XI'AN JIAOTONG-LIVERPOOL UNIVERSITY
西 交 利 物 浦 大 学
COURSEWORK SUBMISSION
COVER Page

Group Number	18				
Students' ID Number	2251461	2251625	2252779	2251603	
Module Code	MEC302				
Assignment Title	Final report				
Submission Deadline	2025.5.11				

By uploading this coursework submission with this cover page, we certify the following:

- We have read and understood the definitions of collusion, copying, plagiarism, and dishonest use of data as outlined in the Academic Integrity Policy of Xi'an Jiaotong-Liverpool University.
- This work is **entirely** our own, original work produced specifically for this assignment. It does not misrepresent the work of another group or institution as our own.
- This work is not the product of unauthorized collaboration between ourselves and others.
- This work has not been shared wholly or in part outside of the group.
- It is a submission that has not been previously published, or submitted to any modules.
- Students who would like to submit the same or similar work from previous years to the current module or other modules must receive written permission from all instructors involved in advance of the assignment due date.
- **All** group members are **equally** and **collectively** responsible for the **entire** submission. Violations of academic integrity including failure to monitor group member contribution originality constitutes negligence.
- Unreported suspicious academic misconduct by one group member will be attributed to ALL members in terms of penalties. **ALL members share the responsibilities**.
- Use of generative AI is strictly prohibited for all assessments involved in this module.

We understand collusion, plagiarism, dishonest use of data, and submission of procured work are serious academic misconducts. **All** group members are held **jointly accountable** for the integrity of the **entire** submission. By uploading or submitting this cover page, we acknowledge that we are jointly subject to penalties and disciplinary actions if we are found to have committed such acts.

We acknowledge that the university late submission policy will be applied if applicable.

Please list the ID number of any group member NOT contributing to the submission:

Please indicate whether based on your individual contributions you meet the learning outcomes covered by this coursework and confirm your submission meets with all above requirements by signing your name, handwritten in ink, in English (Foreign students) or PINYIN (local students):

Student ID	✓(tick) box to confirm you contributed to all cw tasks	✓(tick) box to confirm you meet all LOs covered by this cw	Signed
2251461	☑	☑	Hengzhi Qiu
2251625	☑	☑	Yukun Zheng
2252779	☑	☑	Suheng Ji
2251603	☑	☑	Yaoran Xu

Date: 2025.05.09

DEPARTMENT OF MECHATRONICS AND ROBOTICS

MEC202 2024-25

Final Group Project Report

A Smart Shopping Cart with Gesture and Vision-Based Tracking System

Project Code: MEC202-GEN18

Name	Student ID	Contribution Description for ALL Group Projects	Percentage (%)
Yukun.Zheng22	2251625	ALL	25%
Suheng.Ji22	2252779	ALL	25%
Hengzhi.Qiu22	2251467	ALL	25%
Yaoran.Xu22	2251603	ALL	25%

Contents

I. Introduction

A. Background and Motivation

In today's society, the retail industry is undergoing profound changes, and technological progress is constantly reshaping consumers' shopping experiences [1]. However, despite the increasing popularity of emerging models such as online shopping and smart payment, traditional brick-and-mortar shopping remains an indispensable choice for many people [2]. In brick-and-mortar stores, shopping carts have a long history as auxiliary tools, but the basic model that relies on manual operation has not undergone fundamental changes. For most people, this might not be a problem, but for specific groups, such as the elderly, those with limited mobility, or customers who need to carry a large number of heavy items, pushing a heavy shopping cart can become a significant physical burden and inconvenience [2][3]. This not only affects their shopping experience, but also constitutes a physical environmental obstacle to a certain extent, which is different from the goal of building a more inclusive and barrier-free social environment.

Solving this problem has significant social and commercial value. From a social perspective, paying attention to and improving the shopping experience of vulnerable groups is an important part of demonstrating humanistic care and promoting social equity [4]. Enabling everyone to participate in daily life more easily and freely, including enjoying the pleasure of shopping, inherently carries profound ethical significance. One of the motivations for this project lies precisely in the concern for the challenges faced by vulnerable groups such as the elderly when shopping, and it is hoped to provide them with practical assistance through technological innovation. From a business perspective, providing more convenient and friendly shopping services can attract a broader customer base, enhance customer satisfaction and loyalty, and help retailers stand out in the fierce market competition.

Currently, the retail technology sector has witnessed a variety of attempts to enhance the shopping experience, such as automated checkout systems, in-store navigation applications, and some prototypes of smart shopping carts with specific functions. These efforts have enhanced efficiency and convenience to varying degrees. However, most of the existing solutions have not yet fully addressed the core need of users to "free their hands" throughout the store and move the shopping carts loaded with items easily. Many smart trolleys focus on product recognition and payment functions, but the physical burden of promoting them still exists. Some automated guidance technologies are mainly applied in warehousing or specific industrial scenarios. However, they still face challenges in directly serving ordinary consumers in complex retail environments. Therefore, a shopping assistance tool that can truly achieve user following and intelligent control and significantly reduce physical exertion still has broad application prospects and innovation space.

Based on the above background, ethical concerns about the needs of vulnerable groups, and the limitations of existing technical solutions, this project aims to design and develop an intelligent shopping cart system. The core objective of this system is to achieve automatic user following and have gesture control functions, thereby providing users with a brand-new, effortless and convenient shopping experience, with a particular focus on enhancing the shopping accessibility for groups in need such as the elderly [5].

B. User Needs and Existing Solutions

The main target users of this project include the elderly, those with limited mobility, and ordinary shoppers who need to carry a large number of heavy objects. These groups of people often encounter difficulties when using traditional shopping carts, such as the need to continuously push the shopping cart, the lack of auxiliary support when they have difficulty moving around, and the lack of intuitive and easy-to-use interaction methods. In recent years, some commercial smart shopping carts have emerged in the market, but their focus is mostly on "settlement automation" rather than "mobility assistance".

Table 1. Comparison of Different Existing Popular Smart Shopping Carts

Comparison / Cart	Amazon Dash Cart	Caper Cart	Walmart Smart Cart
Image Example			
Function Overview	Identifies products via cameras and weight sensors; auto checkout via Amazon account [6].	Barcode scanning and touchscreen checkout; user-initiated process [7].	AI-assisted product display and smart inventory [8].
Pros	Fast checkout; no need to queue.	User-friendly interface; intuitive operation.	Personalized experience; smart inventory tracking.
Cons	No movement assistance; manual pushing required.	No auto-movement; lacks support for the elderly.	High cost; complex system; still requires manual effort.
Estimated Cost	\$5,000–\$10,000 [6]	\$5,000–\$10,000 [7]	Not publicly disclosed [8]

In contrast, our design directly addresses a critical blind spot—**movement assistance**—by implementing gesture-based navigation and automatic following capabilities using IMU sensors and wireless communication. While existing commercial smart carts primarily focus on checkout automation, they do not offer mobility support for elderly users or those with physical challenges.

Additionally, our system emphasizes real-time interaction, simplified circuitry, and low production cost. The prototype is built under **¥1000 (approx. \$140)**, significantly lower than the industry-standard smart carts which range from **\$5,000 to \$10,000** [6][7]. This makes our design a practical and accessible solution for small-to-medium retailers and underserved populations.

C. Design Goals and Requirements

To address the shortcomings of traditional shopping carts in assisting with mobility, especially the burden they impose on the elderly and those with limited mobility, this project aims to develop an intelligent shopping cart system with automatic following function and gesture control function. Its core objective is to reduce users' physical burden through technological means and enhance the convenience, safety and universality of the shopping experience. The following are the main design goals of this project and the corresponding functional and non-functional requirements:

C.1 Design Goals

- Realize the automatic following of the shopping cart to the user in the shopping mall environment to reduce the operational burden on the user.
- Support controlling the movement direction of the shopping cart through gestures to achieve more intuitive human-computer interaction.
- Enhance the security during the shopping process, including functions such as obstacle avoidance and anti-deviation.
- Pay attention to the special needs of vulnerable groups such as the elderly and the weak, and enhance social inclusiveness.

- Control the system cost to ensure its feasibility for practical application in ordinary retail scenarios.
- Equip with certain modular expansion capability, providing space for the future integration of technologies such as SLAM and RFID.

C.2 Functional Requirements

- Automatic following Function: The shopping cart can recognize and follow the user, maintaining a reasonable following distance.
- Gesture Control Function: Users can achieve basic operations such as moving forward, backward, and turning through preset gestures.
- Obstacle Detection and Avoidance Function: Uses ultrasonic sensors to identify surrounding obstacles and actively avoid them.
- Security Alarm Mechanism: If the camera on the Raspberry Pi (computer vision) fails to detect the ArUco tags sticked to the customers, the abnormal LED will flash.
- Motion State Feedback: Provide operational status or expression feedback through OLED screens or sound modules to enhance the sense of interaction.

C.3 Non-Functional Requirements

- Low-cost design: The overall hardware cost is controlled within ¥1000, suitable for large-scale promotion.
- Easy to use: The device is easy to operate and suitable for elderly users to get started quickly.
- Communication stability: Wireless data transmission is stable and has low latency, ensuring real-time gesture response.
- Structural Safety: The hardware layout is stable, adaptable to different loads and terrains, and prevents tilting or overturning.
- Energy Consumption and Battery life: Maintain stable performance in shopping scenarios that operate continuously for more than one hour.
- System Scalability: The module design supports the future integration of advanced functions such as SLAM navigation and RFID identification.

The design goals and requirements of this project were formulated with full consideration of user experience, technical feasibility and commercial practicality, aiming to provide a more intelligent, safe and economical shopping solution for specific groups of people.

D. Project Plannings & Work Load Distributions

To ensure the timely and structured progression of the gesture-controlled car project, a detailed weekly schedule and task allocation plan were formulated. These visual plans clearly define each milestone and the responsibilities of all team members.

Table 2. Project Schedule for Gesture-Controlled Car (Weeks 3-13)

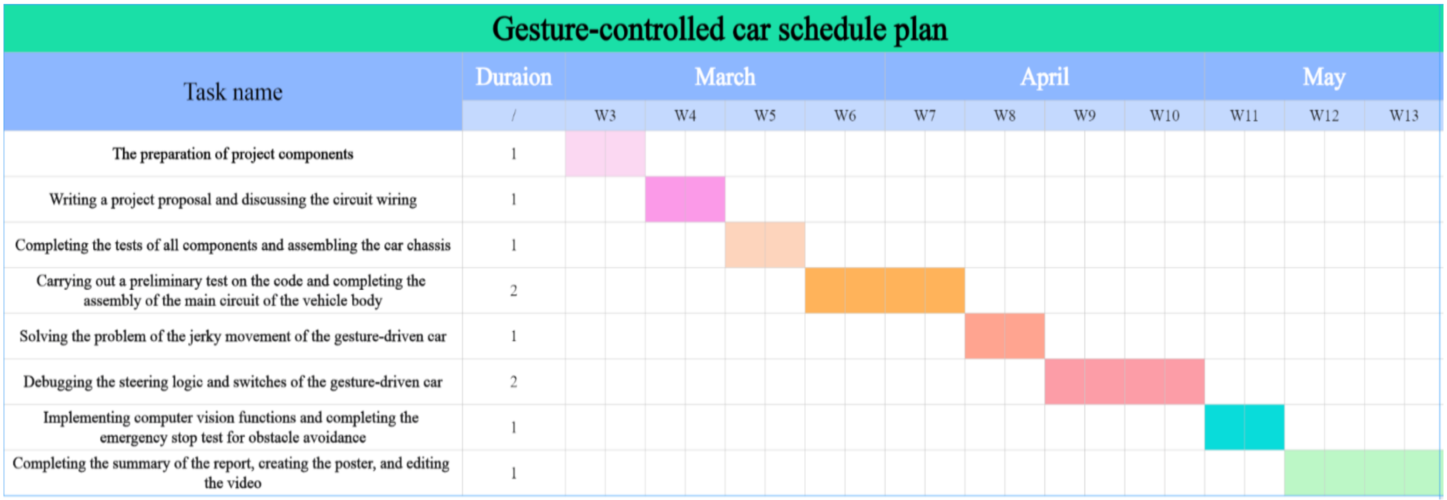


Table 3. Workload Distribution among Team Members

Name	Workload
Hengzhi Qiu (25%)	Assembly of the main circuit and each module of the gesture-controlled car. Help with the construction of the gesture-controlled car code. Write a part of the report.
Suheng Ji (25%)	Build a 3D model of the exterior shell for the gesture-driven car. Assembling OLED for the gesture-controlled car. Shoot a video of the gesture-controlled car. Write a part of the report.
Yaoran Xu (25%)	Editing and production of the video of the gesture-controlled car. Help connect the circuits of some modules of the gesture-controlled car. Write a part of the report.
Yukun Zheng (25%)	Assign tasks to each link in the process. Write the codes for each module of the gesture-controlled car. Assist in the installation of the gesture-controlled car. Write a part of the report.

II. Design Comparison and Selection

A. Initial Design Ideas

Our initial inspiration originated from a project on Arduino Project Hub - Human-Following Bot. This project showed a simple robot following system based on infrared (IR) and ultrasonic sensors, which aroused our interest.

Meanwhile, the four of us participated in the vehicle black-line-tracking experiment based on Arduino programming in the MEC104 course last year. Although the task at that time was mainly to execute preset commands and did not involve material selection or self-determined project design, it laid a foundation for us to develop by ourselves and understand the motion principle of the car.

Since all the members of our team have driving experience and are familiar with the steering logic and perception control of vehicles in daily life, we all agree that developing an intelligent car with autonomous following capability will be an interesting and challenging practical direction.

B. Concept Evaluation against Customer Needs

To achieve user-oriented goals such as "hands-free interaction", "easy to use", "adaptable to a wide range of user groups", and "stable control response", in the early stage of this project, functional evaluations were conducted around three control logic paths. We respectively drew the system control structures of the three schemes through flowcharts (see Figures 1 to 3), and evaluated and compared their respective functional paths, ease of use, maintainability and other aspects.

- Solution One: Bluetooth Cart - Mobile Phone Bluetooth Remote Control (Figure 1)

Fig. 1. Control Flow of Bluetooth Cart Design

Figure 1 shows the control structure of Bluetooth Cart. This solution relies on users holding a smart phone and sending instructions to the Bluetooth module via a Bluetooth App, where Arduino performs motor control.

✧ Pros:

- The system has a simple structure, common modules and a low development threshold.
- The mobile phone serves as the control terminal, facilitating the expansion of interface functions.

✧ Cons:

- Strongly dependent on mobile phones, not suitable for the elderly or users who are not familiar with smart devices.
- Bluetooth signals are prone to interference in crowded environments (such as supermarkets), leading to control delays or failures.
- Users have to hold their mobile phones all the time, which goes against the original intention of "liberating both hands".

- Solution 2: Bionic Glove - Sensor Glove Control (Figure 2)

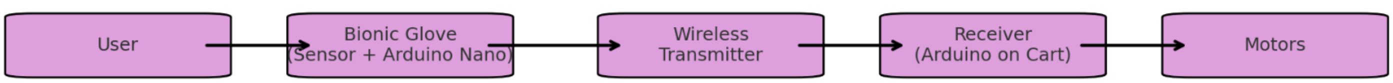


Fig. 2. Control Flow of Bionic Glove Design

As shown in Figure 2, the Bionic Glove solution adopts the "bionic glove" composed of MPU6050 attached to the hand, Arduino Nano and wireless module, which is used to recognize gesture actions and transmit them to the car.

✧ Pros:

- Achieve high-precision gesture recognition with a natural interaction method.
- Smooth control and quick response.

✧ Cons:

- The system is equipped with a large amount of hardware and requires users to wear it continuously, which increases the burden on the hands.
- In the prototype structure, the breadboard and jumper cables are still needed for connection, but the stability in actual operation is insufficient.
- The cost and structural complexity are relatively high, which is not conducive to large-scale deployment.

- Solution 3: Gesture & Target Tracking - Lightweight Sensors + Fusion Perception (Figure 3)

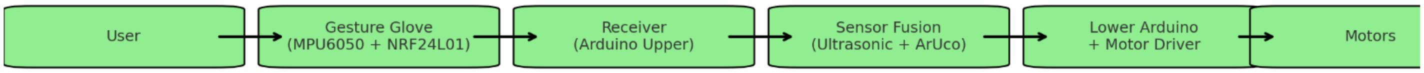


Fig. 3. Control Flow of Gesture & Target Tracking Design

Figure 3 shows the control structure diagram of the final selected scheme for this project. Users only need to wear lightweight gloves integrated with MPU6050 and NRF24L01 to issue basic control instructions. The vehicle body receives information from the upper-level Arduino and combines ultrasonic waves with ArUco visual inspection for action determination. The control signal is transmitted to the lower-level Arduino through the serial port to complete the motor execution.

✧ Pros:

- The gloves are lightweight and pressure-free to wear, suitable for long-term daily use scenarios.
- Supports dual perception of vision and distance, and has failure rollback (such as LED alarm).
- The control link is clear and the upper and lower layers are decoupled. It has strong scalability in the later stage (such as adding SLAM and RFID).
- No need for mobile phone operation. It truly frees your hands and is suitable for the elderly and those with limited mobility.

✧ Cons:

- Huge learning cost for Computer Vision and related Python operations, burdensome for electrical engineering students.
- The layering of the structure increases the difficulty of communication debugging, but it is controllable.

- Summary and decision-making basis

By comprehensively comparing the three schemes, we believe that although Scheme One (Bluetooth Cart) is easy to implement, it has obvious shortcomings in terms of user experience and adaptability. Scheme Two (Bionic Glove) has advantages in recognition accuracy, but the system is complex and the wearing burden is heavy. In contrast, Option Three (Gesture & Target Tracking) achieves the best balance among wearable lightweight, perceptual robustness, control accuracy and user adaptability. It is the implementation option that best conforms to the design concept of this project and the needs of the target users at the current stage.

C. Selection Rationale

After a comprehensive assessment of the performance of various schemes in terms of accuracy, interactivity, technical feasibility and user-friendliness, etc., we finally determined to adopt the following design:

- Users can control the movement of the cart through hand-held devices, achieving start-stop and direction control naturally and with quick response.
- The cart continuously tracks the user through visual recognition, enhancing accuracy and reliability.
- The system structure adopts a layered design, facilitating later maintenance and expansion. Specifically, decoupling the sensor recognition layer, control logic layer and execution driver layer enables the system to expand its functions in the later stage (such as adding SLAM and RFID modules) without making significant changes to the underlying hardware architecture [6][7].

This scene not only has a clear logic and controllable costs, but also highly caters to the usage scenarios of users in actual shopping, especially the operational needs of special groups such as the elderly.

Although the system structure has strong practicability and scalability, it may still be affected by visual recognition interferences such as illumination changes and signal disturbances in the actual environment. We will continuously optimize the relevant algorithms in the subsequent tests and development.

III. Prototyping and Testing

A. Prototyping Method and Structure

The prototype system of this project follows a hierarchical architecture and is divided into the Sensor Recognition Layer, the Control Logic Layer and the Execution Driver Layer. The decoupling and collaborative control of each functional unit are achieved in a modular way.

A.1 Sensor Recognition Layer

In the sensor recognition layer, the system is built based on the Arduino Nano platform worn on the user's hand. The core includes an MPU6050 attitude sensor, a breadboard (for signal distribution and switch connection), an NRF24L01 wireless transmission module, and an independent control switch. This module is responsible for collecting gesture information, identifying user instructions through the pitch Angle and roll Angle obtained by conversion, and transmitting the information to the main body of the car in real time through the wireless module.

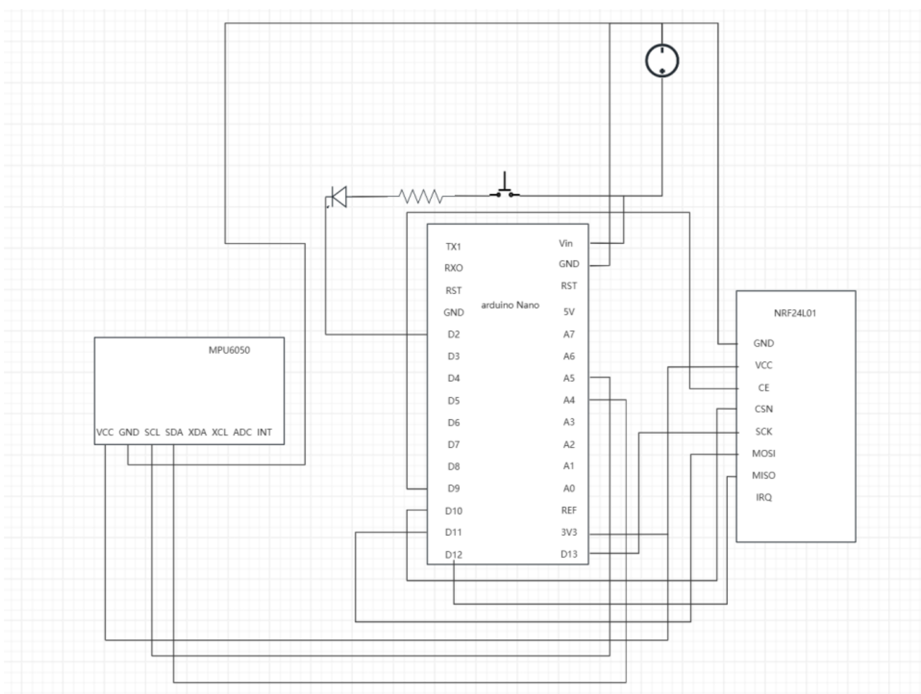


Fig. 4. Electrical Schematic Diagram of Sensor Recognition Layer

A.2 Control Logic Layer

The control logic layer is located in the Upper core section of the car, namely the Arduino Uno (Upper UNO) on the vehicle body. This layer contains two submodules, one is Upper UNO, while another one is Raspberry group. Upper UNO receives the wireless command signals from the hand and combines the information of the HC-SR04 ultrasonic sensor in the front for logical judgment to realize control strategies such as forward movement, obstacle avoidance and steering. In order to interact with the lower-level driver control logic, the upper-level Uno and the lower-level Uno achieve data synchronization and control instruction issuance through the IIC serial communication interface. For Raspberry group, the Raspberry 4B board is connected with three main components, namely another Arduino board (independent of Upper UNO), Camera, and OLED connected to that Arduino. The working mechanism is when the camera does not identify the Aruco code, OLED screen will display stressful emoji, and normal emoji vice versa.

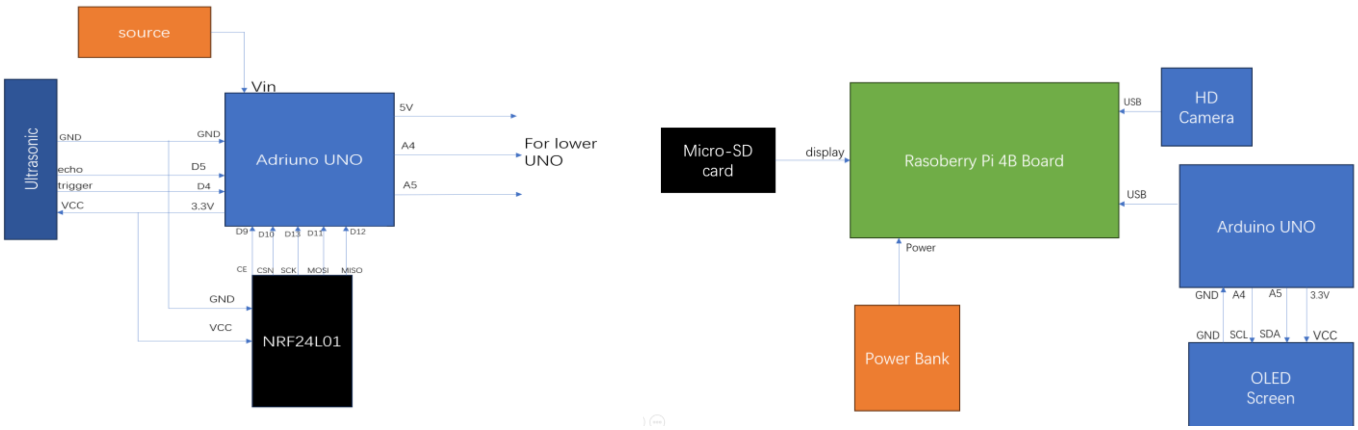


Fig. 5. Electrical Schematic Diagram of Control Logic Layer

A.3 Execution Driver Layer

The execution driver layer is composed of the Lower UNO and the L293D motor driver board. This driver board is directly connected to four TT DC motors and is responsible for executing the action instructions from the upper-level logic to achieve specific motion behaviors, such as acceleration, steering and stopping. This layer has a simple structure and rapid response, and it is the final execution part of the entire system.

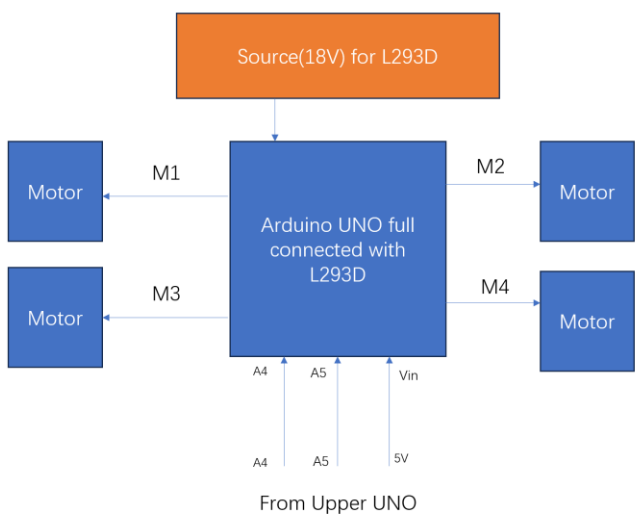


Fig. 6. Electrical Schematic Diagram of Execution Driver Layer

A.4 Overall Structure

The overall structure collaborates through wireless communication and serial communication to achieve a complete control closed loop from user gesture recognition, to logical judgment, and then to the underlying driver control. The connection of the relevant modules and the data path are shown in the figure.

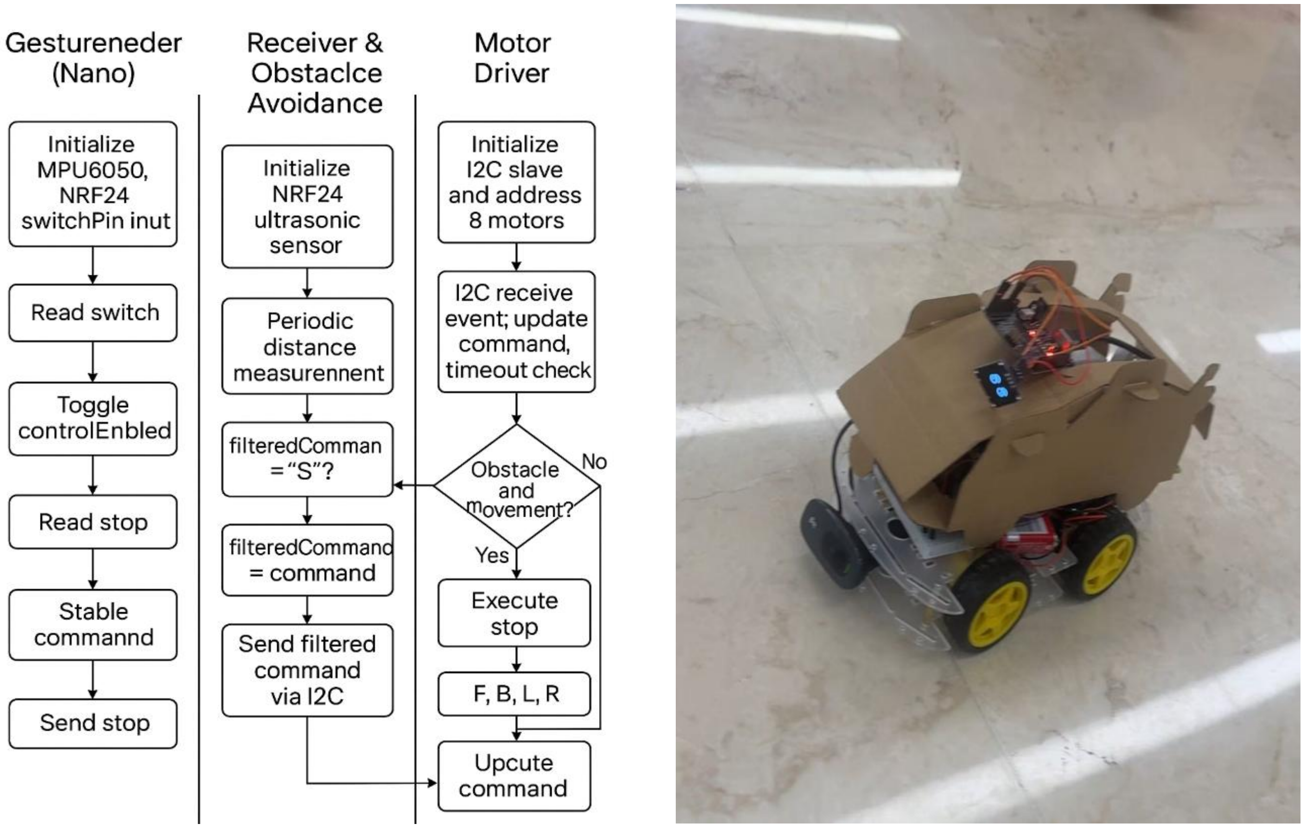


Fig. 7. System software logic (left) and final hardware implementation (right)

B.Testing Plan Based on User-Centered Requirement

To evaluate whether the proposed intelligent shopping cart prototype meets practical usage expectations, a testing strategy was developed based on user-centered requirements. The target users include elderly individuals, people with reduced mobility, and customers who may benefit from hands-free control during shopping. These users typically prioritize ease of operation, physical effort reduction, obstacle safety, and interaction clarity. Accordingly, the testing plan translates these needs into technical metrics that are verifiable through prototype experiments.

The intelligent cart system is structured into three primary functional layers—Sensor Recognition, Control Logic, and Execution Driver—each contributing to the overall user experience. For each layer, a set of design-level requirements was defined to reflect both technical goals and real-world usability considerations. Table 4 summarizes the relationship between user demands, corresponding engineering constraints, and the specific tests that were conducted to verify performance. This plan ensures that the testing activities are not only technically rigorous but also aligned with the social and practical needs of the system’s intended users.

Table 4. Testing Plan in Terms of Users

User Need	**Simple hands-free interaction**	**Avoid bumping into obstacles**	**Visual fallback or privacy safety**	**Real-time wireless signal response**	**Reliable motion execution**

Mapped Design Requirement	Stable gesture recognition (accuracy ≥ 90%)	Immediate stop if obstacle < 10 cm	Camera deactivates when no marker detected	NRF24L01 delay ≤ 300 ms	Command buffer filters out unstable inputs
Planned Testing Approach	MPU6050 angle threshold testing (Section C.1)	Ultrasonic sensor trigger distance sensing test (Section C.2)	ArUco-based LED alert testing (Section C.2)	Arduino controlled Serial port feedback test (Section C.3)	Timeout + Buffer logic validation (Section C.3)

C. Module Testing Results

This section presents the testing outcomes of each core subsystem within the proposed intelligent cart architecture. The system is divided into three primary functional layers: sensor recognition, control logic, and execution driver. For each layer, targeted experiments were conducted to evaluate performance, validate functionality, and identify potential areas for refinement. The results provide empirical support for both the reliability and the limitations of the current design.

C.1 Sensor Recognition Layer: Angle-Based Gesture Signal Processing

The gesture recognition module realizes intuitive and hands-free control of the trolley by converting wrist movements into direction instructions. This module is based on the MPU6050 gyroscope, installed on the back of the user's right hand, and connected to the Arduino Nano.

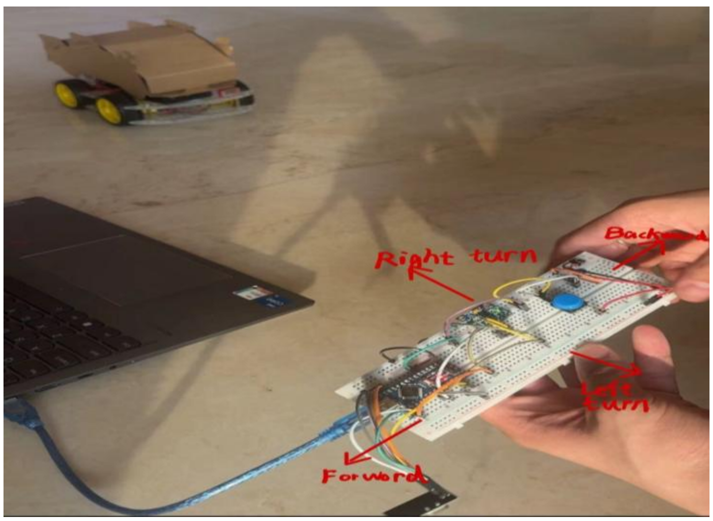


Fig. 8. Sensor Recognition Layer Physical Setup (**Red arrows indicate four gesture control directions**)

To reduce the complexity of interpreting the original three-axis acceleration values, we convert the sensor output into two intuitive variables - pitch and roll, representing the vertical and lateral tilt of the wrist respectively. Based on empirical tests, five core gestures were defined: forward, backward, left turn, right turn and steady. The mapping between gestures and the Angle domain is summarized in Table 5.

Table 5. The Basic Roll and Pitch Angle Components of Gesture

Gesture	**Roll(°)**	**Pitch(°)**
Front	*<15*	>20

Back	<15	<-20
Left	<-20	<15
Right	>-20	<15
Stable	others	others

To enhance robustness, we have set a threshold range for each Angle range to reduce the sensitivity to mild involuntary hand tremors. Furthermore, we have also implemented a signal buffering mechanism: instructions will only be accepted after being received three consecutive times, thereby filtering out transient noise and improving control stability. The anti-noise mechanism has been verified through MATLAB simulation (as shown in Figure 9), indicating that even in the case of small signal deviation, the system can accurately distinguish gesture conversion.

For ease of use, a toggle button is integrated in the glove, allowing users to manually enable or disable signal transmission. Once activated, the gesture signal will be transmitted to the trolley through the NRF24L01 wireless module, and the trolley will receive and execute these signals in real time.

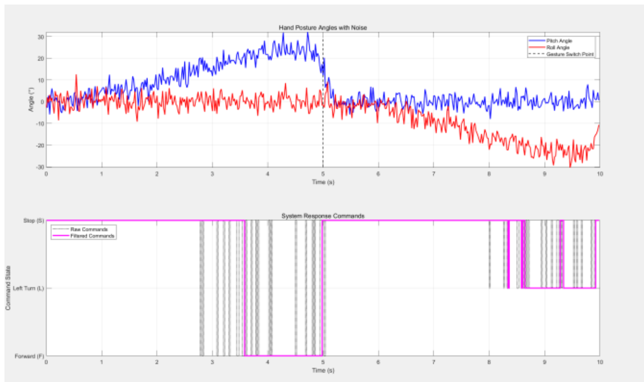


Fig. 9. Simulation Result of Forward and Turn Left Gesture with 3% Error

C.2 Control Logic Layer: Sensor-Visual Coordination for Motion Control

In the control logic layer, we have constructed a basic decision-making system based on ultrasonic ranging + wireless communication control. The car is equipped with ultrasonic sensors, a wireless receiving module (NRF24L01), and four DC motors controlled by the motor drive module (L293D). The working logic of the system is as follows: When receiving motion instructions such as "forward/left turn/right turn", the car will read the distance data ahead in real time and compare the instructions with the ranging results to determine whether it is in a safe state. Through testing, we found that the ultrasonic module can refresh the ranging data once every 100ms. After the distance threshold x was set and tested step by step within the range from 1cm to 20cm, it was finally determined that x=10cm was the safe parking threshold. That is, when the distance ahead is less than 10cm, the car will automatically stop to avoid collision. Meanwhile, the car can still accept the "reverse" command when it is stopped, so that the user can adjust the path.

To further enhance the reliability and multimodal perception capability of the system, we introduced a **visual-assisted recognition mechanism** based on Raspberry Pi and camera modules. This subsystem, consisting of Raspberry Pi 4B, a USB camera, and OpenCV, is responsible for detecting ArUco tags on the user's body. As shown in the **left part of Figure 10**, we installed VS Code and OpenCV in the Raspberry Pi's VNC environment and performed initial recognition tests to verify its detection accuracy and LED signaling logic.

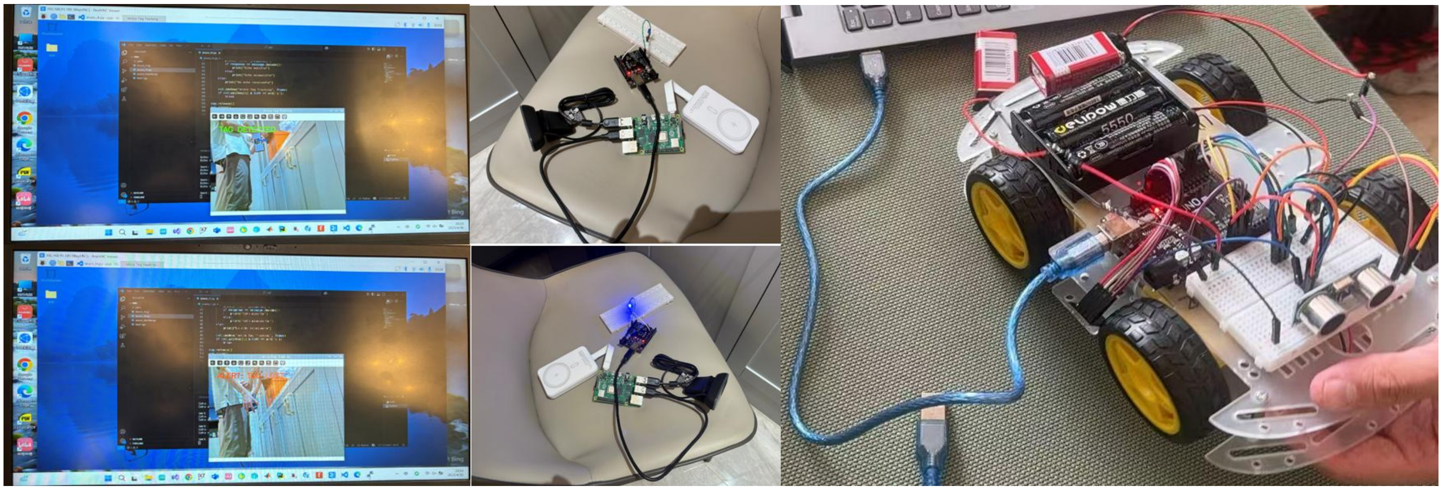


Fig. 10. Hardware setup of the control logic layer, including Raspberry Pi–based visual detection (left) and Arduino UNO–based motion control and wireless communication (right)

Meanwhile, the **right part of Figure 10** illustrates the hardware architecture of the upper-level Arduino UNO system, which independently controls the motor drive logic, ultrasonic obstacle avoidance, and wireless communication. Although the visual and motion control modules are eventually integrated, the two subsystems were tested separately during early development to ensure independent stability and ease of debugging.

To verify the stability and adaptability of the visual recognition system in real-world usage, we conducted a series of tests based on the ArUco tag recognition capability. As shown in **Figure 11**, the recognition accuracy remained above 90% within a distance of 2.0 meters (0.2–2.4 m), covering most practical indoor user-cart distances. Additionally, we evaluated the system's robustness under varying lighting conditions, from very dark (~10 lux) to overexposed (~8000 lux). The results show that while accuracy remained above 95% under typical indoor lighting (300–1000 lux), performance dropped significantly in extremely dim or overly bright settings, highlighting the importance of lighting calibration for reliable visual tracking.

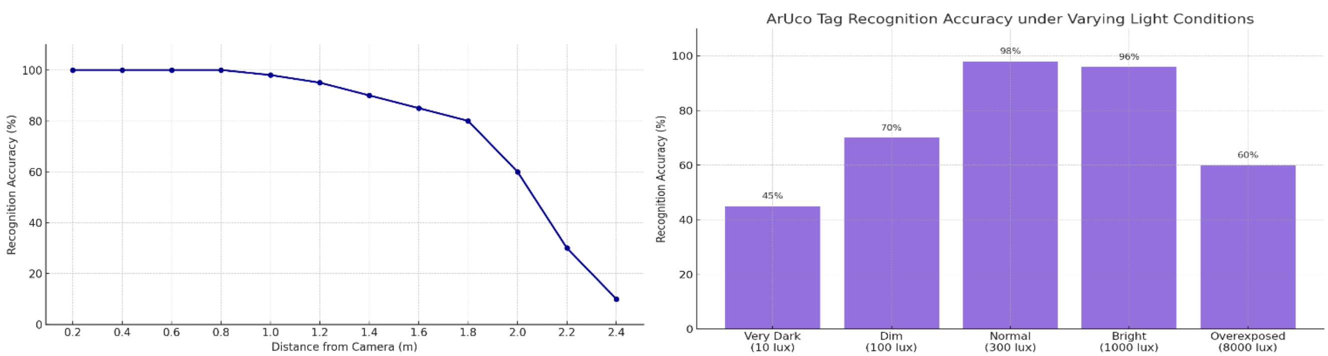


Fig. 11. ArUco Tag Recognition Accuracy: Distance (left) and Marker ID (right)

C.3 Execution Driver Layer: Timeout-Based Motion Control Optimization

For execution driver layer, the driven module had been tested its capability of controlling the motor. The motors are controlled by the Arduino Uno broad inside the cart, only when it receives the moving signal, the car can be driven. The physical connection between the Arduino UNO, the motor driver (L293D), and the DC motors is shown in Figure 12. This setup enables the system to interpret gesture signals and drive the motors accordingly, forming the foundation of the time-based motion continuity mechanism discussed in this section.

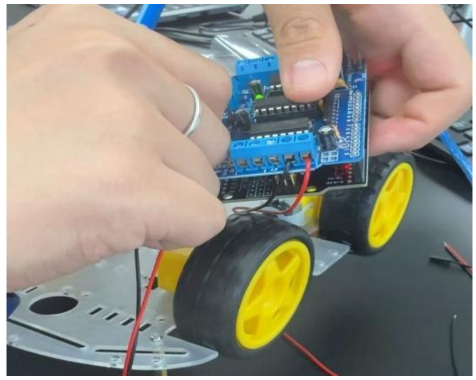


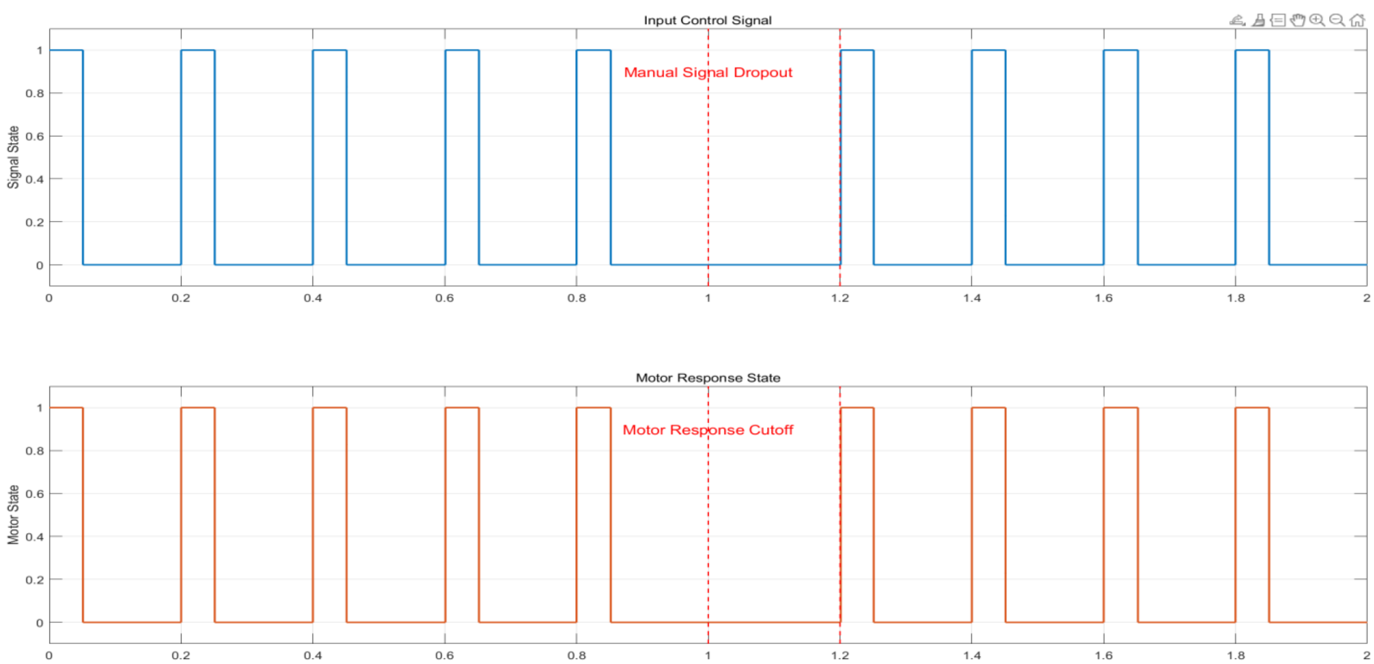
Fig. 12. Execution Driver Layer Hardware Layout: Arduino UNO and L293D

In this case, it is necessary to discuss the duty cycle between the message wave and the response of the motors. Considering the minimum time of sending a message from glove is 300ms. If the motor strictly following the signal it received, there is an obvious discontinuity when it moves. To overcome this issue, it is possible providing a larger acceptable response region for motor, overlapped the interval time of input signal. Mathematically, it can be written as the following formular

$$S(t) = \begin{cases} 1 & t - \tau(t) \leq T_{out} \\ 0 & others \end{cases}$$

$$\frac{d\tau(t)}{dt} = t_k \delta(t - t_k)$$

In this formular, S(t) is the moving state where the output 1 represents the moving case and 0 means the stopping case. The parameter Tout (Tout = 500ms) is the maximum acceptable region of time for motor moving, only when the time distance between instantaneous time t and the last time of signal sending ($\tau(t)$) equal or smaller than Tout, the vehicle is allowed to move. Following the formular, we use the MATLAB to simulate the situation of the time response of the motor and the signal input wave. The result of simulation is shown in Fig. 13. It can clearly illustrate that expending the acceptable region for motor response provide a continuous moving state of motors and avoids it drop into an endless loop of moving. It can ensure the vehicle moving smoothly.



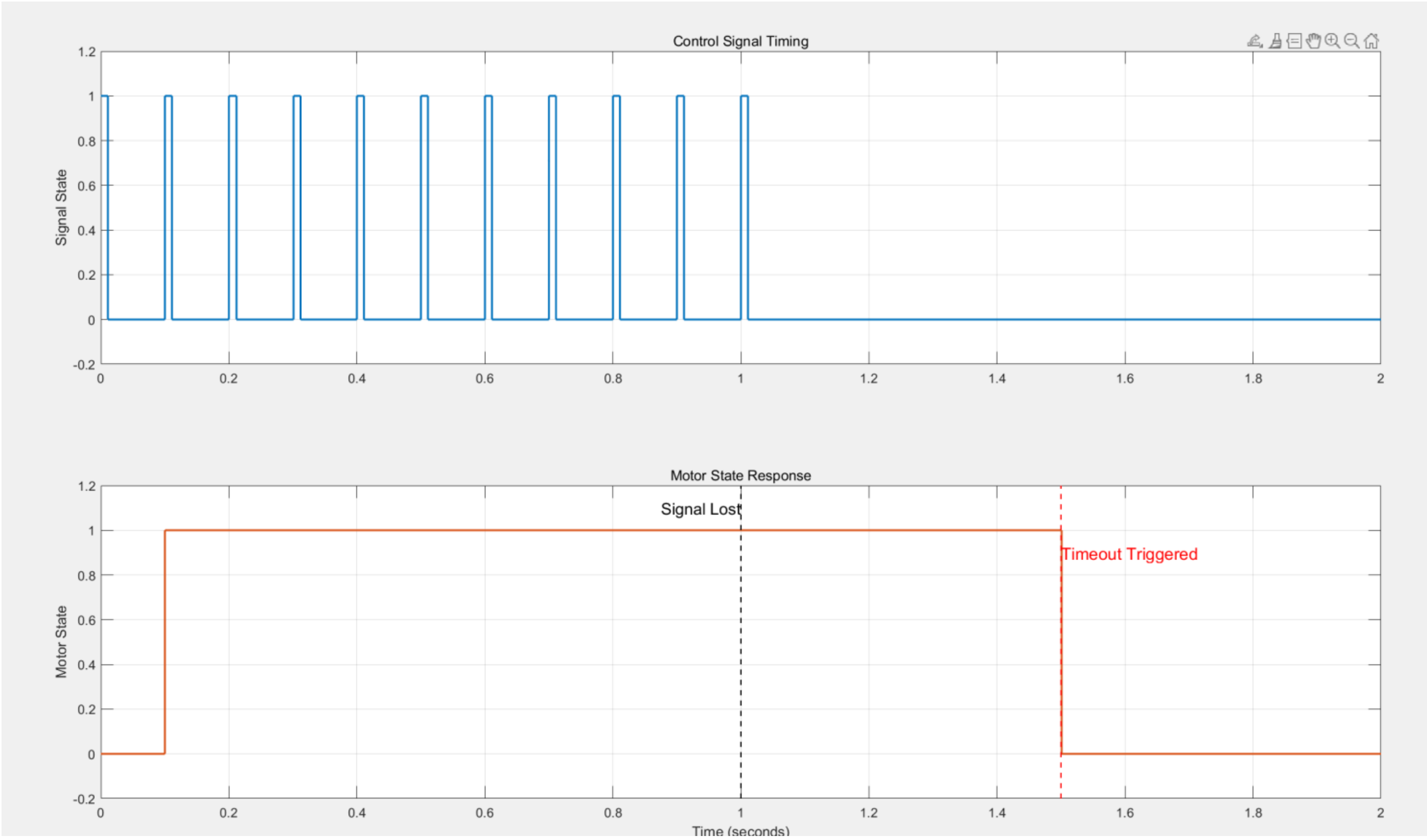


Fig. 13. Comparison of motor response and input signal: without (upper) and with (lower) extending the acceptable region

D. System Debugging and Iterative Optimization

The equations are an exception to the prescribed specifications of this template. You will need to determine whether or not your equation should be typed using either the Times New Roman or the Symbol font (please no other font). To create multileveled equations, it may be necessary to treat the equation as a graphic and insert it into the text after your paper is styled.

D.1 Sensor Recognition Layer: IMU Selection for Gesture Recognition

In order to achieve stable control based on gestures, this system needs to obtain hand movement data in real time through wearable devices. The core technical requirements include three-axis acceleration detection and rotational angular velocity recognition, which are used to distinguish static gestures from dynamic movements. In the early stage, we made a selection trade-off between MPU6050 and ADXL345, and determined the final sensor scheme through experiments and comparative analysis. (Note that ADXL345 has been provided to us in MEC202 Comp.Lab)

Table 6 shows the comparison of the two sensors in terms of key technical parameters and functional support. As a low-power three-axis accelerometer, the ADXL345 offers a high cost-performance ratio and is suitable for basic attitude detection. However, it does not support gyroscope measurement, cannot recognize rotational gestures, and its functional dimensions are limited. The MPU6050 integrates a three-axis accelerometer and a three-axis gyroscope, has a complete 6-degree-of-freedom Motion capture capability, and has a built-in DMP (Digital Motion Processor) module that can be used for attitude fusion and filtering, improving data stability and recognition accuracy.

Table 6. Comparison of MPU6050 and ADXL345 Sensor Features

Feature	Accelerometer	Gyroscope	Sensor Fusion (DMP)	I2C Interface Support	Gesture Recognition Support	Power Consumption	Cost
MPU6050	✓ 3-axis	✓ 3-axis	✓ Yes	✓	✓ (6-DOF)	Medium	¥14.5
ADXL345	✓ 3-axis	✗	✗ No	✓	✗ (Tilt only)	Low	¥6.8

Taking into account factors such as the functional dimension, output stability and engineering scalability, we finally chose the MPU6050 as the main sensor of the sensor recognition layer. It not only meets the basic pose input requirements, but also provides a data support basis for the subsequent more complex human motion recognition.

D.2 Control Logic Layer: Wireless Environment Impact on Development Stability

During the debugging process of the control logic layer of this project, Raspberry Pi 4B was used as the core platform, and remote development was carried out in combination with the camera and VNC virtual desktop. However, limited by the laboratory conditions, the system initially relied on mobile phone hotspots to provide Wi-Fi connections. Although the initial configuration was relatively convenient, serious instability problems were exposed during the subsequent debugging process.

Specifically, in classroom IR112 of the teaching building, there are a large number of terminal devices using the wireless network simultaneously, causing extreme congestion of the 2.4GHz channel resources and frequently leading to problems such as connection interruption, image lag, and remote desktop failure. Experimental analysis shows that with the increase in the number of concurrent devices, the failure rate of VNC connections rises significantly and the communication quality deteriorates rapidly.

Therefore, we suggest that in the future system development and debugging process, priority should be given to using wired networks or independent local area network environments to establish debugging channels, or exploring low-interference communication solutions such as Bluetooth Mesh and LoRa, to enhance the anti-noise ability and ensure the stability and continuity of the remote debugging process.

D.3 Execution Driver Layer Optimization: Motor control stability and response optimization

The execution drive layer is mainly responsible for converting the control instructions from the upper layer into specific motor actions to achieve the propulsion and steering of the vehicle. This layer relies on the Arduino UNO of the lower layer to control four DC motors, and the driving module is L293D. In the initial test, we adopted the traditional PID control strategy to adjust the speeds of the left and right wheels in order to achieve a smoother and more controllable steering. However, in actual operation, this strategy did not achieve the expected effect.

During the experiment, we observed that after receiving the steering command, although the car was able to turn, it often failed to resume straight driving after turning and even spun in place. After a vehicle performs a steering action, it is prone to continuous direction deviation due to inertia or delayed adjustment of the left and right wheels, making it difficult to automatically return to the correct direction. As shown in Figure 14, this is the motion trajectory generated when we execute two different control strategies through a unified control logic. It can be seen that the trajectory under PID control shows continuous deflection after turning, while differential control can stably complete the turning and quickly resume straight forward.

Fig. 14. Motion Trajectory Comparison under Differential Steering (left) and PID (right)

We analyze and believe that the occurrence of this phenomenon is not entirely due to the problem of PID parameter adjustment, but is related to the mechanical differences between the wheels and the support structure of this vehicle body. Unlike real vehicles, the small car we use has an independent four-wheel drive structure. The wheels and the vehicle body are rigidly connected, and natural centering cannot be achieved through the wheel steering mechanism (such as steering knuckles, steering bearings, etc.). As shown in Figure 15, we compared the steering wheel structure used in ordinary cars with the fixed axle structure of the small car in this project. The latter is unable to provide "passive return force" and natural direction adjustment capability during the steering process.

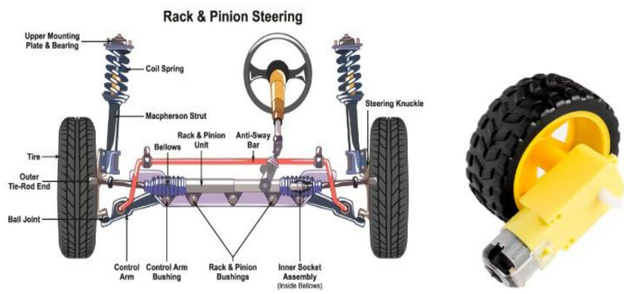


Fig. 15. Comparison of Typical Vehicle Steering Wheel Structure (left) and Our Rigid Wheel Mount (right)

Considering the limitations of the tire structure and the control requirements of the system response, we turned to the "differential steering control logic". Differential control achieves action control with different steering radii by manually setting the speed difference between the left and right wheels, which can avoid the oscillation and unnecessary correction caused by PID regulation. Figure 15 shows the comparison of the left and right wheel speeds under PID control and differential control. Under PID, the left and right wheels have oscillations and frequent fine-tuning, while the differential scheme presents a clean and clear speed switching logic.

E.Evaluation Criteria for User Experience

In order to systematically evaluate the actual performance of this project in terms of user-oriented goals, we referred to the five core user requirements defined in **Section III.B** and based on the data and optimization strategies in **Section III.C** and **Section III.D**, Analyze item by item whether it meets the expected requirements at the initial design. This process involves a comprehensive assessment of the accuracy of sensors, the robustness of visual recognition, the response time of wireless communication, and the smoothness of motion control, ensuring that from the perspective of the actual user experience, the engineering performance of this system is reversely verified to meet the original needs of users.\

Category	**Simple Hands-Free Interaction**	**Avoid Bumping into Obstacles**	**Visual Fallback or Privacy Safety**	**Real-Time Wireless Signal Response**	**Reliable Motion Execution**

Mapped Design Requirement	MPU6050 recognition accuracy ≥ 90%	Immediate stop if obstacle distance < 10 cm	LED alert when no ArUco marker detected	NRF24L01 latency ≤ 300 ms	Timeout mechanism + command buffer filter
Evaluation Outcome	Achieved 95% accuracy in gesture recognition. Requirement met	Cart reliably stopped at 10 ± 1 cm in testing. Requirement met	Visual alerts triggered as expected. See Fig. 11 for robustness verification	Measured average delay: 220 ms. Within acceptable response range	Passed multiple interference scenarios with 98% instruction integrity

Table 7. Evaluation Criteria and Outcomes for User Experience

F. User Feedback

For improving the project in the future, we invited few students in XJTLU to try to use our vehicle and asked them to give our group member comment about the advantages and shortcomings of the vehicle. All of them showed great interests on this new control style for a car. They said this type of controlling function can be easily used by those who have difficulty in moving. Some of them noticed that it can be more convenient if a screen could be added to the glove so that user could see the scene in front of the car. This is a thinkable advice from user in order to enhance the project.

However, they also complained that the car has a low stability for its shell and the electronic elements inside the cart had been displaced which is not elegant. Also, some of they said that this vehicle cannot be charge easily which needs to open the shell and change the battery and the following mode of this vehicle is not very intelligent. These issues are all needed to be considered for improving the project in the future.

The form of the questions we asked to users shows in appendix E. It contains few simple questions of the experience when they used. For protecting the privacy of the users, we had obtained the users' oral permission and do not record their personal information.

Through the feedback we generated, it can give a brief conclusion of our project. The outcome should be improved in the future, especially in enhancing the stability of the car and applying more friendly functions in interaction with customers.

IV. CONCLUSIONS

A. Project Summary and Outcome

This project aims to address the core issues in the traditional shopping model, especially for the elderly, those with limited mobility, and customers carrying a large number of items, such as the physical burden and operational inconvenience caused by pushing shopping carts. Based on this goal, we successfully designed and developed a prototype system of an intelligent shopping cart. This prototype system integrates multiple key functions, including intuitive direction and state adjustment through gesture control, stable automatic following function based on Raspberry PI and computer vision (ArUco tag recognition), as well as obstacle detection and avoidance using sensors. Through the integration and debugging of software and hardware, we have constructed a functional verification platform. After multiple tests, the prototype system verified the effectiveness of its main functions in a controlled environment, including the response to basic gesture instructions, the automatic tracking of targets labeled with specific tags, and the avoidance of simple obstacles. This indicates that the core design goals and key functional requirements of this project have been basically achieved, and a feasible and innovative conceptual prototype of an intelligent shopping assistance system has been successfully constructed.

B. Societal, Ethical and Environmental Impacts

In order to comprehensively evaluate the extended significance of this system in practical applications, we conducted an analysis from three dimensions: social impact, ethical controversy and environmental sustainability. The gesture control and automatic following functions effectively enhance the autonomy and sense of participation of specific groups in the retail environment, help alleviate the usage obstacles caused by physical differences, and reflect the positive role of this design in promoting social inclusiveness and equality. It is also closely related to the directions of the 10th "Reducing Inequality" and the 3rd "Health and Well-being" in the United Nations Sustainable Development Goals (SDGs).

Fig. 16. Sustainable Development Goals

However, in the process of achieving automatic following, this system adopts the camera-based ArUco recognition scheme. Although this approach has high technical practicability, it inevitably raises ethical concerns regarding privacy protection and public acceptance, especially the psychological burden of female users in the photographed positions and usage scenarios.

Furthermore, from an environmental perspective, the 9V disposable battery currently used in the prototype system involves a significant consumption of non-renewable resources and carbon emissions throughout its entire life cycle, posing a challenge to environmental sustainability.

C. Reflection and Future Recommendation

C.1 Project Reflection

In the prototype development and actual testing of this project, we have achieved core functions such as gesture control, ultrasonic following and camera recognition. However, three key issues have also been exposed, which need to be focused on in future optimizations:

1. Insufficient sustainability of the power supply solution: The currently used 9V disposable batteries have environmental burdens and are inconvenient to wear, making them unsuitable for long-term use.
2. Poor load-bearing capacity and adaptability to ramps of the trolley structure: In complex terrains such as supermarket slopes and elevators, the lack of a stable structure and braking control poses a risk of sliding out of control.
3. The camera positioning method has privacy and ethical controversies: The bottom camera recognizes the user's legs, which may raise privacy concerns and affect user acceptance.

C.2 Future Research Orientations

In response to the above problems, we put forward the following optimization suggestions:

1. Power Replacement: Replace disposable batteries with rechargeable lithium batteries and use lightweight button batteries in wearable devices.
2. Structural Improvement: Introduce load detection and improve the suspension system to enhance the stability of the trolley in slope scenarios.

3. Visual Alternative Solution: Replace tag identification with non-contact methods (such as Bluetooth/RFID), and at the same time adopt edge computing technology to prevent image upload and alleviate users' privacy concerns [9].
4. Expansion Capacity Building: Introduce SLAM to achieve autonomous navigation, upgrade the communication module to support encryption and multi-vehicle collaboration, and lay out future supermarket application scenarios [10].

C.3 Commercialization Potential and User Expansion Direction

Although the current project focuses primarily on prototype validation, its design concept and functional framework exhibit clear potential for real-world commercial deployment. To facilitate future market entry and broader user adoption, several strategic directions are proposed:

- **Differentiated Pricing and Modular Design**: Offer both standard and premium versions of the smart cart to cater to different user needs and market segments, with modular components allowing for flexible configuration.
- **Accessibility-Oriented Features:** Incorporate barrier-free functionalities such as voice control, automatic following, and assistance modes designed for elderly users and individuals with mobility impairments.
- **Enhanced User Interaction Interface:** Add touch display screens, real-time voice feedback, and multilingual support to improve the shopping experience across user demographics.
- **Post-Sales Support Mechanism:** Develop a maintenance network, battery replacement plans, and technical support channels to ensure product sustainability and long-term user satisfaction.
- **System Integration for Retail Environments:** Explore integration with supermarket management systems to enable functions like in-store navigation, real-time inventory alerts, and automated checkout, thereby expanding commercial utility beyond cart mobility.

These enhancements are intended not only to increase the system’s practical value but also to lay a solid foundation for scalable commercialization and future smart retail ecosystems.

D. Individual Contribution Description

Name	Workload
Hengzhi Qiu (25%)	Assembly of the main circuit and each module of the gesture-controlled car. Help with the construction of the gesture-controlled car code. Write a part of the report.
Suheng Ji (25%)	Build a 3D model of the exterior shell for the gesture-driven car. Assembling OLED for the gesture-controlled car. Shoot a video of the gesture-controlled car. Write a part of the report.
Yaoran Xu (25%)	Editing and production of the video of the gesture-controlled car. Help connect the circuits of some modules of the gesture-controlled car. Write a part of the report.
Yukun Zheng (25%)	Assign tasks to each link in the process. Write the codes for each module of the gesture-controlled car. Assist in the installation of the gesture-controlled car. Write a part of the the report.

V. REFERENCES

[1] L. Ji-Herr, U. K. Rosli, L. T. Chuan, S. C. Sen, and L. Y. Xia, "Revolution of Retail Industry: From Perspective of Retail 1.0 to 4.0," in *Procedia Computer Science*, vol. 202, pp. 353-359, 2022. doi: 10.1016/j.procs.2022.01.052.

[2] Y. Mercan, K. T. Selcuk, and A. A. Sayilan, "The relationship between types of physical disabilities and the Instrumental Activities of Daily Living (IADL) in the elderly," *Family Medicine & Primary Care Review*, vol. 23, no. 1, pp. 22-28, 2021. doi: 10.5114/fmpcr.2021.103153.

[3] K. Kaman and A. K. Deshmukh, "Universal Access to the Internet for Attaining Responsible Consumption-Based Sustainable Development Goal (SDG)," in *Digital Technologies to Implement the UN Sustainable Development Goals*, W. Leal Filho, S. Kautish, T. Wali, S. Rewhorn, and S. K. Paul, Eds. Springer, Cham, 2024, ch. 22. doi: 10.1007/978-3-031-68427-2_22.

[4] M. Shahroz, M. F. Mushtaq, M. Ahmad, S. Ullah, A. Mehmood and G. S. Choi, "IoT-Based Smart Shopping Cart Using Radio Frequency Identification," in IEEE Access, vol. 8, pp. 68426-68438, 2020, doi: 10.1109/ACCESS.2020.2986681.

[5] H. Basanta, Y. -P. Huang and T. -T. Lee, "Assistive design for elderly living ambient using voice and gesture recognition system," 2017 IEEE International Conference on Systems, Man, and Cybernetics (SMC), Banff, AB, Canada, 2017, pp. 840-845, doi: 10.1109/SMC.2017.8122714.

[6] M. Pomranz, “Amazon's Smart Shopping Carts Are Coming to Whole Foods,” Food & Wine, Jul. 14, 2022. [Online]. Available: https://www.foodandwine.com/news/amazon-smart-shopping-carts-whole-foods

[7] Instacart, “Caper Smart Cart,” caper.ai, Accessed: May 10, 2025. [Online]. Available: https://caper.ai/caper-smart-cart

[8] Shopic, “Walmart Chile Rolls Out AI-Powered Smart Shopping Carts,” Shopic, Dec. 25, 2024. [Online]. Available: https://www.shopic.co/news/walmart-chile-rolls-out-ai-powered-smart-shopping-carts

[9] Z. Ali and R. Sonkusare, "RFID based Smart Shopping: An overview," 2014 International Conference on Advances in Communication and Computing Technologies (ICACACT 2014), Mumbai, India, 2014, pp. 1-3, doi: 10.1109/EIC.2015.7230698.

[10] T. Taketomi, H. Uchiyama, and S. Ikeda, "Visual SLAM algorithms: a survey from 2010 to 2016," *IPSJ Transactions on Computer Vision and Applications*, vol. 9, no. 16, 2017. doi: 10.1186/s41074-017-0027-2.

VI. APPENDICES

Appendix A: Circuit Diagrams

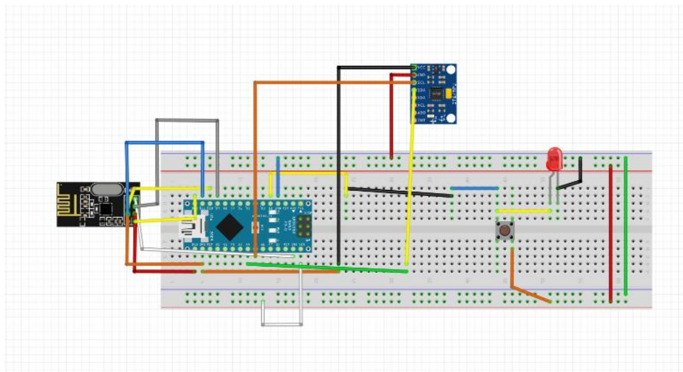


Fig. 17. Circuit Wiring Schematic for Sensor Logic Layer

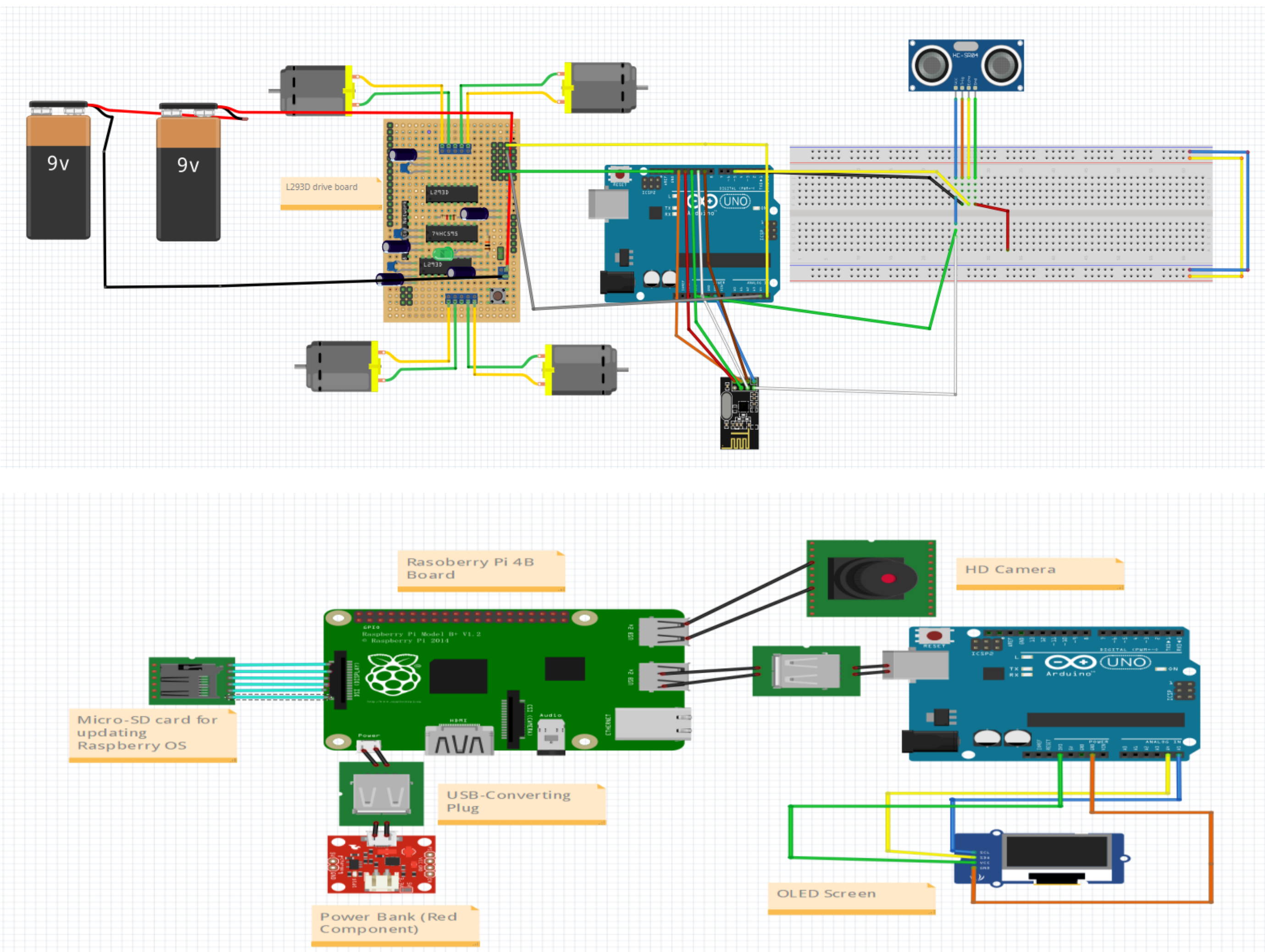


Fig. 18. Circuit Wiring Schematic for Control Logic Layer & Execution Driver Layer

Note that the UNO shown in the figure stands for the Upper UNO, while the lower UNO is placed below the L293D Motor Driven Board. Through observation, we can find that there are some blue colors floating on yellow Motor Driven Board, which stands for the lower UNO.

Moreover, the Raspberry pi 4B board is connected with camera, another UNO and power bank, and the OLED screen is connected to the UNO.

Appendix B: Bill of Materials and Prices

Table 8. Price Lists

Type	Components	Number	Price(¥)
Electronic Components	Arduino uno	4	50.4
	Arduino nano	1	5.88
	MPU6050	1	20
	ultrasonic sensor	1	1.25
	L293D motor drive board	1	11.21
	DC motor	4	12.545
	nRF24L01 Wireless module	2	3.35
	OLED screen	1	11.4
	9V battery	3	3
	18V battery.	1	11.4
	Raspberry Pi 4B	1	429
	Camera	1	15
	Dual-Port line	-	4
	Breadboard	3	40.5
	Resistor	-	2
	LED	1	1.5
Mechanical Components	Button	1	1
	Car Wheels	4	10
	3D Printing Shell	1	-
	Cardboard-Mde Car Shell	1	7
	Power bank	1	119
Total price		759.435	

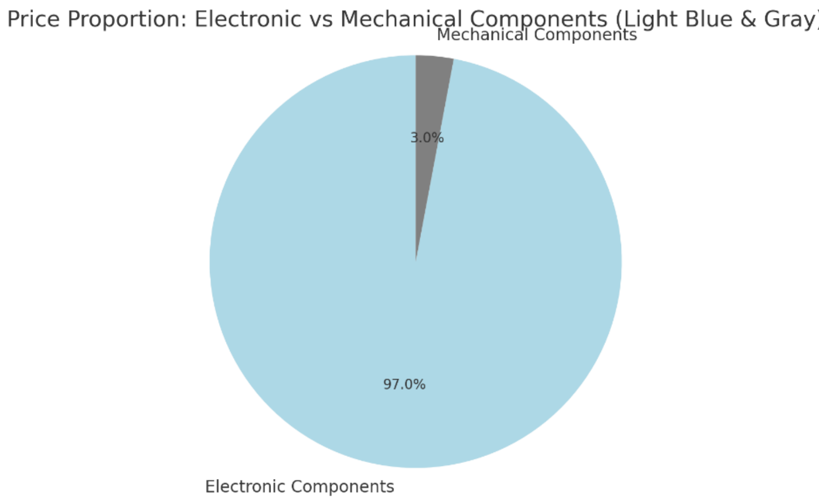


Fig. 19. Comparison Between Electronic & Mechanical Components

Appendix C: Team Workload Distribution Table

Table 9. Team Workload Distribution

Name	Workload
Hengzhi Qiu (25%)	Assembly of the main circuit and each module of the gesture-controlled car. Help with the construction of the gesture-controlled car code. Write a part of the report.
Suheng Ji (25%)	Build a 3D model of the exterior shell for the gesture-driven car. Assembling OLED for the gesture-controlled car. Shoot a video of the gesture-controlled car. Write a part of the report.
Yaoran Xu (25%)	Editing and production of the video of the gesture-controlled car. Help connect the circuits of some modules of the gesture-controlled car. Write a part of the report.
Yukun Zheng (25%)	Assign tasks to each link in the process. Write the codes for each module of the gesture-controlled car. Assist in the installation of the gesture-controlled car. Write a part of the report.

Appendix D: Meeting Minutes

Table 10. Team Meeting Minutes

Meeting Memo #1				
Date	**Time**	**Duration**	**Location**	**Attendance**
February 28th	**2:00**	**3 hours**	**IR112**	**All**
Meeting Summary				
The preparation of project components: ● **Arduino UNO*3 & Arduino Nano** ● **L293d Motor Driver Board** ● **Two IR Sensors** ● **Four Wheels & Motor Assembly** ● **Ultrasonic Sensor** ● **Battery & Battery Box** ● **MPU-6050 Gyroscope**				
Design concept:				

● **Who needs it? (Target Group)** ● **When do they need it? (Application Scenarios)** ● **Why do consumers need your product instead of others? (Competitiveness)**

Meeting Memo #2				
Date	**Time**	**Duration**	**Location**	**Attendance**
March 7th	**2:00**	**3 hours**	**IR112**	**All**
Meeting Summary				
Add two modules: ● **Communication module (NRF24L01)** ● **Visual recognition module (Raspberry Pi & Camera)**				
Discussion on the general idea of the Project Proposal and the wiring strategy for the circuitry of the small vehicle.				

Meeting Memo #3				
Date	**Time**	**Duration**	**Location**	**Attendance**
March 14th	**2:00**	**3 hours**	**IR112**	**All**
Meeting Summary				
● **Complete the tests of all components** ● **Build the base of the car** ● **Clarify the plan for next week such as the connections among various components in the circuit.**				

Meeting Memo #4				
Date	**Time**	**Duration**	**Location**	**Attendance**
March 21th	**2:00**	**3 hours**	**IR112**	**All**
Meeting Summary				
Conduct preliminary testing of the code in each section of the software layer: ● **IR sensor code** ● **master and slave RF Arduino Nano code** ● **master and slave RF Arduino UNO code**				
In terms of hardware, the assembly of the main circuit of the car body has been completed.				

Meeting Memo #5				
Date	**Time**	**Duration**	**Location**	**Attendance**
March 28th	**2:00**	**3 hours**	**IR112**	**All**
Meeting Summary				
No longer use the SG90 servo motor and IR Sensors				
Updated project concept: **Build a small vehicle that can follow the elderly person's movements, support gesture-based start/stop functions, and achieve continuous tracking through visual recognition of leg tags (ArUco) to ensure safe following and avoid misjudgment of the target.**				

Meeting Memo #6				
Date	**Time**	**Duration**	**Location**	**Attendance**
April 4th	**2:00**	**3 hours**	**IR112**	**All**
Meeting Summary				
Problems with the movement: **The movement of the small car is jerky and not smooth enough.**				
Solution approach:				

● **Change the code logic to stop only when a stop signal is received** ● **Set an instruction invalidation timer**

Meeting Memo #7				
Date	**Time**	**Duration**	**Location**	**Attendance**
April 11th	**2:00**	**3 hours**	**IR112**	**All**
Meeting Summary				
The car suddenly failed to respond. After rechecking all the parts, there was no abnormality. Eventually, we believed that the RF communication was affected by temperature and mobile phones, etc., and thus failed to work.				

Meeting Memo #8				
Date	**Time**	**Duration**	**Location**	**Attendance**
April 18th	**2:00**	**3 hours**	**IR112**	**All**
Meeting Summary				
After eliminating the interference from RF communication, we debugged the turning logic and switches of the car based on Week 7's work, and started preparing to write the experimental report.				

Meeting Memo #9				
Date	**Time**	**Duration**	**Location**	**Attendance**
April 25th	**2:00**	**3 hours**	**IR112**	**All**
Meeting Summary				
● **The computer vision function has been successfully implemented.** ● **The Raspberry Pi camera on the computer is linked with the mobile hotspot Wi-Fi.** ● **The part of the car is currently undergoing debugging for the automatic following code.** ● **The obstacle avoidance emergency stop test has been completed.**				

Meeting Memo #10				
Date	**Time**	**Duration**	**Location**	**Attendance**
May 2nd	**1:30**	**1.5 hours**	**SC169**	**All**
Meeting Summary				
● **Summary of experimental report** ● **Start making the poster** ● **Take photos and edit videos**				

Appendix E: Work Logs / Personal Diaries

- **Hengzhi Qiu**

2024.2.28
Participated in the initial hardware preparation and confirmed component list including Arduino, motors, and sensors.
2024.3.14
Helped complete the circuit and physical chassis assembly of the gesture-controlled car.
2024.3.21
Took part in initial code testing for IR sensors and RF communication modules.

2024.4.4
Worked on debugging jerky movement by adjusting motor response timing and implementing stop signal logic.
2024.4.18
Assisted in debugging turning logic and ensured correct execution of gesture commands.

- **Suheng Ji**

2024.3.7
Proposed adding the visual recognition module using Raspberry Pi and camera.
2024.3.28
Contributed to the updated concept of leg-following using ArUco markers, replacing the original IR strategy.
2024.4.11
Investigated issues in RF communication and proposed shielding and interference elimination.
2024.4.25
Helped implement computer vision functionality; tested Raspberry Pi camera for real-time tracking.
2024.5.2
Participated in poster design and video shooting for the final report.

- **Yaoran Xu**

2024.3.14
Participated in early circuit wiring and assembly tasks.
2024.4.4
Helped analyze movement instability issues and assisted in implementing instruction timeouts.
2024.4.25
Assisted with emergency stop test based on obstacle detection using ultrasonic sensors.
2024.5.2
Handled report editing and layout structure discussion.
2024.5.6
Completed final edits of demo video and compiled full experiment summary.

- **Yukun Zheng**

2024.3.7
Responsible for defining communication architecture using NRF24L01; contributed to proposal writing.
2024.3.21
Led the assembly of master-slave control system using two Arduino UNOs and Nanos.
2024.4.4
Installed and configured Raspberry Pi system; installed Visual Studio Code and related extensions.

2024.4.11
Completed OpenCV installation and followed online tutorials to understand image recognition workflows.
2024.4.25
Tested ArUco marker recognition via Raspberry Pi camera; helped refine the automatic following function.
2024.5.6
Coordinated the team for demo filming; finalized the distribution report and task summaries.

Appendix F: Codes

- **Aruco QR Code Detecting Algorithm**

```
import cv2
import cv2.aruco as aruco
import serial
import time

print("OpenCV version:", cv2.__version__)

ser = serial.Serial('/dev/ttyUSB0', 9600)
time.sleep(2)

cap = cv2.VideoCapture(0)

aruco_dict = aruco.getPredefinedDictionary(aruco.DICT_4X4_50)
parameters = aruco.DetectorParameters()
detector = aruco.ArucoDetector(aruco_dict, parameters)

while True:
    ret, frame = cap.read()
    if not ret:
        print("Image capture failed")
        break

    gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)
    corners, ids, _ = detector.detectMarkers(gray)

    if ids is not None and len(ids) > 0:
        aruco.drawDetectedMarkers(frame, corners, ids)
        message = b'Y'
        cv2.putText(frame, "TAG DETECTED", (40, 60), cv2.FONT_HERSHEY_SIMPLEX, 1.2, (0, 255,
0), 3)
    else:
        message = b'N'
        cv2.putText(frame, "ALERT: TAG LOST", (40, 60), cv2.FONT_HERSHEY_SIMPLEX, 1.2, (0, 0,
255), 3)
```

```python
    ser.write(message)
    print(f"Sent: {message.decode()}")
    time.sleep(0.05)

    if ser.in_waiting:
        response = ser.readline().decode().strip()
        print(f"Echo: {response}")
        if response == message.decode():
            print("Echo match\n")
        else:
            print("Echo mismatch\n")
    else:
        print("No echo received\n")

    cv2.imshow("ArUco Tag Tracking", frame)
    if cv2.waitKey(1) & 0xFF == ord('q'):
        break

cap.release()
ser.close()
cv2.destroyAllWindows()
```

- **OLED Response Based On the QR Code Raspberry Detects**

```cpp
#include <Wire.h>
#include <Adafruit_GFX.h>
#include <Adafruit_SSD1306.h>

#define SCREEN_WIDTH 128
#define SCREEN_HEIGHT 64

Adafruit_SSD1306 display(SCREEN_WIDTH, SCREEN_HEIGHT, &Wire, -1);

const int ledPin = 9;  // LED 正极接 D9，负极接 GND

// 哭脸图像数据
static const unsigned char PROGMEM image_data_DISTRESSED_EYESarray[] = {
  // ARRAY for DISTRESSED_EYES
  0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00,
  0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00,
  0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00,
  0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00,
  0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00,
```

```
  0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00,
  0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00,
  0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00,
  0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00,
  0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00,
  0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00,
  0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00,
  0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00,
  0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00,
  0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00,
  0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x07, 0xe0, 0x00, 0x00, 0x00, 0x00,
0x00,
  0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x07, 0xf0, 0x00, 0x00, 0x00, 0x00,
0x00,
  0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x38, 0x00, 0x00, 0x0f, 0xfc, 0x00, 0x00, 0x00, 0x00,
0x00,
  0x00, 0x00, 0x00, 0x00, 0x00, 0x07, 0xf8, 0x00, 0x00, 0x0f, 0xff, 0x00, 0x00, 0x00, 0x00,
0x00,
  0x00, 0x00, 0x00, 0x00, 0x00, 0x3f, 0xfc, 0x00, 0x00, 0x1f, 0xff, 0x80, 0x00, 0x00, 0x00,
0x00,
  0x00, 0x00, 0x00, 0x00, 0x00, 0xff, 0xfc, 0x00, 0x00, 0x1f, 0xff, 0x80, 0x00, 0x00, 0x00,
0x00,
  0x00, 0x00, 0x00, 0x00, 0x01, 0xff, 0xfe, 0x00, 0x00, 0x3f, 0xff, 0xf8, 0x00, 0x00, 0x00,
0x00,
  0x00, 0x00, 0x00, 0x00, 0x03, 0xff, 0xfe, 0x00, 0x00, 0x3f, 0xff, 0xfc, 0x00, 0x00, 0x00,
0x00,
  0x00, 0x00, 0x00, 0x00, 0x3f, 0xff, 0xfe, 0x00, 0x00, 0x3f, 0xff, 0xff, 0x00, 0x00, 0x00,
0x00,
  0x00, 0x00, 0x01, 0xff, 0xff, 0xff, 0xff, 0x00, 0x00, 0x7f, 0xff, 0xff, 0xc0, 0x00, 0x00,
0x00,
  0x00, 0x00, 0x0f, 0xff, 0xff, 0xff, 0xff, 0x00, 0x00, 0x7f, 0xff, 0xff, 0xf8, 0x00, 0x00,
0x00,
  0x00, 0x00, 0x0f, 0xff, 0xff, 0xff, 0xff, 0x00, 0x00, 0x7f, 0xff, 0xff, 0xff, 0xe0, 0x00,
0x00,
  0x00, 0x00, 0x0f, 0xff, 0xff, 0xff, 0xff, 0x00, 0x00, 0x7f, 0xff, 0xff, 0xff, 0xf8, 0x00,
0x00,
  0x00, 0x00, 0x0f, 0xff, 0xff, 0xff, 0xff, 0x00, 0x00, 0x7f, 0xff, 0xff, 0xff, 0xf8, 0x00,
0x00,
```

```
  0x00, 0x00, 0x0f, 0xff, 0xff, 0x8f, 0xff, 0x00, 0x00, 0x7f, 0xf1, 0xff, 0xff, 0xf8, 0x00,
0x00,
  0x00, 0x00, 0x0f, 0xff, 0xff, 0x07, 0xff, 0x00, 0x00, 0x7f, 0xe0, 0xff, 0xff, 0xf8, 0x00,
0x00,
  0x00, 0x00, 0x0f, 0xff, 0xff, 0x07, 0xff, 0x00, 0x00, 0x7f, 0xe0, 0xff, 0xff, 0xf8, 0x00,
0x00,
  0x00, 0x00, 0x0f, 0xff, 0xfe, 0x03, 0xff, 0x00, 0x00, 0x7f, 0xc0, 0x7f, 0xff, 0xf8, 0x00,
0x00,
  0x00, 0x00, 0x0f, 0xff, 0xfe, 0x03, 0xff, 0x00, 0x00, 0x7f, 0xc0, 0x7f, 0xff, 0xf8, 0x00,
0x00,
  0x00, 0x00, 0x0f, 0xff, 0xfe, 0x03, 0xff, 0x00, 0x00, 0x7f, 0xc0, 0x7f, 0xff, 0xf8, 0x00,
0x00,
  0x00, 0x00, 0x0f, 0xff, 0xfe, 0x03, 0xff, 0x00, 0x00, 0x7f, 0xc0, 0x7f, 0xff, 0xf8, 0x00,
0x00,
  0x00, 0x00, 0x0f, 0xff, 0xfe, 0x03, 0xff, 0x00, 0x00, 0x7f, 0xc0, 0x7f, 0xff, 0xf8, 0x00,
0x00,
  0x00, 0x00, 0x0f, 0xff, 0xff, 0x07, 0xff, 0x00, 0x00, 0x7f, 0xe0, 0xff, 0xff, 0xf8, 0x00,
0x00,
  0x00, 0x00, 0x0f, 0xff, 0xff, 0x07, 0xff, 0x00, 0x00, 0x7f, 0xe0, 0xff, 0xff, 0xf8, 0x00,
0x00,
  0x00, 0x00, 0x0f, 0xff, 0xff, 0x8f, 0xff, 0x00, 0x00, 0x7f, 0xf1, 0xff, 0xff, 0xf8, 0x00,
0x00,
  0x00, 0x00, 0x0f, 0xff, 0xff, 0xff, 0xff, 0x00, 0x00, 0x7f, 0xff, 0xff, 0xff, 0xf8, 0x00,
0x00,
  0x00, 0x00, 0x07, 0xff, 0xff, 0xff, 0xfe, 0x00, 0x00, 0x3f, 0xff, 0xff, 0xff, 0xf0, 0x00,
0x00,
  0x00, 0x00, 0x07, 0xff, 0xff, 0xff, 0xfe, 0x00, 0x00, 0x3f, 0xff, 0xff, 0xff, 0xf0, 0x00,
0x00,
  0x00, 0x00, 0x07, 0xff, 0xff, 0xff, 0xfe, 0x00, 0x00, 0x3f, 0xff, 0xff, 0xff, 0xf0, 0x00,
0x00,
  0x00, 0x00, 0x03, 0xff, 0xff, 0xff, 0xfc, 0x00, 0x00, 0x1f, 0xff, 0xff, 0xff, 0xe0, 0x00,
0x00,
  0x00, 0x00, 0x03, 0xff, 0xff, 0xff, 0xfc, 0x00, 0x00, 0x1f, 0xff, 0xff, 0xff, 0xe0, 0x00,
0x00,
  0x00, 0x00, 0x01, 0xff, 0xff, 0xff, 0xf8, 0x00, 0x00, 0x0f, 0xff, 0xff, 0xff, 0xc0, 0x00,
0x00,
  0x00, 0x00, 0x01, 0xff, 0xff, 0xff, 0xf8, 0x00, 0x00, 0x0f, 0xff, 0xff, 0xff, 0xc0, 0x00,
0x00,
  0x00, 0x00, 0x00, 0xff, 0xff, 0xff, 0xf0, 0x00, 0x00, 0x07, 0xff, 0xff, 0xff, 0x80, 0x00,
0x00,
  0x00, 0x00, 0x00, 0xff, 0xff, 0xff, 0xf0, 0x00, 0x00, 0x07, 0xff, 0xff, 0xff, 0x80, 0x00,
0x00,
  0x00, 0x00, 0x00, 0x7f, 0xff, 0xff, 0xe0, 0x00, 0x00, 0x03, 0xff, 0xff, 0xff, 0x00, 0x00,
0x00,
  0x00, 0x00, 0x00, 0x3f, 0xff, 0xff, 0xc0, 0x00, 0x00, 0x01, 0xff, 0xff, 0xfe, 0x00, 0x00,
0x00,
  0x00, 0x00, 0x00, 0x1f, 0xff, 0xff, 0x80, 0x00, 0x00, 0x00, 0xff, 0xff, 0xfc, 0x00, 0x00,
0x00,
```

```
  0x00, 0x00, 0x00, 0x0f, 0xff, 0xff, 0x00, 0x00, 0x00, 0x00, 0x7f, 0xff, 0xf8, 0x00, 0x00,
0x00,
  0x00, 0x00, 0x00, 0x07, 0xff, 0xfe, 0x00, 0x00, 0x00, 0x00, 0x3f, 0xff, 0xf0, 0x00, 0x00,
0x00,
  0x00, 0x00, 0x00, 0x03, 0xff, 0xfc, 0x00, 0x00, 0x00, 0x00, 0x1f, 0xff, 0xe0, 0x00, 0x00,
0x00,
  0x00, 0x00, 0x00, 0x00, 0xff, 0xf0, 0x00, 0x00, 0x00, 0x00, 0x07, 0xff, 0x80, 0x00, 0x00,
0x00,
  0x00, 0x00, 0x00, 0x00, 0x1f, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0xfc, 0x00, 0x00, 0x00,
0x00,
  0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00,
  0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00,
  0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00,
  0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00,
  0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00,
  0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00,


};

// 笑脸图像数据
static const unsigned char PROGMEM image_data_EYES_FRONTarray[] = {
  // ARRAY for EYES_FRONT
  0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00,
  0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00,
  0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00,
  0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00,
  0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00,
  0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00,
  0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00,
  0x00, 0x00, 0x00, 0x00, 0x1f, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0xfc, 0x00, 0x00, 0x00,
0x00,
  0x00, 0x00, 0x00, 0x00, 0xff, 0xf0, 0x00, 0x00, 0x00, 0x00, 0x07, 0xff, 0x80, 0x00, 0x00,
0x00,
```

```
  0x00, 0x00, 0x00, 0x03, 0xff, 0xfc, 0x00, 0x00, 0x00, 0x00, 0x1f, 0xff, 0xe0, 0x00, 0x00,
0x00,
  0x00, 0x00, 0x00, 0x07, 0xff, 0xfe, 0x00, 0x00, 0x00, 0x00, 0x3f, 0xff, 0xf0, 0x00, 0x00,
0x00,
  0x00, 0x00, 0x00, 0x0f, 0xff, 0xff, 0x00, 0x00, 0x00, 0x00, 0x7f, 0xff, 0xf8, 0x00, 0x00,
0x00,
  0x00, 0x00, 0x00, 0x1f, 0xff, 0xff, 0x80, 0x00, 0x00, 0x00, 0xff, 0xff, 0xfc, 0x00, 0x00,
0x00,
  0x00, 0x00, 0x00, 0x3f, 0xff, 0xff, 0xc0, 0x00, 0x00, 0x01, 0xff, 0xff, 0xfe, 0x00, 0x00,
0x00,
  0x00, 0x00, 0x00, 0x7f, 0xff, 0xff, 0xe0, 0x00, 0x00, 0x03, 0xff, 0xff, 0xff, 0x00, 0x00,
0x00,
  0x00, 0x00, 0x00, 0xff, 0xff, 0xff, 0xf0, 0x00, 0x00, 0x07, 0xff, 0xff, 0xff, 0x80, 0x00,
0x00,
  0x00, 0x00, 0x00, 0xff, 0xff, 0xff, 0xf0, 0x00, 0x00, 0x07, 0xff, 0xff, 0xff, 0x80, 0x00,
0x00,
  0x00, 0x00, 0x01, 0xff, 0xff, 0xff, 0xf8, 0x00, 0x00, 0x0f, 0xff, 0xff, 0xff, 0xc0, 0x00,
0x00,
  0x00, 0x00, 0x01, 0xff, 0xff, 0xff, 0xf8, 0x00, 0x00, 0x0f, 0xff, 0xff, 0xff, 0xc0, 0x00,
0x00,
  0x00, 0x00, 0x03, 0xff, 0xff, 0xff, 0xfc, 0x00, 0x00, 0x1f, 0xff, 0xff, 0xff, 0xe0, 0x00,
0x00,
  0x00, 0x00, 0x03, 0xff, 0xff, 0xff, 0xfc, 0x00, 0x00, 0x1f, 0xff, 0xff, 0xff, 0xe0, 0x00,
0x00,
  0x00, 0x00, 0x07, 0xff, 0xff, 0xff, 0xfe, 0x00, 0x00, 0x3f, 0xff, 0xff, 0xff, 0xf0, 0x00,
0x00,
  0x00, 0x00, 0x07, 0xff, 0xff, 0xff, 0xfe, 0x00, 0x00, 0x3f, 0xff, 0xff, 0xff, 0xf0, 0x00,
0x00,
  0x00, 0x00, 0x07, 0xff, 0xff, 0xff, 0xfe, 0x00, 0x00, 0x3f, 0xff, 0xff, 0xff, 0xf0, 0x00,
0x00,
  0x00, 0x00, 0x0f, 0xff, 0xff, 0xff, 0xff, 0x00, 0x00, 0x7f, 0xff, 0xff, 0xff, 0xf8, 0x00,
0x00,
  0x00, 0x00, 0x0f, 0xff, 0xff, 0xff, 0xff, 0x00, 0x00, 0x7f, 0xff, 0xff, 0xff, 0xf8, 0x00,
0x00,
  0x00, 0x00, 0x0f, 0xff, 0xff, 0xff, 0xff, 0x00, 0x00, 0x7f, 0xff, 0xff, 0xff, 0xf8, 0x00,
0x00,
  0x00, 0x00, 0x0f, 0xff, 0xff, 0xff, 0xff, 0x00, 0x00, 0x7f, 0xff, 0xff, 0xff, 0xf8, 0x00,
0x00,
  0x00, 0x00, 0x0f, 0xff, 0xff, 0xff, 0xff, 0x00, 0x00, 0x7f, 0xff, 0xff, 0xff, 0xf8, 0x00,
0x00,
  0x00, 0x00, 0x0f, 0xff, 0xff, 0x8f, 0xff, 0x00, 0x00, 0x7f, 0xf1, 0xff, 0xff, 0xf8, 0x00,
0x00,
  0x00, 0x00, 0x0f, 0xff, 0xff, 0x07, 0xff, 0x00, 0x00, 0x7f, 0xe0, 0xff, 0xff, 0xf8, 0x00,
0x00,
  0x00, 0x00, 0x0f, 0xff, 0xff, 0x07, 0xff, 0x00, 0x00, 0x7f, 0xe0, 0xff, 0xff, 0xf8, 0x00,
0x00,
  0x00, 0x00, 0x0f, 0xff, 0xfe, 0x03, 0xff, 0x00, 0x00, 0x7f, 0xc0, 0x7f, 0xff, 0xf8, 0x00,
0x00,
```

```
  0x00, 0x00, 0x0f, 0xff, 0xfe, 0x03, 0xff, 0x00, 0x00, 0x7f, 0xc0, 0x7f, 0xff, 0xf8, 0x00, 0x00,
  0x00, 0x00, 0x0f, 0xff, 0xfe, 0x03, 0xff, 0x00, 0x00, 0x7f, 0xc0, 0x7f, 0xff, 0xf8, 0x00, 0x00,
  0x00, 0x00, 0x0f, 0xff, 0xfe, 0x03, 0xff, 0x00, 0x00, 0x7f, 0xc0, 0x7f, 0xff, 0xf8, 0x00, 0x00,
  0x00, 0x00, 0x0f, 0xff, 0xfe, 0x03, 0xff, 0x00, 0x00, 0x7f, 0xc0, 0x7f, 0xff, 0xf8, 0x00, 0x00,
  0x00, 0x00, 0x0f, 0xff, 0xff, 0x07, 0xff, 0x00, 0x00, 0x7f, 0xe0, 0xff, 0xff, 0xf8, 0x00, 0x00,
  0x00, 0x00, 0x0f, 0xff, 0xff, 0x07, 0xff, 0x00, 0x00, 0x7f, 0xe0, 0xff, 0xff, 0xf8, 0x00, 0x00,
  0x00, 0x00, 0x0f, 0xff, 0xff, 0x8f, 0xff, 0x00, 0x00, 0x7f, 0xf1, 0xff, 0xff, 0xf8, 0x00, 0x00,
  0x00, 0x00, 0x0f, 0xff, 0xff, 0xff, 0xff, 0x00, 0x00, 0x7f, 0xff, 0xff, 0xff, 0xf8, 0x00, 0x00,
  0x00, 0x00, 0x07, 0xff, 0xff, 0xff, 0xfe, 0x00, 0x00, 0x3f, 0xff, 0xff, 0xff, 0xf0, 0x00, 0x00,
  0x00, 0x00, 0x07, 0xff, 0xff, 0xff, 0xfe, 0x00, 0x00, 0x3f, 0xff, 0xff, 0xff, 0xf0, 0x00, 0x00,
  0x00, 0x00, 0x07, 0xff, 0xff, 0xff, 0xfe, 0x00, 0x00, 0x3f, 0xff, 0xff, 0xff, 0xf0, 0x00, 0x00,
  0x00, 0x00, 0x03, 0xff, 0xff, 0xff, 0xfc, 0x00, 0x00, 0x1f, 0xff, 0xff, 0xff, 0xe0, 0x00, 0x00,
  0x00, 0x00, 0x03, 0xff, 0xff, 0xff, 0xfc, 0x00, 0x00, 0x1f, 0xff, 0xff, 0xff, 0xe0, 0x00, 0x00,
  0x00, 0x00, 0x01, 0xff, 0xff, 0xff, 0xf8, 0x00, 0x00, 0x0f, 0xff, 0xff, 0xff, 0xc0, 0x00, 0x00,
  0x00, 0x00, 0x01, 0xff, 0xff, 0xff, 0xf8, 0x00, 0x00, 0x0f, 0xff, 0xff, 0xff, 0xc0, 0x00, 0x00,
  0x00, 0x00, 0x00, 0xff, 0xff, 0xff, 0xf0, 0x00, 0x00, 0x07, 0xff, 0xff, 0xff, 0x80, 0x00, 0x00,
  0x00, 0x00, 0x00, 0xff, 0xff, 0xff, 0xf0, 0x00, 0x00, 0x07, 0xff, 0xff, 0xff, 0x80, 0x00, 0x00,
  0x00, 0x00, 0x00, 0x7f, 0xff, 0xff, 0xe0, 0x00, 0x00, 0x03, 0xff, 0xff, 0xff, 0x00, 0x00, 0x00,
  0x00, 0x00, 0x00, 0x3f, 0xff, 0xff, 0xc0, 0x00, 0x00, 0x01, 0xff, 0xff, 0xfe, 0x00, 0x00, 0x00,
  0x00, 0x00, 0x00, 0x1f, 0xff, 0xff, 0x80, 0x00, 0x00, 0x00, 0xff, 0xff, 0xfc, 0x00, 0x00, 0x00,
  0x00, 0x00, 0x00, 0x0f, 0xff, 0xff, 0x00, 0x00, 0x00, 0x00, 0x7f, 0xff, 0xf8, 0x00, 0x00, 0x00,
  0x00, 0x00, 0x00, 0x07, 0xff, 0xfe, 0x00, 0x00, 0x00, 0x00, 0x3f, 0xff, 0xf0, 0x00, 0x00, 0x00,
  0x00, 0x00, 0x00, 0x03, 0xff, 0xfc, 0x00, 0x00, 0x00, 0x00, 0x1f, 0xff, 0xe0, 0x00, 0x00, 0x00,
  0x00, 0x00, 0x00, 0x00, 0xff, 0xf0, 0x00, 0x00, 0x00, 0x00, 0x07, 0xff, 0x80, 0x00, 0x00, 0x00,
```

```
  0x00, 0x00, 0x00, 0x00, 0x1f, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0xfc, 0x00, 0x00, 0x00, 
0x00, 
  0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 
0x00, 
  0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 
0x00, 
  0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 
0x00, 
  0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 
0x00, 
  0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 
0x00, 
  0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 
0x00, 

};

void setup() {
  pinMode(ledPin, OUTPUT);
  digitalWrite(ledPin, LOW);

  Serial.begin(9600);

  if (!display.begin(SSD1306_SWITCHCAPVCC, 0x3C)) {
    Serial.println(F("SSD1306 allocation failed"));
    for (;;);
  }

  display.clearDisplay();
}

void loop() {
  if (Serial.available() > 0) {
    char cmd = Serial.read();  // 读取 'Y' 或 'N'

    if (cmd == 'N') {
      digitalWrite(ledPin, HIGH);  // 检测失败 → 点亮 LED
      display.clearDisplay();
      display.drawBitmap(0, 0, image_data_DISTRESSED_EYESarray, SCREEN_WIDTH, SCREEN_HEIGHT, 
1);
      display.display();
    } else if (cmd == 'Y') {
      digitalWrite(ledPin, LOW);   // 检测成功 → 熄灭 LED
      display.clearDisplay();
      display.drawBitmap(0, 0, image_data_EYES_FRONTarray, SCREEN_WIDTH, SCREEN_HEIGHT, 1);
      display.display();
    }
```

```
    Serial.write(cmd);
    Serial.write('\n');
  }
}
```

- **Motor Controlling Code**

```
#include <Wire.h>
#include <AFMotor.h>

AF_DCMotor motor1(1);
AF_DCMotor motor2(2);
AF_DCMotor motor3(3);
AF_DCMotor motor4(4);

char command = 'S';
char currentCommand = 'S';
unsigned long lastCommandTime = 0;
const unsigned long timeout = 500;  // 命令有效期

void setup() {
  Serial.begin(115200);
  Wire.begin(8);
  Wire.onReceive(receiveEvent);

  motor1.setSpeed(200);
  motor2.setSpeed(200);
  motor3.setSpeed(200);
  motor4.setSpeed(200);
}

void loop() {
  unsigned long now = millis();

  // 若超时未更新命令 → 自动停止
  if ((now - lastCommandTime > timeout) && currentCommand != 'S') {
    currentCommand = 'S';
    Serial.println("🛑 超时无新命令，执行停止");
    executeCommand(currentCommand);
  }

  // 若 command 与 currentCommand 不一致（说明刚收到或刚切换）→ 执行新命令
  if (command != currentCommand) {
    currentCommand = command;
    executeCommand(currentCommand);
    Serial.print("⚙️ 执行命令: ");
    Serial.println(currentCommand);
  }
```

```
  delay(20);  // 可选：减少处理延迟
}

void receiveEvent(int howMany) {
  if (Wire.available()) {
    command = Wire.read();
    lastCommandTime = millis();  // 每次接收都刷新时间
    Serial.print("📩 收到命令: ");
    Serial.println(command);
  }
}

void executeCommand(char cmd) {
  int fast = 200;
  int slow = 100;

  switch (cmd) {
    case 'F':
      setAllMotors(fast, FORWARD);
      break;
    case 'B':
      setAllMotors(fast, BACKWARD);
      break;
    case 'L':
      motor1.setSpeed(fast); motor1.run(FORWARD);
      motor2.setSpeed(slow); motor2.run(FORWARD);
      motor3.setSpeed(slow); motor3.run(FORWARD);
      motor4.setSpeed(fast); motor4.run(FORWARD);
      break;
    case 'R':
      motor1.setSpeed(slow); motor1.run(FORWARD);
      motor2.setSpeed(fast); motor2.run(FORWARD);
      motor3.setSpeed(fast); motor3.run(FORWARD);
      motor4.setSpeed(slow); motor4.run(FORWARD);
      break;
    default:
      stopAllMotors();
      break;
  }
}

void setAllMotors(int speed, int dir) {
  motor1.setSpeed(speed); motor1.run(dir);
  motor2.setSpeed(speed); motor2.run(dir);
  motor3.setSpeed(speed); motor3.run(dir);
  motor4.setSpeed(speed); motor4.run(dir);
}
```

```
void stopAllMotors() {
  motor1.run(RELEASE);
  motor2.run(RELEASE);
  motor3.run(RELEASE);
  motor4.run(RELEASE);
}
```

- **Gesture Controlling Code**

```
#include <Wire.h>
#include <SPI.h>
#include <nRF24L01.h>
#include <RF24.h>
#include <Adafruit_MPU6050.h>
#include <Adafruit_Sensor.h>
#include <math.h>

Adafruit_MPU6050 mpu;
RF24 radio(9, 10); // CE, CSN
const byte address[6] = "NODE1";

const int switchPin = 2;
bool controlEnabled = false;
bool lastSwitchState = LOW;
bool stopSignalSent = false;

char lastCommand = 'S';
unsigned long lastSendTime = 0;
unsigned long lastActionTime = 0;
const unsigned long sendInterval = 100;
const unsigned long resendInterval = 300;
const unsigned long holdDuration = 600;

const int bufferSize = 5;
char commandBuffer[bufferSize];
int bufferIndex = 0;

void setup() {
  Serial.begin(115200);
  pinMode(switchPin, INPUT); // 外接下拉，读取高电平启动

  Wire.begin();
  if (!mpu.begin()) {
    Serial.println("❌ MPU6050 初始化失败");
    while (1);
  }
```

```
  mpu.setAccelerometerRange(MPU6050_RANGE_8_G);
  mpu.setFilterBandwidth(MPU6050_BAND_21_HZ);

  if (!radio.begin()) {
    Serial.println("❌ NRF24 初始化失败");
    while (1);
  }

  radio.openWritingPipe(address);
  radio.setPALevel(RF24_PA_LOW);
  radio.stopListening();

  if (digitalRead(switchPin) == HIGH) {
    controlEnabled = true;
    Serial.println("⚡ 通电状态 → 程序已启用");
  } else {
    controlEnabled = false;
    Serial.println("⚡ 通电状态 → 程序已禁用");
  }

  stopSignalSent = false;
}

void loop() {
  bool switchState = digitalRead(switchPin);

  if (lastSwitchState == LOW && switchState == HIGH) {
    controlEnabled = !controlEnabled;
    Serial.print(controlEnabled ? "🟢 控制启用" : "🔴 控制禁用");
    Serial.println();
    stopSignalSent = false;
    delay(200);
  }
  lastSwitchState = switchState;

  if (!controlEnabled) {
    if (!stopSignalSent && lastCommand != 'S') {
      char stopCmd = 'S';
      radio.write(&stopCmd, sizeof(stopCmd));
      Serial.println("⛔ 程序禁用 → 发送停止");
      lastCommand = 'S';
      stopSignalSent = true;
    }
    return;
  }

  if (millis() - lastSendTime >= sendInterval) {
    sensors_event_t a, g, temp;
```

```
    mpu.getEvent(&a, &g, &temp);

    // 使用真实姿态角计算（角度制）
    float pitch = atan2(a.acceleration.y, sqrt(a.acceleration.x * a.acceleration.x + a.acceleration.z * a.acceleration.z)) * 180.0 / PI;
    float roll  = atan2(-a.acceleration.x, a.acceleration.z) * 180.0 / PI;

    char detected = detectCommand(pitch, roll);
    char cmd = getStableCommand(detected);
    bool isAction = (cmd == 'F' || cmd == 'B' || cmd == 'L' || cmd == 'R');

    if (isAction) {
      if (cmd != lastCommand || millis() - lastSendTime >= resendInterval) {
        radio.write(&cmd, sizeof(cmd));
        Serial.print("🔁 指令发送: ");
        Serial.print(cmd);
        Serial.print(" | Pitch=");
        Serial.print(pitch);
        Serial.print("°, Roll=");
        Serial.print(roll);
        Serial.println("°");
        lastCommand = cmd;
        lastSendTime = millis();
      }
      lastActionTime = millis();
    } else {
      if (millis() - lastActionTime > holdDuration && lastCommand != 'S') {
        radio.write(&cmd, sizeof(cmd));
        Serial.println("🛑 姿态回中 → 停止");
        lastCommand = 'S';
        lastSendTime = millis();
      }
    }
  }
}

char detectCommand(float pitch, float roll) {
  if (pitch > 20 && abs(roll) < 15) return 'F';
  if (pitch < -20 && abs(roll) < 15) return 'B';
  if (roll > 20 && abs(pitch) < 15) return 'R';
  if (roll < -20 && abs(pitch) < 15) return 'L';
  return 'S';
}

char getStableCommand(char newCmd) {
  commandBuffer[bufferIndex] = newCmd;
  bufferIndex = (bufferIndex + 1) % bufferSize;
```

```
  int countF = 0, countB = 0, countL = 0, countR = 0;
  for (int i = 0; i < bufferSize; i++) {
    switch (commandBuffer[i]) {
      case 'F': countF++; break;
      case 'B': countB++; break;
      case 'L': countL++; break;
      case 'R': countR++; break;
    }
  }

  if (countF >= 3) return 'F';
  if (countB >= 3) return 'B';
  if (countL >= 3) return 'L';
  if (countR >= 3) return 'R';
  return 'S';
}
```

- **RF Communication Code**

```
#include <Wire.h>
#include <SPI.h>
#include <nRF24L01.h>
#include <RF24.h>

RF24 radio(9, 10); // CE, CSN
const byte address[6] = "NODE1";

// 超声波模块接线
const int TRIG_PIN = 4;
const int ECHO_PIN = 5;
const int obstacleThreshold = 10; // 障碍距离阈值（cm）

char lastCommand = 'S';
char filteredCommand = 'S';

unsigned long lastUltrasonicCheck = 0;
const unsigned long ultrasonicInterval = 100;

void setup() {
  Serial.begin(115200);
  Wire.begin();
  pinMode(TRIG_PIN, OUTPUT);
  pinMode(ECHO_PIN, INPUT);

  if (!radio.begin()) {
    Serial.println("❌ NRF 初始化失败");
    while (1);
  }
```

```
  radio.openReadingPipe(1, address);
  radio.setPALevel(RF24_PA_LOW);
  radio.startListening();

  Serial.println("✅ UNO 接收器启动，支持避障逻辑");
}

void loop() {
  // 读取无线指令
  if (radio.available()) {
    char cmd;
    radio.read(&cmd, sizeof(cmd));
    lastCommand = cmd;
    Serial.print("📡 接收命令: ");
    Serial.println(cmd);
  }
  // 周期性测距避障
 if (millis() - lastUltrasonicCheck >= ultrasonicInterval) {
   lastUltrasonicCheck = millis();
   long distance = getDistanceCM();

    Serial.print("📏 距离: ");
    Serial.print(distance);
    Serial.println(" cm");

    // 对向前移动相关指令启用避障
    if ((lastCommand == 'F' || lastCommand == 'L' || lastCommand == 'R') && distance <
obstacleThreshold) {
      filteredCommand = 'S';
      Serial.println("🛑 避障触发 → 停止小车");
    } else {
    filteredCommand = lastCommand;
    }

    // 发送最终控制命令到下位机
   Wire.beginTransmission(8);
    Wire.write(filteredCommand);
    Wire.endTransmission();
  //}
}

// 获取距离（单位: cm）
long getDistanceCM() {
  digitalWrite(TRIG_PIN, LOW);
  delayMicroseconds(2);
  digitalWrite(TRIG_PIN, HIGH);
  delayMicroseconds(10);
```

```
  digitalWrite(TRIG_PIN, LOW);

  long duration = pulseIn(ECHO_PIN, HIGH, 30000); // 最长 30ms = 约 5 米
  long distance = duration * 0.034 / 2;
  return distance;
}
```

Appendix H: Additional Photos or Videos QR Codes

Fig. 20. QR Code For Video Demonstration

Appendix I: Necessary Theoretical Support & Knowledge

Appendix 1 — Environmental Impact Analysis of 9V Alkaline Battery

- **1.1 Life Cycle Assessment and Sustainability Concern**

The 9V alkaline battery, used in the prototype for powering both the Sensor Recognition Layer (Arduino Nano) and Control Logic Layer (Upper UNO), was identified as the component with the greatest environmental burden. Figure 21 below illustrates the full "Cradle to Grave" life cycle of the battery, highlighting environmental risks at each phase: raw material extraction, manufacturing, transportation, usage, and disposal.

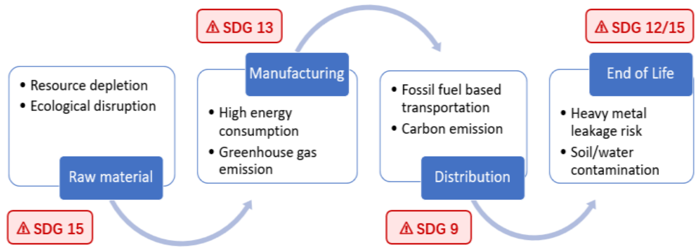


Fig. 21. Cradle-to-Grave life cycle of a 9V alkaline battery.

- **1.2 Stage-Based Environmental Impact Analysis**

1.2.1 Raw Material Stage (SDG 15)

The battery's core components—zinc, manganese dioxide, carbon, and potassium hydroxide—are heavily dependent on mining activities in developing regions. These operations have been associated with soil acidification, surface water contamination, and ecological disturbance, directly contradicting SDG 15 goals on ecosystem protection.

1.2.2 Manufacturing Stage (SDG 13)

The production of 9V batteries is energy-intensive. A lifecycle study published in the *Journal of Cleaner Production* estimates **1.1 kg of CO_2** emitted per battery during manufacturing—significantly exceeding emissions from equivalent rechargeable batteries.

1.2.3 Distribution Stage (SDG 9)

Due to low energy density and widespread shipping, transportation of batteries leads to increased carbon footprints. European logistics data suggests this stage can account for over **10% of total battery-related emissions**, especially during long-range land and air transport.

1.2.4 End-of-Life Stage (SDG 12 & 15)

Improper disposal contributes to heavy metal leaching into soil and water systems. Monitoring studies in urban landfills in southern China have shown elevated concentrations of manganese and zinc ions traced to discarded alkaline batteries.

- **1.3 Recommendations for Sustainable Improvement**

To reduce long-term environmental impact and better align with SDGs:

- Replace 9V batteries with rechargeable lithium-ion cells or supercapacitors.
- Design swappable battery compartments to enable recycling and replacement.
- Introduce disposal guidance and battery return bins to encourage proper classification and collection.

Appendix 2 — Technical Expansion on Vision Recognition and Communication Architecture

- **2.1 SLAM and RFID for Advanced Target Recognition**

For future versions of the system, integrating **Simultaneous Localization and Mapping (SLAM)** can enable autonomous navigation in complex environments such as retail stores. This would allow the cart to map and localize without relying solely on hand gestures or pre-tagged markers. Additionally, replacing ArUco visual tags with **RFID passive identifiers** offers a more user-friendly, contactless, and hygiene-conscious way to detect users without explicit labeling.

- **2.2 Communication Security and Multi-Agent Scalability**

The current implementation using NRF24L01 modules is adequate for single-cart systems but lacks robustness in dense, multi-user environments. Future upgrades could consider:

- **Encrypted communication protocols** such as AES to enhance data security.
- Adoption of **LoRa (Long Range)** modules to increase range and resistance to interference.
- Development of **multi-cart task scheduling and channel coordination mechanisms** to prevent signal collisions and ensure scalable deployment.

These enhancements will contribute to the system's long-term adaptability, security, and commercial viability in real-world applications.

Appendix 3 — Mathematical Basis of MPU6050

To interpret the human hand posture from sensor data, we adopted a simplified angle-based method using the MPU6050 gyroscope. The origin is defined at the center of the user's wrist. The raw acceleration readings along the x, y, and z axes are transformed into the following two angles:

- **Pitch angle** (vertical tilt):

$$pitch = \frac{\arctan\left(\frac{a_y}{\sqrt{a_x^2 + a_z^2}}\right) * 180^{\circ}}{\pi}$$

- **Roll angle** (horizontal tilt):

$$roll = \frac{\arctan\left(\frac{-a_x}{a_z}\right) * 180^{\circ}}{\pi}$$

These formulas provide a computationally efficient approximation of wrist posture in 2D space and allow for robust gesture classification without complex sensor fusion algorithms.

Appendix 4 — Detailed Explanation of the Technical Background in Network Debugging of the Control Logic Layer

In the system development, we constructed a remote debugging architecture based on "Raspberry Pi + camera + VNC virtual desktop + Arduino linkage feedback". Due to the lack of wired network interfaces in the laboratory, we chose to configure the wireless connection of Raspberry Pi through the mobile phone hotspot. This approach is relatively convenient during the initial burning and deployment stage of the system, but it has obvious stability bottlenecks in a dense wireless environment.

- **1. Congestion problems in high-density wireless environments**

Take Classroom IR112 in the teaching building as an example. A large number of students carry devices such as mobile phones, tablets and laptops, and connect to Wi-Fi at the same time, resulting in more than 400 terminals competing for the limited wireless channel resources in the physical space. According to the IEEE 802.11 standard, the 2.4GHz frequency band is divided into a total of 13 channels, among which only channels 1, 6 and 11 are non-interfering primary channels. When multiple hotspots are concentrated in these three channels, the signal spectra among different devices overlap, causing problems such as bandwidth contention, data collision, and increased transmission failure rate (see **Fig. 22.**).

.

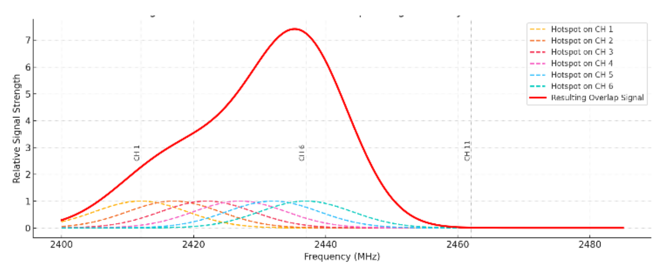


Fig. 22. 2.4GHz Wi-Fi Channel Overlap in High Density Environment

---

- **2. Connection dependencies of VNC services**

The VNC remote desktop mechanism of Raspberry Pi requires the developer's laptop to be in the same local area network hotspot environment as it. Instability of either party's network will lead to connection interruption, causing serious impacts such as debugging interruption and image refresh delay (see **Fig. 23.**).

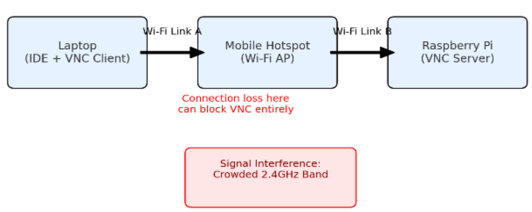


Fig. 23. VNC Debugging Path Dependence on Network Security

- **3. Frequency-domain manifestation of channel interference**

Wi-Fi signals are essentially modulated radio frequency waves with a fixed central frequency, and their spectra can be analyzed through Fourier transform. In a clean channel, its spectrum has obvious main peaks and clear boundaries. However, in a congested environment, the superposition of signals from multiple devices causes the broadening of the main peak of the spectrum, blurred edges, and even the phenomenon of "subcarrier crosstalk" (see **Fig. 24.**), which seriously affects the identification and stable transmission of signals.

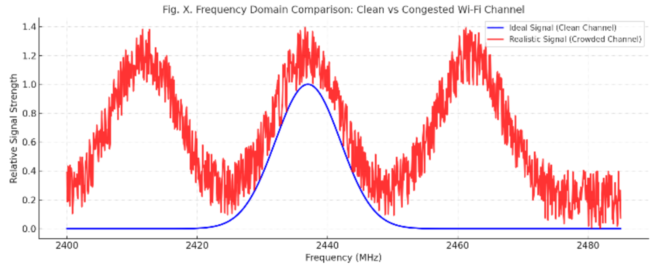


Fig. 24. Frequency Domain Comparison: Clean vs Congested Wi-Fi Channel

---

- **4. Simulation analysis of debugging performance**

We further simulated and tested the stability of VNC connections under different numbers of concurrent connections. The results show that when the number of connected terminals increases from 50 to 450, the connection failure rate rises rapidly and the difficulty of system debugging increases significantly (see **Fig. 25.**). This verifies the practical impact of the high-density wireless environment on the debugging process of the development system.

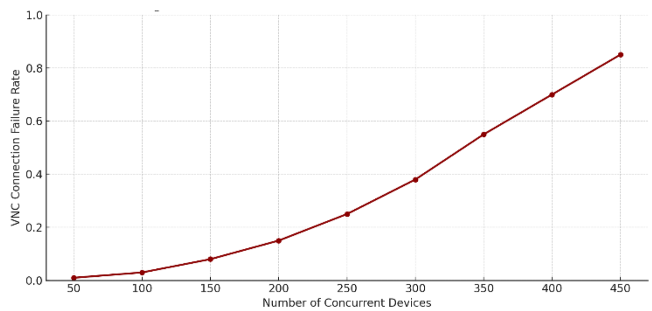


Fig. 25. Simulated VNC Failure Rate with Number of Concurrent Devices

---

Appendix 5 — Detailed Explanation of the Technical Background of Project Reflection

- **1. Analysis of the Sustainability and Wearing Burden of the Power Supply System**

During the prototype stage of this project, the Sensor Recognition Layer and the Control Logic Layer were respectively powered by 9V alkaline batteries. Although this scheme can meet the initial voltage requirements, the following problems have been exposed in the actual usage process:

➢ Heavy environmental burden:
9V batteries are disposable alkaline batteries, and their production, transportation and disposal processes are accompanied by significant carbon emissions and solid waste risks, which violate the sustainable goals in SDG 12/13/15.

➢ Poor wearing experience:
The Sensor Recognition Layer is installed on the user's hand. The 9V battery is large in size and heavy in weight, causing users to feel uncomfortable during long-term wearing and affecting the interaction experience.

- **2. Issues regarding the load capacity and ramp adaptability of the trolley structure**

The current prototype vehicle is mainly used for functional verification and is relatively small in size. There are two deficiencies in its practical application in the retail environment:

- No-load monitoring mechanism:
  It cannot determine the overload status, which may lead to insufficient motor power when the load of the goods is large, causing system vibration or loss of control.
- Poor adaptability to slopes:
  Without a stabilizer and brake control system, it is unable to handle scenarios such as supermarket elevators and slopes. When the slope θ increases, if the gravitational component force $mgsin(\theta)$ along the slope exceeds the maximum static friction force $\mu mgcos(\theta)$, out-of-control sliding will occur. In addition, the structure at the elevator steps (comb plate) is prone to jamming.

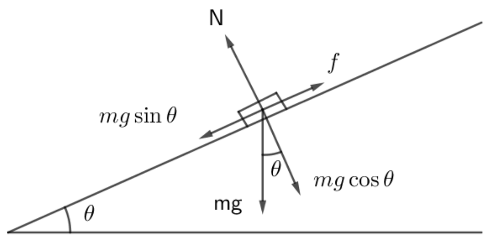


Fig. 26. Force Analysis of Cart on a Downward Inclined Conveyor

---

- **3. Acceptance and Ethical Issues of camera-assisted Recognition**

In the current design, we adopt the bottom camera to recognize the ArUco tag on the user's leg as the following reference point. Although this method has high recognition accuracy, the following problems exist:
Significant individual difference: Different users have different leg shapes, and the label pasting positions are not fixed, which affects the stability of the system.

- Users' hygiene concerns are obvious:
  Users have hygiene resistance to personal labels, especially in the context of the epidemic or among user groups with high cleanliness requirements.
- Privacy Controversy and Ethical Risk:
  The camera's field of view is focused on the legs, which some users may misunderstand as "secret photography", especially causing psychological burden to female users. If the system expands its image storage or remote transmission functions in the future, there is a risk of data abuse or leakage.

- **4. Feasibility of Introducing Autonomous Navigation Algorithm (SLAM)**

SLAM (Simultaneous Localization and Mapping) technology can help small cars achieve in indoor scenes without GPS. Visual SLAM is currently a relatively easy-to-deploy solution on embedded environments (such as Raspberry Pi), especially when combined with the camera module (already available in this system) and a lightweight IMU, it can achieve initial environment mapping and navigation without increasing additional hardware costs.

- Application potential after introduction:
  Fully autonomous obstacle avoidance shopping cart: No longer relying on user location guidance, but actively cruising and following within the supermarket.
- Shelf identification and navigation:
  Combined with QR codes or markers, for area positioning and precise parking.
- Multi-vehicle path planning:
  Laying the spatial coordinate foundation for the future "collaborative scheduling of multi-vehicle systems".
- Environmental memory and map reuse: Supports map loading and optimization in repetitive scene navigation (such as similar structures on each floor of a supermarket).

- **5. Comparison and Encryption Suggestions for Communication Modules**

Given the high terminal density observed in classroom IR112, it is crucial to compare wireless modules in terms of their robustness and scalability. Table 11 provides a summary of three candidate modules and their suitability for various deployment needs.

Table 11. Comparison of Future Extended Communication Methods

Module Type	**Range**	**Power Consumption**	**Interference Resistance**	**Encryption Capability**	**Application Scenario**
NRF24L01	Medium	Low	Moderate	Weak	Single-vehicle debugging
LoRa	Long	Low	Strong	Supports AES	Multi-vehicle scenarios
Wi-Fi	Medium	High	Poor	Extensible	Image transmission