Lightweight Autonomous Soccer Robot: Team Radian

RoboCupJunior 2022 Competition

Jieruei Chang (jchang@princetonk12.org),
Shrey Khetan (shrey@khetans.com),
Yangwenbo Yao (william.us35@gmail.com),
Luna Shoval (ashoval@princetonk12.org)

Princeton High School, NJ, USA

May 2022

1 Abstract

This paper provides an in-depth explanation of the software, mechanical, and electrical aspects of the soccer robots made by Team Radian, which will be competing in the 2022 National RoboCupJunior Soccer Lightweight competition. It also provides an overview of our development process in the creation of the robots.

2 Introduction

Team Radian is composed of four high school students with an enthusiasm for robotics. Our team members are Jieruei Chang, Shrey Khetan, Luna Shoval, and Yangwenbo (William) Yao; we specialize in software, vision, mechanical and electrical, respectively. This is our first year of participating in RoboCup and we did not have much prior experience with robotics before. We have all learned a lot this year about programming, CAD modeling, and working with a sophisticated robotics platform.

The road to here was anything but smooth, but we are proud to present our two robots for the 2022 Robocup season.

Figure 1: Offense Robot

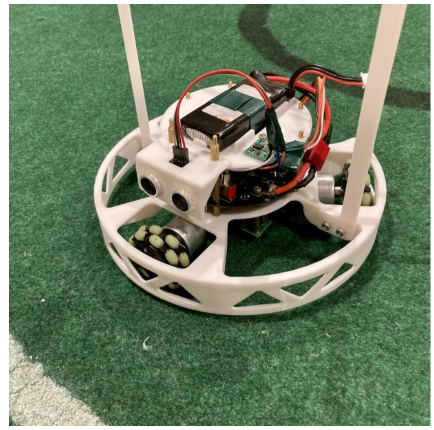

Figure 2: Defense Robot

Below are listed the parts we used to construct our two robots.

Part Name	Description/Purpose	Quantity
Zircon	Main PCB with motor drivers and voltage regulators	2
Teensy (3.5 & 3.6)	Powerful and small microcontroller	2
Omniwheel	Allow for rapid translational/rotational movement	6
Daisen 6V Motor	Allow robot motion	6
OpenMV H7 R2	Programmable camera for goal detection	1
HC-SR04 Ultrasonic	Allow for goalkeeper stationkeeping	1
TSSP4038 IR Sensor	Detect the ball	16
BNO055 IMU	Determine which way is forward	2
3D Printed Parts	Lightweight parts for structural support	8

3 Project Planning & Development Cycle

3.1 Determination of Necessary Functionalities

The process of creating the robot began with determining the features desired. We decided to aim for the following:

1. Two robots with omnidirectional translational movement capabilities
2. IR sensors to detect the direction of the ball
3. Gyroscope/Compass to maintain a forward direction
4. Orbit and damping algorithms to efficiently capture the ball
5. Line sensors to stay inside the field boundaries
6. A camera (with mirror tower) to determine the position of the goal
7. A kicker to launch the ball into the goal

Due to time and experience constraints the eventual design included the first six features. Later into the development process, we decided to attach an ultrasound sensor to aid the goalkeeper robot from going into the box around the goal and to maintain a constant distance from the field walls.

3.2 Initial Testing and Redesigning

The 3D printed parts went through multiple redesigns. Specifically, the main chassis was difficult to design. We ran into issues such as weak structural integrity, misplaced screw holes, complications with 3D printing and difficulty with robot assembly. All these issues were slowly mitigated with each new prototype, although even the newest designs are not perfect.

The software also developed through multiple iterations. The first few iterations of the attacker could barely score a goal, and the first prototypes of the goalie algorithm were very fragile and would cause the robot to crash into the walls when presented with rapidly changing data. The final versions are much more robust against differing conditions, and are far more reliable in attacking and defending.

3.3 Final Assembly

The final product of the robot provides reasonably strong structural integrity, uses three DC gearbox motors, a camera with onboard computer vision capabilities, ultrasound for the goalkeeper and IR sensors to track the ball.

4 Software

4.1 Movement

The robots use a triad of omniwheels for unconstrained lateral translation. Trigonometric equations are used to determine how quickly each of the three motors must rotate in order to move in any given direction. Rotational force can be applied simply by adding an additional constant speed to each motor.

To maintain a consistent orientation, the robot uses a gyroscope for angle correction. To avoid overshoot while simultaneously providing rapid and efficient error correction, a proportional-integral-derivative (PID) control system [1] is used to determine the strength of the angle correction based on the magnitude of the difference between the desired and current angles:

$$u(\theta) = K_p \cdot \theta + K_i \cdot \int_0^t \theta \cdot dt + K_d \cdot \frac{d\theta}{dt}$$

where θ is the error angle and K_p, K_i and K_d are the experimentally determined proportional, integral and derivative gains respectively. The proportional term applies a corrective force that is matched by the resistive force of the derivative force (to prevent overshoot).

4.2 Ball Detection

Detection of the infrared ball is conducted through analyzing the outputs of eight IR sensors positioned equidistantly around the main circuit board. The data from each sensor can be interpreted as a vector whose direction is that of the sensor's orientation, and whose magnitude is the strength of the infrared signal received by that sensor. By taking the summation of all eight vectors, we can determine the precise direction in which the ball is located with the equations

$$v_x = \sum_{j=0}^{7} m_j \cdot \cos\left(\frac{\pi}{4} j\right)$$
$$v_y = \sum_{j=0}^{7} m_j \cdot \sin\left(\frac{\pi}{4} j\right)$$

where v_x and v_y are the $x-$ and $y-$ coordinates of the resultant vector respectively, and m_j are the magnitudes of the signal from each sensor.

4.3 Line Tracking

Line detection is done with a triad of line sensors attached to the underside of the robot. The line detection algorithm is extremely conservative as it is better to lose possession of the ball than to have the robot be removed from the field.

Currently, line detection is only run on the attacker robot, as the penalty-box lines tend to confuse the robots, and the goalie can stay inside the field boundaries relatively well using solely the readings from the ultrasonic sensor.

4.4 Goal Detection

Utilizing the OpenMV camera [2], the scoring goal is detected via computer vision techniques: The camera runs a threshold filter over the input image to detect all pixels within a certain (calibrated) color range. Then, it determines the locations of large patches of the color, and the coordinates of the center of the largest patch are sent to the robot via UART [3] communication. The offense robot uses this information to decide its heading once it is in possession of the ball.

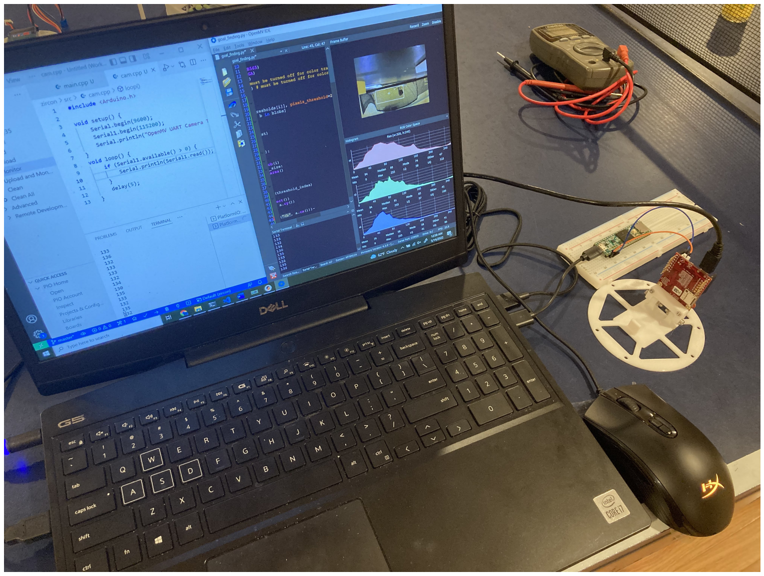

Figure 3: Demonstration of goal detection and UART communication

4.5 Orbit

Orbit is the primary attack functionality, allowing the robot to make a clean and efficient curve towards the ball to kick it. Orbit is done with the mathematical function

$$f(\theta, d) = \theta + j \cdot d \cdot \min(k^{\theta}, 90)$$

where j and k are constants determined experimentally, θ is the angle from the robot to the ball, and d is the robot's distance to the ball. Essentially, the robot will head in the general direction, with an offset corresponding to the angle difference between the ball and robot heading.

4.6 Goalie

The Goalie algorithm attempts to deflect the ball away from the goal by matching the ball's x-position. However, simply moving in the direction of the ball is inadequate as the robot will not react quickly enough if the ball has a large horizontal velocity. Therefore, the goalie calculation also takes into consideration the rate at which the ball's angle vector to the robot is changing.

To keep the goalie in front, an ultrasound sensor is used to keep a constant distance from goal. If the ultrasound detects that the robot is much farther from the goal than previously (indicating that it has crossed the threshold of the goal, it immediately moves in the reverse direction to prevent the robot from leaving the field. This is enough to stop most shots, but preventing the goalie from leaving the area directly in front of the goal allows for opponent robots to score from the sides. In the future we aim to mitigate this, perhaps with more numerous line sensors to follow the arc of the penalty box.

4.7 Development Tools

The software was primarily written in C++, with the computer vision code for the camera being written in MicroPython. Platformio was used to simplify development.

In addition, we developed a visualizer using Pygame, to facilitate debugging of the various sensors on the robot.

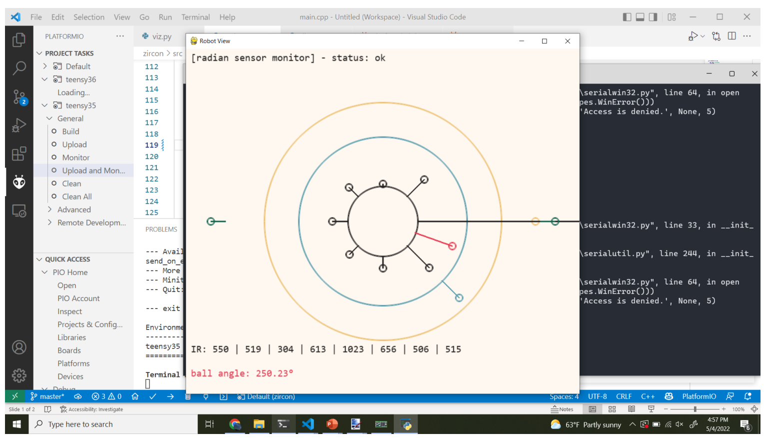

Figure 4: Our visualizer. Here it is revealing an issue with an IR sensor.

5 Hardware

5.1 Mechanical Design

We used the CAD software Onshape [4] to design the various 3D-printed parts on the robot, and printed them using a Prusa i3 MK3S printer. PLA was chosen for its ease of printing; however this means that the printed parts are not incredibly strong and are more prone to damage.

The printed components consist of a main chassis and sensor assembly board, which are identical between the two robots.

The offense robot has an additional camera mount, and the defense robot has a backward-facing ultrasound mount. The printed pieces are designed in such a way that it is simple to replace or swap components between robots if necessary.

5.2 Electrical Design

In our robot design, we use a single main PCB to save space, resulting in a cleaner overall design. We use a Teensy [5], a small but powerful microcontroller, to control the robot. Each

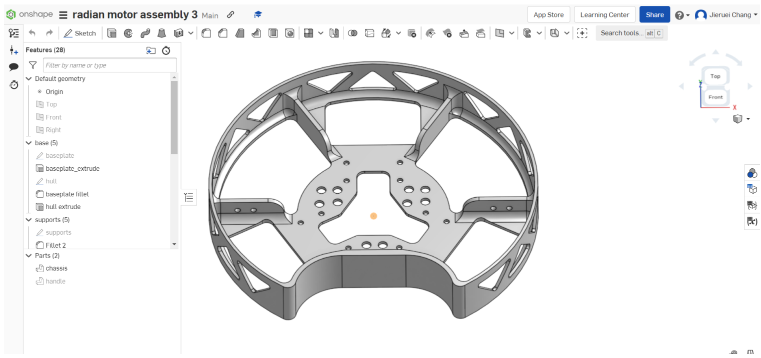

Figure 5: Main Chassis viewed in Onshape

robot has eight infrared sensors mounted symmetrically around the exterior perimeter of the PCB in order to determine the angle to the ball. Both robots also have three line sensors on the underside to avoid overstepping the field boundaries. Our goalkeeper robot contains a backward-facing ultrasonic sensor, in order to maintain a constant distance in front of the goal. We also installed a camera on our attacking robot in order to determine the position of the goal.

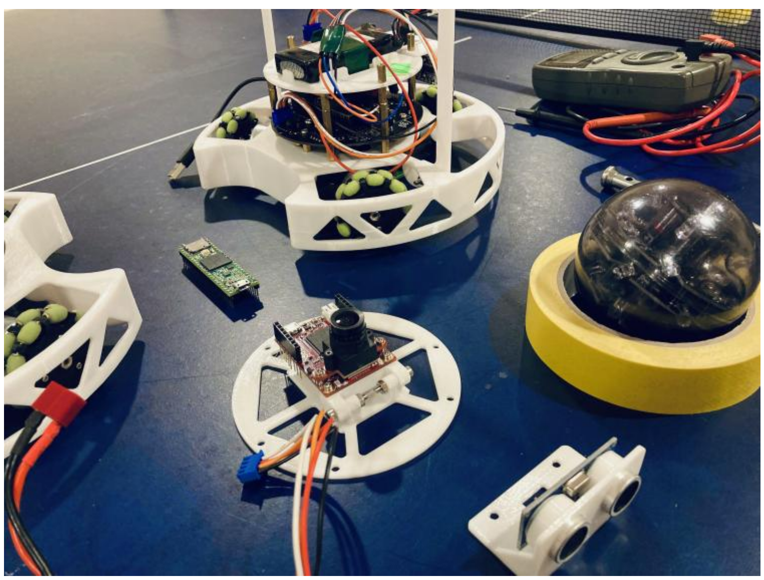

Figure 6: View of 3D printed parts and most of the electronics

6 Reliability

Our OpenMV camera is relatively reliable but sometimes ends up having some difficulty pinpointing the largest patch of color if the background is not completely uniform. The line sensors are relatively reliable given that there are only three sensors and the IR ball sensors are reliable and can be used to detect the angle of the IR ball with substantial accuracy. However, using their measurements to estimate the distance to the ball is not very reliable;

thus the damping function sometimes causes errors and detection of ball possession is not always dependable.

7 Limitations and Further Research

Due to time and monetary constraints we had to make several sacrifices in the construction of the robot. The primary limitations of our robot are the lack of a kicker, dribbler or line sensor ring. In addition, each robot only has three motors; however this makes the robot less efficient and less powerful. Although we attempted to compensate for these shortfalls in hardware with more reliable and effective software, the robot could be drastically improved with such improvements with hardware.

8 Conclusions

Radian's robots are surprisingly powerful and intelligent soccer robots, despite their relatively low cost and incredibly low weight. Our robots feature infrared ball tracking, efficient orbit algorithms and vision-based navigation, along with effective ultrasound-assisted goalkeeping strategies, and PID angle correction.

We are confident that we can improve our robots even more for the next competition, as we have gained a significant amount of experience from our work this year.

9 Acknowledgements

We are grateful for all of the help from everyone who assisted us in making this possible. We would like to thank our sponsor OpenMV for providing us with their cameras, and Team Orion for answering our technical questions. We would also like to thank our parents for their support throughout this endeavor.

References

[1] `https://www.ni.com/en-us/innovations/white-papers/06/pid-theory-explained.html`

[2] `https://openmv.io/collections/cams/products/openmv-cam-h7-r2`

[3] `https://www.analog.com/en/analog-dialogue/articles/uart-a-hardware-communication-protocol.html`

[4] `https://www.onshape.com/en/`

[5] `https://www.pjrc.com/teensy/`