ECE445

SENIOR DESIGN LABORATORY

FINAL REPORT

Dodgeball Bot

Team #8
Haoxiang Tian [ht13@illinois.edu]
Yuxuan Xia [yuxuanx9@illinois.edu]
Yujie Pan [yujiep2@illinois.edu]
Chengyuan Fang [cfang14@illinois.edu]

Advisor: Timothy Li

TA: Liu Ningfang

May 29, 2025

Abstract

Dodgeball Bot is an innovative fixed robot system, aiming to address the limitations of traditional dodgeball gameplay, including inconsistent human bodies, safety risks, and the lack of adaptive training scenarios. By integrating real-time computer vision, dynamic tracking and adjustable propulsion mechanisms, this system can achieve precise, safe and customizable gameplay for competitive training or entertainment purposes. Ethical considerations, such as privacy protection and bias mitigation, as well as robust mechanical safety measures, emphasize the project's commitment to responsible innovation. The Dodgeball robot demonstrated the feasibility of an automated motion system in enhancing training consistency, safety and adaptability, paving the way for the future development of entertainment robot technology.

Key words:

Dodgeball Bot, Real-time Human Tracking, PID Control, Computer Vision, Dynamic Target Locking, Modular Subsystem Design.

Contents

1.Introduction

1.1 Problem Statement

Traditional dodgeball gameplay faces challenges due to human physical limitations, inconsistent skill levels, and safety risks when targeting opponents at varying distances. Existing automated sports systems lack integration of real-time human tracking, adjustable launch power, and rotational targeting in a fixed body design, limiting their utility for competitive training or adaptive recreational play. The Dodgeball Bot aims to address these issues by enforcing controlled force limits for safety, overcoming human physical inconsistencies with automated precision, and enabling adaptive gameplay through dynamic tracking and adjustable intensity. It will detect and track human torsos in real time using computer vision and depth sensors, rotate a motorized turret to align the launcher with moving targets, and propel balls at adjustable speeds (10–20m range) while maintaining safety compliance. This system will provide consistent, repeatable training scenarios for skill development, enhance safety by eliminating erratic throws and enforcing force limits, and allow customizable difficulty levels for recreational or competitive use. The Dodgeball Bot combines fixed-body rotational targeting with real-time tracking and variable power control, filling a gap in existing sports robotics, and prioritizes safety without sacrificing performance, differentiating it from static or unregulated systems. At the same time, we solved the limitation of serving speed caused by limited capacity and realized the fast automatic continuous serving design that cannot be achieved by human power.

1.2 Solution Overview & Visual Aid

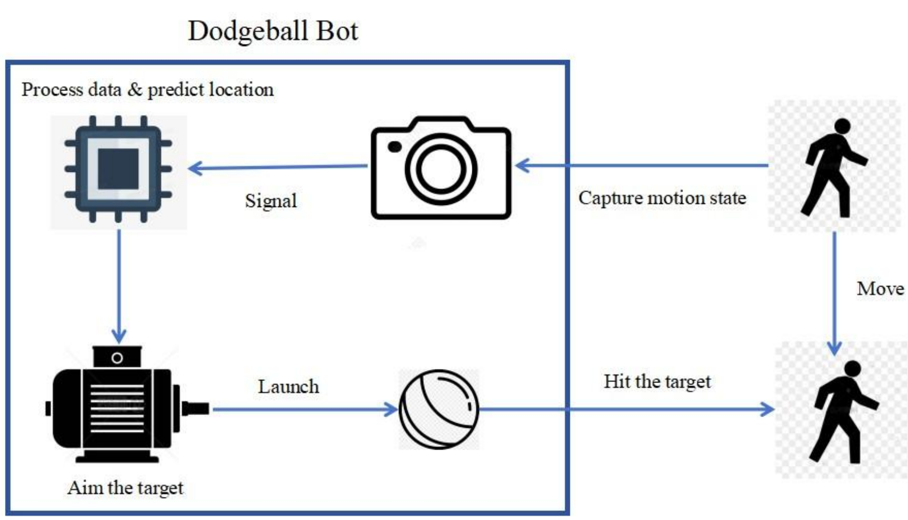

Figure 1: Visual Aid of the whole system

The Dodgeball Bot is a stationary robotic system designed to simulate competitive dodgeball scenarios with precision and adaptability. A you can see in Figure 1, the total system is supposed to have Aim, Rotate, Power and Control subsystem. It employs computer vision and depth sensing through a camera and depth sensor to detect human torsos and calculate target distance. A motorized turret, powered by a high-torque servo or stepper motor, rotates the launcher horizontally to align with moving targets. The adjustable launch mechanism uses rubber wheels to propel balls at controlled velocities, modulated by distance-based PID control. A centralized controller manages sensor input, turret rotation, and launch parameters to ensure seamless operation. The system dynamically adjusts aim and power in real time, offering a safe yet challenging experience for users.

1.3 High-Level Requirements List

The Dodgeball Bot system shall meet the following requirements:

(1) **Accuracy Requirement**:
- Achieve **80% accuracy** in hitting stationary human-sized torso targets at distances of **5–10 meters**.
- Have a **50% accuracy** of hitting a target below 10 kilometers an hour at distances of **5–10 meters**.

(2) **Speed Configuration**:
- Propel balls at configurable speeds between **20–50 km/h**.

(3) **Mobility Performance**:
- Turret rotation capability of **120° within 2 second.**
- Target reacquisition time for moving targets under **1 seconds.**

(4) **Safety & Durability**:
- Compliance with biomechanical force limits to prevent injury.
- Mechanical reliability for **beyond 1,000 consecutive launches** without failure.

2. Design

2.1 Block Diagram

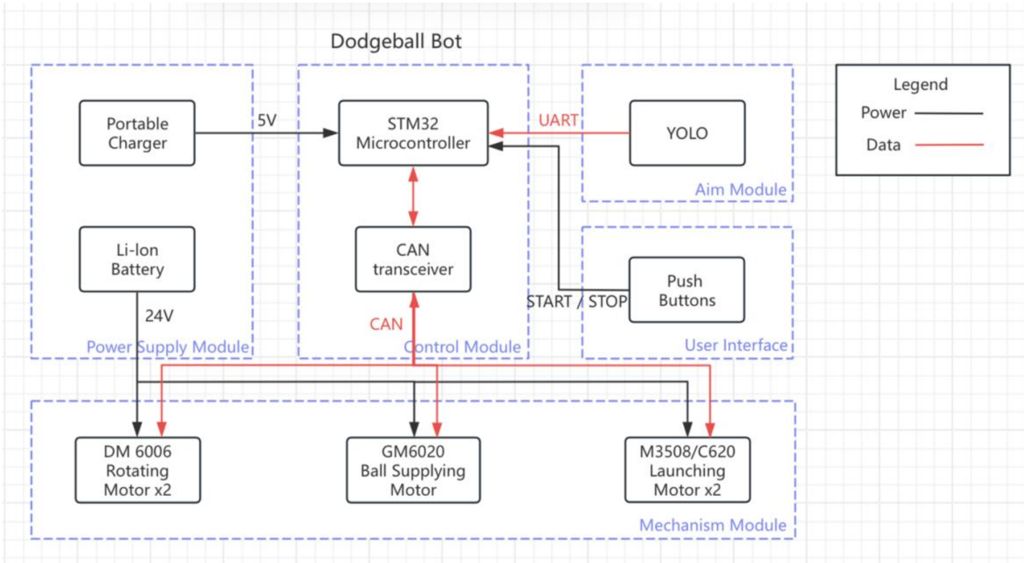

Figure 2: The block diagram of the whole system

In Figure 2, the Dodgeball bot is supposed to have four subsystems such as Aim, Mechanism, Power Supply and Control. Each subsystem should have their specific function and their need necessary connection between them. Here is our simple Block diagram.

2.2 Physical Diagram

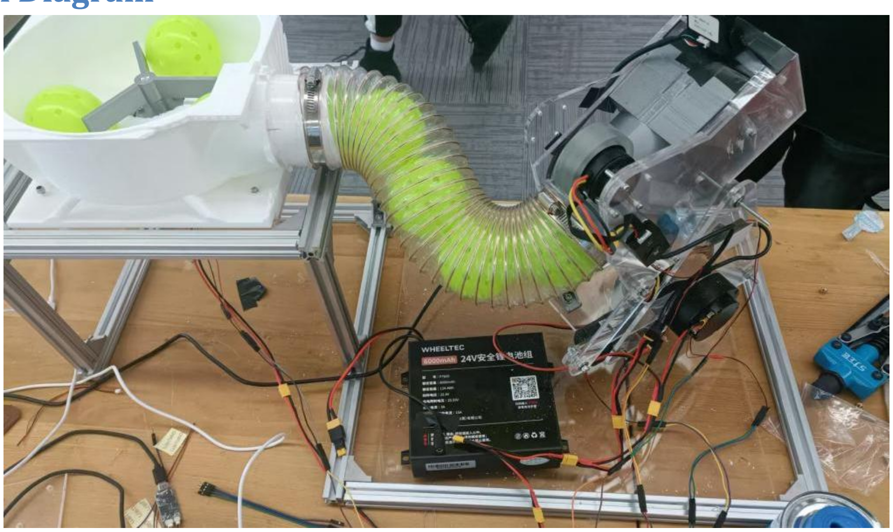

Figure 3: The physical diagram of the whole system

In Figure 3, you can see the general structure of our Dodgeball Bot, the positions of each motor, the connection of power lines and CAN signal lines.

2.3 Subsystem

2.3.1 Aim System

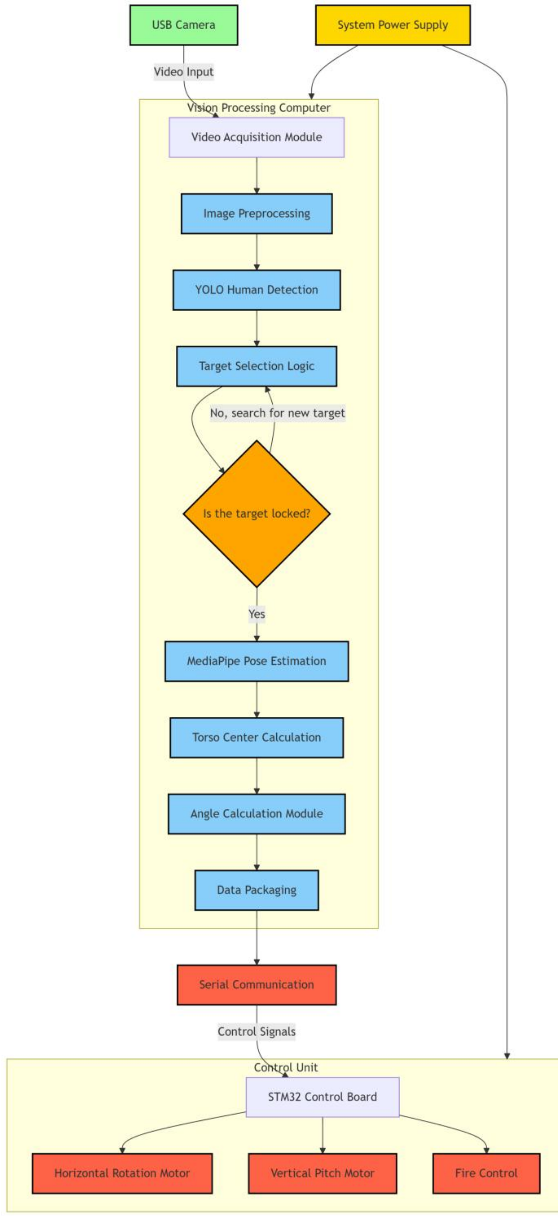

Figure 4: The block diagram of the aiming system

As shown in Figure 4, the aiming system operates through a structured pipeline: beginning with system power supply management (including parameter generation and data acquisition), it progresses to image processing stages featuring VEO stream detection with light-based target selection, new target search functionality, and back-end indexing. The system then performs angle calculations and data packaging before entering the smart communication phase, which handles signal processing through dedicated circuit boards.

2.3.1.1 Hardware Support

Figure 5: RGB Camera

Visual Sensor: RGB Camera (1080p, 3.6 mm, 120°) as we can see in Figure 5, connected via USB.
Processing Unit: A personal computer running Python scripts.
Communication Interface: USB-to-TTL Serial Module (connected to STM32).

2.3.1.2 Human Detection

The system employs the YOLOv8 model for real-time human detection (as illustrated in Figure 6), where the input video stream undergoes multi-layer processing: initial image representation is analyzed through YOLO inference, followed by target management and a locking state machine for tracking, with pose estimation and geometric calculations enabling precise localization. Processed data is then packaged and transmitted serially, while auxiliary layers handle performance monitoring, system configuration, user interface interactions, status display, and parameter adjustments to maintain operational efficiency.

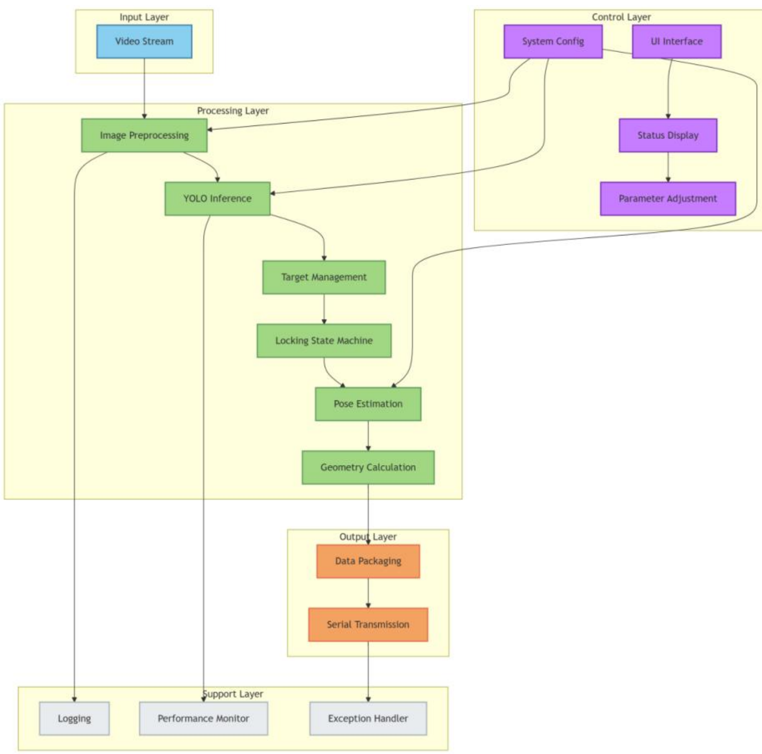

Figure 6: Working principle of YOLOv8 model

2.3.1.3 Target Locking Logic

Once a human is detected, the code calculates the bounding box of the human and determines the distance between the human and the camera's center point. Initially, the code selects the person closest to the camera's center point as the locked target. After locking onto a target, the system maintains this lock for a set duration (15 seconds) without switching targets. After this duration, regardless of whether the current target is still the closest, the code switches to lock the next closest person. If only one person is present in the camera's view, they remain the locked target. However, if this person has been locked for over 15 seconds and another person enters the frame, the code immediately switches to the newly entered person.

2.3.1.4 Distance Estimation

The distance to the target is estimated using the height of the YOLOv8 detection box (already implemented in the code). The formula used is:

$$\text{distance} = \frac{\text{AVG\_HUMAN\_HEIGHT} \times \text{FOCAL\_LENGTH}}{\text{box\_height}}$$

where `AVG_HUMAN_HEIGHT` is the average human height (1.7 meters), and `FOCAL_LENGTH` is the focal length of the camera in pixels.

2.3.1.5 Dynamic Tracking

After locking a target, the code uses MediaPipe for the pose estimation, calculating the shoulder and hip keypoints to determine the torso center of the target. The code calculates the offset of the torso center from the camera's center point and converts this offset into an angle (in radians), which is used to dynamically adjust the aiming direction. When the offset consistently remains below a set threshold, the system determines that the target is successfully locked and sends a signal indicating that firing is possible.

2.3.2 Mechanism System

2.3.2.1 Rotate System

The core concept behind the rotating mechanism is to replicate the dual-axis movement found in tripod head designs, enabling precise control over both horizontal and vertical rotation. This design ensures smooth and stable adjustments, allowing the system to track targets or reposition efficiently. To achieve omnidirectional aiming, the yaw and pitch axes will be driven by GM6020 DC motors [1] as we can see in Figure 7. These motors are selected for their high torque output and precise control, making them well-suited for handling dynamic movement requirements. Their robust performance ensures that the aiming mechanism remains responsive and accurate, even under varying operational conditions. By integrating this dual-axis rotational system, the firing mechanism gains full freedom of movement, significantly enhancing its versatility and effectiveness. Whether for manual control or automated tracking, this design allows for rapid and accurate adjustments, ensuring optimal targeting capabilities. If things get out of hand, consider using an DM-J6006-2EC motor as we can see in Figure 8, which costs similarly but provides greater torque, as backup plan.

GM6020 SPECS

Performance Parameters

Rated voltage: DC 24 V

Rotational Speed (without payload): 320 rpm

Rated torque (max continuous torque): 1.2N·m

Max speed at rated torque: 132 rpm

Rated current (max continuous current): 1.62 A

Operating Temperature Range: 32°~ 131°F (0°~55°C)

Max Permissible Winding Temperature: 257°F (125°C)

Structure Parameters

Motor weight: Approx. 468 g

Hallow shaft inner diameter: 18 mm

Hallow shaft outer diameter: 22 mm

Motor diameter: 66.7 mm

Total height: 45 mm

XT30 power cable: 500 mm

CAN cable: 500 mm

PWM cable: 500 mm

Figure 7: GM6020

Rated voltage	24-48V
Rated current	4A
Max current	13A
Rated torque	4NM
Max torque	12NM
Max speed	150rpm
Weight	0.335kg

Figure 8: DM6006

(1) **Requirement 1**: Sufficient torque

The weight change caused by the recoil and the difference in the amount of ammunition requires sufficient torque to offset.

(2) **Requirement 2**: Sufficient steering angle

The yaw axis needs to have a rotation capability of at least 150°, and the pitch axis should also have a pitch angle range of at least 60°.

2.3.2.2 Launch System

The firing mechanism is designed based on the principles of a ball launcher, utilizing friction wheels as the primary propulsion method to ensure that the projectile's velocity remains both stable and controllable. This approach allows for precise speed adjustments and consistency in performance, reducing variability in projectile trajectory.

In the initial design, projectiles are intended to be manually loaded one at a time, ensuring simplicity and reliability in early stage testing and operation. However, if conditions allow, an automatic feeding system using a bullet tray will be implemented to enable rapid, continuous firing. This would significantly enhance the system's efficiency, reducing the need for manual intervention while increasing the rate of fire.

Regarding the friction wheels' power source, the selection of motors is still under consideration. The preliminary plan is to use two M3508 motors as we can see in Figure 9 and Figure 10, which are expected to provide sufficient torque for the system's operational demands. These motors come with a well-developed speed controller, which not only ensures precise speed regulation but also simplifies the coding process. This integration should enhance system stability while minimizing development effort.

(1) **Requirement 1**: Stable friction wheel rotation center
In order to ensure that the launching mechanism provides constant momentum, the distance between the friction wheels must be fixed to maintain a reliable and sufficient initial velocity.

Figure 9: M3508

M3508 Brushless DC Gear Motor Combo

Rated Voltage：24 V

Rotational Speed (without payload)：482 rpm

Max continuous torque：3N·m

Max rotational speed with a continuous torque of 3N·m：469rpm

Operating Temperature Range：32°~122° F (0~50° C)

Figure 10: The simple datasheet of M3508

(2) **Requirement 2**: Reliable shock absorption system

The vibration during the acceleration process will cause serious momentum waste and directional deviation, resulting in the inability to achieve the required stable initial velocity and direction. At a long distance, this degree of deviation is enough to affect the launch accuracy.

2.3.3 Control System

We predict to select STM32 as our microcontroller[3]. As the core hub of Dodgeball Bot, the control subsystem coordinates the operation of each subsystem through hardware interface and algorithm. It receives json data from the aiming subsystem in real time through the UART interface, analyzes the target coordinates and speed information, and converts it into the control instructions of the mechanism subsystem. The control subsystem connects the GM6020 motor, two DM J6006 2EC motors and two M3508 friction wheel motors through the dual CAN bus (CANH/CANL), and sends the angle setting value and speed command respectively. CAN network controls the horizontal/pitch shaft motor to realize the fast steering of 60°/ sec. CAN network adjusts the speed of the friction wheel to maintain the ejection speed of 20-50 km/h; At the same time, the temperature and current sensor data are monitored and fed back to the microcontroller in real time, triggering an emergency stop or power limit to meet biomechanical safety standards.

As you can see in Figure 11, the control subsystem serves as the central hub of the Dodgeball Bot, integrating data from the aim subsystem (via RGB camera) and orchestrating the mechanism subsystem through precise motor control. Using the STM32F103ZET6 microcontroller, it processes real-time target coordinates, calculates motor commands, and ensures seamless communication across subsystems. The block diagram below illustrates its role:

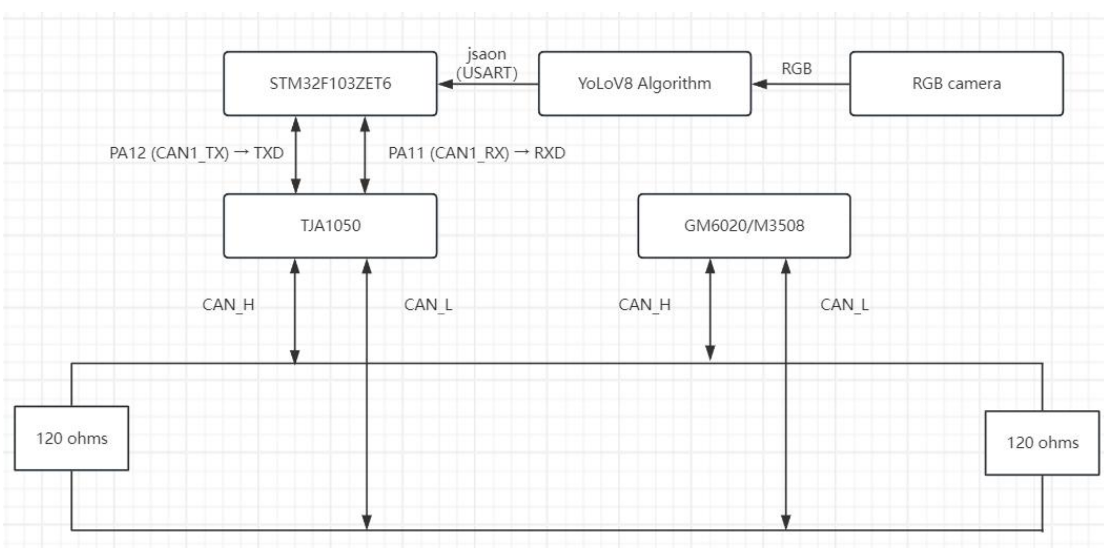

Figure 11: Control subsystem diagram

2.3.3.1 Microcontroller

In the Dodgeball robot system, the STM32F103ZET6 microcontroller in Figure 12 is the core computing unit integrating data processing, real-time control and system coordination. Its 32-bit ARM Cortex-M3 architecture and rich peripherals perfectly meet the project requirements. Become a key hub connecting the "perception layer - decision layer - implementation layer". Its efficient data processing capability, multi-protocol communication interface and hardware-level security mechanism not only support the key functions of real-time target tracking and closed-loop control, but also become the ideal choice for system design with high cost performance and stability.

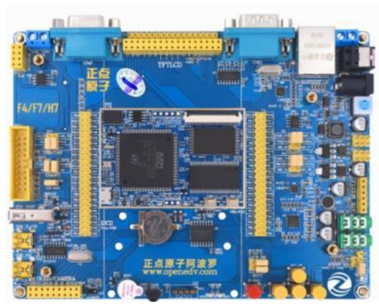

Figure 12: An image of STM32

We have understood the basic principle of the single-chip microcomputer, among which the CAN module and the UART module are the main parts we use. Figure 13 shows the main interfaces we use. PA9 and PA10 are used for UART transmission, and PA11 and PA12 are used for CAN transmission. The chip is powered by 3.3 volts. The Figure 14 shows the single-chip microcomputer uses AMS1117 to achieve the conversion from 5 volts to 3.3 volts.

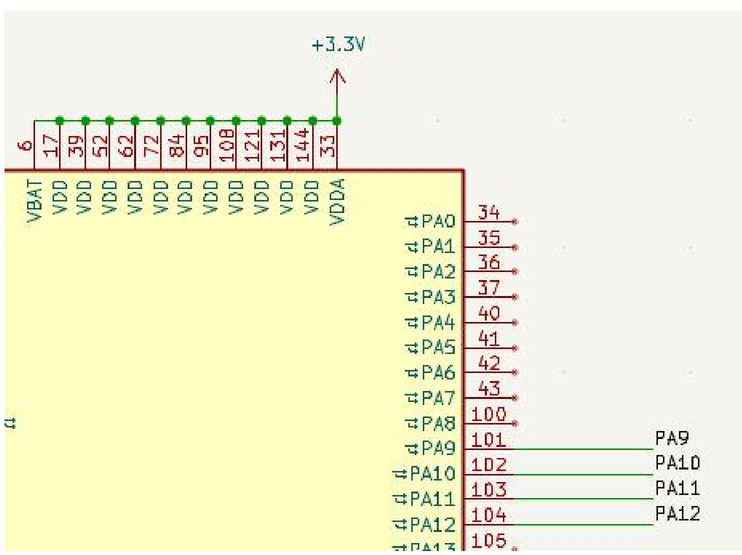

Figure 13: STM32 Connection

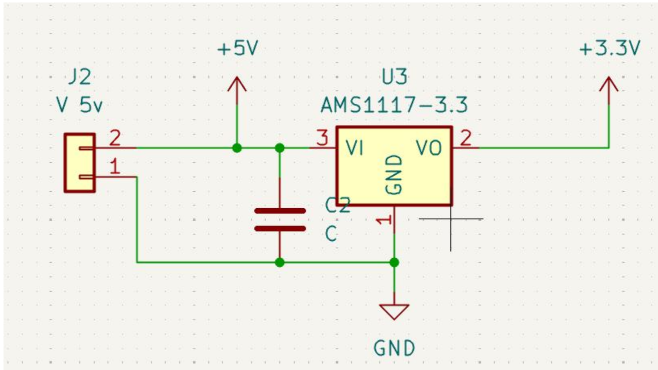

Figure 14: Power supply

2.3.3.2 CAN Bus Module

In the Dodgeball robot system, the closed-loop high-speed CAN Bus module is the key data channel connecting the control core (STM32 microcontroller) and the actuator (GM6020 motor, two DM J6006 2EC motors and two M3508 friction wheel motors). Its core function is to ensure the accurate tracking and striking of the dynamic target by high-speed and reliable two-way data interaction and real-time closed-loop control. Its high speed and low delay two-way data interaction capability not only provides real-time guarantee for dynamic target tracking, but also ensures high precision and stability of mechanical execution through closed-loop feedback mechanism, becoming a key bridge connecting "perception - decision - execution", supporting the high performance of the system in competitive training and entertainment scenes.

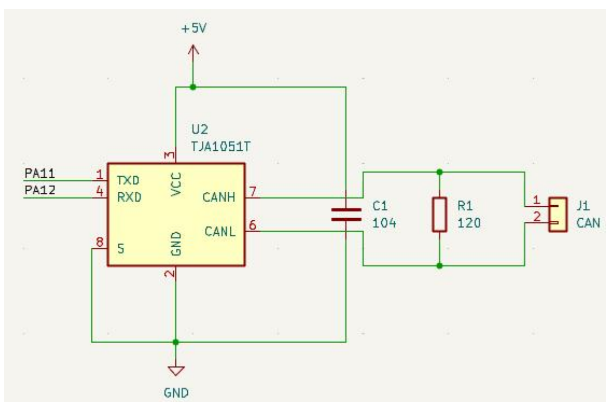

Figure 15: CAN Communication

Just like in the Figure 15, we designed and built the CAN transceiver module by ourselves. The TJA1050T is a high-speed CAN transceiver. It is mainly used for the interface between the Controller Area Network (CAN) protocol and the physical bus, and is suitable for use in partial power supply networks.

In addition, we used PCB to build the connections among each node in the CAN bus in Figure 16. The upper end is CANH, the lower end is CANL, and the middle part is the connection between the CAN transceiver module and the single-chip microcomputer.

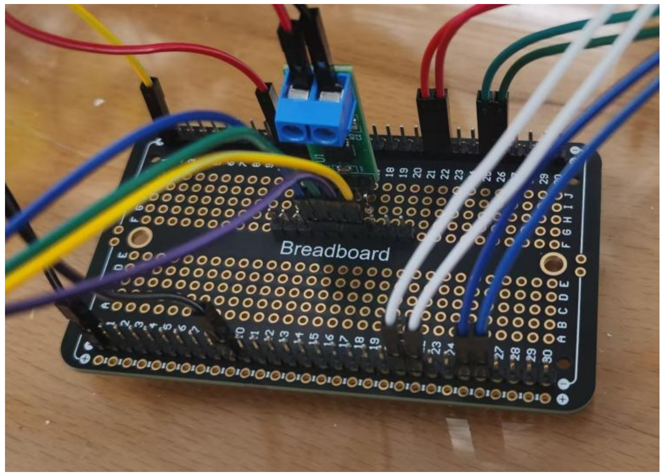

Figure 16: CAN PCB

2.3.3.3 UART Signal Transfer

In Dodgeball robot system, UART signal transmission is a key data link connecting the PC-side vision processing unit (YoloV8 algorithm) and the STM32 control core. Its core role is to accurately transmit the target information extracted by the vision system to the control layer through real-time and reliable one-way data interaction. Support full process control of dynamic target tracking and attack. From pixel-level image recognition to physical space perspective calculation, from algorithm output to hardware execution of cross-system collaboration, the stable operation of UART signal transmission ensures that the robot maintains high precision strike capability in complex environments.

The UART module has been integrated in our single-chip microcomputer. The following Figure 17 is its schematic diagram.

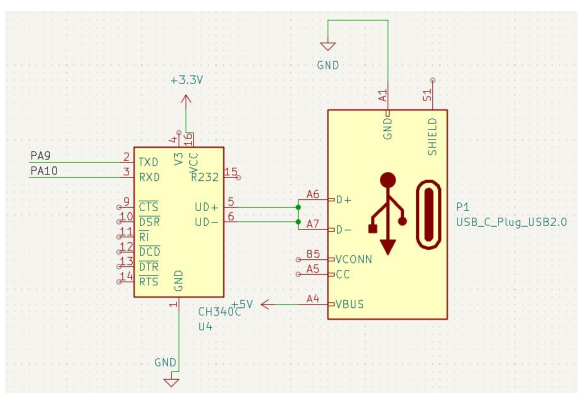

Figure 17: UART receive

2.3.4 Power Supply System

Figure 18: 24V 6000mAh LiFePO4 Battery

The 24V 6000mAh LiFePO4 battery (Figure 18) provides high-current power for the Dodgeball Bot's motors (GM6020 and M3508), enabling rapid rotation and launches. Its XT60 connectors and a XT60 to XT30 converter ensure robust, low-resistance connections (rated for 20A+), minimizing energy loss and heat generation during high-intensity operations. With a lifespan of 2000+ cycles and stable voltage output, it supports long-term testing and repetitive use. The integrated BMS communicates via CAN bus with the STM32, enabling real-time monitoring of battery health and safety parameters. Its lightweight design and wide operating range (-20°C to 60°C) make it ideal for both lab and field deployments.

Figure 19: Portable Power Bank

A compact power bank (Figure 19) supplies backup power to low-power components, including the STM32 microcontroller, sensors, and vision-processing PC, via USB connections. This ensures uninterrupted operation during main battery charging or maintenance.

2.4 Tolerance Analysis

The most critical part of the Dodgeball Bot project is the **human movement prediction and accurate ball launching system**. This component integrates sensor data processing, real-time target tracking, movement prediction, and precise launch mechanism control. Failure in this subsystem would render the entire robotic system ineffective, as it directly impacts the system's primary function of simulating competitive dodgeball scenarios.

(1) Design-Level Risks

(a) **Sensor Data Accuracy**: The precision of the LD2450 radar, camera, and depth sensor data directly affects the system's ability to track targets and predict their movements accurately.

(b) **Algorithm Robustness**: The movement prediction algorithm must efficiently process large volumes of data in real-time while maintaining accuracy.

(c) **Dynamic Environment Handling**: The system must manage multiple moving targets and potential obstructions, increasing the complexity of target tracking and prediction.

(d) **Mechanical Precision**: The launch mechanism must convert predicted target positions into accurate launch parameters, considering factors like distance, angle, and ball velocity.

(2) Movement Prediction

For a target moving with constant velocity:

$$\mathbf{p}(t) = \mathbf{p}_0 + \mathbf{v}_0(t - t_0) \quad (1)$$

For a target with constant acceleration:

$$\mathbf{p}(t) = \mathbf{p}_0 + \mathbf{v}_0(t - t_0) + \frac{1}{2}\mathbf{a}(t - t_0)^2 \quad (2)$$

(3) Launch Parameter Calculation

The required launch velocity *v* to hit a target at distance *d* and height *h*:

$$v = \sqrt{\frac{gd^2}{2\cos^2\theta(d\tan\theta - h)}} \quad (3)$$

(4) Simulation and Validation

Physical calculations validate the system's ability to track moving targets, predict positions, and calculate launch parameters. Results show the system can accurately track targets and predict movements within required tolerances. Launch simulations confirm ball trajectories can hit predicted target positions.

(5) Conclusion

The Dodgeball Bot's design for predicting human movement and launching balls accurately is feasible. Mathematical analysis and simulations demonstrate the system can achieve required accuracy and reliability. Future work will focus on hardware implementation and real-world testing to refine performance.

3. Verification

3.1 Verifications for Aim System

Requirements	Verification
1. The camera shall maintain 120° ± 5° horizontal FOV.	1. Measure with a protractor at 1m distance, edge-to-edge angle = 120° ± 3°
2. Processing unit shall execute YOLOv8 at ≥ 15 fps.	2. Run benchmark.py with monitoring and frame time measurement, average inference time ≤66ms.
3. STM32 shall receive angle data within 50ms of capture.	3. With timestamped packets and logic analyzer trace, end-to-end latency ≤ 50 ms.
4. The system shall detect human figures within 10 - 20 m range.	4. Place test subjects at varying distances (5 m, 10 m, 15 m). There should be ≥ 90% detection rate at 5 m, ≥ 80% at 10 m.
5. Processing latency shall be ≤ 50 ms per frame.	5. Measure inference time using system timestamps. The average latency should ≤ 50 ms over 1000 frames.
6. The data transfer rate should remain ≥ 15 fps.	6. Run continuous detection for 5 minutes. The sustained FPS ≥ 15 with < 10% variance.
7. Initial lock on closest target to image center.	7. Introduce multiple test subjects simultaneously. It should selects a target with min(pixel_offset) in first frame.
8. Immediate switch to new targets after lock timeout when available.	8. Introduce new target at >15s mark, switches within 3 frames (≤200ms) of new target appearance.
9. Single target persistence when no alternatives exist.	9. Single subject test for > 15 s. The machine should maintains lock indefinitely until target exits frame.
10. Accurately track and lock onto human targets within 20m range.	10. Achieve 100% hit rate for stationary single targets within 20 m. Maintain 80% hit rate against targets moving at 1 m/s.
11. Update rate ≥10Hz for moving targets.	11. Sinusoidal target motion at 1 m/s. The tracking lag ≤ 0.1 s phase delay.

Table 1: Verifications for Aim System

3.2 Verifications for Mechanism System

Requirements	Verification
1. Need to achieve satisfactory movement without damaging the motor.	1. The movement capacity ensures that the design requirements are met.
2. Minimize vibration that affects performance during movement.	2. There is slight vibration, which is considered acceptable after testing.
3. Have sufficient durability to support normal use for a medium period of time.	3. No obvious structural damage was found during use.
4. The shape, volume and weight are within the range that can be transported by one person	4. Try to be lightweight and modular, enough for one person to carry.
5. Do not cause unplanned damage to the user or other institutions during movement.	5. After calculating and observing the range of motion, it is believed that no accidental injury will occur.

Table 2: Verifications for Mechanism System

3.3 Verifications for Control System

Requirements for complete control	Verification
1.The time interval between receiving RGB Camera information and sending control instructions via the CAN bus is less than 50ms 2.The communication rate of CAN reaches 1Mbps, and the communication rate of UART reaches 115,200bps 3.Run 1000 consecutive control instruction sending operations without crash or stalling	1.Multiple measurements, the average time interval is less than 50ms, and the maximum value of each measurement is not more than 60ms 2.The rate of the CAN bus signal is stable in the range of 1Mbps±5%, and the rate of the UART bus signal is stable in the range of 115,200bps ±5% 3.During use, the number of crashes or stalling is zero

Table 3: Verifications for complete control

Requirements for MCU	Verification
1.The error rate of control signal transmission is less than 5%, that is, no more than 10 error signals in every 1000 control signals. 2.The communication delay with STM32F103ZET6 is less than 10µs.	1.When encountering a stationary or moving person at less than 1m/s, the engine can only fail five times out of 100 aiming and firing processes. 2.For multiple measurements, the average delay is less than 10µs, and the maximum value of each measurement is not more than 15µs.

Table 4: Verifications for MCU

Requirements for CAN and UART	Verification
1.Data is transmitted continuously for 10 minutes. No data is lost. 2.The data transmission is correct, and the horizontal and vertical deviation error of the character from the camera center is less than 5%.	1.In the process of use, continuous 10 minutes, the received data is complete, no packet loss phenomenon. 2.In the process of use, the actual measurement of the character and the camera position deviation. | Data deviation - Actual deviation |/ Actual deviation is less than or equal to 0.05.

Table 5: Verifications for CAN and UART

4. Costs and Schedule

4.1 Costs

Part	Cost (rmb)
1. RGB Camera (1080P, 3.6mm, 120°), with 5m USB connect wire	90
2. USB-to-TTL Serial Module (connect PC to STM32)	30
3. All PCB components (STM32, TJA1051, CH340 and others)	164
4. Portable power source	319
5. Motors	1700
6. Carbon plate	300
Total cost	2603

Table 6: Costs

4.2 Schedule

<table>
<tr><th>Week</th><th>Haoxiang Tian</th><th>Yuxuan Xia</th><th>Yujie Pan</th><th>Chengyuan Fang</th></tr>
<tr><td>2/24</td><td>Select STM32F103ZET6 chip to control</td><td>learn the power requirements of components</td><td>Vision Recognition Solutions Design</td><td rowspan="3">Parameter calculation, component selection</td></tr>
<tr><td>3/3</td><td>Learn CAN communication basic knowledge</td><td>Understand how the components are connected</td><td>Infrared Vision Solution Research</td></tr>
<tr><td>3/10</td><td>Establish CAN closed loop</td><td rowspan="2">design the circuit to meet the power requirements of components</td><td>Human Tracking Software Solution 1 Selected: YOLO</td></tr>
<tr><td>3/17</td><td>Learn USAT vision transfer signal</td><td>Version 1 Hardware Selected: Raspberry Pi + RGB Camera</td><td rowspan="5">CAD Model drawing, 3D printing and adjustment</td></tr>
<tr><td>3/24</td><td>Establish total PCB to control</td><td rowspan="2">Complete circuit design and PCB plate welding</td><td>Completed and ordered Version 2 hardware: PC + RGB Camera</td></tr>
<tr><td>3/31</td><td>Build the engineering framework of STM32 HAL library</td><td rowspan="2">Develop human tracking software solution: YOLO + MediaPipe</td></tr>
<tr><td>4/7</td><td colspan="2">Learn HAL library and STM32CubeIDE software</td></tr>
<tr><td>4/14</td><td colspan="2">Write total code to receive, process and send data</td><td rowspan="2">Conduct PC-side testing for human tracking software</td></tr>
<tr><td>4/21</td><td colspan="2">Joint debugging with aiming and motors</td><td>Make core parts prepared</td></tr>
<tr><td>4/28</td><td colspan="2">Actually test and improve the code</td><td>Achieve STM32-compatible output</td><td>Whole model under testing</td></tr>
<tr><td>5/5</td><td colspan="2">Verify key indicators against requirements documents</td><td>Prototype testing</td><td>Prototype testing with code</td></tr>
<tr><td>5/12</td><td colspan="4">Final presentation preparation</td></tr>
</table>

Table 7: Schedule

5. Conclusion

This project successfully designed the core subsystems of Dodgeball Bot, including target locking and tracking based on YOLOv8, real-time signal processing of STM32 controller, and dual-axis rotation and transmission mechanism driven by GM6020/M3508/DM6006 motors. Finally, it was verified through practical use that the system met the set high requirements.

5.1 Ethics

The Dodgeball Robotics program is designed to create a stationary dodgeball launcher for training and children's play, and as such must adhere to a code of ethics.

Privacy is the primary concern. The machine is currently designed to use radar sensors for tracking and therefore does not involve image data collection functions. However, if we were to increase the use of cameras at a later stage, we would need to anonymize the data, ensure consent is obtained, and limit the use of the data to the scope of the study.

We are committed to fairness and inclusion. The targeting system needs to be designed to prevent bias based on body size, gender or ethnicity[2]. This requires rigorous testing with diverse participants and improved targeting features to ensure fair performance.

Environmental impact is also considered. We will use environmentally friendly materials, design energy-efficient solutions, and plan for proper disposal and recycling to minimize harm to the environment. In addition, we will ensure that machines are used responsibly as training or recreational tools, promoting positive values rather than violence. We will work with the community to listen to feedback and comply with regulations. In short, we are committed to creating a safe, inclusive and responsible program.

5.2 Safety

The feeding mechanism and the barrel must be equipped with protective shielding to eliminate the risk of accidental injuries caused by unintended discharge or exposure to moving parts. This shielding should be robust enough to withstand potential malfunctions while ensuring that personnel operating or working near the system are not endangered.

All other electronic components should be strategically positioned in areas that remain inaccessible during regular operation, only becoming reachable when maintenance procedures are explicitly carried out. This design choice prevents unauthorized tampering or accidental interactions that could compromise the system's integrity and safety.

While the mass of the projectile is constrained within a safe limit, ensuring it does not pose an inherent danger upon impact, the velocity at which the projectile is launched remains a critical factor. Specifically, the rotational speed of the firing mechanism must be capped at a predefined threshold to mitigate the risk of unintended injuries. A projectile moving at excessive speeds, even within a controlled mass range, could still cause harm in certain scenarios, especially in confined spaces or near personnel. By implementing these limitations, the system can operate within a controlled and predictable safety margin, reducing the likelihood of unforeseen hazards.

5.3 Accomplishment

AI-Based Automatic Real-Time Human Locking: The code successfully captures camera images and achieves real-time human tracking and locking using the YOLO model and MediaPipe. A finite state machine is employed to handle target locking and switching, enabling the recognition of all human targets in the frame and selective tracking. The system generates movement and firing signals for the machine. The code maintains a stable performance of over 10 FPS even on a standard laptop, demonstrating good versatility.

Mouse-Controlled Manual Locking Method: In addition to real-time camera image processing, users can manually track targets and control firing using the mouse. This method replicates the operational flow of an FPS game, providing an additional human-computer interaction approach and enhancing playability.
Data Transmission: Regardless of the locking method used, the code can transmit movement and firing signals to the Control Subsystem in real time, directing the robot to rotate, track targets, and fire.

Real-Time Vision-Motor Integration: The motor control code successfully achieves real-time processing of camera data to dynamically adjust motor operations. By integrating RGB camera input with the STM32 microcontroller, the system processes target coordinates within 50ms latency and generates precise CAN bus commands for GM6020 and M3508 motors.

Custom CAN Transceiver PCB Implementation: The custom-designed CAN transceiver PCB demonstrates robust signal integrity and low-latency communication (<10μs). Leveraging TJA1051 transceivers and optimized trace routing, the PCB ensures reliable bidirectional data exchange between the microcontroller and motors, even under high-frequency operation (1Mbps). This achievement validates the subsystem’s ability to handle closed-loop control demands.

5.4 Uncertainties

Limitations in Distance Estimation Accuracy: The current system relies on a simplified formula based on image height to estimate target distance, yielding only rough approximations. To achieve precise real-world distance measurements, integration of depth sensors (e.g., LiDAR) or high-precision ranging modules (e.g., ToF sensors) is required. This upgrade would incur additional costs, including hardware procurement, development of sensor data processing algorithms, and increased demands on system power consumption and computational resources.

Limitations in Simplified Aiming Logic: The current system employs a basic screen-center reticle approach that fails to account for critical ballistic factors including target distance, projectile velocity, and gravitational drop. To achieve proper trajectory compensation, the implementation requires real-time acquisition of distance metrics and ballistic parameters coupled with a predictive aiming algorithm (e.g., parabolic modeling). This enhancement necessitates integrating distance measurement sensors and projectile velocity detection systems, which would significantly increase computational complexity for ballistic calculations. Potential operational challenges include servo motor oscillation from continuous elevation adjustments and system latency due to heightened real-time processing demands. The technical upgrade consequently involves balancing precision improvements against hardware constraints and response time degradation.

Lack of Distance-Dependent Pitch Calibration: The current motor control logic only adjusts horizontal and vertical angles based on target pixel coordinates, neglecting distance-based elevation calibration. While the code estimates target distance via bounding box height, this data is not yet integrated into the pitch axis (DM-J6006 motor) control loop. Without dynamic elevation adjustment, launch trajectories may deviate at varying distances, compromising accuracy.

Code-to-Motor Distance Data Pipeline Gap: A resolution requires modifying the Aim subsystem code's pipeline to transmit real-time distance data via UART to the STM32. The microcontroller must then calculate the required pitch angle using kinematic equations and send updated commands to the DM-J6006 motor. Potential challenges include latency introduced by additional calculations and sensor inaccuracies propagating into elevation errors. Prototyping and iterative PID tuning will be critical to mitigate these risks.

Material and Manufacturing Constraints in Structural Durability: Due to limited funding, most locations only use acrylic plates as the main material, and long-term use may cause a burden on some structures. In addition, with the limited production process of the laboratory, the rivet structure of some joints cannot reach the best level. Although it can work normally, it can only meet lower expectations.

Platform-Dependent Performance Limitations: Due to the launch elevation angle and the need for an external control system, it still needs to be placed on a relatively flat platform such as a desk to achieve the best range and angle as well as the user experience.

References

[1] Ahmet Can Ceceloglu and Tulay Yildirim. Modeling and simulation of turret stabilization with intelligent algorithms. *Procedia Computer Science*, 154:377–382, 2019. Proceedings of the 9th International Conference of Information and Communication Technology [ICICT-2019] Nanning, Guangxi, China January 11-13, 2019.

[2] IEEE Global Initiative on Ethics of Autonomous Systems. Ethically aligned design: A vision for prioritizing human well-being with autonomous and intelligent systems. Technical report, IEEE Standards Association, 2021.

[3] STMicroelectronics. *STM32H7 Series Reference Manual: Advanced Arm-Based 32-Bit MCUs*, 2024. Version 3.0.