ECE 445

SENIOR DESIGN LABORATORY

PROJECT PROPOSAL

Project Proposal for ECE 445

Team #28

CHENG ZHENG
(cz77@illinois.edu)
YUXUAN WU
(yuxuan59@illinois.edu)
ZHEWEI ZHANG
(zheweiz3@illinois.edu)
ZIYANG JIN
(ziyang3@illinois.edu)

Sponsor: Liangjing Yang

March 24, 2026

Contents

1 Introduction

Portable robotic assistants have strong potential in everyday settings, but their compact form factor often leads to a highly constrained user interface. Most palm-size robots depend on a mobile app or a small set of physical buttons, which makes interaction indirect and less intuitive. In practice, users must frequently switch between the phone and the robot to understand system status and trigger actions, increasing both learning cost and operational friction. Some solutions add external displays to improve usability, but this adds bulk and setup complexity, undermining portability.

To address these limitations, we propose a portable robotic assistant that projects an interactive interface onto any flat surface (e.g., a desk or a wall) and supports both projected "tap-like" interaction and gesture-based commands. By combining dynamic projection with real-time vision-based gesture recognition and feedback, the system aims to deliver a more natural, direct, and engaging human–robot interaction experience while preserving the core advantage of portability.

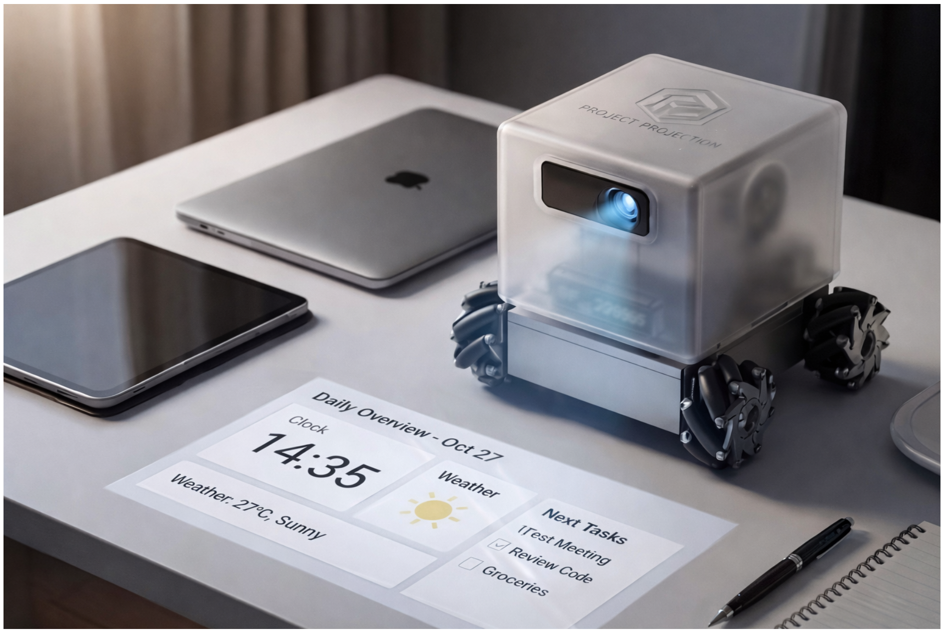

Figure 1: Design Conceptual Graph.

1.1 Objective and Background

- **Goals:** The goal of this project is to improve the usability of a palm-size robotic assistant by providing an on-demand, surface-based interactive interface without requiring an external screen. Specifically, we aim to enable users to access key robot functions through a projected UI and to issue quick commands via hand gestures, while keeping the system compact and portable.
- **Functions:** The robot projects a main interface containing essential icons such as

Weather, Clock, and Exit onto a flat surface. Users can select icons through projected "tap-like" interaction to navigate between pages and query information. In parallel, the system recognizes predefined gestures (e.g., thumbs-up, thumbs-down, heart) and triggers real-time visual and audio feedback for affective interaction. In addition, the robot supports basic mobility under user commands as a closed-loop demonstration of "interaction to command to execution."

- **Benefits:** Compared with app-only or button-only interaction, a projected surface interface provides a more direct and intuitive way to understand system state and trigger actions. Because the interface is projected onto the environment, no additional display hardware is required, preserving portability and reducing setup effort. Gesture shortcuts further lower the interaction barrier by enabling quick, hands-free commands and expressive feedback.
- **Features:** The key differentiators of this project are: (1) transforming ordinary surfaces into an interactive UI through dynamic projection, (2) supporting two complementary interaction channels—projected UI selection and gesture-based commands—and (3) providing immediate multimodal feedback (animation + audio) for a more engaging user experience. These features collectively target a compact, "interface-anywhere" robotic assistant designed for practical desktop scenarios.

This project is motivated by the need to improve the usability and friendliness of portable robots. By turning the surrounding physical environment into an interactive space, the robot can offer a richer and more flexible user experience.

1.2 High-Level Requirements List

The system should satisfy the following high-level requirements:

1. The robot shall project an interactive user interface onto flat surfaces (e.g., a desk or wall) without relying on any external display device, while maintaining overall portability.
2. The system shall support real-time interaction through both projected point-and-click selection and gesture-based commands, enabling users to reliably operate the interface.
3. The main interface shall provide core navigation and information-query functions, including at minimum Weather, Clock, and Exit, and shall display current weather and time information upon user selection with a consistent return-to-main mechanism.
4. The system shall provide affective feedback for predefined gestures (thumbs-up, thumbs-down, and heart) by displaying corresponding animations and playing distinct sound cues.

2 Design and Requirements

2.1 Physical Diagram and Block Diagram:

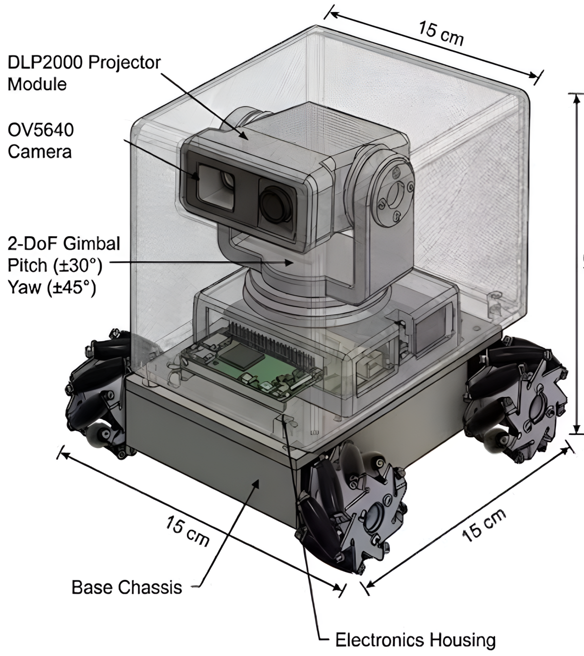

Figure 2: Physical Overview of the Palm-Size Robotic Assistant.

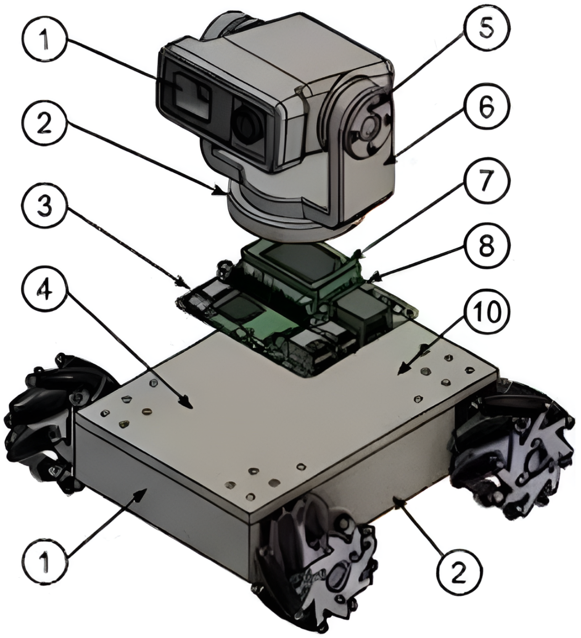

Figure 3: Physical Layout and Major Components.

Table 1: Bill of Materials (BOM)

No.	Part Name	Qty	Description
1	Rectangular Base Block	1	–
2	Mecanum Wheel Assembly	4	Wheel diameter: 5 cm
3	2-DoF Gimbal	1	–
4	DLP2000 Projector Module	1	–
5	OV5640 Camera Module	1	–
6	Raspberry Pi Zero 2 W	1	–
7	Power Management Module	1	–
8	Speaker Module	1	–
9	3D-Printed Enclosure Assembly	1	Overall size: 25 cm × 25 cm × 25 cm
10	Misc. Connectors / Screws	1 set	Fasteners, wiring, and adapters

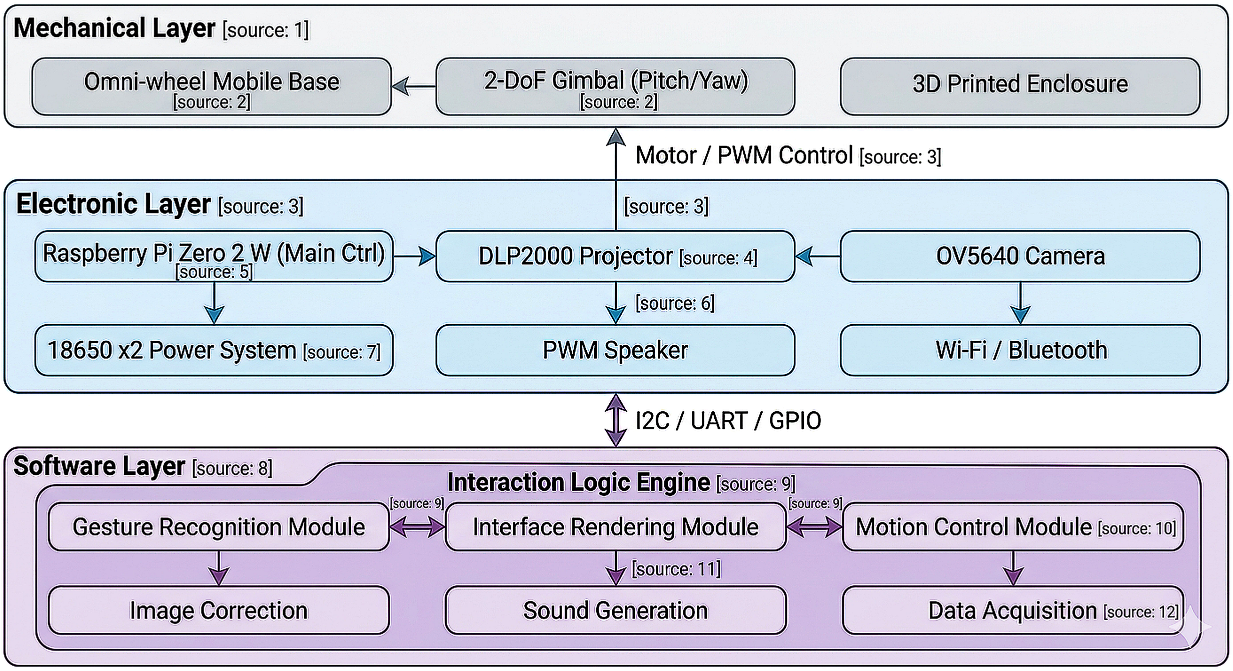
Mechanical Layer [source: 1]
Omni-wheel Mobile Base [source: 2]
2-DoF Gimbal (Pitch/Yaw) [source: 2]
3D Printed Enclosure
Motor / PWM Control [source: 3]
Electronic Layer [source: 3]
[source: 3]
Raspberry Pi Zero 2 W (Main Ctrl) [source: 5]
DLP2000 Projector [source: 4]
OV5640 Camera
[source: 6]
18650 x2 Power System [source: 7]
PWM Speaker
Wi-Fi / Bluetooth
I2C / UART / GPIO
Software Layer [source: 8]
Interaction Logic Engine [source: 9]
[source: 9]
[source: 9]
Gesture Recognition Module
Interface Rendering Module
Motion Control Module [source: 10]
[source: 11]
Image Correction
Sound Generation
Data Acquisition [source: 12]

Figure 4: System Block Diagram.

2.2 Subsystem Requirements:

2.2.1 Mechanical Module Requirements

Table 2: Mechanical Module Requirements

ID	Component	Requirement	Verification Method
M1	Omni-wheel Base	Maximum linear speed ≥ 0.1 m/s; capable of planar translation in arbitrary directions	Mark a known distance on the desk and measure travel time
M2	Omni-wheel Base	Positioning accuracy $\leq \pm 2$ cm (open-loop control)	Command a fixed travel distance via software and measure the actual displacement
M3	Enclosure	Overall dimensions $\leq$ 25 cm × 25 cm × 25 cm	Measure using a caliper or ruler
M4	Enclosure	Total mass ≤ 2.5 kg	Measure using an electronic scale

2.2.2 Electronic Module Requirements

Table 3: Electronic Module Requirements

ID	Component	Requirement	Verification Method
E1	DLP2000 Projector	Projection resolution $\geq 854 \times 480$; maintains clear focus within a projection distance of 20–50 cm	Visual inspection plus a resolution test pattern
E2	OV5640 Camera	Gesture-recognition processing frame rate ≥ 15 fps	Log and report the frame rate in software
E3	OV5640 Camera	Reliable gesture recognition within a working distance of 30–60 cm	Test recognition success rate at multiple distances within 30–60 cm

2.2.3 Software Module Requirements

Table 4: Software Module Requirements

ID	Module	Requirement	Verification Method
S1	Gesture Recognition	Supports four gesture types: point-and-click, thumbs-up, thumbs-down, and heart	Functional test case coverage
S2	Interface Rendering	UI page switching response time ≤ 2 s (covered by F2/F4/F7)	Frame-by-frame video analysis
S3	Interface Rendering	Icon size ≥ 3 cm $\times$ 3 cm (at 40 cm projection distance)	Measure the projected UI using a ruler
S4	Motion Control	Motion command response latency $\leq$ 500 ms (from trigger to base movement start)	Frame-by-frame video analysis

2.3 Tolerance Analysis

2.3.1 Critical Risk Identification

The most critical subsystem in this project is the projection-vision closed loop, which encompasses camera capture, gesture recognition, and projected interface update. This closed loop directly determines the responsiveness and reliability of the user interaction and poses the greatest technical risk to successful project completion. The primary sources of risk are threefold. First, misalignment between the projection area and the camera field of view can cause positional mismatch where a user's gesture over a projected icon does not register correctly in the camera coordinate system. Second, variations in ambient lighting conditions may degrade projection visibility or reduce gesture recognition robustness. Third, accumulated system latency from gesture completion to interface update, if exceeding perceptual thresholds, can significantly impair the user experience.

2.3.2 Projection-Camera Alignment

A critical design parameter is the alignment between the projector′s output and the camera's field of view. The projector and camera are mounted side by side with an optical center separation distance d of no more than 5 cm. Under typical operation with a projection distance $L = 40$ cm, the physical offset introduces a geometric discrepancy that must be calibrated out. The angular offset between the two optical axes is given by:

$$\theta = \arctan\left(\frac{d}{L}\right). \tag{1}$$

Assuming a projected image width of approximately 60 cm at 40 cm distance (based on the DLP2000 throw ratio), this angular offset would theoretically be fully compensable through homography calibration if the projector and camera are aligned on the same horizontal axis. However, residual calibration error must be managed. After calibration, the reprojection error between the projector coordinate system and the camera image coordinate system shall not exceed 5 pixels in the camera frame. Given that the minimum icon size of 3 cm $\times$ 3 cm corresponds to approximately 42 $\times$ 42 pixels in the projected image (based on 854 pixels spanning 60 cm), a 5-pixel reprojection error constitutes roughly 12% of the icon width, which falls within the design tolerance of 20% of the icon size. This margin ensures that even under worst-case calibration error, the click detection region remains sufficiently reliable.

2.3.3 End-to-End Latency Budget

To satisfy the success criterion that all gesture-triggered responses occur within 2 seconds, a detailed latency budget is established for the projection-vision closed loop. Camera frame acquisition and transmission to the processor are budgeted at 50 ms, assuming a 30 fps capture rate with approximately 33 ms per frame plus transmission overhead. Gesture recognition processing, using MediaPipe or a lightweight OpenCV-based pipeline on the Raspberry Pi Zero 2 W, is expected to require 100 ms per frame. The interaction logic engine, responsible for state machine transitions and command mapping, is allocated 50 ms. Finally, interface rendering and projection output are budgeted at 100 ms for frame generation and HDMI transmission. The end-to-end latency of the projection-vision loop can be expressed as

$$T_{\text{e2e}} = T_{\text{cap}} + T_{\text{vis}} + T_{\text{logic}} + T_{\text{render}}, \tag{2}$$

and must satisfy the requirement

$$T_{\text{e2e}} \leq T_{\text{req}}, \tag{3}$$

where $T_{\text{req}} = 2$ s is the maximum allowed response time for gesture-triggered feedback. The corresponding timing margin is

$$M_T = T_{\text{req}} - T_{\text{e2e}}. \tag{4}$$

The total estimated end-to-end latency is therefore 300 ms, substantially below the 2-second success threshold, providing a comfortable design margin.

The primary risk to meeting this latency budget lies in the computational limitations of the Raspberry Pi Zero 2 W when simultaneously executing gesture recognition, interface rendering, and network data acquisition. To mitigate this risk, the software architecture will separate vision processing and rendering into distinct threads to prevent blocking. Interface rendering will leverage pre-rendered frame buffers where possible to reduce real-time computational load. Additionally, if preliminary performance measurements indicate excessive latency, the gesture recognition frame rate can be reduced from 30 fps to 15 fps as a controlled trade-off between responsiveness and computational feasibility.

Through the quantitative alignment analysis and the latency budget presented above, this design demonstrates that the core projection-vision interaction loop is technically feasible within the selected hardware constraints, provided that systematic calibration and software optimization are properly implemented.

3 Ethics and Safety

3.1 Ethics

This project involves human-computer interaction, gesture recognition, and projection-based interface technology, thus requiring consideration of several ethical issues.

Firstly, there are concerns regarding privacy. Since the system may use cameras or visual sensors to recognize gestures, the collected image data should only be used for real-time interaction and should not be stored, transmitted, or shared without the user's consent. Users should also be clearly informed when the system is performing gesture recognition.

Next are issues of fairness and accessibility. Due to the varying action habits and physical conditions of different users, the system should adopt simple and intuitive interaction methods and avoid relying on overly complex gestures, so that more users can use it conveniently.

Furthermore, the system should not mislead users. The purpose of this project is to provide convenient auxiliary interaction rather than to exaggerate the intelligence level of the robot. Therefore, in the design and presentation, the description of the system's functions should remain truthful and clear.

Finally, since this device may be used in shared spaces such as classrooms, dormitories, or offices, attention should also be paid to its social impact in order to avoid invading others' privacy or causing unnecessary interference. In summary, we hope that the design and use of this project comply with the engineering ethical principles advocated by IEEE and ACM.

3.2 Safety

During the design and testing phases of this project, various safety issues need to be considered.

Firstly, there are electrical safety issues. The system consists of a power module, control circuit, sensors, and projection equipment. Therefore, short circuits, overheating, and power abnormalities should be avoided. All circuit connections should be correct, the power supply should operate within the rated range, and necessary insulation protection measures should be taken.

Secondly, there are optical safety issues. Since the system uses a projection display interface, the projection brightness and direction should be controlled to prevent direct strong light from shining into the user's eyes, thereby reducing eye discomfort.

Mechanical safety constitutes the third critical consideration. To maintain operational stability, the robot's projection module, camera, and associated components must be securely affixed, thereby preventing dislodgement, falling, or instability.

In addition, attention should also be paid to heat dissipation safety. Processors, projection modules, and power components generate heat during operation, so basic heat dissipation conditions must be ensured to prevent excessive temperature.

Finally, during the development and testing phases, laboratory safety regulations should be followed. Tools, power sources, and soldering equipment should be used with caution, and experiments should be conducted in as controlled an environment as possible.

In summary, this project does not involve high-risk medical or invasive functions. With basic engineering safety measures in place, the safety risks are low and controllable.