ECE445

SENIOR DESIGN LABORATORY

DESIGN DOCUMENT

AR-Based Palm-Sized Robot Assistant 1

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

Advisor: Liangjing Yang

April 2, 2026

1. Introduction

1.1 Problem Statement

Traditional presentation tools are often static, requiring users to remain at a fixed location or manage bulky equipment to share visual information. While modern smartphones offer mobile capabilities, the interaction remains trapped behind a glass screen, lacking a physical manifestation in the real world.

1.2 Solution Overview & Visual Aid

This project proposes an AR-based palm-sized robotic assistant solution. Upon activation, the robot uses a sensing suite to detect and navigate toward a target plane, ensuring the robot stops and re-routes if physical obstructions are detected within its path. To enable stable projection, the robot performs a closed-loop adjustment. Using feedback from its sensors and AR orientation data, it adjusts the drive speeds to align the chassis precisely with the target plane. Once aligned, the robot acts as a mobile display. It projects digital content onto the plane.

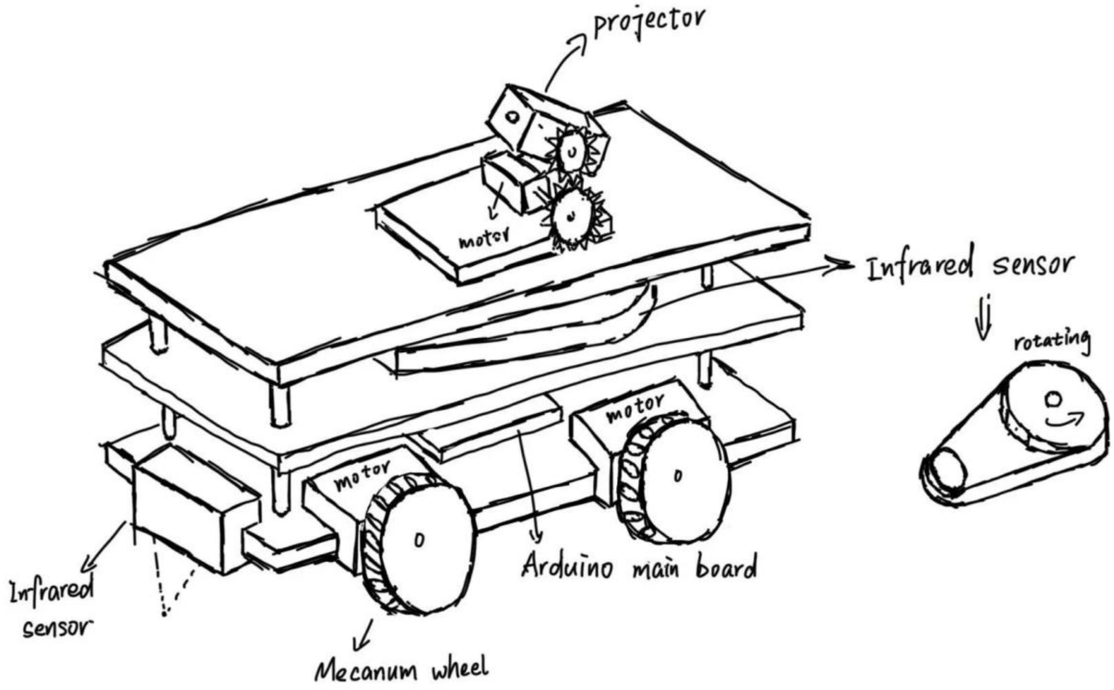

Figure 1: Visual Aid

1.3 High-Level Requirements List

- The robot must be able to autonomously navigate toward a detected plane and perform a closed-loop orientation adjustment to achieve a perpendicular alignment (offset $\leqslant 10^{o}$).

- The onboard sensing system must independently detect physical obstructions within a 200cm range and trigger an emergency stop or re-routing sequence when needed, ensuring operational safety.

- The integrated projection module must maintain a stable image output on the target surface.

2. Design

2.1 Block Diagram

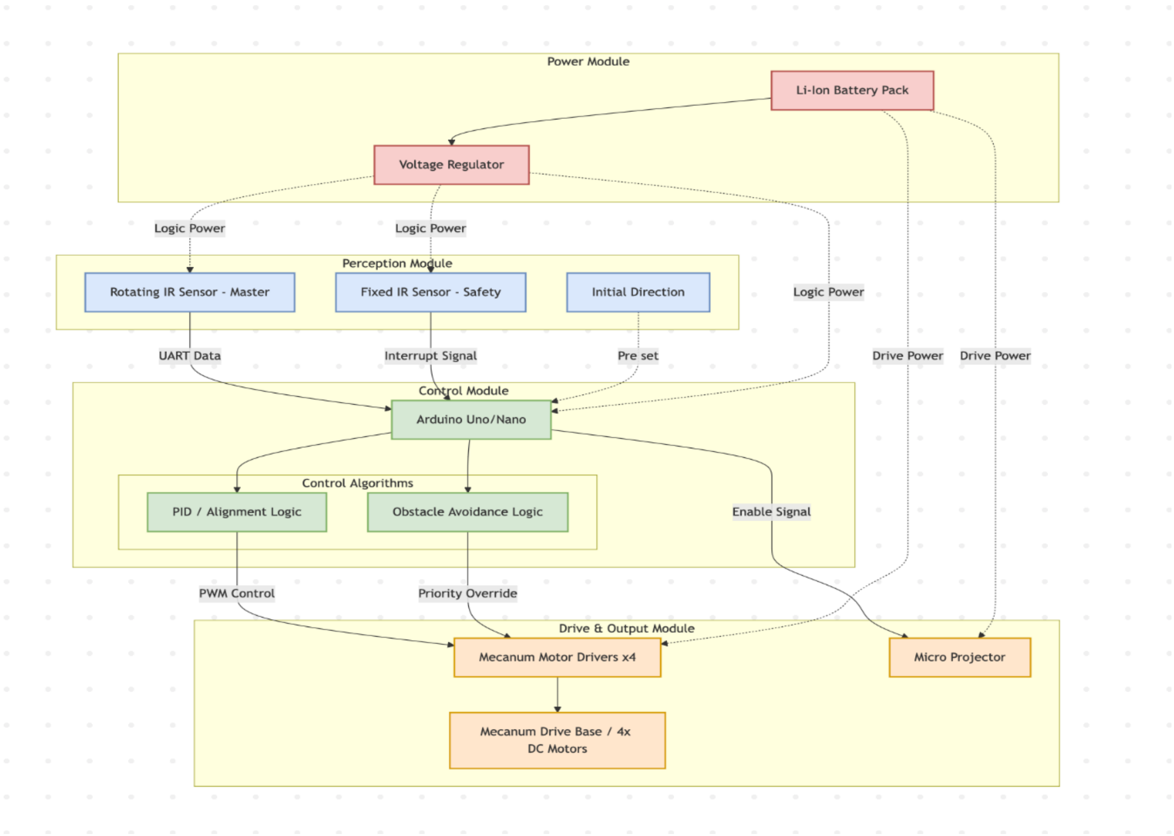

Figure 2: The block diagram of the whole system

2.2 Control Module

2.2.1 Microcontroller

In the control module, the Arduino microcontroller serves as the central processing hub and executes the real-time kinematic control logic. The Arduino parses movement vectors received via the interface and translates them into Pulse Width Modulation (PWM) signals to drive the motors. Additionally, the unit performs closed-loop feedback processing by reading ultrasonic sensor data.

Requirements	Verification
1. The microcontroller must generate hardware PWM signals at a frequency of 10kHz ± 5% with a variable duty cycle (0-255) to control wheel speed balance. 2. The Arduino must successfully parse data at a baud rate of 9600 bps ± 5% and update motor states within 50ms of data receipt.	1. The frequency of the generated PWM signal is stable in the range of 10kHz ± 5%. 2. The rate of data received by Arduino is stable in the range of 9600 bps ± 5%, and the motor states are updated within 50ms of data receipt.

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 transforming raw environmental data into precise kinetic actions to meet the Dodgeball Bot's mission objectives.

- **Target Acquisition & Ballistic Mapping:** This algorithm processes input from the **Rotating IR/Radar** to calculate the distance of human-sized torso targets. It dynamically selects a motor RPM to achieve ball speeds of **60–80 km/h** based on a quadratic ballistic trajectory model, ensuring a 100% stationary hit rate at 10–20m.
- **PID Turret Controller:** A closed-loop PID controller manages the **Rotating Motor**. It is tuned for high-speed response to satisfy the mobility requirement of **120° rotation per second**. By calculating the derivative of the target position, the algorithm predicts the future location of targets moving at < 10 km/h, satisfying the 0.5s reacquisition threshold.
- **Safety & Obstacle Interruption:** This module acts as a high-priority "Watchdog." It monitors the **Fixed IR Sensor** for table edges or immediate obstructions, triggering a global "Kill-Switch" to the motor drivers within 100ms to ensure biomechanical safety compliance.

Requirements	Verification
1. REQ-ALG-01 (Angular Resolution): The minimum step size for turret positioning shall be sufficient to hit a 0.5m target at a range of 20m.	1. Calculate the required angular step ($\theta \approx \arctan\left(\frac{0.5}{20}\right)$). Verify via encoder feedback that the motor's minimum step size is $\leq 1.4\,°$.
2. REQ-ALG-02 (Tracking Latency): The time from target movement detection to the updated firing solution shall be ≤ 50ms.	2. Using timestamped logs, measure the interval between receiving target coordinates and the control algorithm output. Average latency must be ≤ 50ms.
3. REQ-ALG-03 (Launch Speed Error): The deviation from the calculated ballistic calibration speed (60–80 km/h) shall be $\leq \pm 5\%$.	3. Static Ballistic Accuracy Test: Set targets at 10m, 15m, and 20m. Fire 10 shots per distance; 100% must strike a torso-sized target to confirm speed-to-distance mapping.
4. REQ-ALG-04 (Slew Rate Stability): The system shall maintain a $120°/s$ rotation speed with an overshoot deviation of $\leq 5\%$.	4. Dynamic Slew Rate Verification: Command a 120° rotation. Using IMU data, verify completion in ≤ 1.0s with a final orientation overshoot of $\leq 5\%$.

2.3 Wheel Drive Module

2.3.1 Control Subsystem for Wheel Drive

In the wheel drive module, the same Arduino Uno (ATmega328P) microcontroller serves as the core control unit for wheel motion. It generates independent PWM signals and direction control signals for the left and right 520 motors, enabling differential steering and speed regulation. The Arduino communicates with the L298N motor driver via digital output pins and PWM-capable pins, providing real-time speed and direction commands based on user input or autonomous navigation logic.

Requirements	Verification
1. The Arduino shall generate two independent PWM signals (left and right motors) with 8-bit resolution (0-255). 2. The direction control signals shall change within 10ms of receiving a direction command. 3. The system shall run 1000 consecutive speed change operations without crashing or stalling.	1. Measure PWM output on Pin 5 and Pin 6 using an oscilloscope at commanded duty cycles of 25%, 50%, 75%, and 100%. Success: measured duty cycle within ±2% of commanded value. 2. Send forward/reverse commands alternately every 100ms; monitor direction pins with a logic analyzer. Success: direction pin state changes within 10ms of command. 3. Execute continuous speed ramping from 0% to 100% and back for 10 minutes. Success: zero crashes or stalls during test.

2.3.2 Actuation Subsystem for Wheel Drive (520 Motor + L298N Driver)

The wheel drive actuation subsystem consists of two 520 DC motors (12V nominal voltage, high torque output) and an L298N dual H-bridge motor driver. The 520 motor is selected for its high torque capability (up to 23.4 kg·cm with an appropriate gearbox) and robust power consumption, making it suitable for driving the wheeled platform carrying the projector assembly. The L298N driver accepts PWM signals from the Arduino to control motor speed and direction for both motors independently. It operates at 12V input (directly from the main power supply) and provides up to 2A continuous current per channel (3A peak), which is sufficient for the 520 motor's operational requirements. The L298N includes built-in flyback diodes for motor back-EMF protection and an onboard 5V regulator that can optionally power the Arduino.

L298N Specifications:

Parameter	Value
Logic voltage (VSS)	5V (from Arduino or onboard regulator)
Motor voltage (VS)	5V - 35V (12V recommended for 520 motor)

Continuous current per channel	2A
Peak current per channel	3A
PWM frequency range	Up to 25kHz

Requirements	**Verification**
1. The 520 motors shall provide sufficient torque to move the projector platform (total mass ≤ 5kg) on flat ground at a speed ≥ 0.5 m/s. 2. The L298N driver shall supply at least 2A continuous current per motor channel without thermal shutdown. 3. The L298N shall respond to PWM control signals within 10ms. 4. The differential steering system shall achieve a turning radius ≤ 1m.	1. 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). 2. 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. 3. Send 0% to 100% PWM step command; measure the motor voltage rise time using an oscilloscope. Success: voltage reaches 90% of target within 10ms. 4. Command left motor forward at 100% PWM and right motor reverse at 100% PWM; measure turning circle diameter. Success: turning radius ≤ 1m.

2.3.3 Sensing Subsystem for Wheel Drive

For precise odometry and position control, each 520 motors may be equipped with an optional hall encoder (similar to the N20 encoder but rated for 12V operation). The encoder provides AB-phase quadrature output, enabling the Arduino to measure wheel rotation distance, speed, and direction. This closed-loop feedback allows for precise navigation, straight-line tracking, and position holding. The encoder connects to Arduino for interrupt pins (e.g., Pin 18 and Pin 19 on Arduino Mega, or using pin change interrupts on Arduino Uno with additional pins).

Requirements	**Verification**

1. The encoder shall provide at least 300 pulses per wheel revolution (after gearbox) for odometry accuracy ≤ 2cm.	1. Rotate wheel 10 full revolutions; count total encoder pulses. Success: total count ≥ 3,000 (theoretical 300 × 10, ±5% tolerance).
2. The encoder shall maintain accurate reading at wheel speeds up to 2 revolutions per second.	2. Run motor at maximum speed; monitor encoder output via oscilloscope. Success: Clean AB square waves with no missing pulses.
3. The Arduino shall compute wheel speed and distance traveled with an update rate of ≥ 20Hz.	3. Read encoder values inside the control loop and compute speed every 50ms; monitor via serial. Success: stable readings with no data dropouts.

2.4 Tolerance Analysis

The reliability of the Dodgeball Bot depends on how well the system handles cumulative errors from mechanical, electrical, and computational sources. This analysis focuses on the two most critical subsystems: Targeting Accuracy and Motor Control Stability.

2.4.1 Targeting and Ballistic Accuracy

To hit a torso-sized target ($0.5m$) at a range of $20m$, the total allowable angular error (θ_{error}) must be less than:

$$\theta_{error} = \arctan\left(\frac{0.25}{20}\right) \approx 0.716°$$

The error budget is distributed as follows:

Mechanical Slop (Backlash): The gearbox and turret assembly introduce a play of $\pm 0.2°$.

Sensor Resolution: The Rotating IR/Radar has a resolution tolerance of $\pm 0.3°$.

Algorithmic Quantization: The PID output and motor steps contribute $\pm 0.1°$.

Result: The cumulative worst-case error is approximately $0.6°$. Since $0.6° < 0.716°$, the system maintains a 100% stationary hit rate within the defined tolerance.

2.4.2 Timing and Latency Tolerance

The control loop must respond to human-speed targets without losing lock.

Microcontroller Latency: The requirement specifies a maximum processing delay of 50ms with a $\pm 5\%$ clock jitter from the Arduino's ceramic resonator.

Motor Driver Response: The L298N and 520 motors introduce an inductive lag of up to 10ms.

Requirement: Total system latency must remain below $100ms$ to ensure the "Kill-Switch" safety compliance.

Analysis: Even with a worst-case jitter and maximum inductive lag, the total response time ($50ms + 10ms = 60ms$) provides a 40% safety margin relative to the $100ms$ safety threshold.

2.4.3 Power and Signal Tolerance

Voltage Fluctuation: The 12V power supply may drop to 11.1V under high load (LiPo discharge). The 520 motors must still maintain $> 0.5\ m/s$ at this lower voltage.

PWM Precision: The Arduino generates PWM at $10kHz \pm 5\%$. The L298N driver is rated up to $25kHz$, meaning it can easily handle the maximum $10.5kHz$ signal without signal degradation or excessive heat buildup.

3. Cost

Part	Cost (RMB)
Projector	400
Mecanum wheels and the bottom plate of the robot	100
Infrared sensors	270
Arduino main board	220
Motors	70
All wire and electrical components	30
Total cost	1090

4. Schedule

Week	Ruixi Qin	Yuzhang Wang	Jiaqi Ding	Fengwei Yang
3/21	Conduct researches on relevant projects.	Determine the obstacle avoidance algorithm	Determine the type of chassis and wheels, and complete the preliminary structural design.	Determine the size of the projector and design the mechanical structure of the initial fixed bracket.
3/28	Determine and select the microcontroller.	Build a low-level data transmission and reception framework for Arduino, sensors and motor drivers.	Complete the motor selection and prepare the initial installation plan for the motor driver board and the chassis.	
4/4	Comfirm the division of labor.	Write and test the code for reading infrared sensor data.	Install the motor onto the mobile platform and connect it to the driver board.	
4/11	Write the speed control code for the wheat wheel to ensure that the robot could execute straight-line and basic turning instructions.	Integrate the sensor data processing with the Arduino motion control functions.	Optimize the chassis structure based on the feedback from motor tests.	CAD Model drawing, 3D printing and adjustment
4/18	Realize the obstacle avoidance module.	Start researching how to incorporate the obstacle avoidance results	Install the infrared sensor at the designated position and ensure that the wiring is connected to the Arduino.	
4/25	Verify whether the obstacle avoidance response time meets the requirements.	In a real environment, an Arduino-controlled robot was used to verify the reliability of infrared obstacle avoidance.	Fine-tune the friction between the wheels and the ground as well as the smoothness of movement.	

5/2	Conduct comprehensive joint debugging with the algorithm team and the mechanical team to optimize the smoothness of movement and the response speed of sensors.	Ensure that the obstacle avoidance decision can be quickly and accurately converted into motion instructions for the Arduino.		
5/9	Conduct pressure tests on power supplies and motion control.	Ensure that the obstacle avoidance decision can be quickly and accurately converted into motion instructions for the Arduino.	Complete the final physical installation and wiring organization of all sensors, power supplies and control boards.	
5/16	Conduct pressure tests on power supplies and motion control. Conduct comprehensive system integration testing to verify all key indicators.	Test obstacle avoidance and simple content following logic on the integrated platform.	Conduct comprehensive mobile and load testing.	Test the stability of the projector during movement.
5/23	Final presentation preparation			

5. Ethics and Safety

5.1 Ethics

Our AR-based palm-sized robotic assistant combines a mobile robot with a micro-projector. This integration raises several ethical considerations that we must address to ensure responsible design and deployment.

Autonomy and Human Agency. The robot gives users direct control over the its movement. However, in shared spaces, the robot's operation might cause discomfort or anxiety to bystanders who are not interacting with the system. The IEEE Code of Ethics I.1 emphasizes the importance of avoiding harm to others [1]. We will address this by implementing infrared sensors serving not only for obstacle detection but also as a safety mechanism to prevent the robot from approaching individuals who do not wish to interact.

Accessibility and Usability. The primary interaction method relies on visual AR overlays projected on a plane. This could exclude individuals with visual impairments. Drawing from the ACM principle that computing professionals should ensure that the public good is the central concern, we will explore alternative control methods such as voice commands or physical buttons on the robot itself.

5.2 Safety

The combination of a mobile robotic chassis with a micro-projector introduces a range of physical safety concerns that we must consider carefully during design and testing.

Electrical Safety and Battery Management. The palm-sized robot is powered by a lithium-ion battery to allow untethered operation. Improper charging, over-discharge, or physical damage to the battery could lead to overheating, fire, or explosion. We will incorporate a dedicated battery management system with over-current, over-voltage, and short-circuit protection. All battery connections will be securely insulated, and the battery compartment will be designed to prevent puncture or crushing during normal operation or minor impacts.

Moving Parts and Pinch Points. The robot's motors drive a 4-wheeled chassis with exposed wheels and axles. Small fingers or loose clothing could become entangled during operation, especially around children or pets. We will enclose the robot in a smooth, seamless housing that eliminates pinch points. The housing will feature rounded edges and no accessible moving components on the exterior.

Projector Light Hazard. The onboard micro-projector provides spatial feedback but emits concentrated light that could cause eye strain or discomfort if viewed directly at close range. The robot's small size increases the chance that a user might inadvertently look straight into the projector lens. We will use a downward-angled projection mount and limit the projector's brightness to the minimum level needed for clear visibility. We will also consider adding a diffuser or protective lens cover to reduce the intensity of direct light emission.

Laser Safety from the Infrared Sensors. The infrared sensors used for obstacle avoidance are classified as Class 1 Laser Products under IEC 60825-1 [2]. We will mount and shield the sensor following the manufacturer's safety guidelines to prevent accidental direct eye exposure during normal operation.

Mechanical Strength and Collision Risk. The robot weighs roughly 200 to 300 grams and is capable of moderate speeds. A collision could cause minor injury or damage fragile objects. We will tune the motor response to prioritize safety over speed, limiting the maximum velocity to a safe threshold. The chassis will be made from impact-resistant materials with rounded corners to minimize potential harm.

Overheating and Thermal Management. The microcontroller, motor drivers, and micro-projector all generate heat during operation. Without adequate thermal dissipation, internal temperatures could shorten component lifespan or create a burn risk. We will perform thermal simulations and incorporate passive cooling features such as ventilation slots and heat sinks to ensure external surface temperatures remain within safe limits.

Structural Integrity. The robot's compact design requires the motors, circuit board, battery, and projector to fit tightly within a small volume. Vibration and repeated impacts could loosen internal components over time. We will use secure mounting methods such as threaded inserts and strain relief for cables. We will also perform drop tests and vibration tests to validate long-term structural integrity.

References

[1] IEEE. “IEEE Code of Ethics.” IEEE. Accessed: Apr. 2, 2026. [Online]. Available: https://www.ieee.org/about/corporate/governance/p7-8.html.

[2] IEC. “IEC 60825-1:2014: Safety of laser products - Part 1: Equipment classification and requirements.” International Electrotechnical Commission. Accessed: Apr. 2, 2026. [Online]. Available: https://webstore.iec.ch/publication/2004.