ECE 445

SENIOR DESIGN LABORATORY

FINAL REPORT

# Autonomous Vehicle with Sign Recognition and Obstacle Clearing through Wi-Fi

**Team #19**

PAI ZHANG (paiz4)
RUI ZHANG (rui14)
WENDI WANG (wendiw2)
PENGYU ZHU (pengyuz4)

TA: Ruihao Du

May 18nk,b ; 2025

## Abstract

Autonomous vehicles have become popular in industrial research and daily life for a long time, with advanced functions like reading signal lights and giving ways to pedestrians. In recent years, many efforts have been devoted to introducing autonomous vehicles to urban traffic and domestic life. However, among those universalized autonomous vehicles like Tesla and Cruise, they are seldom brought to campus as shuttle buses or trash carts due to high costs. It would bring much convenience to students' campus life if some low-cost and efficient autonomous vehicles help students commute and provide some safety guarantees in obstacle detection and self-speed control by reading speed limit signs. We aim to develop the core of such low-cost autonomous vehicles to potentially take the responsibilities of commuting and obstacle-clearing, enabling wireless transmission to provide efficiency via campus Wi-Fi.

# Contents

**1 Introduction** **1**
1.1 Purpose . . . . . 1
1.2 Functionality . . . . . 1
1.3 Subsystem Overview . . . . . 2

**2 Design** **5**
2.1 Equations & Simulations . . . . . 5
2.2 Design Alternatives . . . . . 5
2.3 Design Description & Justifications . . . . . 5
2.3.1 Image Capture and Simple Analysis . . . . . 5
2.3.2 Microcontroller . . . . . 8
2.3.3 Thorough Image Analysis . . . . . 9
2.3.4 Design Output . . . . . 9

**3 Costs and Schedules** **10**
3.1 Costs . . . . . 10
3.2 Schedules . . . . . 10

**4 Requirements & Verification** **11**
4.1 Requirements . . . . . 11
4.2 Verification Procedures . . . . . 11
4.3 Quantitative Results . . . . . 12

**5 Conclusion** **13**
5.1 Accomplishment . . . . . 13
5.2 Uncertainties . . . . . 13
5.3 Future Work . . . . . 13
5.4 Ethical Considerations . . . . . 13

**References** **15**

# 1 Introduction

In the following sections, we will elaborate on the problem and offer our solutions, including visual aids and high-level requirements.

## 1.1 Purpose

Based on our personal life experience and class schedules on campus, we observe that most undergraduate students have a tight class schedule, resulting in insufficient time to transfer from one classroom to another between consecutive classes. Some choose to ride a bicycle for commuting. However, the number of bicycles around each teaching building is often limited and unevenly distributed. Classrooms with a large student current might be matched to an area with only a few available bicycles. Consequently, manually moving and re-distributing those bicycles to support student commuting needs places an additional burden on campus staff. In the meanwhile, we also notice that the ratio of cleaning staff and the campus area is quite low. The campus is too vast compared to the number of personnel responsible for cleaning, laying down a heavy workload to clean trash and obstacles on the campus, especially in the leaf-falling season or rainy and snowy weather.

According to two observations mentioned above, we came up with the idea of incorporating autonomous driving systems into our campus life and study as shuttle buses or trash carts. However, the most popular autonomous driving vehicles in the market are expensive and suitable for more complex situations other than campus life. Therefore, considering the special properties of the campus (everywhere Wi-Fi coverage and relatively simple road conditions), we aim for a low-cost autonomous vehicle designed and constructed using basic mechanical and electronic components like Arduino board, Wi-Fi modules, robot arms, etc.

## 1.2 Functionality

Specifically, our solution of an autonomous vehicle supporting speed limit sign recognition and obstacle-clearing elaborates as follows with a brief overview in Figure 1. Considering that the speed limit on campus is usually 30 km/h and a safe distance of 2 meters, the camera contained in the Raspberry Pi captures an image every other 0.24 seconds (i.e. at a frequency of 4.167 Hz). Another specific factor on the campus is Wi-Fi signals and availability. Each time the camera snaps a shot, we consider two following cases:

- When Wi-Fi is available, the captured image is sent to the remote server for thorough powerful image analysis and obstacle/speed limit sign recognition. After the remote server returns the corresponding obstacle type or the speed in km/h, such information is sent to the microcontroller (Arduino Uno R3) via Wi-Fi again. On receiving obstacle or speed information, Arduino Uno R3 transmits it to the vehicle motors or the robot arm for self-speed adjustments or obstacle clearing via Universal Asynchronous Receiver/Transmitter (UART) protocols and Wi-Fi modules respectively. For visual aid of this situation, please refer to Figure 3 and Figure 2a.

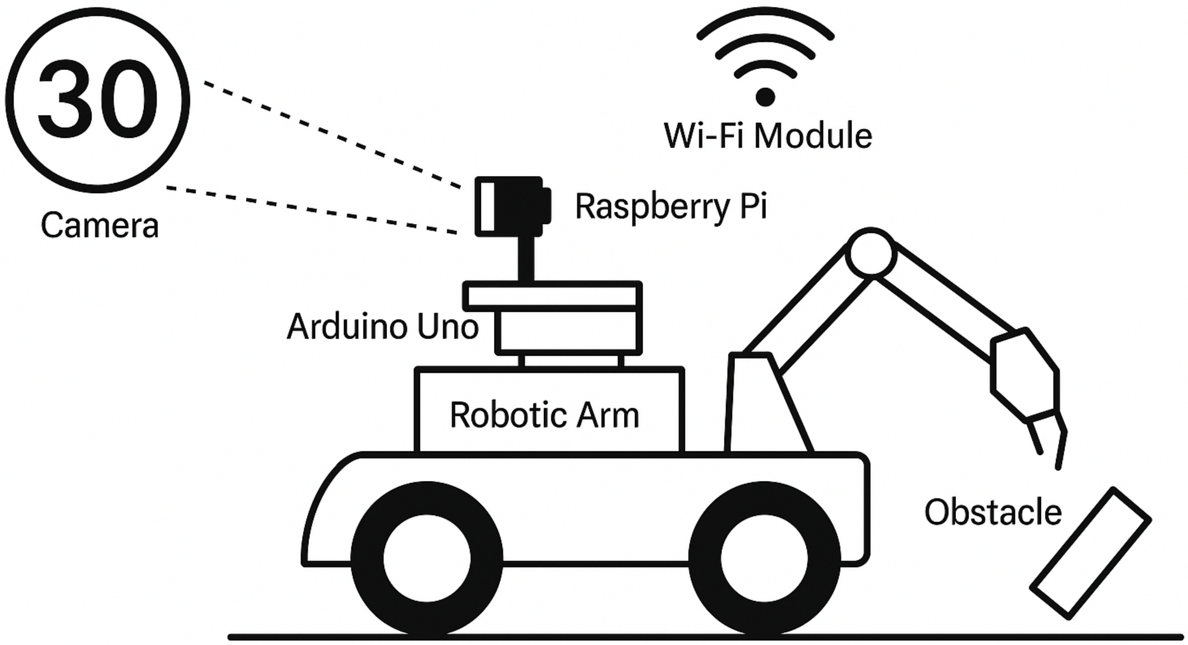


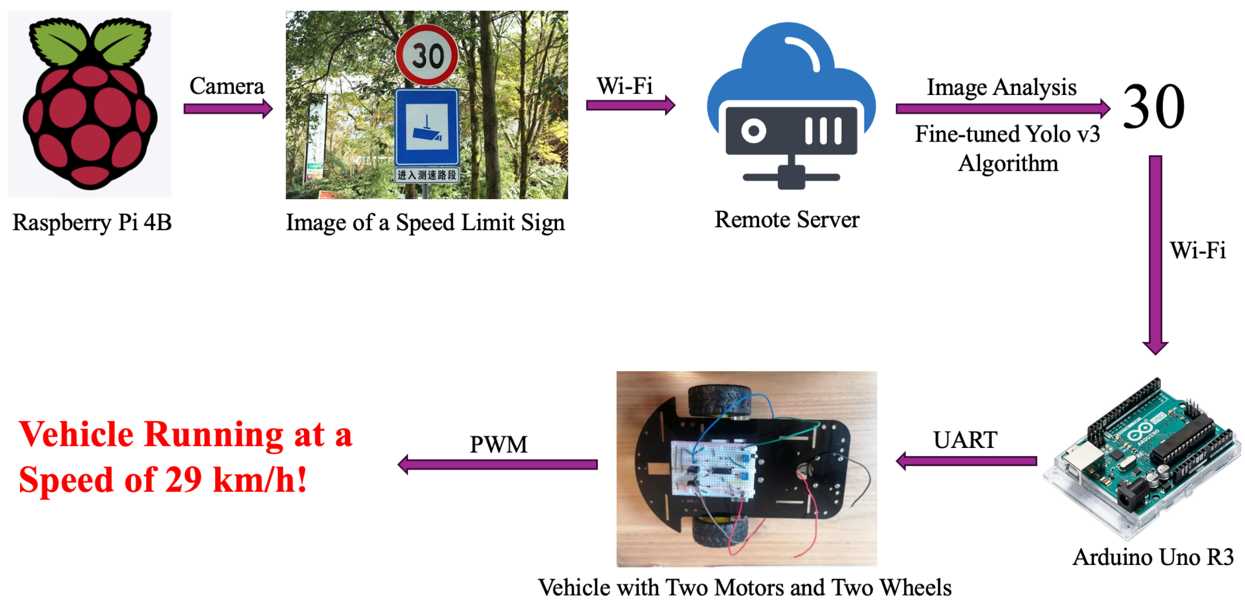

Figure 1: A visual aid of our general solution.

- When Wi-Fi is unavailable in Figure 2b, the Raspberry Pi takes the responsibility to analyze the image using simpler computer vision algorithms like CNN [1] compared to the remote server due to limited GPU capability. After the Raspberry Pi returns the corresponding speed in km/h, such information is sent to the microcontroller (Arduino Uno R3) via UART protocols. On receiving speed information, Arduino Uno R3 transmits it to the vehicle motors for self-speed adjustments via UART protocols again.

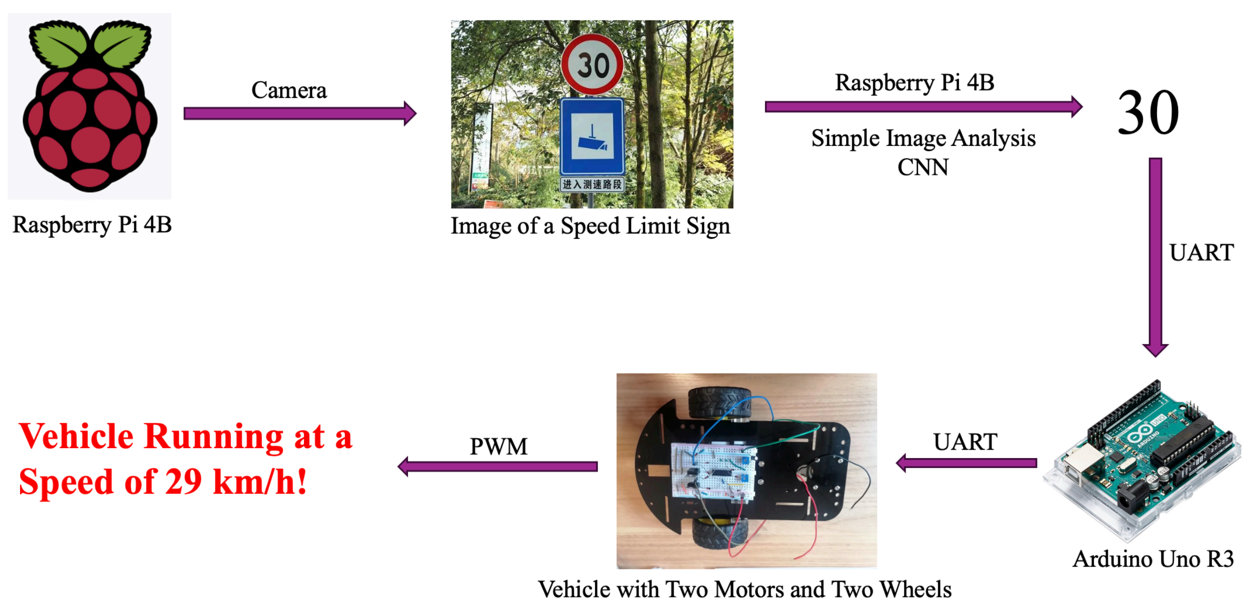
## 1.3 Subsystem Overview

The main component in the *Image Capture and Simple Analysis Subsystem* is the Raspberry Pi 4B. With a camera of high resolutions and wide angles, it captures images of the speed limit signs and obstacles every 0.24 seconds to ensure safety and promptness. When Wi-Fi signals are unavailable, it can also take the responsibility of simple image analysis to ensure the self-speed adjustment of the autonomous vehicle. This subsystem is closely related to the *Microcontroller Subsystem* as the Raspberry Pi 4B communicates with the microcontroller Arduino Uno R3 via UART for image and speed information transmissions.

The main component in the *Microcontroller Subsystem* is the Arduino UNO R3 with a schematic in Figure 7. With a power supply of 5 volts direct current, it connects the components of the Raspberry Pi 4B, the remote cloud server, the robot arm, and the vehicle across all other subsystems. It enables simultaneous reading and writing via UART to receive information from the Raspberry Pi 4B in the *Image Capture and Simple Analysis*




(a) Speed limit sign recognition when Wi-Fi is available.




(b) Speed limit sign recognition when Wi-Fi is not available.

Figure 2: A visual aid of speed limit sign recognition.

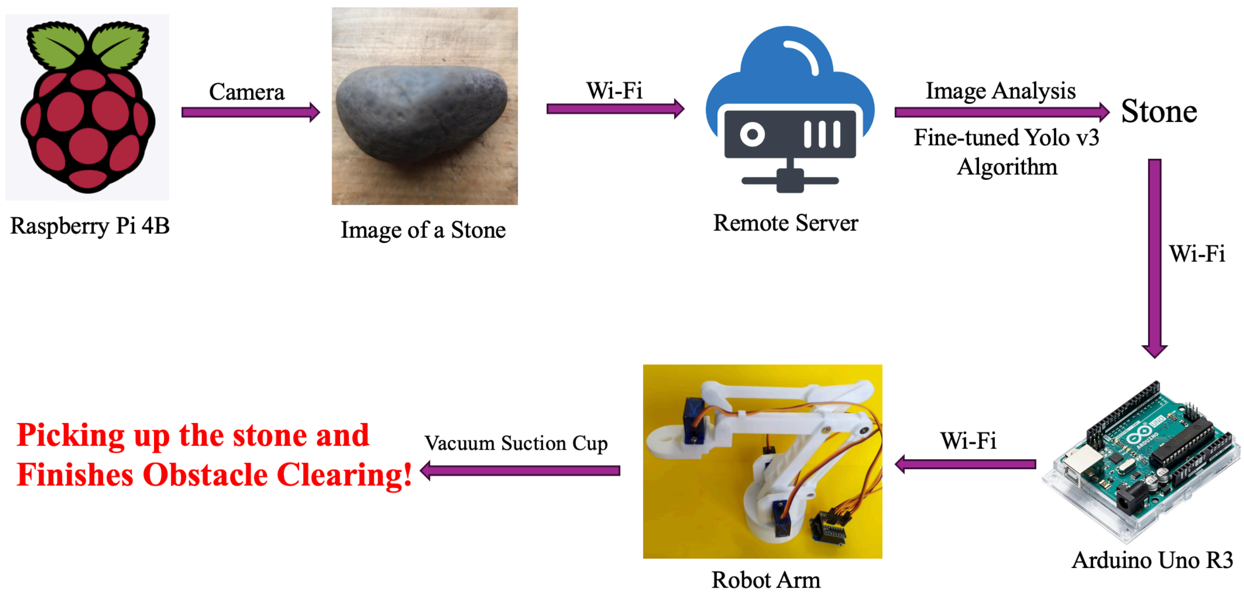



Figure 3: A visual aid of obstacle clearing.

*Subsystem* and delivers it to the vehicle to adjust motor speed via PWM. It also enables Wi-Fi transmission with the remote cloud server in the *Thorough Image Analysis Subsystem* by controlling the Wi-Fi module.

The main component in the *Thorough Image Analysis Subsystem* is the remote cloud server containing a powerful GPU to perform more complex and more accurate image analysis and object detection. When Wi-Fi signals are available, it can also take the responsibility of thorough image analysis to ensure the self-speed adjustment and obstacle detection of the autonomous vehicle. This subsystem is closely related to the *Microcontroller Subsystem* as the remote cloud server communicates with the microcontroller Arduino Uno R3 via Wi-Fi modules for image and speed/obstacle information transmissions. The software algorithm used by both the remote cloud server can be generalized and summarized in Figure 8.

The main component in the *Output Subsystem* is a 4-axis 4-degree robot arm and a vehicle with two DC motors. With a power supply of 12 volts direct current, both are controlled by the Arduino Uno R3 in the *Microcontroller Subsystem*. On receiving the clearing command from the Arduino Uno R3 via the Wi-Fi module, the robot arm picks up the obstacle via a vacuum suction cup and completes the obstacle-clearing task. If the command of speed adjustment is received via UART protocols, the vehicle adjusts the speed of its DC motors via PWM.

# 2 Design

Our design is based on Arduino Uno R3 (as the microcontroller), with the aid of the Raspberry Pi 4B as the camera and simple image analyzer, the Wi-Fi module ESP8266 as the information transmission media, a 4-axis 4-degree robot arm as the obstacle-clearing tool, and a vehicle with two DC motors carrying all components as an autonomous driving system. The detailed workflow, block diagram, and the physical design of our design can be found in Figure 4.

## 2.1 Equations & Simulations

Considering that the speed limit on campus is usually 30 km/h and a safe distance of 2 meters, the camera contained in the Raspberry Pi captures an image every other 0.24 seconds (i.e. at a frequency of 4.167 Hz). This frequency is calculated based on a simple formula:

$$v = \frac{s}{t}. \tag{1}$$

We use accuracy and time to evaluate the performance of this subsystem as the simulation before we load the model to the autonomous vehicle. If the time of speed limit sign recognition is less than $0.08 \pm 0.001$ seconds and the accuracy of speed limit sign recognition is larger than $80\% \pm 0.1\%$, we consider this subsystem to be successfully constructed and functioning. The tolerance scale is derived from the following experimental conclusions by averaging the classification results of 100 test samples on the remote cloud server with a RTX 3090 GPU, as shown in Figure 6.

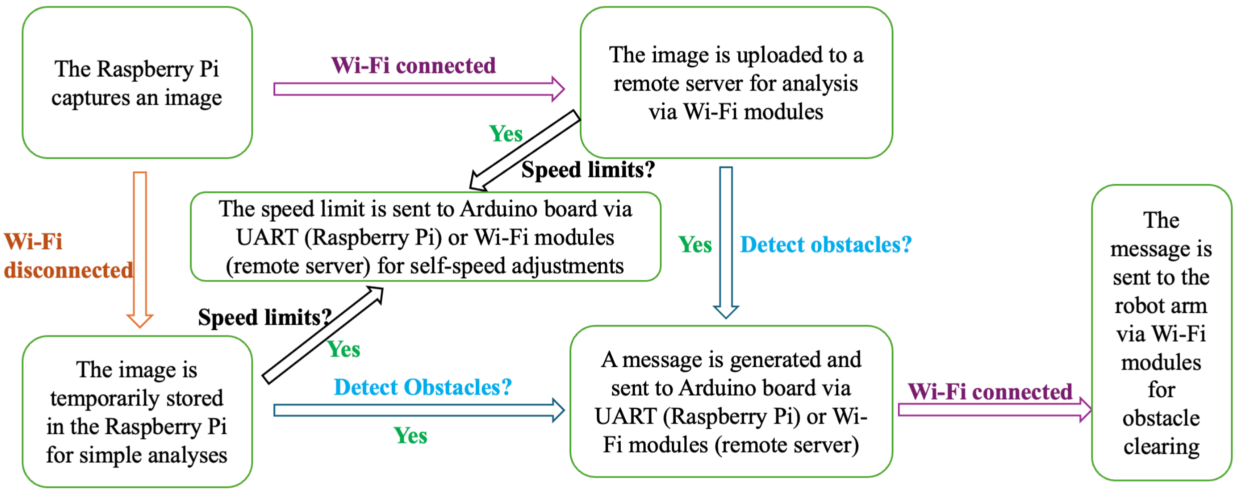
## 2.2 Design Alternatives

We initially planned to use hardware ports for data transmission between the Arduino Uno R3 board and the ESP8266 Wi-Fi module. However, we encountered difficulties in simultaneously controlling motor movement and data transmission. As a result, we switched to software ports as an alternative solution. Ultimately, this allowed us to successfully handle data transmission while the vehicle was in motion.

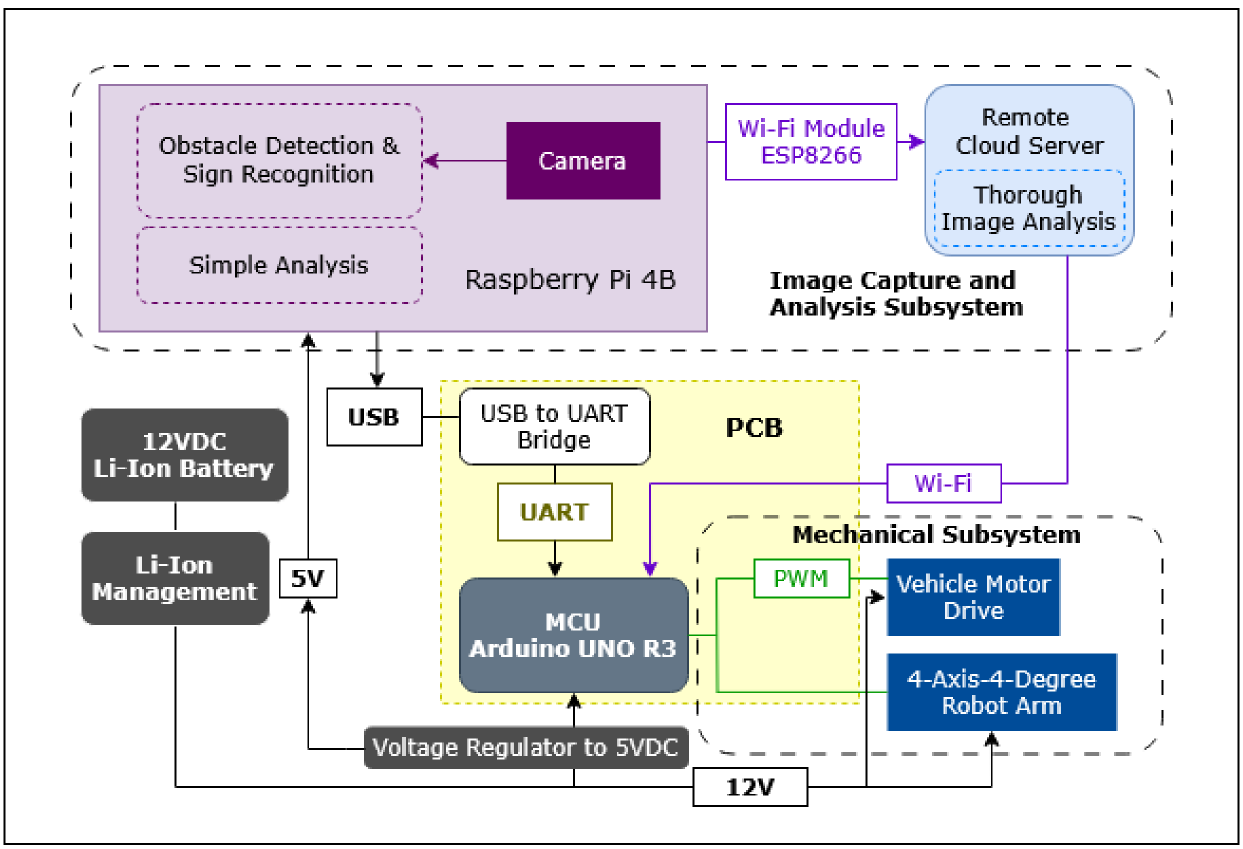
## 2.3 Design Description & Justifications

### 2.3.1 Image Capture and Simple Analysis

The main component serving for this functionality is the Raspberry Pi 4B. With a camera of high resolutions and wide angles, it captures images of the speed limit signs and obstacles every 0.24 seconds to ensure safety and promptness. When Wi-Fi signals are unavailable, it can also take the responsibility of simple image analysis to ensure the self-speed adjustment of the autonomous vehicle. This subsystem is closely related to the *Microcontroller Subsystem* as the Raspberry Pi 4B communicates with the microcontroller Arduino Uno R3 via UART for image and speed information transmissions.




(a) Workflow of our proposed solution.




(b) Block diagram of our proposed solution consisting of four subsystems and a power supply module.

Figure 4: Block Diagrams in Hardware and Software Design.

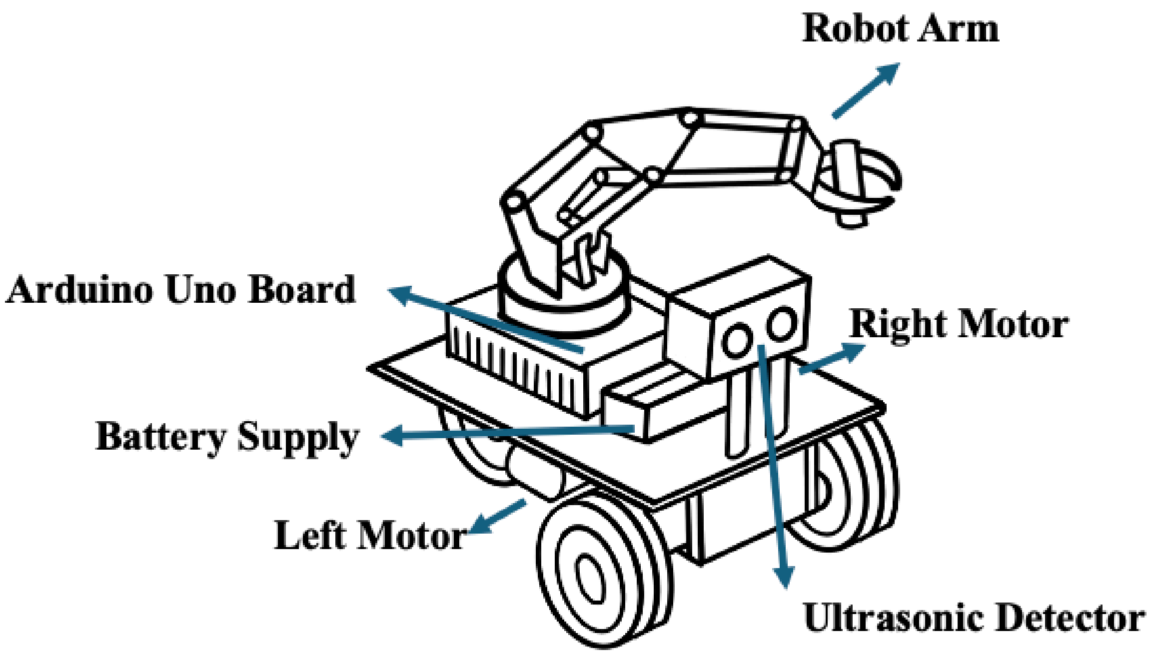



Figure 5: Physical Diagram in Hardware and Software Design.

```python
times = []
for _ in range(100):
    start_time = time.perf_counter()
    model.predict(test_sample)
    times.append(time.perf_counter() - start_time)
avg_time = np.array(times).mean()
std_time = np.sqrt(np.array(times).var())

print(f"\nAverage inference time: {avg_time*1000:.2f}+-{std_time*1000:.2f} ms")
```

```
Average inference time: 75.08+-4.98 ms
```

Figure 6: Average recognition time is $0.075 \pm 0.005$ seconds.

Since the GPU capability and storage space of the Raspberry Pi 4B is limited (4 GB), finding and finetuning a CNN-based algorithm to detect obstacles and speed limit signs while satisfying the accuracy requirement of 80% might take some time of careful consideration.

### 2.3.2 Microcontroller

The main component serving for this functionality is the Arduino UNO R3 with a schematic in Figure 7. With a power supply of 5 volts direct current, it connects the components of the Raspberry Pi 4B, the remote cloud server, the robot arm, and the vehicle across all other subsystems. It enables simultaneous reading and writing via UART to receive information from the Raspberry Pi 4B in the *Image Capture and Simple Analysis Subsystem* and delivers it to the vehicle to adjust motor speed via PWM. It also enables Wi-Fi transmission with the remote cloud server in the *Thorough Image Analysis Subsystem* by controlling the Wi-Fi module.

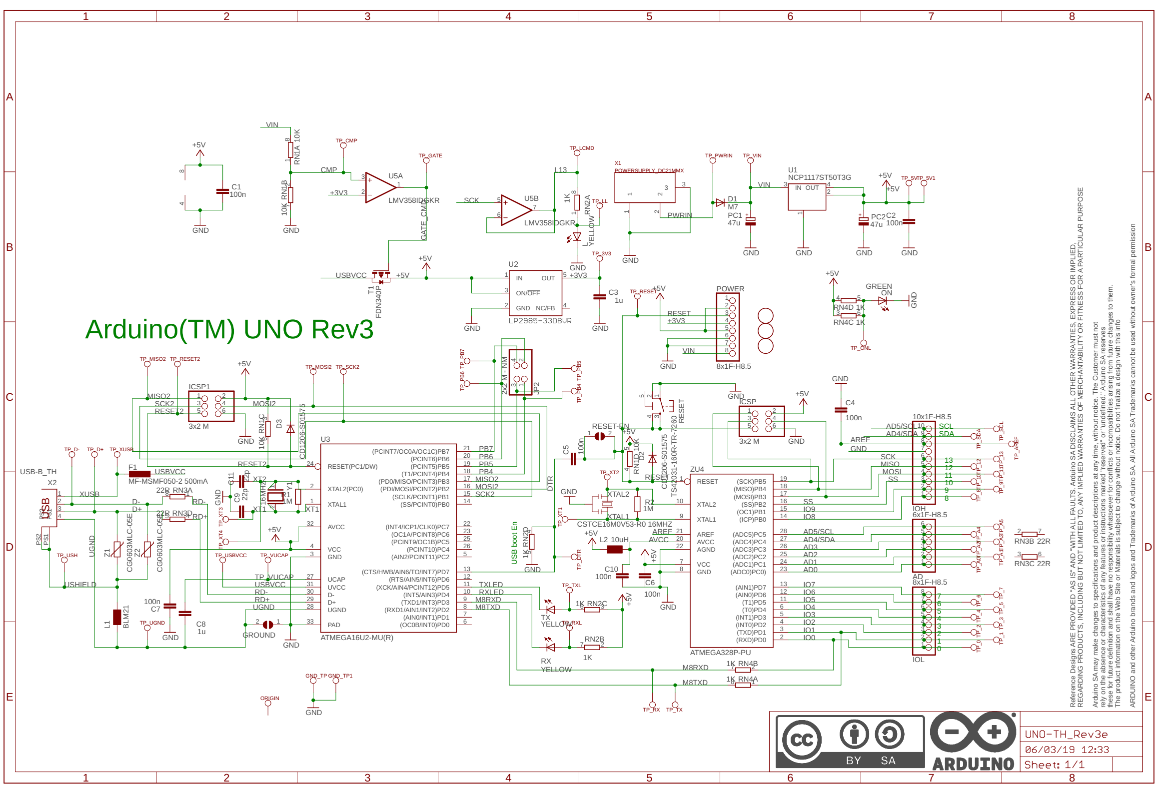



Figure 7: Arduino Uno Schematic Diagram referring to official Arduino Board information website [2].

Since the microcontroller links all other subsystems and components in this design, it is essential to avoid any potential misusing or misleading practices on the Arduino Uno R3.

#### 2.3.3 Thorough Image Analysis

The main component serving for this functionality is the remote cloud server containing a powerful GPU to perform more complex and more accurate image analysis and object detection. When Wi-Fi signals are available, it can also take the responsibility of thorough image analysis to ensure the self-speed adjustment and obstacle detection of the autonomous vehicle. This subsystem is closely related to the *Microcontroller Subsystem* as the remote cloud server communicates with the microcontroller Arduino Uno R3 via Wi-Fi modules for image and speed/obstacle information transmissions. The software algorithm used by both the remote cloud server can be generalized and summarized in Figure 8.

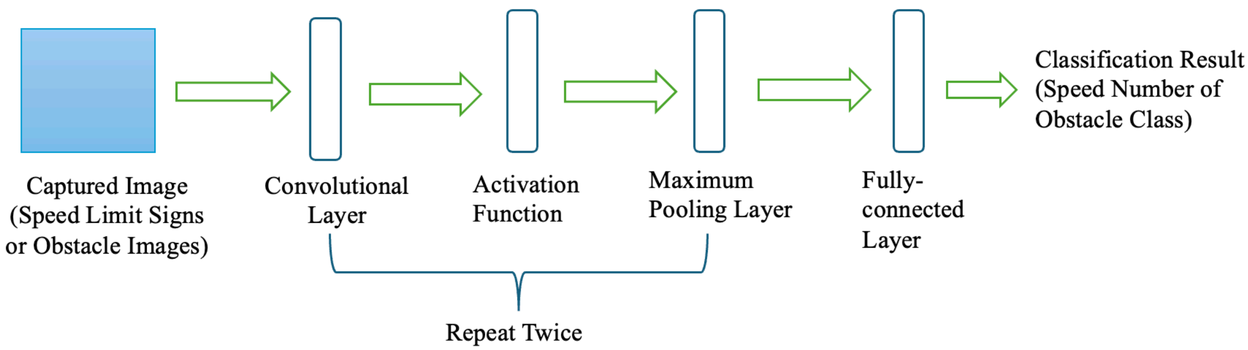



Figure 8: The training flow chart of speed limit sign and obstacle recognition.

Since the computer vision algorithms implemented on the remote cloud server might be complex to enhance the accuracy of object and speed sign detection, finding and finetuning a powerful algorithm like Yolo v3 [3] to detect obstacles and speed limit signs while satisfying the time constraint of 0.08 seconds might take some time of careful consideration.

#### 2.3.4 Design Output

The main component serving for this functionality is a 4-axis 4-degree robot arm and a vehicle with two DC motors. With a power supply of 12 volts direct current, both are controlled by the Arduino Uno R3 in the *Microcontroller Subsystem*. On receiving the clearing command from the Arduino Uno R3 via the Wi-Fi module, the robot arm picks up the obstacle via a vacuum suction cup and completes the obstacle-clearing task. If the command of speed adjustment is received via UART protocols, the vehicle adjusts the speed of its DC motors via PWM.

Since the vehicle carries all other subsystems and components in this design, it is essential to avoid any potential misusing or misleading practices on the DC motors.

# 3 Costs and Schedules

## 3.1 Costs

Please refer to Table 1 for the cost of our project.

| **Component** | **Cost** | **Cost Type** |
|---|---|---|
| Raspberry Pi 4B | $54.6224 | Electronic Consumables |
| ESP8266 Wi-Fi Module | $4.5165 | Electronic Consumables |
| Arduino Uno Board with Vehicle Components | $85.9914 | Electronic Consumables |
| Robot Arm Components | $28.8828 | Mechanical Consumables |
| 4 Persons with Equal Contributions | $30 per hr * 30 hr/week * 20 week/person * 4 persons = $72000 | Labor |
| Summary | $72174.0131 | Total |

Table 1: Detailed cost analaysis including Consumables and Labor Costs

## 3.2 Schedules

Please refer to Table 2 for detailed schedules of our project.

# 4 Requirements & Verification

## 4.1 Requirements

Here are four quantitative high-level requirements for our proposed solution, aiming to solve students' commuting issues and the heavy workload of trash clearing on the campus:

- The vehicle must navigate autonomously with the aid of speed limit signs only on the campus, for at least 30 minutes.
- Total time of each image capturing, image analysis, and information transmission must be less than 0.24 seconds, the time interval between two shots of the Raspberry Pi camera.
- The robot arm must clear an obstacle within 0.24 seconds, which is the time interval between two shots of the Raspberry Pi camera.
- The accuracy of recognizing speed limit signs and obstacles must be above 80%.

## 4.2 Verification Procedures

- *Requirement 1*: The accuracy of speed limit sign recognition must be larger than 80%.
- *Verification 1*: Generate images of different speed limits ranging from 0 to 30 km/h (according to the rules on the campus) to test the recognition algorithm on the Raspberry Pi 4B and record the accuracy of recognition.
- *Requirement 2*: The time of speed limit sign recognition must be less than 0.08 seconds, one-third of the time interval between two shots of the Raspberry Pi camera.
- *Verification 2*: Generate images of different speed limits ranging from 0 to 30 km/h (according to the rules on the campus) to test the recognition algorithm on the Raspberry Pi 4B and record the average recognition time.
- *Requirement 3*: The time of UART transmission must be less than 0.08 seconds, one-third of the time interval between two shots of the Raspberry Pi camera.
- *Verification 3*: Record the average UART transmission time by recording the time when the Raspberry Pi 4B generates the result and the time Arduino receives.
- *Requirement 4*: The time of Wi-Fi transmission must be less than 0.08 seconds, one-third of the time interval between two shots of the Raspberry Pi camera.
- *Verification 4*: Record the average Wi-Fi transmission time by recording the time when the remote cloud server returns the result and the time Arduino receives.
- *Requirement 5*: The accuracy of speed limit sign recognition must be larger than 80%.
- *Verification 5*: Generate images of different speed limits ranging from 0 to 30 km/h (according to the rules on the campus) to test the recognition algorithm on the remote cloud server and record the accuracy of recognition.

- *Requirement 6*: The time of speed limit sign recognition must be less than 0.08 seconds, one-third of the time interval between two shots of the Raspberry Pi camera.
- *Verification 6*: Generate images of different speed limits ranging from 0 to 30 km/h (according to the rules on the campus) to test the recognition algorithm on the remote cloud server and record the average recognition time.

## 4.3 Quantitative Results

- *Requirement 1*: The accuracy of speed limit sign recognition must be larger than 80%.
- *Result 1*: Accuracy on 10 images: 90.70%.
- *Requirement 2*: The time of speed limit sign recognition must be less than 0.08 seconds, one-third of the time interval between two shots of the Raspberry Pi camera.
- *Result 2*: Average recognition time is 0.075 ± 0.005 seconds.
- *Requirement 3*: The time of UART transmission must be less than 0.08 seconds, one-third of the time interval between two shots of the Raspberry Pi camera.
- *Result 3*: Average UART transmission time is 0.02 seconds.
- *Requirement 4*: The time of Wi-Fi transmission must be less than 0.08 seconds, one-third of the time interval between two shots of the Raspberry Pi camera.
- *Result 4*: Average Wi-Fi transmission time is 0.02 seconds.
- *Requirement 5*: The accuracy of speed limit sign recognition must be larger than 80%.
- *Result 5*: Accuracy on 10 images: 90.70%.
- *Requirement 6*: The time of speed limit sign recognition must be less than 0.08 seconds, one-third of the time interval between two shots of the Raspberry Pi camera.
- *Result 6*: Average recognition time is 0.075 ± 0.005 seconds.

# 5 Conclusion

## 5.1 Accomplishment

We have successfully met all high-level requirements specified in our design document. Our autonomous vehicle can accurately detect speed limit signs and determine the maximum permitted speed. Upon identifying the speed limit, the microcontroller promptly adjusts the motor speed. When the vehicle detects obstacles near the road, the microcontroller directs the robotic arm to remove them via Wi-Fi communication. This low-cost autonomous vehicle represents a successful implementation of our design objectives.

## 5.2 Uncertainties

We cannot use wireless Wi-Fi signals (without connecting the Wi-Fi modules to power) when transmitting data.

## 5.3 Future Work

We can enable more flexible autonomous vehicle with less USB connections and wires in the circuits.

## 5.4 Ethical Considerations

For the privacy part, image capturing might inevitably incorporate students' faces, which requires us to manually blur or remove such private information to protect students (Term I. 4 of IEEE Code of Ethics [4]). For the social effects, the introduction of autonomous cleaning vehicles might reduce the needs of cleaning staff on the campus, which requires us to collaborate with campus administration to reassign affected staff to other roles or help them become the technical controller of such autonomous vehicles. In our solution, we will not leak any images with students' faces, even the blurred one. And our algorithm will not misuse the images except from speed limit sign and obstacle recognition. We will immediately delete the images once the analysis is done and the action is delivered to either the vehicle or the robot arm. This avoids information misconduct and abusing images from the source. In the future, anti-attack techniques will also be added to the vehicle to prevent any abnormals' attacks.

| Time | Pai Zhang | Rui Zhang | Wendi Wang | Pengyu Zhu |
|---|---|---|---|---|
| Feb 24th-28th, Mar 3rd-7th | Get familiar with the project topic. | Get familiar with the project topic. | Get familiar with the project topic. | Get familiar with the project topic. |
| Mar 10th-14th, Mar 17th-21th | Start searching suitable computer vision algorithms for speed limit sign recognition. | Start searching suitable components for constructing the robot arm. | Start searching suitable components for the autonomous vehicle and the microcontroller. | Start searching suitable computer vision algorithms for obstacle detection. |
| Mar 24th-28th, Mar 31th-Apr 4th | Start testing suitable computer vision algorithms for speed limit sign recognition. | Start constructing the robot arm. | Start constructing the autonomous vehicle. | Start testing suitable computer vision algorithms for obstacle detection. |
| Mar 31th-Apr 4th, Apr 7th-11th | Modifying computer vision algorithms for speed limit sign recognition. | Testing the robot arm to pick up obstacles and controlled by the Arduino Uno Board (the microcontroller). | Testing the autonomous vehicle to change its speed and instruct the robot arm's behavior. | Modifying computer vision algorithms for obstacle detection. |
| Apr 14th-18th, Apr 21th-25th | Collecting real images of speed limit signs to attain 80% recognition accuracy. | Enabling data transmission between the robot arm and the microcontroller. | Enabling data transmission among the robot arm, the vehicle, and the microcontroller. | Collecting real images of diverse obstacles to attain 80% recognition accuracy. |
| Apr 28th and after | Unifying all functions and preparing for the demo. | Unifying all functions and preparing for the demo. | Unifying all functions and preparing for the demo. | Unifying all functions and preparing for the demo. |

Table 2: Detailed Schedule and Task Plan

# References


[1] R. Girshick, "Fast r-cnn," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 1440–1448. [Online]. Available: https://openaccess.thecvf.com/content_iccv_2015/papers/Girshick_Fast_R-CNN_ICCV_2015_paper.pdf (visited on 04/14/2025).

[2] "Arduino uno rev3 documentation," Arduino. (2019), [Online]. Available: https://content.arduino.cc/assets/UNO-TH_Rev3e_sch.pdf (visited on 04/14/2025).

[3] J. Redmon and A. Farhadi, "Yolov3: An incremental improvement," *arXiv preprint arXiv:1804.02767*, 2018. [Online]. Available: https://ask.qcloudimg.com/draft/2661027/i16j8zgndj.pdf (visited on 04/14/2025).

[4] "Ieee code of ethics," IEEE. (2025), [Online]. Available: https://www.ieee.org/about/corporate/governance/p7-8.html (visited on 04/14/2025).