ECE445

SENIOR DESIGN LABORATORY

Final Report

AR-Based Palm-Sized Robot Assistant 1

Team #25
Ruixi Qin [ruixiq2]
Fengwei Yang [fengwei3]
Yuzhang Wang [yw91]
Jiaqi Ding [jiaqi23]

Advisor: Liangjing Yang

May 15, 2026

Content

1. Introduction 3

    1.1 Problem Statement and Solution 3

    1.2 Functionality 3

    1.3 Subsystem Overview 4

2. Design 4

    2.1 Block Diagram 4

    2.2 Control Module 5

        2.2.1 Raspberry Pi 5

        2.2.2 Control Algorithms (Software Logic) 6

        2.2.3 The PCB Schematic and Layout (Hardware Circuit) 7

    2.3 Wheel Drive Module 8

        2.3.1 Actuation Subsystem for Wheel Drive (JGB37-520 Motor + TB6612 Driver) 8

    2.4 Perception Module 9

        2.4.1 The YDLiDAR X3 Pro lidar 9

        2.4.2 TCRT5000 10

    2.5 Interactive Projection Module 11

        2.5.1 The OpenMV H4 Plus Camera Module 11

        2.5.2 The Communication between The Interaction Module and PC 12

    2.6 Enclosure and Mounting Module 12

    2.7 Tolerance Analysis 13

        2.7.1 Motion Control Accuracy Tolerance 13

        2.7.2 Signal Transmission and Latency Tolerance 13

        2.7.3 Power Supply and Signal Tolerance 13

3. Cost 15

4. Schedule 16

5. Conclusion 17

    5.1 Accomplishments 17

    5.2 Uncertainties 18

    5.3 Future Work / Alternatives 19

    5.4 Ethical Considerations 19

1. Introduction

1.1 Problem Statement and Solution

Problem Statement: Traditional presentation tools are typically static, requiring users to remain at a fixed location or manage bulky equipment to share visual information. While modern smartphones offer mobile computing capabilities, the user's experience remains confined to small, two-dimensional screens. This creates a disconnect between digital content and the physical environment, limiting the ability to dynamically share information in collaborative or spontaneous settings.

Solution: This paper presents a palm-sized mobile robotic assistant that bridges digital content with physical space. The robot is controlled by users via an interactive interface. Users can guide the robot towards a target direction, and the robot can be stopped by another user by stretching their hand. After proper positioning, the robot projects an interactive display, enabling users to navigate PPT slides and control the robot’s motion effortlessly via intuitive hand gestures.

1.2 Functionality

Safety and Obstacle Avoidance Module: Employs a hybrid IR sensing strategy to monitor the environment within a 200cm range. The system triggers emergency stops or re-routing sequences upon detecting potential drops, ensuring operational safety.

Projection and Interaction Module: Uses a micro-projector for visual output and a camera-based vision system to track user gestures. This module enables real-time interaction, such as PPT slide transitions and directional control of the robot.

Central Processing Module: Acts as the "brain" of the system, processing sensor data and running computer vision tasks for gesture recognition.

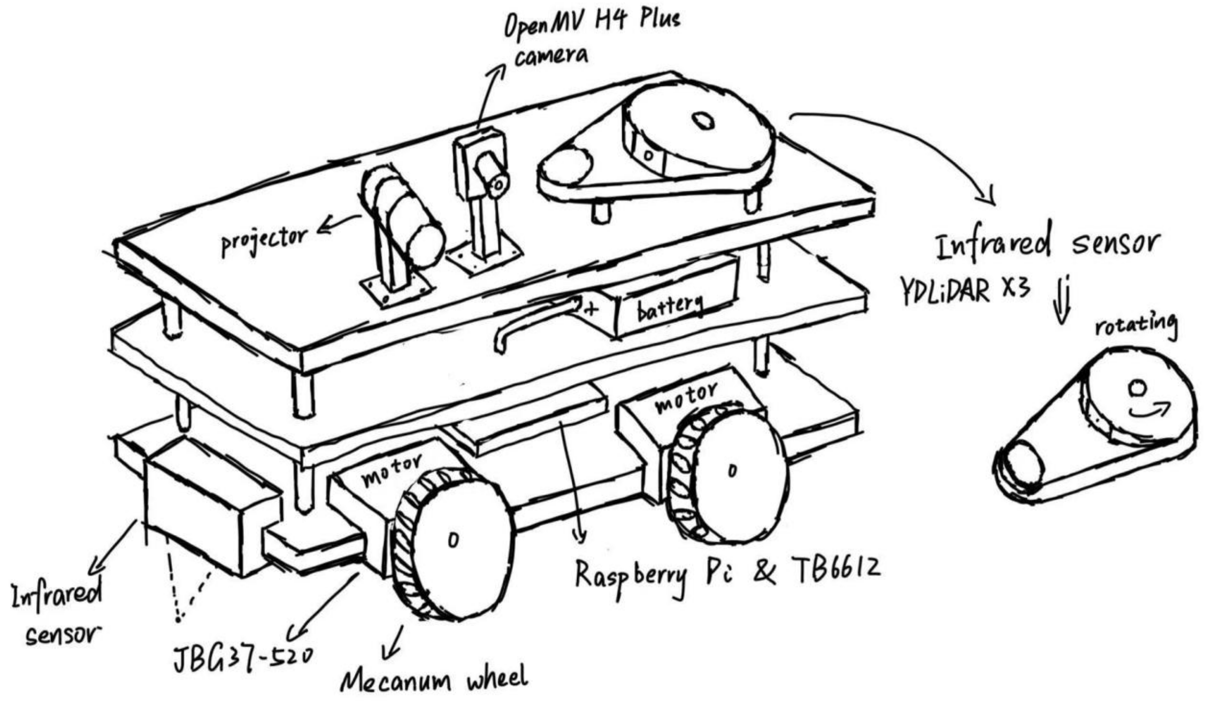

Figure 1: Visual Aid

1.3 Subsystem Overview

- **Perception Subsystem:** To ensure operational safety, the robot employs a hybrid infrared (IR) sensing strategy. A rotating LiDAR sensor provides long-range detection (up to 8m) for interactive proximity halts, allowing users to stop the robot by simply stretching out their hand. Simultaneously, a fixed array of TCRT5000 IR sensors is installed near the wheels to provide real-time edge-drop prevention, triggering an emergency "Kill-Switch" within 500ms if a table edge is detected.
- **Projection and Vision Subsystem:** This module serves as the primary human-machine interface. A micro-projector displays interactive elements (directional arrows and PPT control buttons) onto the desktop. An integrated OpenMV H4 Plus camera continuously monitors the projection area to track a red finger cot worn by the user. This vision system translates touch or hover gestures into control signals, which are then transmitted to the host computer via Bluetooth to execute PPT page transitions or robot movement commands.
- **Central Processing Module:** Powered by a Raspberry Pi 4B, this module acts as the system's "brain." It manages multi-threaded operations, including sensor fusion for safety, real-time computer vision processing for gesture recognition, and the execution of motor control logic. It coordinates the state machine to ensure smooth transitions between manual navigation, alignment, and interactive projection modes.

2. Design

2.1 Block Diagram

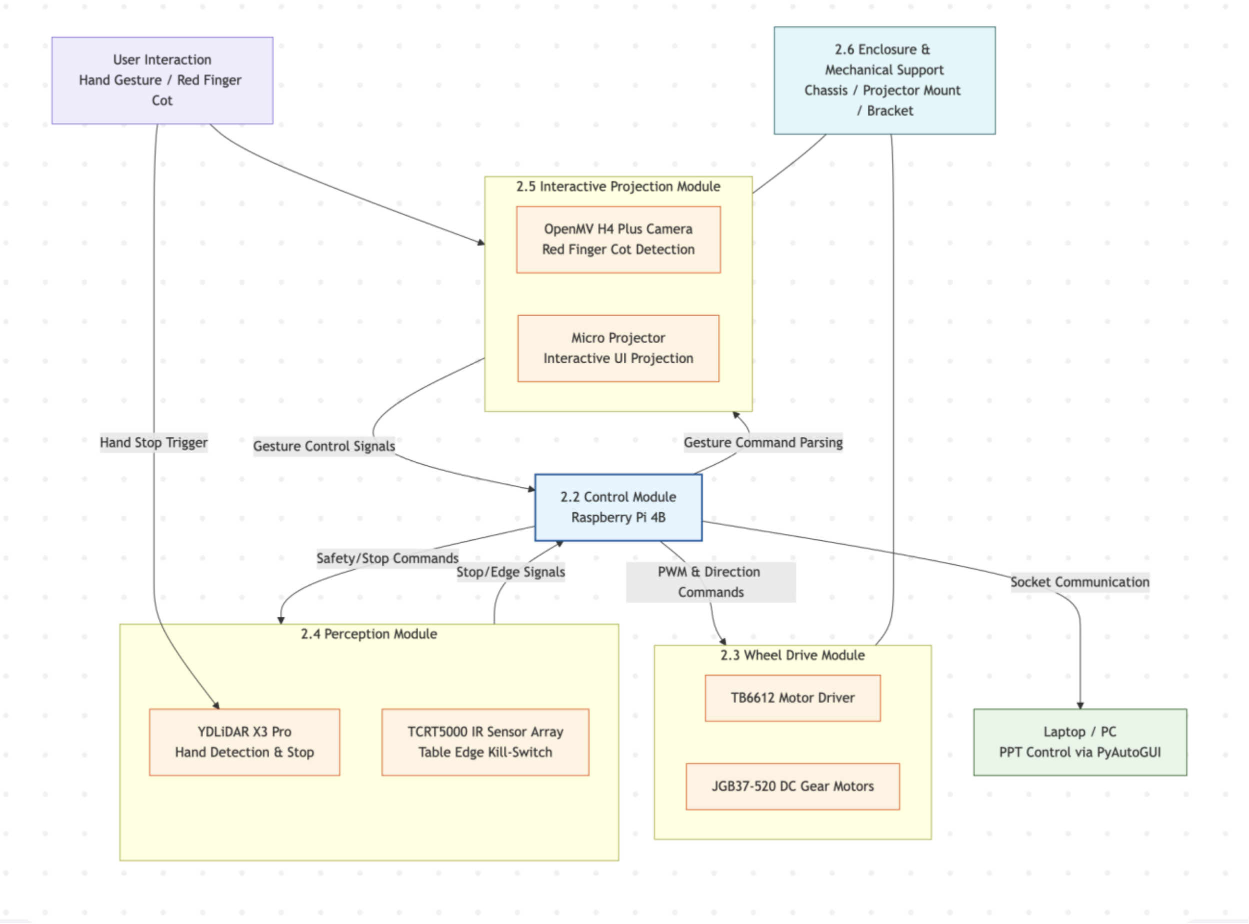

Figure 2: The block diagram of the whole system

2.2 Control Module

2.2.1 Raspberry Pi

We initially considered using Arduino as the microcontroller, but we finally chose the Raspberry Pi because it is available to us through our mentor. In the control module, the Raspberry Pi serves as the central processing hub. It is a small single-board computer that is widely used for robotics applications. As shown in Figure 3, the Raspberry Pi 4B has 4 USB ports and 2GB RAM, thus it can satisfy our needs to cooperate with the OpenMV camera and YDLider well.

The Raspberry Pi has two roles in the whole system. The first is to receive signals from the lidar and IR sensors and control the wheel motors, so that the robot assistant can move on the desk. The second is to provide the interface to the camera and send the signal from the OpenMV module back to our laptop, so that the PowerPoint pages on the computer can be flicked.

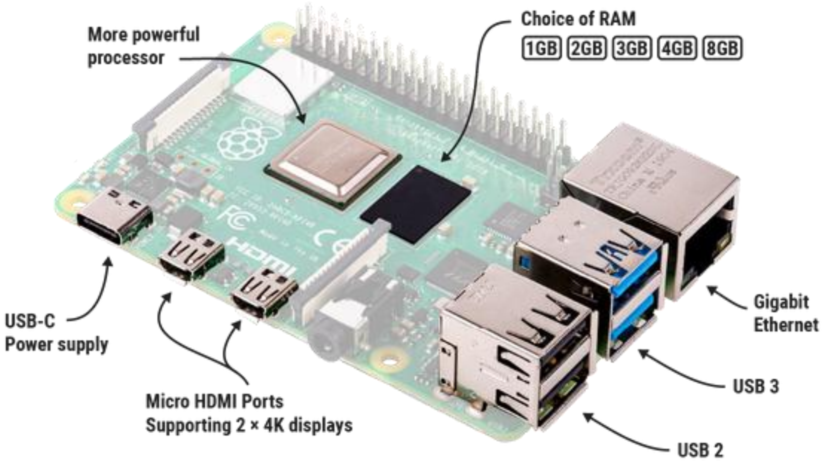

Figure 2: The picture of Raspberry Pi 4B

Requirements	Verification
1. The Raspberry Pi must be connected to the same Wifi as our laptop to achieve communication.	1. To verify this, we can connect both the Raspberry Pi and our laptop to our personal hotspot.

2.2.2 Control Algorithms (Software Logic)

The Control Algorithms represent the high-level decision-making layer of the Control Module. This sub-module is responsible for controlling the robot's movement.

- **Directional Mapping:** The program interprets inputs from the interactive projection to determine the movement direction of the robot. It then calculates the required PWM duty cycles. The output signal is sent to the TB6612 motor drive board by the GPIO pins of the Raspberry Pi.
- **Safety & Table Edge Interruption:** This sub-module is a high-priority monitoring and protection mechanism. We use TCRT5000 IR sensors as fixed infrared detection components, and one is installed near each of the 4 wheels of the robot to detect the table edge in real time. Once any TCRT5000 IR sensor detects the critical position of the table edge, it immediately triggers a global Kill-Switch to the motor drivers to ensure the robot will never fall off the table.
- **Interactive Proximity Halt:** This module enables a physical interaction-based stop mechanism using the LiDAR sensor, allowing the user to intercept the robot manually by stretching out their hand. When the LiDAR detects an object (the user's hand) within the designated interception range, the program triggers an immediate stop. This provides a natural, intuitive way to halt the robot's movement during a presentation.

Requirements	Verification

1. The PWM and direction control signal should be sent to the motor driver board within 1s since the Raspberry Pi receives input signals from the OpenMV module.	1. Record the interval between the Raspberry Pi's receiving input and the wheels' starting to spin. The interval must be <1s.
2. To prevent false triggering, the Raspberry Pi shall only execute the "Halt" command if the lidar detects a hand within the interception range for a continuous duration of at least 500ms.	2. Perform a Pulse Test: Briefly pass an object through the LiDAR zone and verify the car does not stop; then hold the hand steadily to confirm the car stops after the required duration.
3. The TCRT5000 IR sensors shall trigger the motor emergency stop within 500ms when detecting the table edge, with no false triggering.	3. Move the robot slowly towards the table edge, record the time from the TCRT5000 sensor detecting the edge to the motor stopping, which must be ≤500ms; test in non-edge areas to confirm no false triggering.

2.2.3 The PCB Schematic and Layout (Hardware Circuit)

We implement a PCB board to integrate the Raspberry Pi, the TB6612 motor driver board, and the IR sensors together. So that the Raspberry Pi can receive input signals from the IR sensors and send output signals to the TB6612 motor drive board to drive the robot. The sensors can also be powered by the 3.3 V pin of the Raspberry Pi. The connection between them and the PCB layout is shown as follows:

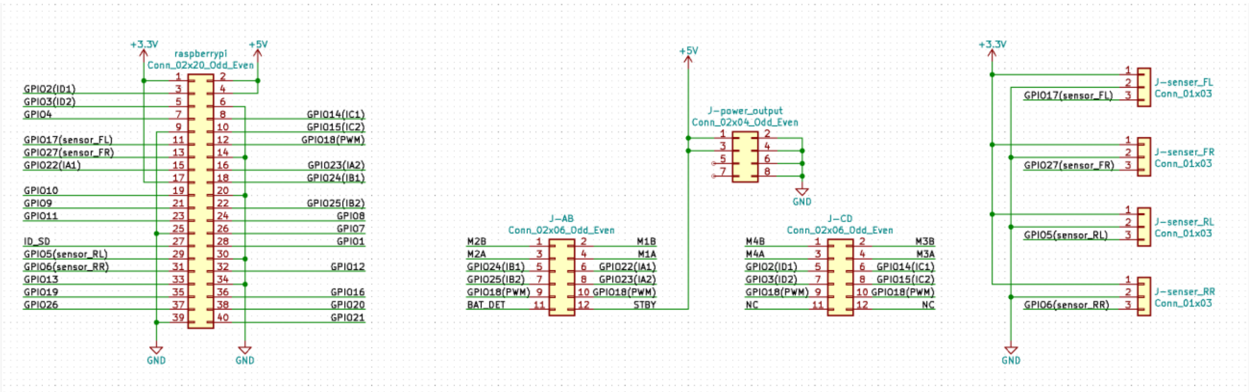

Figure 3: The PCB Schematic

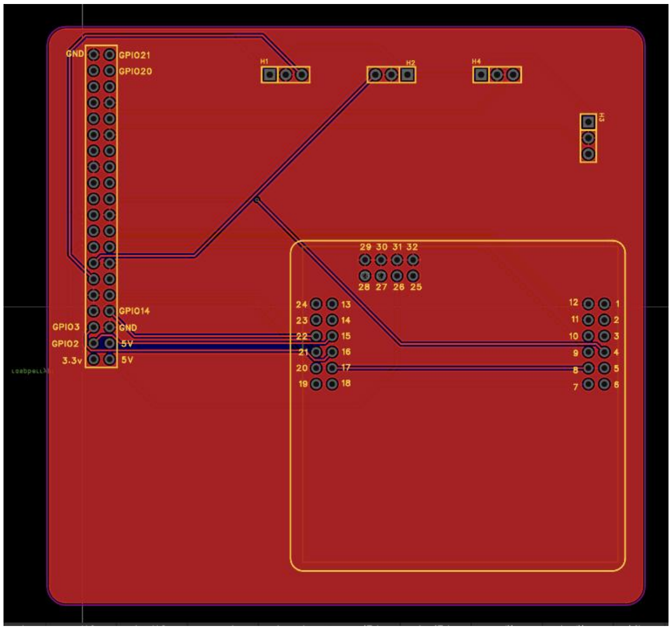

Figure 4: The PCB Layout

2.3 Wheel Drive Module

2.3.1 Actuation Subsystem for Wheel Drive (JGB37-520 Motor + TB6612 Driver)

The wheel drive actuation subsystem consists of four JGB37-520 DC gear motors (12V nominal voltage, with Hall encoder) and a TB6612FNG dual-channel full-bridge motor driver. The JGB37-520 motor is selected for its high torque capability, making it suitable for precise speed control and positioning.

The TB6612FNG driver accepts PWM signals from the Raspberry Pi to control motor speed and direction for both motors independently. It operates at 12V input and provides up to 1.2A continuous current per channel. The TB6612 is more compact and efficient, featuring an integrated MOSFET H-bridge structure that requires no external heatsinks or flyback diodes.

TB6612FNG Specifications:

Parameter	Value
Logic voltage (VSS)	3.6V
Motor voltage (VS)	12V (12V recommended for JGB37-520)
Continuous current per channel	1.2A

Peak current per channel	2A (continuous pulse) / 3.2A (single pulse)
PWM frequency range	Up to 100 kHz

Requirements	Verification
The JGB37-520 motors shall provide sufficient torque to move the projector platform (total mass ≤ 5kg) on flat ground at a speed of ≥ 0.5 m/s.	Assemble a complete platform with a projector; measure the time to travel 5m on a flat surface. Success: travel time ≤ 10 seconds (0.5 m/s).
The TB6612 driver shall supply at least 1.2A continuous current per motor channel without thermal shutdown.	Run both motors at 100% duty cycle under load for 10 minutes; measure driver case temperature with an infrared thermometer. Success: temperature ≤ 70°C with no thermal shutdown.
The TB6612 shall respond to PWM control signals within 10ms.	Send 0% to 100% PWM step command; measure the motor voltage rise time using an oscilloscope. Success: voltage reaches 90% of target within 10ms.
The differential steering system shall achieve a turning radius of ≤ 1m.	Command left motor forward at 100% PWM and right motor reverse at 100% PWM; measure turning circle diameter. Success: turning radius ≤ 1m.

2.4 Perception Module

2.4.1 The YDLiDAR X3 Pro lidar

As shown in Figure 5, the YDLIDAR X3 Pro lidar uses triangular ranging technology and serial communication. The detection radius is about 8m, which is enough for our robot, which is designed to move on a table. It comes with a motor driver that can automatically adjust the scanning frequency, and the sampling frequency is 3000 times per second.

In our design, the YDLIDAR X3 Pro lidar is implemented to detect the user’s hand. When a user stretches their hand near the lidar, the robot should stop to allow the user to interact with its display.

Figure 5: The YDLIDAR X3 Pro lidar

Requirements	Verification
1. The robot assistant should be stopped when a hand is stretched within a distance of 20cm near the lidar on it. The error of distance should be < 10%.	1. Approach the lidar with a hand and record the distance when the robot is stopped. The actual threshold distance should be within $\pm 10\%$ of 20cm.

2.4.2 TCRT5000

The TCRT5000 is a reflective infrared sensor that integrates an infrared emitter and a phototransistor, adopting reflective ranging technology to detect the presence of objects and their relative distance. It features low power consumption, small size, and stable performance, making it ideal for table edge detection in our desktop robot design. The sensor has a detection distance range of 1mm to 10mm, which is perfectly suited for identifying the boundary between the desktop surface and the air (i.e., the table edge) in our project.

In our design, four TCRT5000 sensors are installed—one near each wheel of the robot—to achieve full coverage of table edge detection. The relative position of each sensor is fixed and calibrated to ensure that no blind spots exist around the robot. When any TCRT5000 sensor detects the table edge (i.e., when the reflective signal weakens significantly due to the absence of a desktop surface), it immediately sends a high-priority signal to the Raspberry Pi, which then triggers a global emergency stop for the wheel motors to prevent the robot from falling off the table.

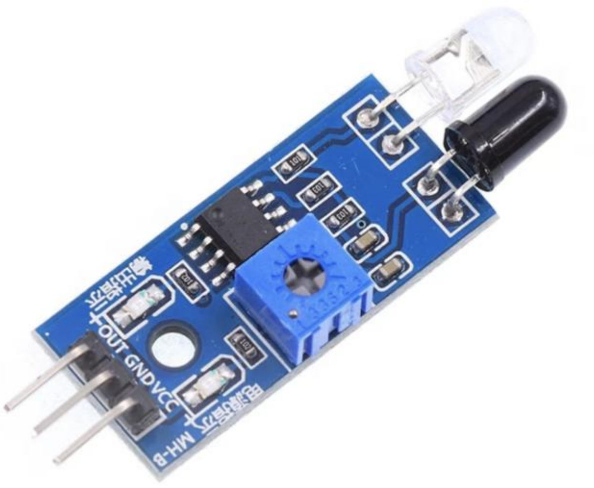

Figure 6: TCRT5000

2.5 Interactive Projection Module

2.5.1 The OpenMV H4 Plus Camera Module

The interactive projection is a core human-computer interaction component of the robot assistant, responsible for providing an intuitive operation interface and collecting user input signals. Specifically, a projection lamp is fixed on the robot car, which projects four direction arrows (up, down, left, right) and two function buttons (Page Up, Page Down) onto the desktop. An OpenMV H4 Plus camera is also fixed on the car, and its relative position to the projection lamp is strictly calibrated and fixed to ensure accurate capture of the projection area.

The OpenMV H4 Plus camera continuously captures the projection area in real time, with the main function of identifying the red finger cot worn on the user's finger and its real-time coordinates within the projection area. When the user's finger with the red finger cot touches or hovers over a specific projection area (e.g., the Page Up button area), the OpenMV H4 Plus immediately identifies the target area, generates a corresponding control signal (e.g., Page Up signal), and transmits this signal to the Raspberry Pi.

Requirements	Verification
1. The projection lamp shall clearly project four direction arrows and two Page buttons on the desktop, with no obvious distortion or blur. 2. The relative position between the OpenMV H4 Plus camera and the projection lamp shall remain fixed, ensuring the camera can stably capture the entire projection area.	1. Observe the projected image on the desktop to check for clarity, distortion and completeness. 2. Fix the camera and projection lamp, then move the robot slightly to verify that the camera still captures the entire projection area without deviation. 3. Wear the red finger cot and move it within the projection area 50 times, record the identification time and success rate,

3. The OpenMV H4 Plus shall accurately identify the red finger cot within 300ms, with an identification accuracy of ≥95%. 4. The OpenMV H4 Plus shall transmit the correct control signal to the Raspberry Pi within 200ms after identifying the target projection area.	ensuring the accuracy is ≥95% and response time ≤300ms. 4. Record the time from the camera identifying the target area to the Raspberry Pi receiving the signal, which must be ≤200ms.

2.5.2 The Communication between The Interaction Module and PC

The OpenMV camera module is connected to the Raspberry Pi by the USB port, thereby realizing the robot movement direction control for the four arrow signals. After receiving the slide page turning signals sent by the camera, the Raspberry Pi forwards the slide page turning signals to the personal computer via sockets. The program on the laptop reads the signals sent by the Raspberry Pi and translates them into commands to turn the pages of the PowerPoint slides using pyautogui. So the slides on the computer can be flicked by the user.

Requirements	Verification
1. The Raspberry Pi shall forward the received signal to the computer via Socket within 1s, and the computer shall successfully execute the corresponding PPT page turning command.	1. Test the signal transmission process, record the time from the Raspberry Pi receiving the signal to the computer executing the command, ensuring it is ≤1s and the command is executed correctly.

2.6 Enclosure and Mounting Module

The mechanical architecture of this project adopts a three-layer stacked structure. Each layer uses an acrylic sheet with dimensions matching the robot's chassis. The three layers are supported and secured using copper standoffs and screws, ensuring reasonable space allocation between modules, firm connections, and convenient wiring and heat dissipation.

Layer	Mounted Modules	Description
Layer 1 (Bottom)	Diffuse reflection infrared sensor	Mounted under the chassis to detect table edges and prevent falls
Layer 2 (Middle 1)	Raspberry Pi and TB6612 driver board	Close to motors for easy connection of motor control wires
Layer 3 (Middle 2)	Lithium battery (12V)	Provides power to the robot; placed in the middle to lower the center of gravity
Layer 4 (Top)	Camera, Micro-projector and Infrared sensor	Used for button projection, user input recognition, and hand detection

2.7 Tolerance Analysis

The reliability of the AR-based palm-sized robot assistant depends on mitigating cumulative errors from mechanical, electrical, and computational subsystems. This analysis focuses on three critical performance aspects: motion control accuracy, signal latency, and power stability, verifying compliance with design requirements for safe, effective operation in presentations.

2.7.1 Motion Control Accuracy Tolerance

Motion control accuracy is critical for precise projection positioning and safe desktop navigation, with a core requirement of ≤5mm positional error. The error budget breaks down as follows:

- **Mechanical Tolerances**: Mecanum wheels and chassis introduce ±2mm backlash, minimized by precision mounting and proper fastener torque.
- **Sensor Measurement Errors**: Wheel encoders and TCRT5000 sensors contribute a combined ±1.5mm error (encoder resolution and IR detection precision).
- **Algorithmic Errors**: Path-planning and motor control algorithms introduce ±1mm quantization error from discrete PWM adjustments.

Result: Cumulative worst-case positional error is ~4.5mm, well within the ≤5mm tolerance, ensuring accurate positioning and safe table edge avoidance.

2.7.2 Signal Transmission and Latency Tolerance

Low latency is key for intuitive interaction, with a requirement of ≤300ms total system latency (gesture detection to response). Latency breakdown:

- **Vision/Gesture Latency**: OpenMV H4 Plus processes gestures in 100ms±10ms, optimized via lightweight algorithms and 10 FPS frame rate.
- **Communication Latency**: USB (OpenMV-Raspberry Pi) adds 50ms±5ms; socket (Raspberry Pi-laptop) adds 30ms±5ms, optimized via high-speed connections.
- **Motor/Control Latency**: Raspberry Pi and TB6612 driver add 40ms±5ms and 20ms±5ms; L298N driver has ≤10ms inductive lag.

Result: Total worst-case latency is 240ms, below the 300ms threshold, ensuring immediate gesture responses.

2.7.3 Power Supply and Signal Tolerance

Stable power and signal integrity prevent component failure. Below is analysis of component tolerance to voltage fluctuations and interference:

- Voltage Tolerance: 12V Li-ion battery discharges to 11V under load; 520 motors maintain ≥0.5m/s, and Raspberry Pi (5V±0.2V) is stabilized by onboard regulators.
- PWM Signal Tolerance: Raspberry Pi’s 10kHz±5% PWM is well within the TB6612/L298N’s 25kHz limit, ensuring accurate motor control.
- Interference Tolerance: Shielded IR sensors, anti-noise LiDAR algorithms, and shielded communication links minimize ambient and EMI interference.

Result: All components operate reliably within tolerance, ensuring consistent performance and reducing system failure risk.

In summary, the tolerance analysis confirms that the robot meets all critical performance and safety requirements, with cumulative errors well within allowable limits for reliable, intuitive operation.

3. Cost

Part	Cost (RMB)
Projector and plate	195
Mecanum wheels and the bottom plate of the robot (include the motors)	265
TDLIDAR X3 infrared sensor	189
TB6612 motor drive module	58
Diffuse reflection infrared sensor	33.4
All wire and electrical components	70
Batteries	174
Projection and camera stand	97
Car floor panel	48
OpenMV4 H7Plus 500W camera	169
Total	1298.4

4. Schedule

Week	Ruixi Qin	Yuzhang Wang	Jiaqi Ding	Fengwei Yang
3/21	Research relevant projects.	Research relevant projects.	Determine the type of chassis and wheels and complete the preliminary structural design.	Completed the project bill of materials, compared prices from multiple suppliers, and finished procurement of all electronic components and structural parts.
3/28	Determine and select the microcontroller.	Confirm the division of labor.	Complete the motor selection and prepare the initial installation plan for the motor driver board and the chassis.	Completed the preliminary structure design of the robot in Fusion 360, determined the layout of each module and the overall dimensions.
4/4	Confirm the division of labor.	Confirm the division of labor.	Install the motor onto the mobile platform and connect it to the driver board.	Designed motor mounts and driver board brackets verified the accuracy of mounting hole positions in the model.
4/11	Install the Raspberry Pi system. Connect it to Wifi. Write the robot movement program controlled by Bluetooth.	Discuss sensor-motion control integration, deliberate TCRT5000 algorithm ideas, and determine interactive projection design.	Optimize the chassis structure based on feedback from motor tests.	Exported the 3D printing model of the chassis and performed post-processing on printed parts (support removal, sanding).
4/18	Connect the Raspberry Pi with the motor driver board and motor. Ensure it spins successfully.	Research obstacle avoidanceanti-fall function integration, debug TCRT5000 algorithm, and test projection lamp image clarity.	Install the infrared sensor at the designated position and ensure that the wiring is connected to the Raspberry Pi.	Designed projector bracket and sensor bracket, ensuring adjustable projection angle and secure fixation.
4/25	Connect the Raspberry Pi with the motor driver board and all 4 motors on the car. Ensure the car can move in the correct direction.	Verify infrared anti-fall reliability and preliminarily calibrate projection lamp and	Fine-tune the friction between the wheels and the ground as well as the smoothness of movement.	Completed full robot assembly, checked connection strength and interference between

	Write the program to send signals from the Raspberry Pi to the laptop to flick the slides	OpenMV camera position.		modules.
5/2	Implement the lidar and write relevant code. Ensure the car can be stopped when I stretch my hand near the lidar.	CConvert anti-fall decisions to motion instructions, fix and calibrate projection lamp and OpenMV camera.		Modified the Fusion 360 model based on assembly feedback, optimized the battery compartment and wiring channels.
5/9	Draw the PCB schematic and layout to replace the breadboard. Send the file to the Taobao seller to get the PCB board printed	Convert anti-fall decisions to motion instructions, write and test OpenMV red finger cot recognition code.	Complete the final physical installation and wiring organization of all sensors, power supplies and control boards.	Completed robot weight balance testing, adjusted battery and projector positions to lower the center of gravity.
5/16	Solder the PCB board and connect it with the Raspberry Pi, the motor driver board, and the sensors. Final integration.	Test anti-fall function on the integrated platform and debug the interactive projection system for stable operation.	Conduct comprehensive mobile and load testing.	Organized mechanical design documentation, exported final engineering drawings and bill of materials.
5/23	Final presentation preparation			

5. Conclusion

5.1 Accomplishments

The proposed AR-based palm-sized robotic assistant for meeting room environments has been successfully implemented and demonstrates full end-to-end functionality. The system successfully integrates a mobile robotic chassis, a micro-projector, a camera, and infrared sensors to enable intuitive human-robot interaction.

The following key accomplishments were achieved:

Projection and Recognition: Control buttons (forward, backward, next slide, previous slide, etc.) are stably projected onto the desktop. The onboard camera reliably identifies user inputs with an estimated accuracy exceeding 95% under normal indoor lighting conditions.

Mobile Control: The robot accurately executes movement commands (forward, backward) based on user input from the projected buttons.

User Detection and Docking: The infrared sensors detect a user's hand from up to 1 meter away. Upon detection, the robot successfully stops within approximately 10 cm of the user.

PPT Control Transfer: Once docked, the robot reliably transfers presentation control rights to the user via Socket, allowing the user to control the meeting room's PPT slides (next/previous page) via the projected buttons.

The complete workflow—from button projection, user input detection, robot navigation, hand detection, docking, to PPT control handover—was successfully demonstrated, confirming the feasibility of using a palm-sized robotic assistant to facilitate shared control in collaborative environments.

5.2 Uncertainties

Despite the successful demonstration of core functionalities, some uncertainties and limitations remain, which are discussed quantitatively below.

Docking Accuracy: While the infrared sensor reliably detects a hand at up to 1 meter, the robot stopping distance from the user shows a tolerance of approximately ±10 cm from the intended docking point. This uncertainty primarily stems from the sensor's beam width, the robot's kinetic inertia during braking, and the unstructured meeting room for the desktop environment (e.g., varying surface friction).

Projection Visibility on Dark Surfaces: The visual clarity of the projected buttons is reduced on dark or black desktops. However, since the system relies on the camera to recognize button coordinates rather than a user's visual confirmation, this limitation does not affect the functional execution of commands. It remains a usability rather than functional uncertainty.

Multi-User Scenario: When multiple users extend their hands simultaneously, the robot currently stops at the first hand detected by its infrared sensor. The system does not differentiate between users, resulting in non-deterministic behavior in crowded scenarios.

"Go to Opposite Side" Command: The implementation of the "go to opposite side" command is preliminary. The robot performs a spin to reorient and then moves straight until a hand is detected. Robust path planning or obstacle avoidance was not implemented for this specific command, making its performance less predictable in cluttered environments.

No system failures (e.g., crashes, overheating, or loss of control) were observed during normal testing. However, the docking uncertainty and the rudimentary implementation of the "go to opposite side" command represent the primary areas where design requirements were not fully met.

5.3 Future Work / Alternatives

Based on the uncertainties identified above, the following improvements and alternative design solutions are proposed for future iterations:

Improved Docking Precision: To reduce the stopping distance tolerance, future versions could incorporate a time-of-flight sensor or ultrasonic sensor with a narrower field of view and more precise distance feedback, enabling proportional speed control during the final approach.

Enhanced Visual Feedback: To address poor projection visibility on dark surfaces, a higher-lumen projector or a projector with adaptive brightness could be used. Alternatively, a small e-ink display under the robot could replace projection entirely for fixed button layouts.

Multi-User Differentiation: A camera-based gesture recognition system or an array of directional sensors could be added to allow the robot to detect which user is extending a hand, enabling user selection or queue management.

Voice Control as an Alternative Modality: To increase accessibility and provide a backup control method, voice commands can be integrated using a simple offline speech recognition module. This would also assist users with visual impairments.

5.4 Ethical Considerations

Our design and implementation adhere to the IEEE Code of Ethics. Specifically:

Avoiding Harm: The hand detection program serves a purpose as a safety mechanism to prevent the robot from approaching individuals who do not wish to interact.

Accessibility: We acknowledge the limitations of visual-only interfaces and have proposed voice control as a future alternative to ensure the public good remains a central concern.

Risk Mitigation: The physical safety risks associated with the mobile chassis, lithium battery, projector, and infrared sensors were systematically addressed, including the use of downward-angled projection, and Class 1 laser safety compliance for sensors.

All design decisions were made with due consideration for user safety, bystander comfort, and equitable accessibility. The robot is intended to augment, not replace, human agency in meeting room environments.