uses sophisticated machine learning to predict people and vehicles behaviors
- Planning and Controls: Consider the perception and prediction data to determine how to safely and comfortably drive to the destination
- Low-level Controls: Responsible for regulating acceleration, braking, and steering
- TeleGuidance: Provides assistance from a human TeleOperator in face of "edge cases" - rare complex driving situations.

## 1.2 Project Definition

The Society of Automotive Engineers (SAE) defines 6 levels of vehicular autonomy, starting from Level 0 to Level 5: no automation, driver assistance, partial automation, conditional automation, high automation, and full automation [15]. Among these, a fully automated vehicle needs to fulfill these capabilities [8]:

- Operating and navigating to a predetermined location in all conditions and scenarios without human involvement, except for setting a destination
- Sensing the environment: intelligent algorithms process the data from installed sensors (Including but not limited to: camera, radar, lidar, and ultrasonic sensor). From that, the sample continuously draws maps of its surroundings, which include all visible and predicted obstacles, and uses machine learning to determine the identity of certain objects, whether they be lanes, people, bumpers, traffic signs and lights, etc.
- Controlling all aspects of driving: plot a quick and safe path; command actuators of acceleration, braking, and steering; avoid obstacles; etc.

Driving systems achieving Level 5 of Automation are undergoing development across the world. Nevertheless, they are not yet available for use by the general public due to a variety of reasons including risk management and supply chain complications. An astronomical amount of research and testing will be required to bring us closer to achieving Full Driving Automation. However, with continuous effort and determination to reach the ultimate target of better-serving society and the automotive industry, such a mission is not "science fiction." At a smaller scope, our team wants to simulate self-driving cars through a mini sample.

## 1.3 Project Objectives

In response to the defined project statement, we construct a miniature autonomous vehicle that would successfully perform the following missions to navigate across a track:

- Retrieve images from the camera and send them to the primary processing system for analysis in order to detect its trajectory
- Identify its position and automatically progress between the two black lane lines from the starting point to the finishing destination on the given track. This may consist of performing left and/or right turns accordingly

Through the process of designing and implementing a simple automotive system that meets practical requirements, we yearn to understand the applications of vital engineering and computer science concepts. Acquiring crucial knowledge regarding programming, electrical, and mechanical engineering will help us achieve this course's learning outcomes and become the fundamentals for us to grow in future career paths.

## 1.4 Project Specifications

### 1.4.1 Non-functional Requirement Specifications

Performance

Our control computer has high recognition accuracy and fast processing speed, while the self-driving car model has fast transmission speed and timely response to commands from the control computer. These primary operating characteristics set a premise enabling the sample to run efficiently.

Reliability

The probability of the sample malfunctioning is observed to be pretty high. Some of the components, namely the motors and the camera, have unstable performance. During the experimentation process, one of the motor's rotations cannot keep up with the other's for a few times. Whereas the camera WIFI signal strength fluctuates.

Durability

External frame of the model is robust enough to protect fragile electric components inside. The presence of the heatsink plate assists in protecting the camera from overheating. The Li-ion 18650 batteries have a cycle life of 800 times, which we believe is sufficient for the model to work throughout the span of this project. Yet, since there is a fixed life cycle, we attempted our best to limit recharges and ensure adequate rest time for the batteries between each trial. On top of that, sturdy wheels with rubber tires designed for friction that can be driven independently also play a role in the durability factor.

Serviceability

Components all have great replacing flexibility. Most of them are easy to find and affordable, also uncomplicated and non time-consuming to repair after breakdown. The only exception would be the Power Module VIA B Banh Mi Que: there is extremely limited information about this component online, resulting in difficulties to research its technical specifications or to replace it.

Conformance

As for the image quality, Espressif is among one of the most reputable producers on the chip market, therefore ESP32 Camera has a relatively good performance. This camera model brings together impressive features, including a powerful processor and dual Bluetooth and WIFI connectivity.

Safety is one of the biggest concerns regarding driverless automobiles, so we always take this factor into serious consideration. Our team tried to be cautious in all situations involved in constructing this model, from assembling electrical components to designing.

We utilized the LM 2596 DC Buck Converter to keep the voltage constant to guarantee for the batteries to run efficiently, avoiding potential scenarios of explosion that pose a danger for our team members.

On top of that, we also tried our best abilities to meet environmental standards. We believe that our computation is energy-efficient, alongside the use of recyclable materials for the external design. Furthermore, we aim to reuse the rechargeable batteries even after this project.

<u>Aesthetics</u>

For our external design, we picked lightweight and recyclable materials to minimize the cost as well as to maximize the performance, since the weight will be a burden to the model's efficiency.

### 1.4.2 Electrical Component Specifications

**Camera**[3]

| | |
|---|---|
| Type | Monolithic IC |
| Supply Voltage | 18 V(max) |
| Output Current | 5 V |
| Operating Voltage | 3 V – 15 V |
| Weight | 10 g |

**Motor Driver Circuit RZ 7889**[12]

| | |
|---|---|
| Type | Monolithic IC |
| Supply Voltage | 18 V(max) |
| Output Current | 5 V |
| Operating Voltage | 3 V – 15 V |
| Weight | 10 g |

**Yellow DC Motor**[16]

| | |
|---|---|
| Dimensions | 65*37*22 mm |
| Operating Voltage | 3 V – 12VDC |
| Maximum Torque | 800 g/cm max. @ 3V |
| Gear Ratio | 1 : 48 |
| Load Current | 70 mA(250 mA max. @ 3V) |
| Weight | 29 g |

**High Drain 18650 Battery**[1]

| | |
|---|---|
| Type | Li-ion |
| Dimensions | 18*65 mm |
| Nominal Voltage | 3.7 V |
| Capacity | 3000mAh |
| Charging Voltage | 4.2 V |
| Discharge Cut-off Voltage | 2.75 V |
| Cycle Life | 800 times |
| Weight | 48 g |

**LM 2596 DC Buck Converter**[7]

| | |
|---|---|
| Dimensions | 43*21*14 mm |
| Conversion Efficiency | 92%(max) |
| Switching Frequency | 150KHz |
| Input Voltage | 4 V $\sim$ 35 V |
| Output Voltage | 1.23 $\sim$ 30 V (adjustable) |
| Output Current | 3 A(max) |

# 2 Project Management

## 2.1 Project Plan

During this project, there are many tasks we need to do. These tasks are arranged in sequence and distributed among members to perform. The tasks and completion times are listed in the table below.

| No | Task Name | | | | Description | Duration | Start Date | End Date |
|---|---|---|---|---|---|---|---|---|
| 1 | Create specification | | Create functional requirements | Control Computer | Design Control software that contains Perception, Prediction, Planning, Locating, and Supervising System | 4 days | 14/11/2022 | 17/11/2022 |
| | | | | Autonomous Car Model | Implement Sensors, 3D-printing Frame, ... | 4 days | 18/11/2022 | 21/11/2022 |
| | | | Create non-functional requirements | Performance | The primary operating characteristics of a product. This dimension of quality can be expressed in measurable quantities. | 3 days | 22/11/2022 | 24/11/2022 |
| | | | | Reliability | The probability of a product failing or malfunctioning within a specified time period | 3 days | 22/11/2022 | 24/11/2022 |
| | | | | Durability | A measure of the amount of use one gets from a product before it breaks down and replacement is preferable to continued repair. Durability is a measure of product life. Durability and reliability are closely related. | 3 days | 22/11/2022 | 24/11/2022 |
| | | | | Serviceability | Ease and time to repair after breakdown. Other issues are courtesy and competence of repair personnel and cost and ease of repair. | 3 days | 22/11/2022 | 24/11/2022 |
| | | | | Conformance | the degree to which a product's design and operating characteristics meet both customer expectations and established standards. These standards include industry standards and safety and environmental standards. | 3 days | 22/11/2022 | 24/11/2022 |
| | | | | Aesthetics | How a product looks, feels, sounds, tastes, and smells. The customer response in this dimension is a matter of personal judgment and individual preference. | 3 days | 22/11/2022 | 24/11/2022 |
| | | | | Others | Power consumption | 3 days | 22/11/2022 | 24/11/2022 |
| 2 | Create Plan | | | | Tasks are assigned and deliverables and milestones are determined. | 1 day | 23/11/2022 | 23/11/2022 |
| 3 | Perform block design | | Analyze the requirements | | This step involves high level functional design. A number of designs are considered and evaluated. | 1 day | 24/11/2022 | 24/11/2022 |
| | | | Design block diagram of the system | | | 2 days | 25/11/2022 | 26/11/2022 |
| 4 | Design each block and select best alternatives | Control computer | Design | | -Design each block: In this step, detailed design with alternatives for each part of the system are carried out.<br>-Select best alternatives: In this step, the most promising design among the alternatives is selected. In order to make an informed choice, engineers perform very detailed calculations and analysis to verify that the designs will work as planned. | 4 days | 27/11/2022 | 30/11/2022 |
| | | | Test | | | 1 day | 01/12/2022 | 01/12/2022 |
| | | | Select the best alternative | | | 2 days | 02/12/2022 | 03/12/2022 |
| | | Self-driving Car model | Design | | | 4 days | 04/12/2022 | 07/12/2022 |
| | | | Test | | | 1 day | 08/12/2022 | 08/12/2022 |
| | | | Select the best alternative | | | 2 days | 09/12/2022 | 10/12/2022 |
| 5 | Test | Integrate the system | | | This step involves intensive test on the product including functional testing, performance, durability and safety testing ... under different testing conditions. | 5 days | 11/12/2022 | 16/12/2022 |
| | | Test using simulation | | | | 10 days | 17/12/2022 | 27/12/2022 |
| | | Test with prototype | Order components | | | 5 days | 27/12/2022 | 31/12/2022 |
| | | | Create prototype | | | 10 days | 01/01/2023 | 10/1/2023 |
| | | | Test the prototype | | | 10 days | 11/01/2023 | 20/01/2023 |
| 6 | Manufacture | Manufacture subsystem 1 | | | In this step, the product is fabricated/implemented. | 10 days | 21/01/2023 | 31/01/2023 |
| | | Manufacture subsystem 2 | | | | 10 days | 01/02/2023 | 10/02/2023 |
| 7 | Deliver | Write report (user manual) | | | Write the eventual report of the whole products | 10 days | 11/02/2023 | 21/02/2023 |
| | | Present and demo the system | | | Present the final with the customers | 1 day | 24/02/2023 | 24/02/2023 |

## 2.2 Contribution of Team Members

In this project, all tasks and workloads were carefully assigned to each individual. Each team member was assigned tasks depending on their capacity to perform and deliver them on time.

The table below displays the assignments, the individuals that completed them, and the percentage of contributions from their side in each section.

| No | Task Name | | | | Delivarables | Assigned to | Contribution | Predecessors |
|---|---|---|---|---|---|---|---|---|
| 1 | Create specification | | Create functional requirements | Control Computer | High accuracy control | Khoa | 100% | Research, analyze, and brainstorm |
| | | | | Autonomous Car Model | A frame from 3d printer | | 100% | |
| | | | Create non-functional requirements | Performance | The primary operating characteristics of a product. | Ngọc | 100% | |
| | | | | Reliability | The probability of a product failing or malfunctioning within a specified time period | | 100% | |
| | | | | Durability | A measure of the amount of use one gets from a product before it breaks down and replacement is preferable to continued repair. | Dũng | 100% | |
| | | | | Serviceability | Good preparation | | 100% | |
| | | | | Conformance | Understand the requirements | Linh | 100% | |
| | | | | Aesthetics | Completed car model design | | 100% | |
| | | | | Others | A sustainable power | Thanh | 100% | |
| 2 | Create Plan | | | | Detail plan | Ngọc | 100% | Discussion and suitable-task distribution |
| 3 | Perform block design | | Analyze the requirements | | Detail requirment and detail design | Khoa | 100% | Research for data and then apply to create system |
| | | | Design block diagram of the system | | | Dũng | 100% | |
| 4 | Design each block and select best alternatives | Control computer | Design | | Best alternative of control computer | Thanh | 100% | Idea, references, list of criteria to choose best block |
| | | | Test | | | Linh | 100% | |
| | | | Select the best alternative | | | | 100% | |
| | | Self-driving Car model | Design | | Best alternative of car model | Ngọc | 100% | |
| | | | Test | | | Khoa | 100% | |
| | | | Select the best alternative | | | | 100% | |
| 5 | Test | Integrate the system | | | Necessary data | Thanh | 100% | Simulation, data, outcome standard |
| | | Test using simulation | | | | | 100% | |
| | | Test with prototype | Order components | | | Dũng | 100% | |
| | | | Create prototype | | | | 100% | |
| | | | Test the prototype | | | | 100% | |
| 6 | Manufacture | Manufacture subsystem 1 | | | Completed product | All | 100% | Supportive machines (3D printer, electrical devices, sensors), algorithms, blueprint |
| | | Manufacture subsystem 2 | | | | | 100% | |
| 7 | Deliver | Write report (user manual) | | | Final report | | 100% | Recorded data, images, description of products, plan |
| | | Present and demo the system | | | | | 100% | Scripts and slides |

## 2.3 Project Execution Monitoring

To make the process of completing the project go smoothly, we have held many online and offline meetings. Through the meetings, we have more time to discuss how to proceed with the project thoroughly, the problems we encountered during the process, and how to overcome them. The table below shows our group's events and meeting times.

| Time/Date | Activities/Events |
|---|---|
| Biweekly | Meet with group members |
| 10/01/2023 | Finish 1st prototype |
| 10/02/2023 | Test the system |
| 22/02/2023 | Finish Final Prototype |
| 24/02/2023 | Demo the final product |

## 2.4 Challenges and Decision Making

### 2.4.1 Challenges

**Camera**:
The camera is one of the most important components of an autonomous car. We faced several challenges while testing the camera. The camera's wifi was intermittent, which caused inconvenience during our test drive. In addition, the camera's poor quality caused road detection to be delayed, leading to the car going out of the lane. This delay is due to the camera overheating when it works continuously. Moreover, we must find a suitable angle to capture the most appropriate line for the car to follow.
In addition, there was a challenge while testing the car: the camera fixed axle is very brittle and easy to break; during the test, the car had a few problems causing the shaft to break.

**Battery**:
The battery is a crucial part of the car's circuit; and it has the potential to affect the whole project, including software and hardware. The 18650 battery is a 3.7V battery, which is insufficient to power the car's circuit. The circuit needs 5V to function correctly. Using only one 18650 battery caused the motor to run weakly and unevenly between the two motors. However, using a voltage higher than 5V caused the camera and circuit to overheat and burn wires.

**3D Printing Technique**:
3D printing is an essential technique in the creation of an autonomous car model. However, we faced challenges while using it. We were unfamiliar with the technique, and it took us a long time to become accustomed to it. This unfamiliarity caused delays in our project.

**Time Constraints**:
The final exams coincided with the period we were creating the autonomous car model, which made it overwhelming to complete the project. As a result, we had less time to devote to the project, which resulted in a less-than-satisfactory final product.
Creating an autonomous car model has its challenges. Our team faced several challenges

during the project. The camera's intermittent wifi, poor quality, overheating, and battery voltage were some of the issues we faced. We also needed help with 3D printing and time constraints due to overlapping final exams. These challenges affected the quality of our final product; therefore, we came up with some decisions that improved our model.

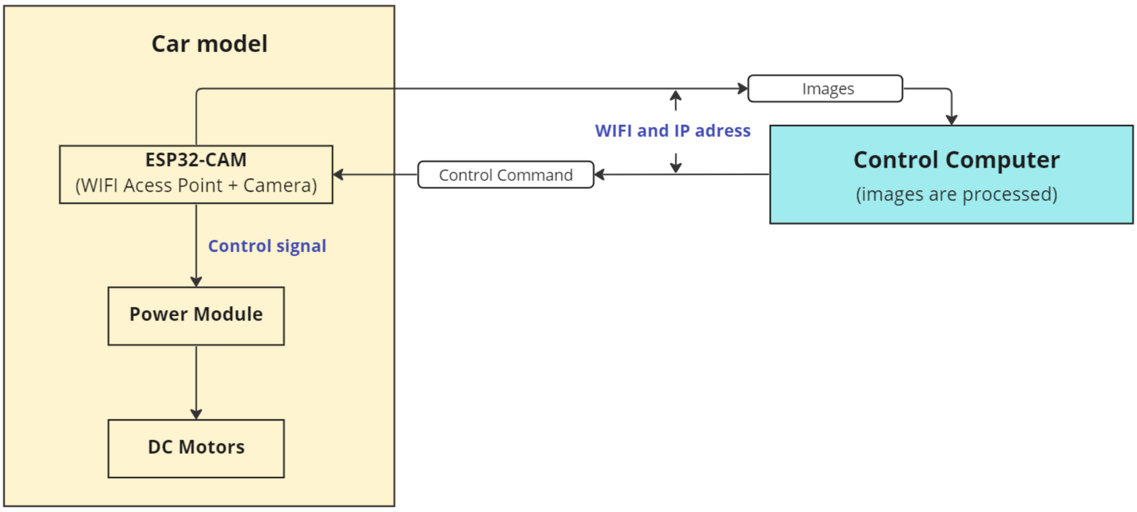
### 2.4.2 Decision Making

**Camera**:
One of the challenges we faced was lag and intermittent wifi connectivity with the camera, despite our code being reasonable. To address this issue, we contacted the Teaching Assistant to request a new camera. The new camera was more sensitive and reduced delay time. Additionally, we found a suitable angle for the camera to detect the best path, and we fixed it. These changes resulted in more accurate road detection and improved performance of the car.
To solve the problem of the camera shaft, we had to use a glue gun to fix the shaft, but we found that this also did not make the shaft too stable, so we planned to reprint the shaft with a 3D printer.
**Battery**:
To power our car, we decided to use two 18650 batteries with a voltage of 3.7V each. However, since the car's circuit needs 5V to function properly, we used the DC-DC low-voltage module LM2596 to provide the required voltage. After changing the power supply to 5V, we noticed a significant improvement in the car's performance. It ran much smoother, and the camera lag and delay were significantly reduced.
**Design and 3D Printing Technique**:
We tried to learn 3D printing techniques from upper-level students and learn more independently. After studying and researching, our team was also able to understand 3D printing technology. We also use stronger plastic straps to increase the durability and beauty of the car.
**Teamwork and Time Management:**:
By dividing the work properly, we were able to pass the final exam smoothly and focus more on the project. We also had more time to perfect our car, thanks to the extended deadline checkpoint and car demo.

Overcoming the challenges in developing an autonomous car requires problem-solving skills, communication, and teamwork. By making the right decisions, learning from others, and proper time management, we were able to create a successful autonomous car project.

# 3 System Design

## 3.1 Block diagram of the system




We use PlatformIO to install the firmware for the board, which is in Python, to connect the ESP32-CAM to the board and Control Computer through WIFI and IP address. ESP32-CAM on the car will receive images and send them back to the main computer where they are processed by OpenCV and algorithms, which is followed by control commands. These commands are created from the computer and the car will receive them through the ESP32-CAM. Then Control signals will be created and directed to the power module, finally the DC Motors will operate according to signals.

## 3.2 Design of each block and select the best alternative

**SELECTION MATRIX**

| Criteria | Alternatives | | |
|---|---|---|---|
| | First selection | Second selection | Third selection |
| Battery | Use one battery (3.7V, 3000mAh) | Use two batteries and a Step-down module | Use one big battery |
| Lane detection algorithm | Use one horizontal line | Use two horizontal lines | |
| Car cover frame | 3D printing | Carton design | Nothing |
| Motors | 4 Wheels | 2 Wheels with 1 front ball | |
| Camera | Old camera | Brand new one | |

The second selections are those that we chose optimally for each category

- Battery: We tried to use one chargeable battery but it did not provide enough power to run the car smoothly so we changed it to a set of 2 batteries. As a result, this made our input voltage 7.4V but the needed output voltage is 5V so we use a step-down module which fits reasonably. Big battery may also offer enough energy for running. However, this will increase the total weight of the car and not bringing too many differences whereas we need to make some changes to the design. Therefore, it is not the best option.

- Algorithm: The original algorithm was unstable which made the car sometimes can not turn at curves properly. We then developed the algorithm by adding one more detection line which then worked perfectly.

- Frame: For the frame of the car, we can use 3D printing or design a carton case or simply not use anything. We firstly chose to print a 3D model because although it is heavier than the cartone one, it is way more durable and long lasting. It is also more flexible to design and better-looking. However, due to technical issues of the 3D printer, we cannot print the car frame before the deadline. Therefore, we decide to change to the carton design with the advantages of easy to build and convenient.

- Motors: Although the participating of 4 wheels give a better control for the car and make it move better on different types of surfaces, it is more complicated to code for running the system. In addition, when we only have two wheels and one ball (code both wheels), we have to redesign the board if we want to change it to 4 wheels and the difference between two designs is not worth it. The basic design with 2 wheels can handle the competition well.

- Camera: Our first camera was terribly delayed after using it for a while and its connection was somewhat unstable. So we changed it to a brand new one which works perfectly fine.

## 3.3 Testing of each block

### 3.3.1 Battery

**a. Voltage**

First of all, our team was received a single battery with the voltage of 3.7V. However, after the process of testing, we found out that the car model did not perform well as it just ran for a short distance before immediately stopping. Our team think that this was the problem with the power supply as when we connected the model with the laptop power, the problem seemed to be solved. Then, after researching more information, we realized that the board generally operates at 5V. Because of the gap between the preferred and actual voltage, as well as the fact that the ESP-32CAM and 2 DC Motors require a certain amount of electricity, the performance did not meet our expectation. Therefore, we decided to increase the voltage by adding on another battery with the same voltage improving the total voltage to 7.4V. Then, we connect the set using Double battery base 18650.

However, to ensure that the voltage did not overcome the limitation, which is 5V, we utilize a module DC-DC LM2596 (image), which could reduce the output voltage to the suitable level.


For testing the output, we use the Multi-meter and adjust the variable resistor until the outcome met our expectations.
There is one note that we need to aware about the LM2596, which is it only maintains one limit. While the input voltage is lower than 5V, the output will fall either (for the LM2596, the input voltage must be higher than the output). As a result, the whole

system will be affected. For example, when the requirement of voltage is not satisfied, the camera may get disconnected. When we ran our car in a duration of time, roughly about 1 hour, the battery was seemed to be exhausted, thus led to the decline in overall performance. Therefore, carefully calculating the execution time is extremely important in order to avoid the mentioned issue.

**b. Capacity**

The capacity of one single battery is 3000mAh, which illustrates the maximum power it can contain. In this case, 3000mAh means that the supplier can provide the electricity of 3000 milliamps in 1 hour before totally running out of energy. However, to ensure the performance, we use two cells that contains 2 batteries, bringing the total capacity to 6000mAh.

Regarding the drain (load) of the battery set, it is quite complicated to calculate exactly. However, based on the process of running, our team saw that this power supply could ensure for about 1 hours of operating normally. The actual duration could be more than this assumption, however, long time execution accompanied with high temperatures of the module as well as the decline of image quality from the camera. Therefore, we set this limitation for our execution. In case we have the exact information about the drain (load) of the battery set, we could implement this calculation:

Time discharging= Capacity of the battery / load rate (in the same unit)

### 3.3.2 Motor and Wheel

In our design, DC Motors and Back Wheels are in charge of physical movement of the car model. To be more specifically, the two yellow motors would receive the signal from the controller and based on that to rotate the rear wheel. At the front of the car, there is no rear wheel but a rotating wheel (a small ball) to corporate with the movement force from the back wheels. However, this design has some disadvantages. First of all, the car model will meet some difficulties when moving through rough or slippery environments. Furthermore, because of the differences in fundamental construction, the two wheels do not moving in a straight line even if they have the same rotating speed. Therefore, we need to use many set of parameters for testing to figure out what is the set that brings the most stable performance. In addition, because the direction of the car is mainly based on two back wheels, it is complicated to calculate the suitable speed of those two when turning. Doing so requires one wheel running with low speed while the other one accelerating. We also needed to implement many trial runs to find the most suitable speed attribute.

```python
def calculate_control_signal(left_point, right_point, im_center):
    if left_point == None or right_point == None:
        left_motor_speed = right_motor_speed = 80
        return left_motor_speed, right_motor_speed

    # Calculate difference between car center point and image center point
    center_point = (right_point + left_point) // 2
    center_diff = center_point - im_center

    # Calculate steering angle from center point difference
    steering = -float(center_diff * 0.03)
    steering = min(1, max(-1, steering))
    throttle = 0.8

    # From steering, calculate left/right motor speed
    left_motor_speed = 0
    right_motor_speed = 0

    if steering > 0:
        left_motor_speed = throttle * (1 - steering)
        right_motor_speed = throttle
    else:
        left_motor_speed = throttle
        right_motor_speed = throttle  * (1 + steering)

    # Adjust the parameters accordingly to the requirements
    left_motor_speed = int(left_motor_speed * 100)
    right_motor_speed = int(right_motor_speed * 100)
    diff = abs(left_motor_speed - right_motor_speed)
    if left_motor_speed < 80:
        left_motor_speed = 80
        right_motor_speed+= diff // 5.25

    if right_motor_speed < 80:
        right_motor_speed = 80
        left_motor_speed+= diff // 5.25

    return int(left_motor_speed), int(right_motor_speed)
```

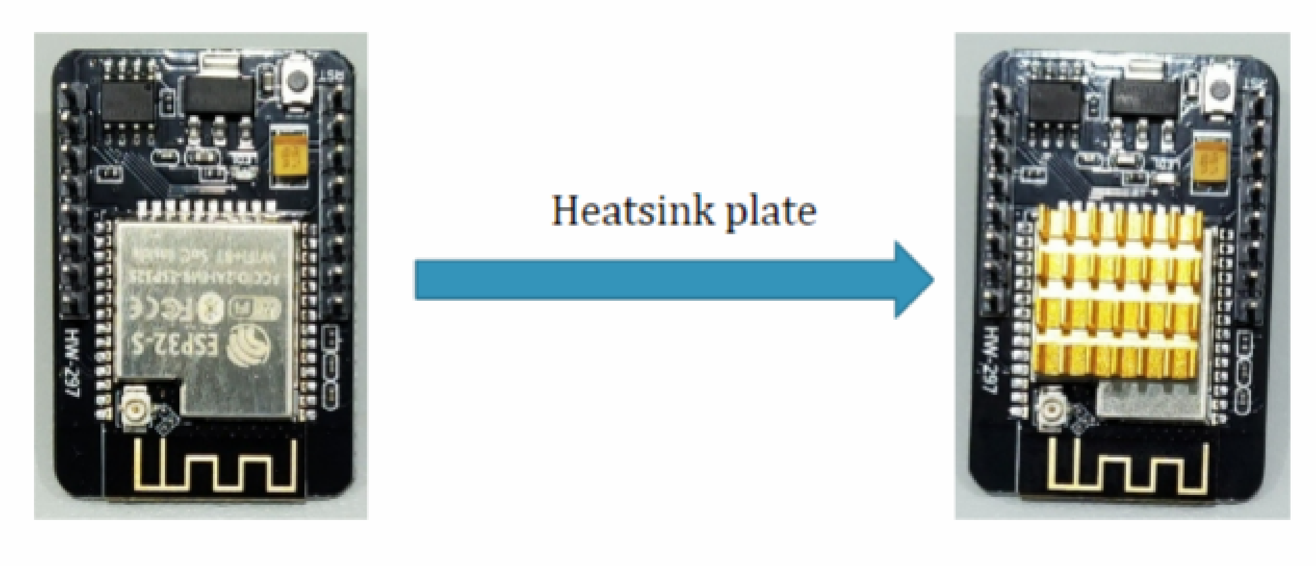
### 3.3.3 Camera

Initially, the camera had two main challenges. First of all, in order to track the black line, the camera need to stay in the correct height and angle where it can catch the lane easily. If the camera vision is too short, it may run away from the road before recognizing the corner. In the other hand, if it is too far, the camera may receive many distracting objects and cannot observe the surrounding lines, which may make it run into errors. Therefore,

the camera needs to be placed highly but bowing down in a suitable corner. Doing so will ensure that it can track the line correctly and instantly if there are any upcoming lane switches. After many modifications, we have found a suitable position for our camera, which is approximately 15 cm height from the base and bowing down at an angle of 34 degrees. In this case, the visible distance is about 20cm far from the head of the car.
Another problem was the delay of received images from the camera. Since an autonomous car requires instant and continuous updating of data from the sensors, the difference in terms of time between the input images and the reality is a huge obstacle. Therefore, we have tried to solve this issue in many ways such as using the car in an acceptable period of time because the longer the running time is, the worse camera performs. However, the issue seems to be addressed when we modified the "platformio.ini" file of the firmware and replaced the line "platform= espressif32" with "platform = espressif32@3.5.0". After reloading the firmware to the camera, the performance has improved significantly.
There are also some troubles with the WIFI connection. Sometimes we could not access to the WIFI of the camera, or its quality was terrible. Then, we figured out that using smartphone nearby might affect the connection as well as long distance between the camera and the laptop. When we avoided doing so, the running quality was improved clearly.

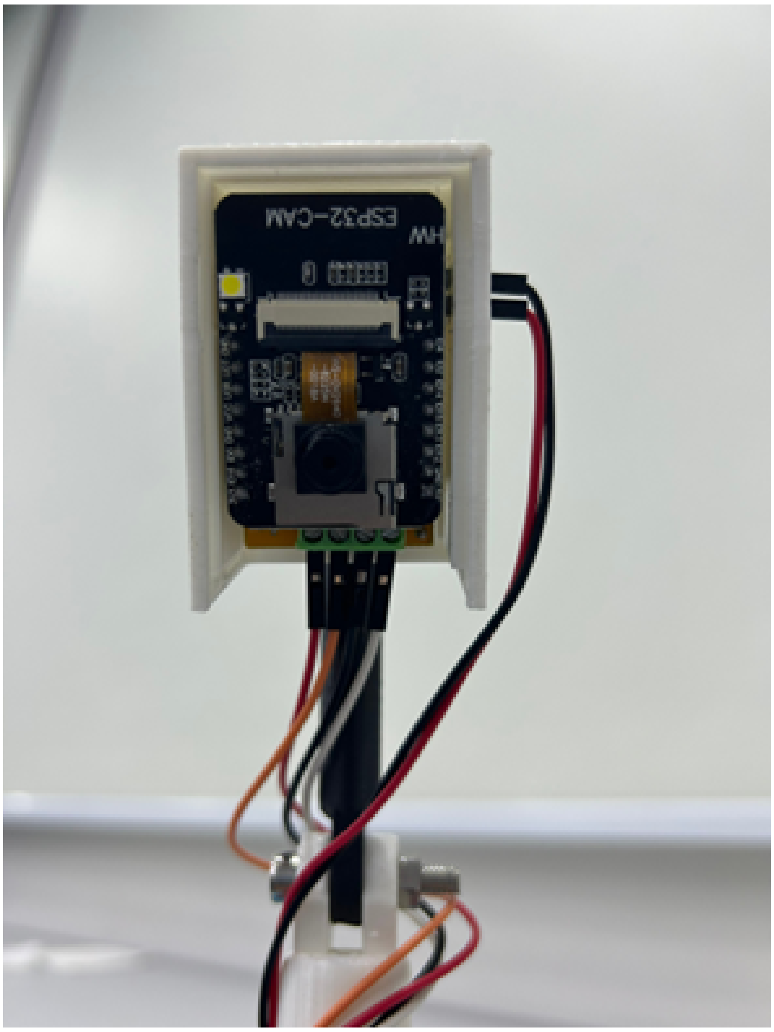
## 3.4 Manufacturing and assembly

First of all, we place the heatsink plate on the Module ESP32 CAM in order to reduce the heat when operating.

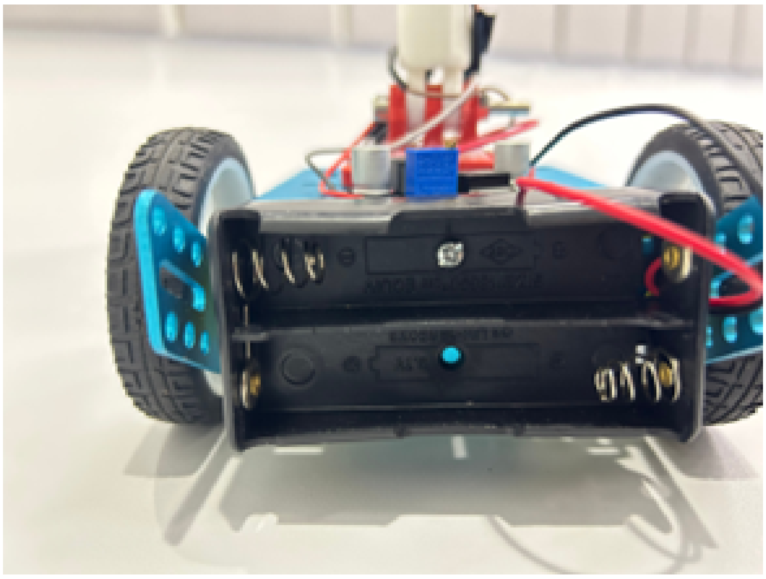



Then, we attach the ESP32 Module with the Power module VIA B.

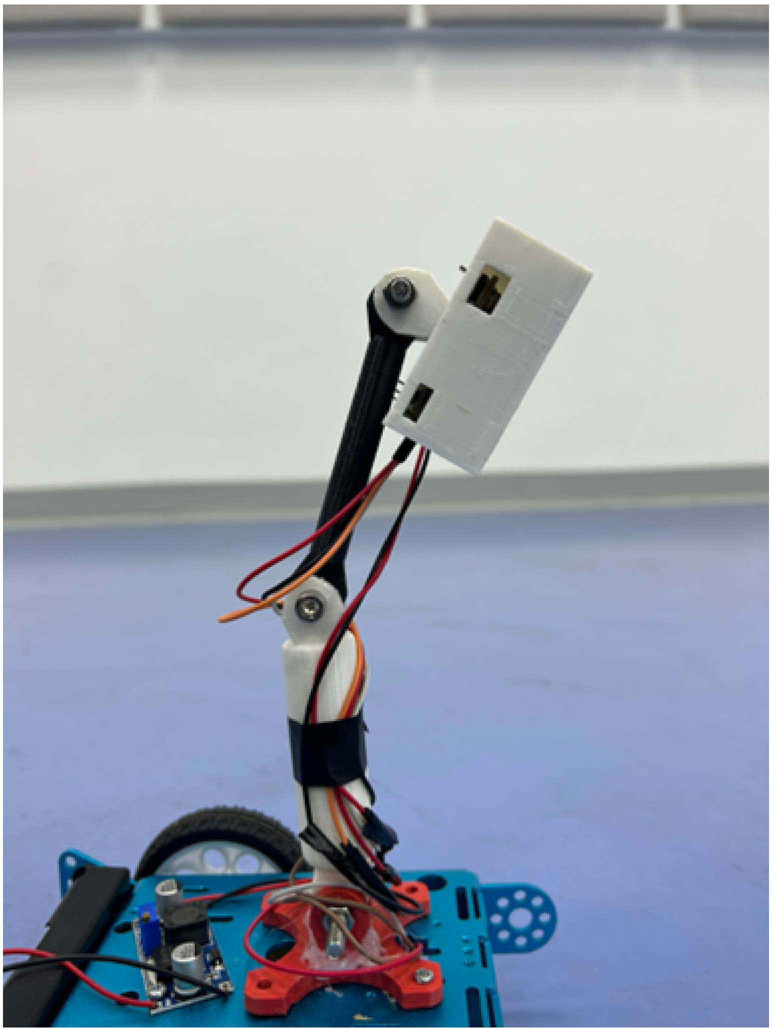

After that, we use the wire to link the power supplier and two motors with the module.



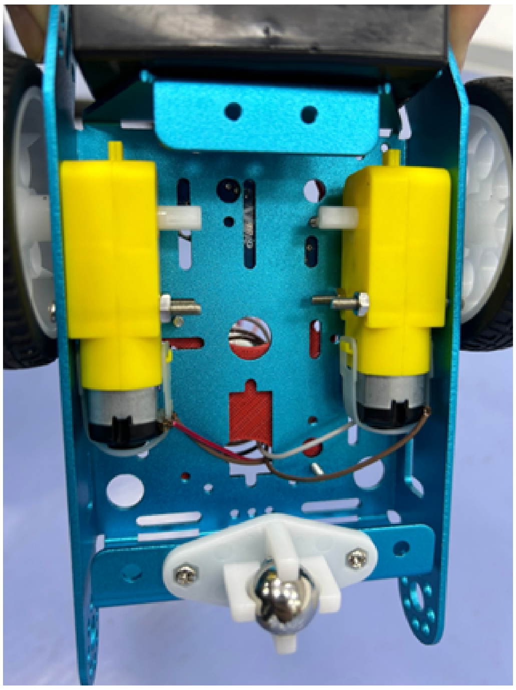
Subsequently, we fixed the power cell at the back of the car.



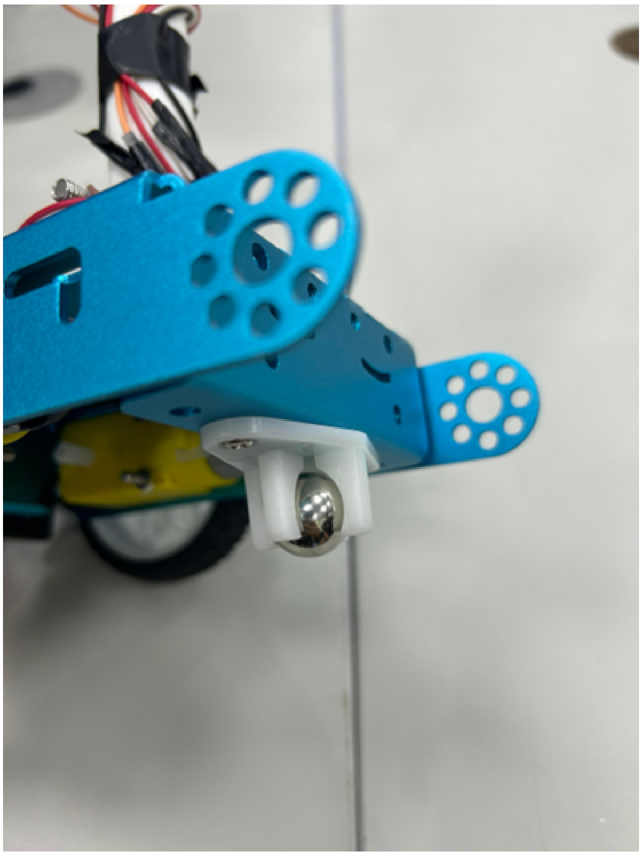
Then, we stick the ESP32 module to the support shaft and modify the height and angle that satisfies the requirement of camera image.



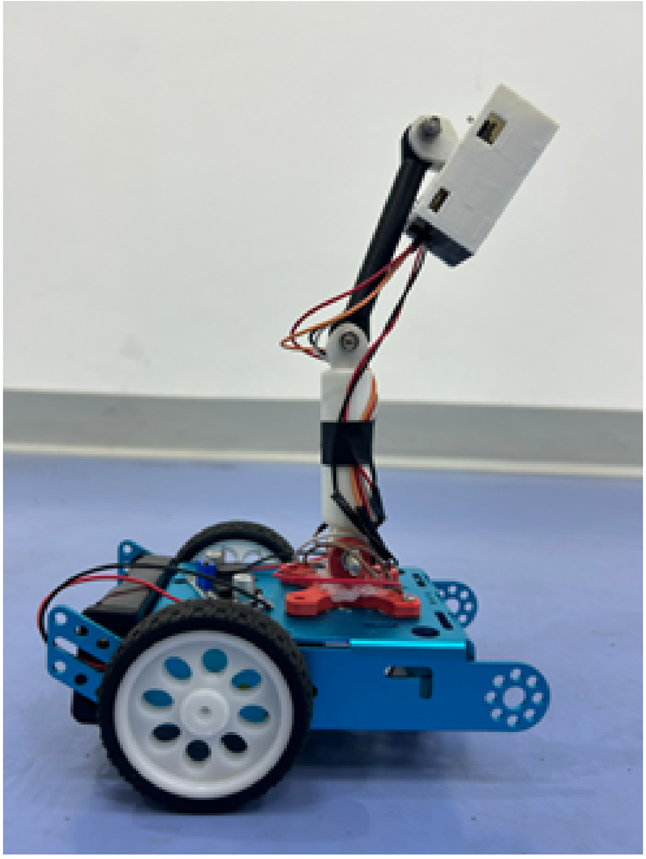
Meanwhile, we also attach the two motors and wheels with the car base, while connecting to the power module by wire.



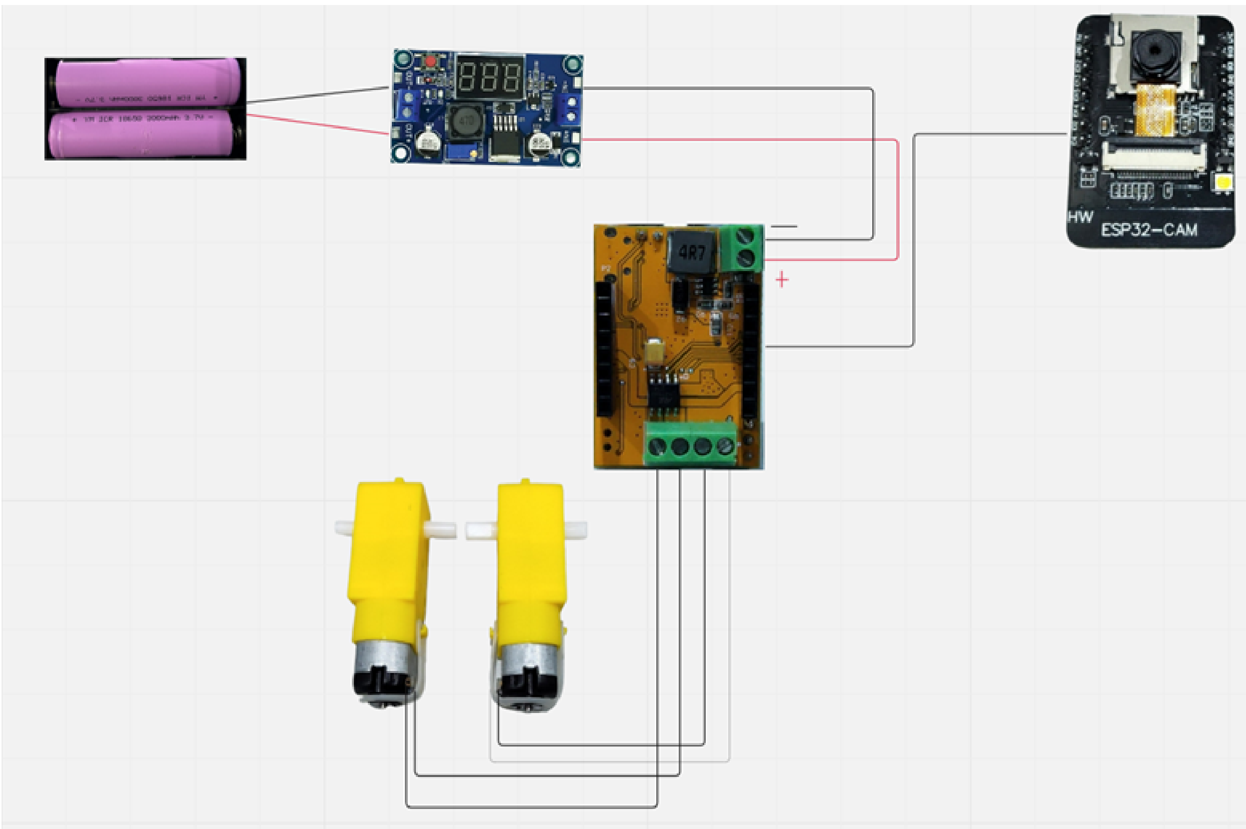
In addition, because we only use two motors at the back, the responsibility of supporting the front part is given to the rotating wheel.



This is the basic structure of the car with all components attached.



Below is the diagram of all the electrical components linked together.




Finally, we prepared the cover frame using carton paper. Then, we made some aesthetic features and added some colors for making the whole car become realistic and satisfying.

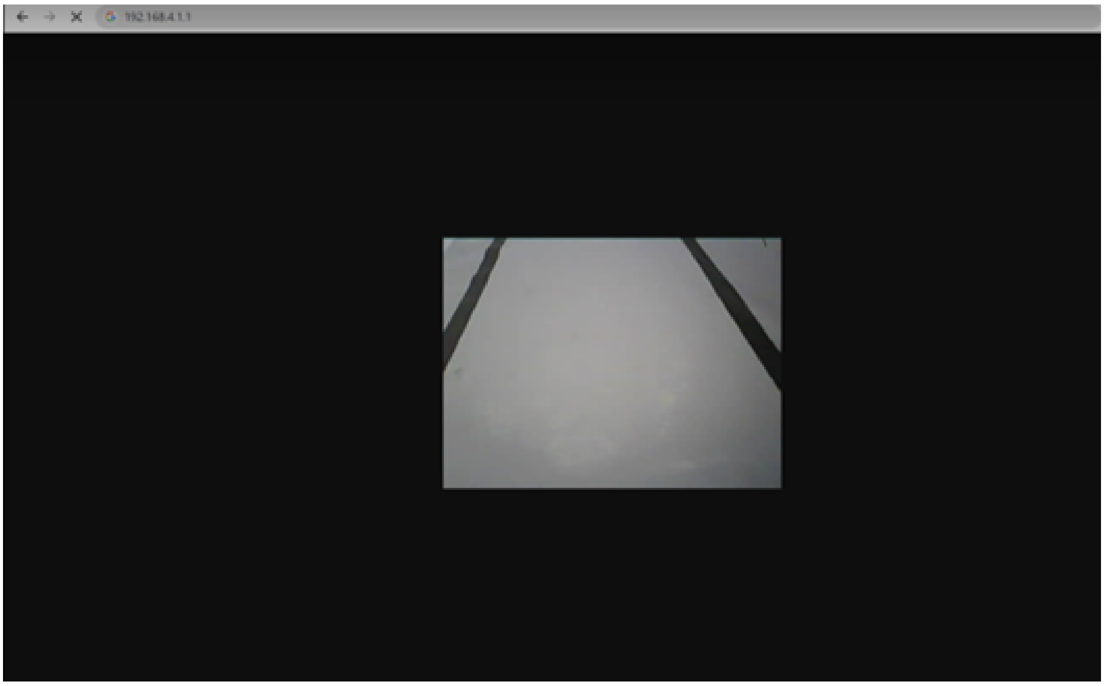
The final version of our car is shown below.

# 4 System Testing and Analysis

## 4.1 System testing

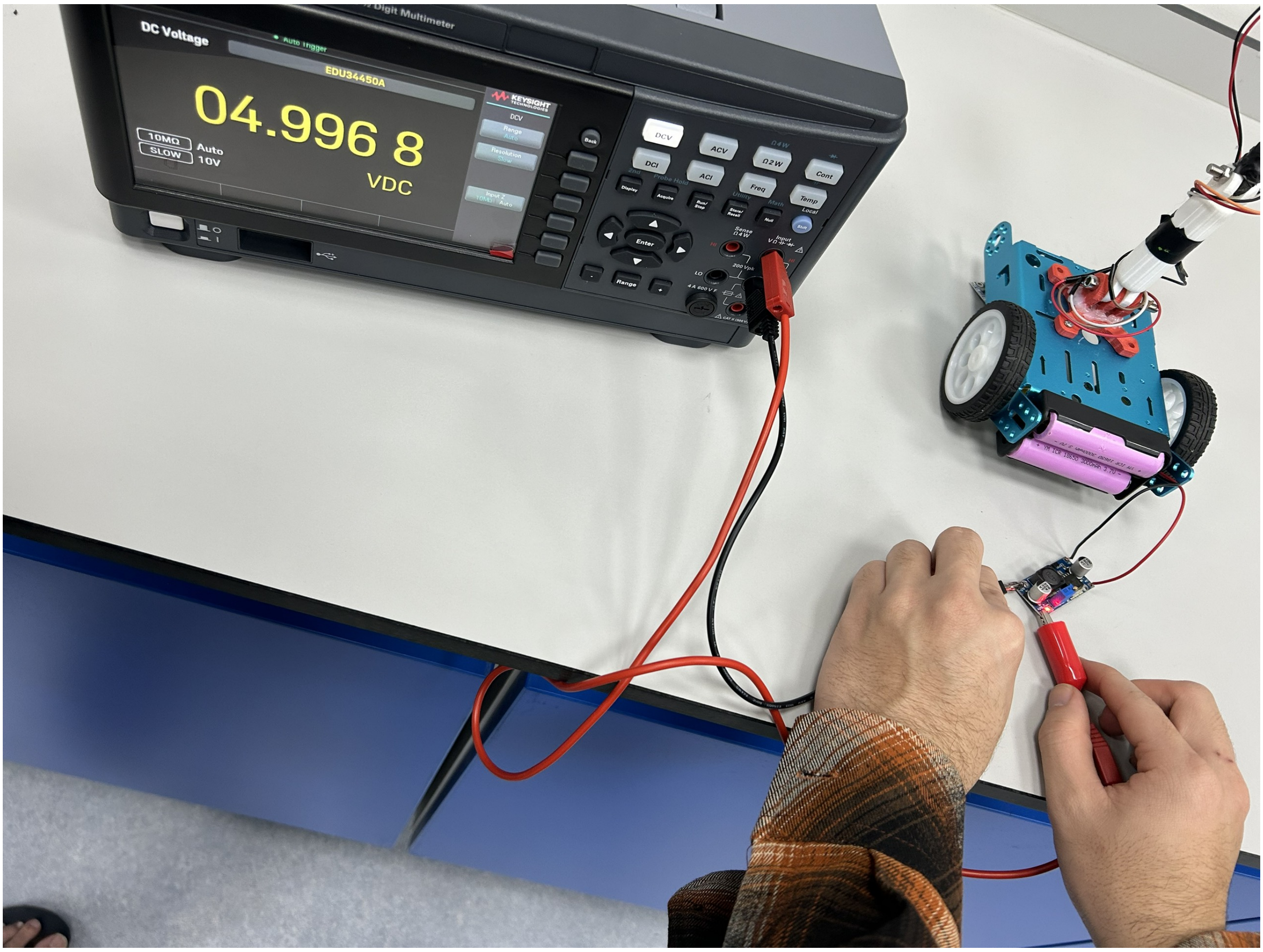
### 4.1.1 Test the image quality

Initially, we need to check the quality of the camera outcome. To do this, we need to connect to the WIFI outputted by the module. Then, we access the IP address: 192.168.4.1, which can help us observe the image directly as below.




### 4.1.2 Test the power supplier

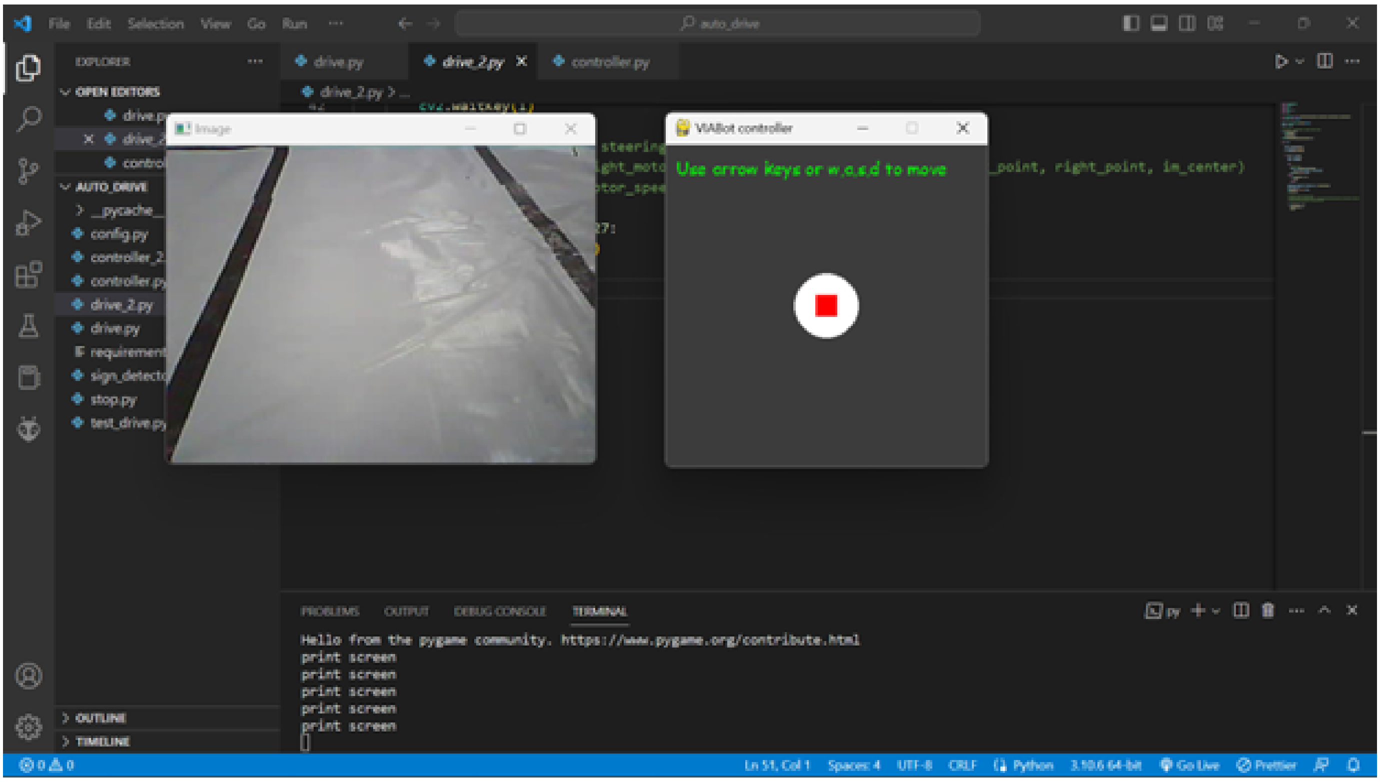
With the LM2596 module, we can assure the input voltage to a suitable level for the system to operate. Because the limitation of the decided voltage is 5V, while the total of the battery set is 7.4V, we need to modify it until reaching the target. For checking this requirement, we utilize the multi-meter for testing. The measurement is demonstrated below.




### 4.1.3 Test the operation

This is one of the most important steps for starting the car. First of all, we need to connect to the ESP32-CAM's WIFI and evaluate the connection quality. In some cases, we even could not access the WIFI or it would automatically disconnect. Therefore, carefully checking this step is essential for risk management when implementing.

Then, we run the keyboard control example using Python. If the car performs smoothly and moves in the same direction as the keyboard arrows without any unexpected errors, it is ready for the next phase.




### 4.1.4 Overall test

After finishing the initial test for each component, we combine them all and begin to upload the sample code for the real implementation on the map. However, there are still further steps to making this car model completely autonomous. These phases will be mentioned in the next part.

## 4.2 Results, Analysis and Discussion

### 4.2.1 Original algorithm

**Step 1:** Preprocess the image received from the camera:

```python
def preprocess(img):
    """Preprocess image to get a birdview image of lane lines"""
    img = grayscale(img)
    img = gaussian_blur(img, 3)
    img = canny(img, 100, 200)
    img = birdview_transform(img)
    return img
```

**Step 2:** Find lane lines:

```python
    # Determine left point and right point
    left_point = -1
```

```python
    right_point = -1
    lane_width = 100
    center = im_width // 2
    for x in range(center, 0, -1):
        if interested_line[x] > 0:
            left_point = x
            break
    for x in range(center + 1, im_width):
        if interested_line[x] > 0:
            right_point = x
            break
```

On a horizontal line, the left and right points will be the closest non-black pixel to the left and right, respectively, of the middle point (the center of the image).
If one of them is not detectable, we will use the detectable one to predict it.

t

```python
if left_point != -1 and right_point == -1:
        right_point = left_point + lane_width
if left_point == -1 and right_point != -1:
       left_point = right_point - lane_width
```

**Step 3:** Calculate control signals

t

```python
def calculate_control_signal(left_point, right_point, im_center):
    # Calculate difference between car center point and image center point
    center_point = (right_point + left_point) // 2
    center_diff = center_point - im_center
    # Calculate steering angle from center point difference
    steering = -float(center_diff * 0.03)
    steering = min(1, max(-1, steering))
    throttle = 0.5
    # From steering, calculate left/right motor speed
    left_motor_speed = 0
    right_motor_speed = 0
    if steering > 0:
        left_motor_speed = throttle
        right_motor_speed = throttle * (1 - steering)
    else:
        left_motor_speed = throttle * (1 + steering)
```

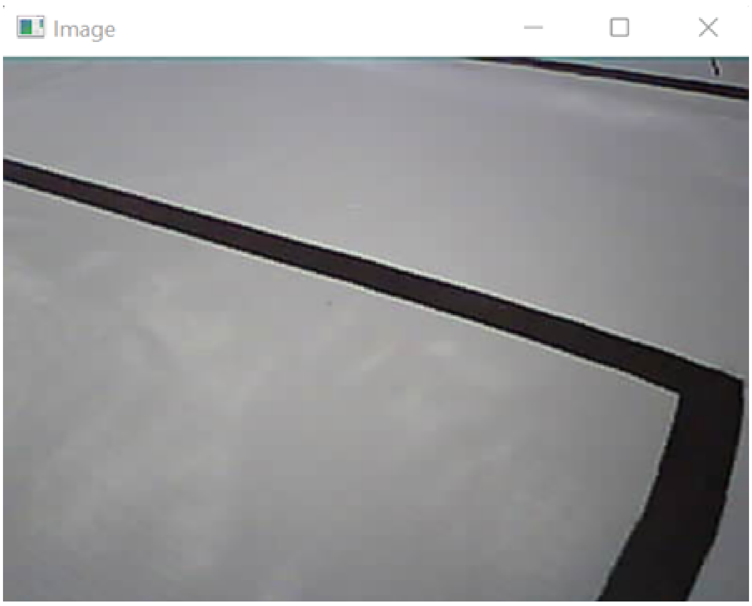
```python
            right_motor_speed = throttle

    left_motor_speed = int(left_motor_speed * 100)
    right_motor_speed = int(right_motor_speed * 100)

    return left_motor_speed, right_motor_speed
```

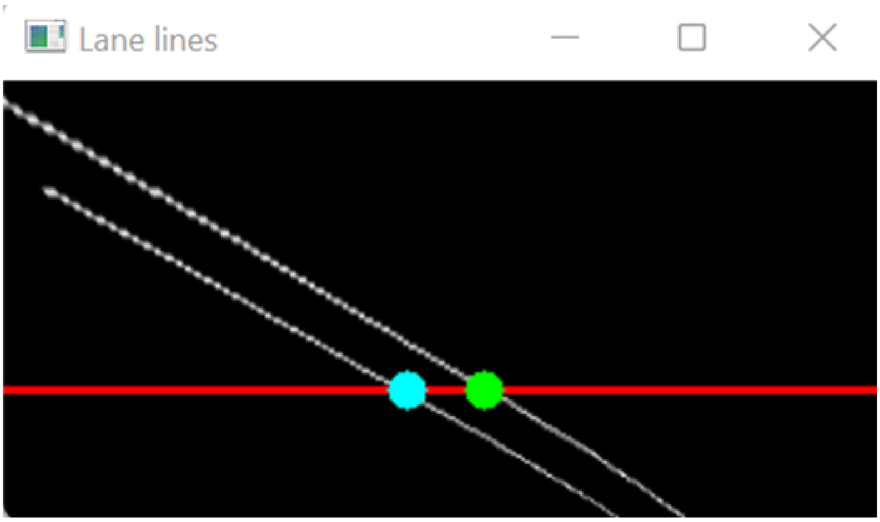
**Step 4:** Send control signals
t

```python
def send_control(left_motor_speed, right_motor_speed):
    control_msg = "CONTROL_WHEEL {} {}".format(
        left_motor_speed, right_motor_speed).encode('ascii')
    sk.sendto(control_msg, (CONTROL_IP, CONTROL_PORT))
```

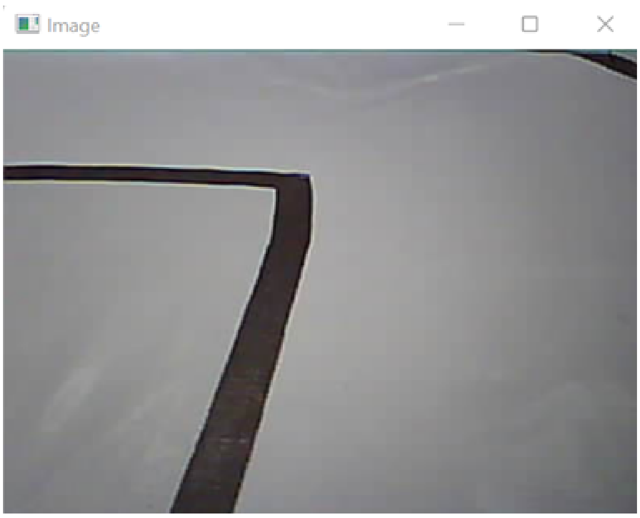
### 4.2.2 Problems Solutions

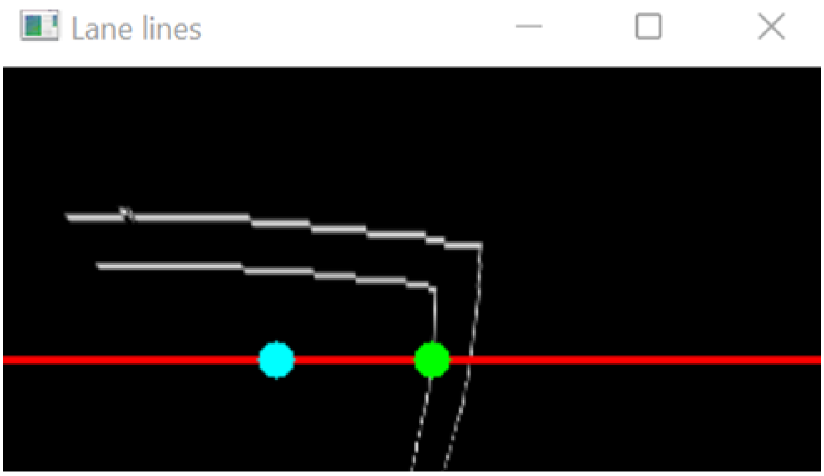
At turns, lane lines will be falsely detected. Because in the step 2 of the original algorithm, the left and right points are the nearest non-black pixels found from the middle point, so when the middle point lies at inappropriate spots, the left and right endpoints will also be falsely detected. The figures below show that the middle point lies on one of the lane lines.



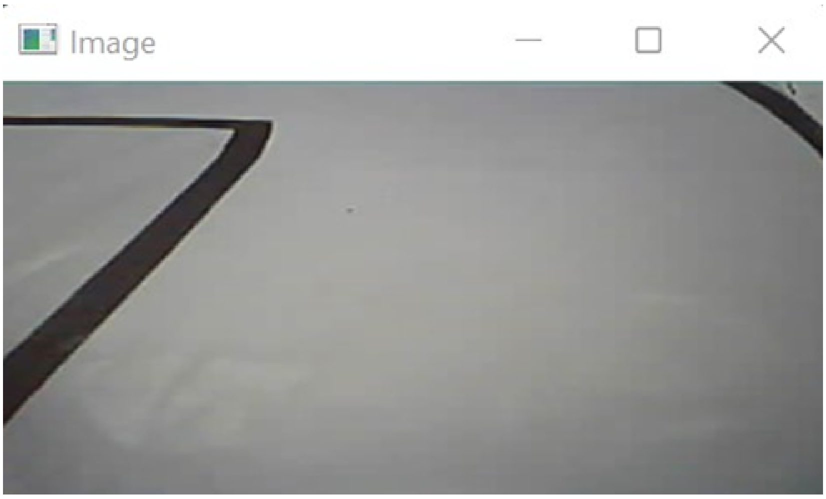


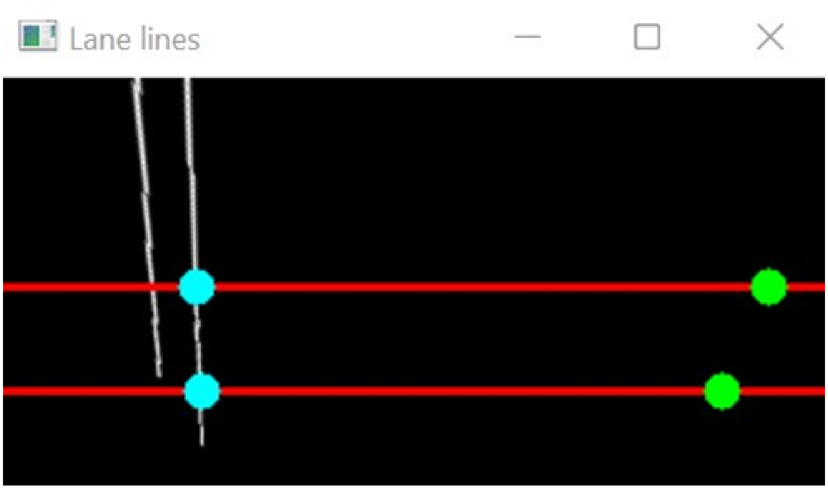

Or sometimes the middle point is outside of the lane:




At turns, the newly detected left and right points will diverge greatly from the previous ones. This will lead the steering angle to suddenly change significantly, resulting in going off the lane. To prevent such occurrence, multiple horizontal lines are used to better predict an appropriate path.




The goal of our solution is to ensure left point and right point are seen at all the time. In order to improve the accuracy of the lane lines detection in the 'current frame', we use the data we had in the 'previous frame'.

Let 'curLeft' and 'curRight' are two points detected in the 'current frame', 'preLeft' and 'preRight' in the 'previous frame'. We have to ensured that the distance between two points is large enough, to avoid the case in Figure 2. Otherwise, we will estimate new two points that are as close as possible to the previous 'preLeft' and 'preRight' points.

We also handle cases where 'curLeft' or 'curRight' is not detected, the stored data of 'preLeft' and 'preRight' will be used to ensure that the difference between the positions of the current points and the previous points are not too large.

t

```
preLeft = [None] * number_of_lines
preRight = [None] * number_of_lines
def fix_point(type, curLeft, curRight, lanewidth):
    left = curLeft
    right = curRight
    if preLeft[type] != None and preRight[type] != None:
        if curLeft != None and curRight != None:
```

```
        if curRight - curLeft >= lanewidth - 50:
            left = curLeft
            right = curRight
        else:
            curMin = abs(curLeft - preLeft[type])
            left = curLeft
            right = left + lanewidth
            if abs(curRight - preRight[type]) < curMin:
                curMin = abs(curRight - preRight[type])
                right = curRight
                left = right - lanewidth
            if abs(curLeft - preRight[type]) < curMin:
                curMin = abs(curLeft - preRight[type])
                right = curLeft
                left = right - lanewidth
            if abs(curRight - preLeft[type]) < curMin:
                curMin = abs(curRight - preLeft[type])
                left = curRight
                right = left + lanewidth
    else:
        if curLeft == curRight == None:
            left = preLeft[type]
            right = preRight[type]
        elif curLeft == None or curRight == None:
            if curLeft == None:
                if (abs(curRight - preRight[type])) <= abs((curRight - preLeft[type])):
                    right = curRight
                    left = right - lanewidth
                else:
                    left = curRight
                    right = left + lanewidth
            else:
                if (abs(curLeft - preLeft[type])) <= abs(curLeft - preRight[type]):
                    left = curLeft
                    right = left + lanewidth
                else:
                    right = curLeft
                    left = right - lanewidth
        else:
```

```
                left = curLeft
                right = curRight
    else:
        if curLeft == curRight == None:
            return None, None
        if curLeft == None or curRight == None:
            if curRight == None:
                left = curLeft
                right = left + lanewidth
            else:
                right = curRight
                left = right - lanewidth
    preLeft[type] = left
    preRight[type] = right
    return left, right
```

# 5 Conclusion and Recommendation

## 5.1 Conclusion

The position of automated driving systems is undeniable in our 4.0 era and their application is warmly welcomed in the new normal. There exists the need to continuously research and develop self-driving automobile products for mankind to stride towards a future of prosperity and sustainability. In this small-scaled project, we were granted the opportunity to reconstitute a simulation of an autonomous car and through that, gained numerous insights that we have not yet been exposed to. This experience is a hoard of novel knowledge, scientific comprehension, and advanced extracurricular competencies.

**Teamwork:**

Since the first few days informed of the project, we were guided to set up working principles and strengthen our bonds through several in-class activities. On the basis of effective communication, we have worked together to produce a team contract that all team members agreed upon. We were also efficient with task distribution, making use of the beneficial tools introduced. One of these was the RACI Matrix, a responsibility assignment chart that mapped out every task, milestone or key decision involved in completing a project. It helped us appoint which roles are Responsible for each action item, which personnel are Accountable, and, where appropriate, who needs to be Consulted or Informed. We tried to discover the strong suits of each individual and distribute tasks accordingly. So although our members' background and skill sets vary greatly, we managed to find an equilibrium point and complement each other. On top of that, we were also able to divide big missions into smaller steps and work more effectively thanks to a vigorous Task Table. These two instruments assisted in ensuring the punctuality and quality of each small work by enforcing a high sense of responsibility within each member.

**Time-management:**

Coming along with a detailed and high quality project is an extensive amount of tasks and deadlines. That is why time-management skills are extremely vital. Key milestones and expected outcomes were set very clearly since the beginning to give us a well-defined vision of what is ahead. We prioritized tasks based on their importance and urgency and allocated resources accordingly. We tried to allow sufficient resting breaks between each big stage while making sure that the gap is not too large that our flow is disconnected. Accountability did play a part in encouraging us to have our contribution completed on time. Nevertheless, preplanning is definitely the biggest dimension making up this successful picture.

**Hardware-related skills:**

The first half of this project relies greatly on the understanding of mechanical and electrical engineering. Aside from assembling the components to make up a functionable vehicle, hardware skills were also needed in several other activities including 3D designing and using measurement tools. An adequate understanding of the camera, the motors, the batteries, and others are needed to not only put together the modules, but also to fix them during unexpected situations. Despite the fact that the majority of the participants had never used these components previously, this project actually provided us a good chance to conduct research and continually discover relevant uses for them.

**Software-related skills:**

The project would have not been completed without software contribution. Optimizing the given source code was the first challenge we had to face. Lots of efforts were required to understand the algorithms and functions used, as only deep understanding could lead us to efficiently utilize them. There is also a demand for knowledge regarding data structures and design principles to develop software that is reliable, scalable, and maintainable. Besides, autonomous vehicles rely heavily on sensors to collect data about their surroundings, and we would need to have the skills to process this data to make informed decisions. Constructing software for a self-driving model requires a high degree of precision and attention to detail. Numerous testings and debugging were carried out to ensure that the model works correctly and is free from errors.

**Decision-making:**

In every scenario, we would need to be able to identify problems and propose solutions to address them. This problem-solving process involved conducting research, analyzing data, and collaborating with other team members. Any autonomous vehicle is a complex system that concerns a range of risks, including technical failures and safety concerns. We have realized the need to assess these risks and develop strategies to mitigate them.

To sum up, our team has had a puzzling yet enjoyable hands-on learning experience. This project took us through various phases of observing, questioning, experimenting, and much more. The joy of working and the thrill involved while tackling the various problems and challenges gave us a feel of the industry. We believe that by the end of this project, all participants will acquire a basic grasp of engineering and computer science, providing us with the fundamentals for future triumph in studying and on our diverse career paths.

## 5.2 Future recommendation

In this project, our main problems are related to the connection between the ESP32-CAM with the power module, the quality of the camera image, and the power provider. Therefore, to improve these issues in the future, these are our recommendations:

- Using an antenna: Utilizing a camera in our design can better enhance the speed of information transmitting between our model and the computer, which will then increase the speed and performance of running.
- Change the camera module: Since we discover that the ESP32-CAM is not the ideal choice for this project, we can then find another alternative to better improve the line detection. Another option that I find pretty potential is the use of an Ultrasonic distance sensor, for example, the "PING))) Ultrasonic Distance Sensor" as below.


- Choose another power provider: Because one battery is not qualified for running the car, but two ones overcome the requirement, therefore a voltage reducer is needed.

However, in the end, this approach is not optimal. Hence, we suggest finding another power supplier to increase the operation in general.

## References


[1] 18650 3000mah li ion battery cell 3.7v 3000mah li ion battery cell. (n.d.). himax electronics professional manufacturer of lifepo4 battery. retrieved february, 2023, from https://himaxelectronics.com/product-item/18650-3000mah-li-ion-battery-cell/ automated vehicle safety. (n.d.). national highway traffic safety administration. retrieved february, 2023, from https://www.nhtsa.gov/technology-innovation/automated-vehicles-safety.

[2] Autopilot and full self-driving capability. (n.d.). tesla. retrieved february, 2023, from https://www.tesla.com/en$_eu/support/autopilotdebord, m.(2018, october 10). waymo, cruise, mobileye, andteslaapproachestoself - drivingcars.businessinsiderfromhttps : //www.businessinsider.com/waymo - cruise - mobileye - and - tesla - approaches - to - self - driving - cars - 2018 - 10cruise - has - the - major - advantage - of - being - part - of - one - of - the - worlds - biggest - automakers - 2.$

[3] Esp32-cam module overview features. (n.d.). lobo. retrieved february, 2023, from https://loboris.eu/esp32/esp32-cam%20product%20specification.pdf.

[4] The 'full-stack' behind autonomous driving. (n.d.). zoox. retrieved february, 2023, from https://zoox.com/autonomy/.

[5] https://www.parallax.com/product/ping-ultrasonic-distance-sensor/.

[6] Lambert, f., lewis, m., johnson, p., weintraub, s., dow, j. (n.d.). tesla: Current and upcoming models, prices, specs, and more. electrek. retrieved february, 2023, from https://electrek.co/guides/tesla/.

[7] Lm2596 lm2596s dc-dc step-down power supply module. (n.d.). kuongshun electronic shop. retrieved february, 2023, from https://kuongshun.com/products/lm2596-lm2596s-dc-dc-step-down-power-supply-module.

[8] Meyer, s. (2022, july 7). how do self-driving cars work? the zebra. retrieved february, 2023, from https://www.thezebra.com/resources/driving/how-do-self-driving-cars-work/.

[9] Most valuable car brands worldwide by brand value 2022. (2022, june 16). statista. retrieved february, 2023, from https://www.statista.com/statistics/267830/brand-values-of-the-top-10-most-valuable-car-brands/.

[10] The next step on our journey to public roads. (n.d.). zoox. retrieved february, 2023, from https://zoox.com/journal/self-certification/.

[11] Paris, f. (2018, december 23). zoox inc. snags first california permit to transport passengers in self-driving cars. npr. https://www.npr.org/2018/12/23/679705342/zoox-inc-snags-first-california-permit-to-transport-passengers-in-self-driving-c.

[12] Rz7889 circuit datasheet pdf - driver circuit. equivalent, catalog. (n.d.). retrieved february, 2023, from https://datasheetspdf.com/pdf/1053775/wuximicroelectronics/rz7889/1.

[13] Waymo driver – waymo. (n.d.). waymo. retrieved february, 2023, from https://waymo.com/waymo-driver/.

[14] Waymo stock ipo. (2021, june 17). economagic. retrieved february, 2023, from https://www.economagic.com/waymo-stock-ipo/.

[15] What is an autonomous car? – how self-driving cars work. (n.d.). synopsys. retrieved february, 2023, from https://www.synopsys.com/automotive/what-is-autonomous-car.html.

[16] Yellow dc motor 3-12vdc 2 flats shaft. (n.d.). wiltronics. retrieved february, 2023, from https://www.wiltronics.com.au/product/10137/yellow-motor-3-12vdc-2-flats-shaft/.

[17] Zoox autonomous robotaxi first look: Gad-zoox, amazon goes mobile! (2020, december 21). motortrend. retrieved february, 2023, from https://www.motortrend.com/news/zoox-autonomous-robotaxi-amazon-first-look-review/.