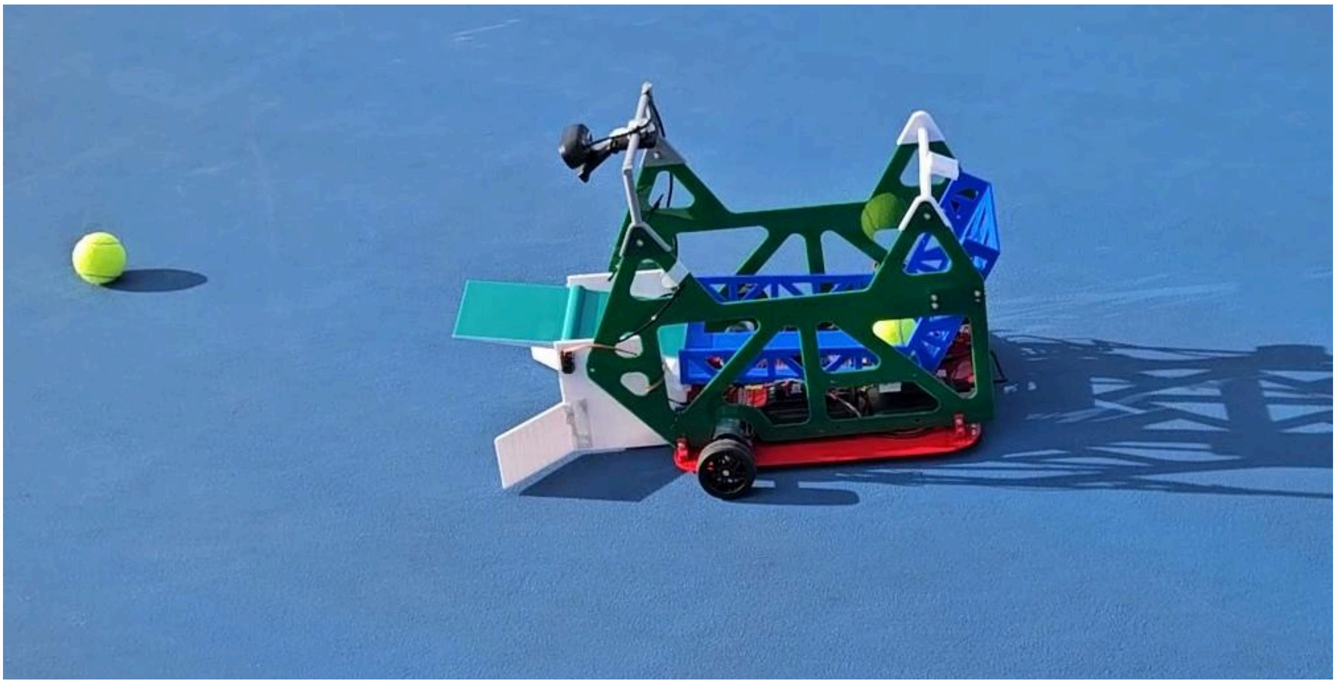
getPaddled
BALL COLLECTING BOT

ECE4191 Final report



Team Members

Joseph Darling 30566703 | Isabelle Hee 31445462 | Edric Lay 32471491

Aryaman Kathuria 31462367 | Sadman Arafat 31866905 | Rohit Phalke 31442552

EXECUTIVE SUMMARY

This project aims to develop an autonomous robotic system capable of picking up and depositing tennis balls in a tennis court quadrant. The finalised tennis ball collection system is a differential wheel-driven robot with a cost of $351.39. The components necessary for building this robot have been provided in assembly and engineering drawings, along with a comprehensive bill of materials. The developed robot can identify tennis balls using an onboard camera, and is capable of movement towards balls and picking them up with a paddle to be stored in a bucket. This bucket then drops the balls into the storage box in the centre of the court.

During the competition, the system's robot navigation, movement, ball and line detection, and tennis ball pickup operated successfully as intended. However, the robot could not deposit the maximum load of 5 tennis balls due to the replacement depositing servo motor lacking sufficient torque. The servo motor replacement was necessary due to damage to the original servo motor. While the original servo motor was operational, the system was tested in multiple environments, including on the blue outdoor tennis court. These tests assessed the functionality of all subsystems and confirmed that the system met the client's requirements.

In retrospect, the team deemed the project overall successful. The collaborative efforts of the mechanical, electrical, and software sub-teams contributed to the cohesive final design. However, the team recognized that the project could have been more successful with better preparation and access to spare components.

1.0 PROBLEM STATEMENT

The client requested the development of a tennis ball collection robot capable of recognizing, navigating and delivering tennis balls to a designated gathering box.This need arises from the notion that picking up tennis balls during training is not only monotonous and time-consuming but can also lead to player fatigue and injury, ultimately reducing training efficiency.

The stakeholder required the robotic system to be financially viable for both players and organisations, as existing robotic systems with similar features, such as the Tennibot [1], have a minimum price of $2,995. Additionally, the system needed to ensure safety, as it could collide with people on court, including children, necessitating an even higher standard of safety.

To provide a solution to the overarching design problem given by the client, the team developed functional requirements for user operation, and robot functionality. Key functionalities include collecting tennis balls present on the court, and depositing collected balls into the designated storage box, whilst complying with the cost requirements, safety needs, and sustainability requirements.

2.0 DESIGN EVALUATION

Table 1: List of requirements and project implementation. Green indicates a fully met requirement, yellow a partially met requirement, and red a failed requirement

	Requirements	Project Resolution
	Functional Requirements	
F.1.	The robot shall be able to collect a tennis ball with a diameter in range 6.5 cm - 7.0 cm without damaging the balls	Collection mechanism can act on balls with a diameter 6-8 cm. No damage to collected balls
F.2.	The robot shall be able to unload up to 5 stored tennis balls into an 18 cm tall box	Large storage compartment can store up to 6 balls. The dumping mechanism was not reliable with loads of more than 4 balls due to torque limitations.
F.3.	The robot shall be able to autonomously explore a tennis court quadrant in under 120 seconds from within the boundaries of the quadrant, actively avoiding exiting the boundaries of the quadrant	Efficiently and reliably achieves full coverage of a tennis court quadrant within 110 seconds with no obstacles. With 2 obstacles, the robot can take up to 180 seconds. The robot exited quadrant boundaries in 2 out of 20 tests due to tennis court lines not being detected.
F.4	The robot shall be able to identify tennis balls, tennis court boundaries and a cardboard box of dimensions 60 cm x 45 cm x 16 cm with a minimum precision of 90% and recall 80% at distances of up to 6 m.	Successfully identified objects with required precision and recall up to 4 m. Objects at 4-6 m were recalled at a rate of 50 %, bringing average recall below 80%
F.5	The robot shall sustain typical functionality (F.1-F.4) for over 30 minutes.	A fully-charged battery powers the robot for 60 minutes of typical use. In stress testing, the drive motors operated continuously for 30 minutes off battery power with charge remaining. There were no instances of overheating in typical use or stress testing.
	Non-Functional Requirements	
N.1.	The robot shall have a modular design, allowing seamless component replacement to alter/improve performance	Large size of the robot allows for flexibility in subsystem size and arrangement. All electrical components are mounted via screws such that they are easily removable

N.2.	The total cost for all components in a single robot shall be less that $400	Total cost of $323.40
N.3	All components on the robot shall be easily accessible for repairs or replacement, with the entire assembly and disassembly of the robot taking one person under 45 minutes to complete without the use of power tools.	Large components, such as the bucket, can be easily removed in under 3 minutes allowing access to the electronics. The robot can be fully disassembled or assembled by a familiar team-member in under 40 minutes.
N.4	The robot shall have a footprint smaller than 50 cm x 48 cm x 38cm (L x W x H) to ensure stowability and portability	Final dimensions of 60 cm x 39 cm x 38 cm
N.5	The robot shall weigh less than 2.5 kg to allow easy portability and safe, one-person operation	Final weight of 2.45 kg
N.6	The fully-loaded robot shall have a maximum speed of 1.0 m/s, be able to accelerate from stationary to 1.0 m/s in under 0.5 seconds, and be able to decelerate from 1.0 m/s to stationary in under 0.5 seconds.	Maximum tested speed of 1.2 m/s with acceleration to this speed from stationary taking 0.45 seconds. Braking from 1.0 m/s took 0.35 seconds.
	Interface Requirements	
I.1.	The robot shall use standard connectors to permit interchangeability	All electronics use a standard header pin connection to interface with the power supply and Raspberry Pi. Appropriate wiring colours used.
I.2	The battery shall supply a voltage above 14 V throughout its entire discharge cycle in order maintain consistent motor characteristics	Above 18 V for the entirety of the discharge cycle.
I.3	The combined current draw from all Raspberry Pi GPIO pins shall remain under 50 mA	Pi powers only the encoders and IR sensors, drawing negligible current
I.4	The combined current draw from servos shall remain under 3 A at any one time in order to remain within voltage regulator specifications	Stall current of the bucket servo is 3 A, causing a potentially unreliable interface with the voltage regulator
I.5	Each wheel motor shall draw no more than 2 A to remain within motor driver specifications	1.2 A per motor while under load
	Safety and Regulatory Compliance	
S.1	The robot shall not damage the tennis court that it is operating on	Rubber wheels and castor balls did not leave any marks on any testing surface
S.2	The robot shall be able to safely navigate around the tennis court, stopping if an unexpected obstacle is within 5 cm of the robot's direction of travel.	Infrared sensors were unreliable in full sunlight, preventing the obstacle collision avoidance in this environment. Navigation algorithm was adjusted to operate with more caution in order to avoid obstacles.
S.3	Electronics shall be developed and tested in accordance with IEEE Std 1573-2021[2]	The design and testing of electronics followed this standard

3.0 DESIGN OVERVIEW

3.1 KEY INNOVATIONS AND DESIGN OVERVIEW

The final design consists of a paddle mechanism for collecting tennis balls and a dump-truck-like collection bucket for storing and tipping balls into the designated box, all housed within a sturdy acrylic-based frame. The electronic circuitry complies with IEEE Std 1573-2021 standards, and is also contained within the frame. The on-board computing is conducted entirely on a Raspberry Pi 4b, with vision provided by a webcam. Several design and process innovations were implemented to address the physical and non-physical requirements set out by the client. Some of these key innovations include:

- Low production costs for structural materials, despite custom designs
- High quality camera that performs well in various lighting conditions and court environments
- Large ball collection capacity whilst maintaining low weight
- Paddle mechanism that collects tennis balls of varying sizes

By incorporating these key innovations into a well-designed and aesthetically pleasing form factor, our robot aims to not only meet but exceed client and market expectations at an exceptionally affordable price point.

3.2 OVERVIEW OF SELECTED CONCEPT

3.2.1 Subsystems

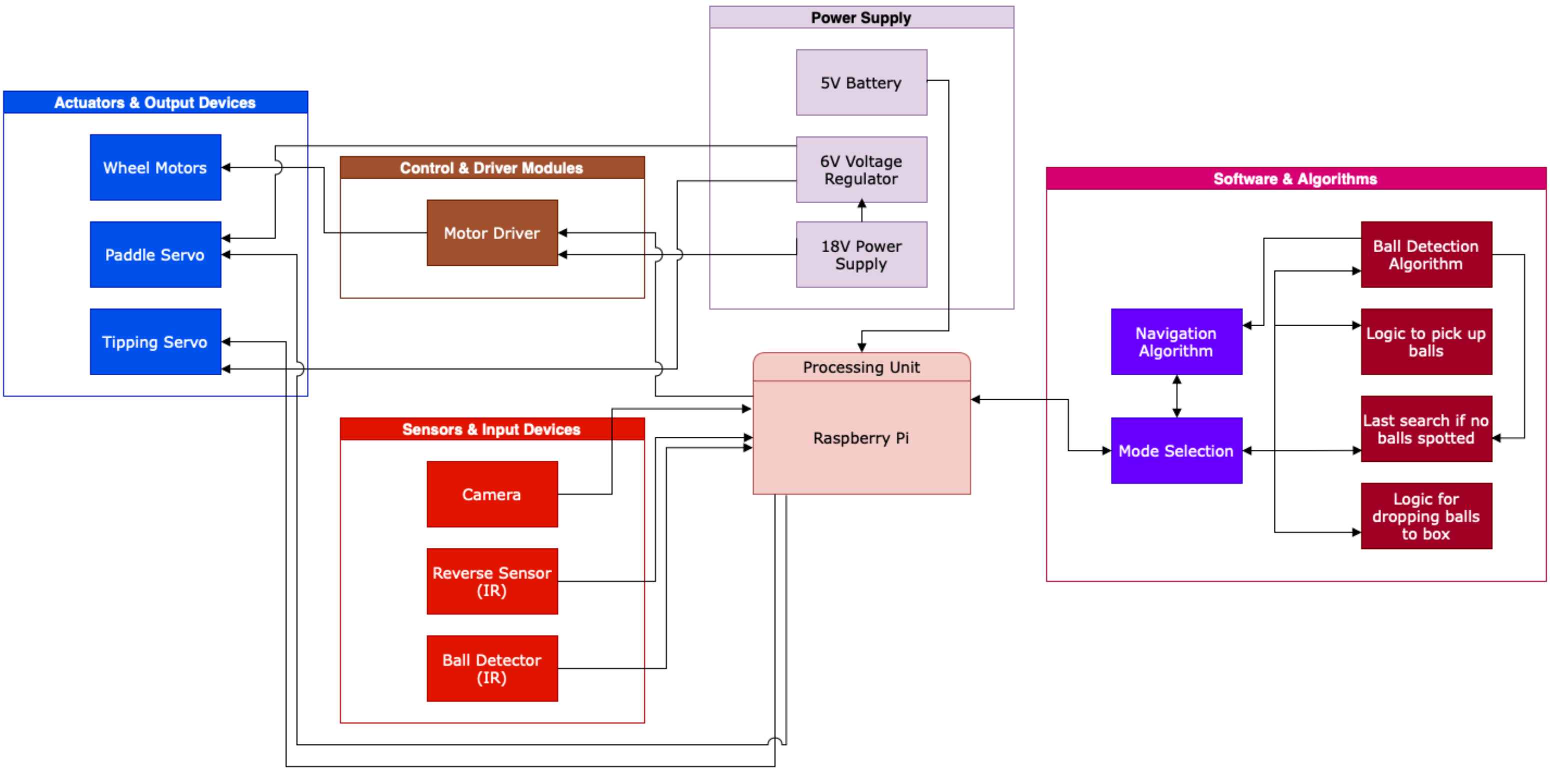


Figure 1: Functional block diagram of subsystems included in the final robot design

This section discusses the functional architecture of our tennis ball collecting robot, simplifying the operation and interaction of its core components that integrate to solve the given project task efficiently. The functional block diagram of our design shows the operation flow from sensory input through actuator responses that is handled by a centralised processing unit. Brief explanation of each segment within our system:

1. Sensors and Input Devices:

- Camera: Positioned to provide a frontal view, this sensor inputs visual data to detect tennis balls and other objects within a set field of vision.
- Reverse Sensors (Infrared): These sensors measure the distance to objects and determine the location of the collection box.

- Ball Detector (Infrared): It is crucial for triggering the paddle servo when the ball is within reach.

2. Processing Unit:

- Raspberry Pi: Acts as the central control system of our robot by processing input signals from various sensors. It runs the software algorithms responsible for decision-making processes such as navigation and ball collection strategies.

3. Control & Driver Modules:

- Motor Driver: Receives commands from the Raspberry Pi and controls the wheel motors and facilitates movement and steering of the robot.

4. Actuators & Output Devices:

- Wheel Motors: Drive the robot across the tennis court.
- Paddle Servo: Operates the paddle mechanism for collecting tennis balls.
- Bucket Servo: Controls the tipping action of the bucket for unloading collected balls into the designated box.

5. Power Supply:

- 5V Battery: Specifically caters to the Raspberry Pi by providing it constant uninterrupted power. Provides the primary power source for the robot's motor and servo systems.
- 18V Power Supply: Provides the primary power source for the robot's motor.
- 6V Voltage Regulator: Stepped down from the 18V Power supply to feed the servo systems.

6. Software & Algorithms:

- Ball Detection Algorithm: Analyzes visual data to identify and locate tennis balls.
- Navigation Algorithm: Processes sensor data to navigate the robot around the court efficiently.
- Mode Selection: Allows switching between different operational modes, such as active collection, searching, or dumping of the balls.

The integration of these components ensures our robot can autonomously navigate a tennis court, identify and collect tennis balls, and deliver them to a specified location without human intervention. The operational efficiency and autonomous features were tested and refined through multiple development cycles, ensuring robust performance under varying real-world conditions. This strategic integration aligns with our goal to enhance the training experience by reducing downtime and physical strain for players.

3.2.2 Sustainability

In the final design of our tennis ball collecting robot, we have addressed the ethical and security concerns alongside sustainability. All components used in the robot are sourced from trusted organizations, ensuring ethical procurement and high standards of quality. These components are designed to be reusable across both prototype and final versions of the robot, promoting sustainability and reducing waste. To safeguard privacy, our robot's camera, which is crucial for the operation of detecting tennis balls and determining their location, does not store any images or information. This ensures that no data about the environment or individuals present is retained.

Aligning with the UN's Sustainable Development Goal 12 for Responsible Consumption and Production, our design principles steer clear of environmentally harmful mass production methods. The robot is built to be cost-effective, enhancing accessibility for a broad range of users from individual enthusiasts to larger educational institutions. Throughout the development lifecycle, from initial testing to the final prototype, we are committed to refining the robot's efficiency and sustainability. These enhancements focus on leveraging clean and affordable energy solutions, ensuring our robot operates effectively while minimizing its environmental footprint.

3.3 SYSTEM INTEGRATION

The robot can be described in a total of four subsystems which include the camera system, motor control, servos and actuators, and supplementary sensors. Each of these subsystems need to be controlled, or used as inputs to the Raspberry Pi.

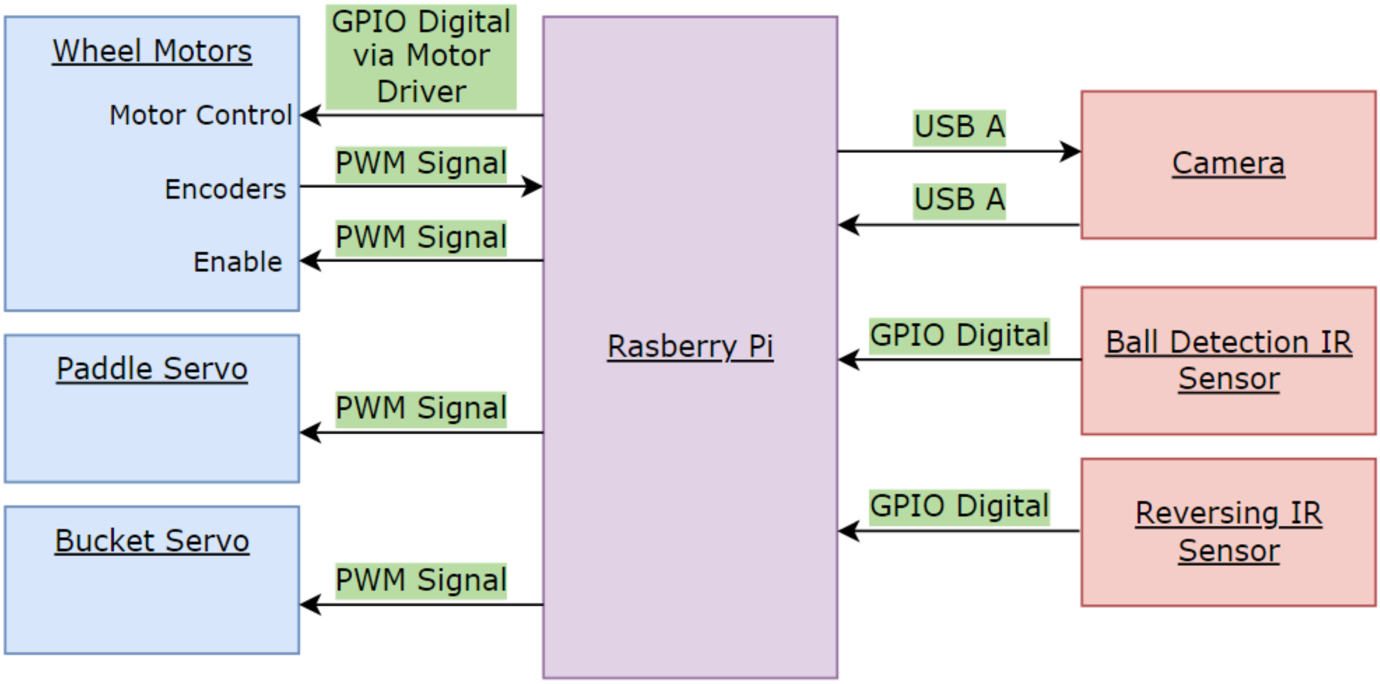


Figure 2: Interconnection block diagram of all components used in the final design of the robot and their communication channels

The wheel motors have a total of six pinouts, five of which are used in the final robot design.

- **Motor Positive and Motor Negative pins:** These digital signals are connected to the L298N motor driver, which connects to the Raspberry Pi. The motor driver amplifies the two Raspberry Pi digital signal output pins to control the motor's direction, which allows the robot to move freely within the court.
- **Motor Enable pin:** This is connected to a GPIO pin on the Raspberry Pi that outputs PWM signals which controls the motor's speed.
- **Encoder A pin:** This pin outputs a PWM signal that indicates the speed and direction of the motor, which the Raspberry Pi reads and utilises in its navigation calculations.

Both the paddle and bucket servos use the same type of communication via the same single signal pins provided. A PWM signal sent from the Raspberry Pi which controls their speed, and desired movement angle.

The infrared sensors also communicate similarly to each other. They act purely as sensory inputs to the Raspberry Pi, and use a single GPIO pin to send a digital "high" signal when an object is detected within the adjustable threshold distance. This threshold can be set using a small flathead screwdriver, much like adjusting a potentiometer.

The camera connects to the Raspberry Pi via a USB A cable, which is simply plugged into the provided port on the Raspberry Pi. All communication and control between the camera and Raspberry Pi are managed through software.

A more detailed excel sheet containing exact pin assignments can be found in the google drive. For power distribution, the veroboard diagram in the appendix shows how each subsystem connects to its respective power rails.

3.4 DETAILED OVERVIEW OF SUBSYSTEMS

The overarching design problem has been broken down into smaller tasks, with each subsystem contributing to their completion. Integrating these subsystems with the achieved deliverables ultimately solves the overarching design problem given. The key tasks we have chosen to focus on include: robot navigation, robot movement, tennis ball pickup, and tennis ball delivery.

3.4.1 Camera System for Robot Navigation

The camera mounted on the top of the robot at a 25° downward angle, is a crucial input to the Raspberry Pi in the robot's design. It serves as the robot's primary vision system, capturing images after each motor movement is completed. These images are sent to the Raspberry Pi, where they are processed by the software to help the robot navigate the court, determining its position, rotation, and distance.

The camera is straightforward to integrate and delivers high-resolution images, enabling efficient and accurate navigation analysis by the software. However, a limitation of this setup is that the camera must initialize when the robot is powered on. If this connection fails, the robot will be unable to navigate the court, as it will not be able to calculate or send movement and rotation commands to the wheel motors.

The camera can be easily substituted and does not require a specific model to interface with the Raspberry Pi. As long as the chosen camera is able to interface to the Raspberry Pi via one of the available ports—USB 2, USB 3, or the flex cable connection—it will be able to function as intended with the robot and its navigation. This flexibility supports the UN Sustainable Development Goal for responsible consumption and production, as it eliminates the need for mass-producing a single camera type, reducing waste by allowing for easy substitutions. Regarding ethical and security concerns, the camera and software only store images for debugging purposes. Additionally, the camera is always angled downward, ensuring it never captures anyone's identity.

3.4.2 Motor Control for Robot Movement

The robot features two wheel motors located at the front, beneath the base plate. At the back, caster wheels are positioned at each corner to support the robot's weight. The wheel motors can operate independently, simultaneously, and can rotate in opposite directions. This design enables the robot to pivot around a central point. For our robot, this central point is where the camera is mounted, allowing the camera to rotate directly on its axis. As a result, calculations for distance, rotation, and location within the tennis court quadrant do not need to be adjusted based on changes in the camera's position during rotation.

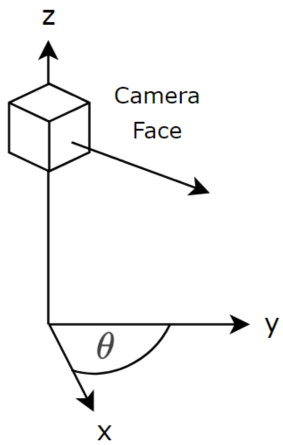


Figure 3: Visualization of the robot pivot point aligning with the camera position

In our final design, we utilize the motors and their encoders to move the robot forward to a specified position, rotate around a pivot point, and reverse into the box to deposit the collected balls. The Raspberry Pi controls the adjustable speed, allowing us to select an optimal pace for the most accurate encoder readings and effective navigation around the court. As the robot approaches the box, we use this speed control to gradually reverse into it, stopping the motors when it encounters resistance from the box.

Since the motors are solely responsible for robot movement, it's essential that the final design incorporates robust, reliable, and efficient motors. To achieve this requirement, we have selected a metal DC geared motor with a 251 RPM rating and a required supply voltage of 12V. This motor, featuring internal metal gears, produces a higher torque than the motors included in the kit, making it a stronger, heavy-duty, and dependable option for robot movement. Additionally, the motor encoders we chose are significantly more reliable and accurate compared to those provided with the kit, enhancing energy efficiency, complying with the UN Sustainable Development Goal for Responsible Consumption.

As discussed in Section 3.3, the motors are controlled through the motor driver connected to the Raspberry Pi. This setup provides current and voltage protection, ensuring safe motor operation and reducing the risk of stalling or damaging components, while also amplifying control signals from the Raspberry Pi to the motors.

Each motor draws 350 mA, and requires 12V to operate, with a 5V encoder operating rating that can be easily supplied with the chosen power supply and distribution circuit. Additionally, each motor weighs 205g, contributing to the counterweight needed at the front of the robot to maintain stability when the bucket tips backward to deposit balls into the box behind it.

3.4.3 Servos and Actuators for Tennis Ball Pickup and Delivery

The robot is equipped with two servos. The first servo actuates the paddle, which picks up balls and guides them up the ramp at the front of the robot into the bucket. The second, larger and more powerful servo lifts the bucket and tips it backward to deposit the balls into the box. Both servos have three pins: 5V, ground, and signal. The signal pins are connected directly to the Raspberry Pi, while the 5V and ground pins connect to the power distribution board. The Raspberry Pi sends a PWM signal to control each servo’s speed and rotation angle.

The paddle servo operates only when a ball is detected at the front of the robot, where the paddle can scoop the ball into the onboard bucket. Since the robot funnels balls into a central area, the paddle is designed to pick up just one ball at a time, reducing the overall current draw and allowing for a smaller, more cost-effective servo.

Similarly, the bucket servo activates when the software detects that five balls have been collected and the robot is positioned to deposit them into the box. The selected servo meets the increased torque requirements and effectively delivers the collected balls into the box.

3.4.4 Supplementary Sensors for Tennis Ball Pickup and Delivery

The supplementary sensors consist of two infrared sensors: one at the front of the robot to detect when a ball is in position for the paddle to activate and guide it into the bucket, and another at the back to determine when the robot is close enough to deposit balls into the box. Each sensor has 5V, ground, and signal pins, connected similarly to the servos used in the final design. The 5V and ground pins connect to the power distribution board, while the signal pin connects directly to the Raspberry Pi.

The Raspberry Pi reads the front infrared sensor only when the camera detects that a ball is within 30 cm of the robot. The back infrared sensor is only read when the robot reverses into the box. Due to this selective activation period, this subsystem becomes an energy efficient addition that operates best indoors. Since the demonstration took place on the outdoor tennis court, the infrared sensors were not utilized in the competition. This design flexibility provides a robust solution to the overarching design problem in various environments. All circuit diagrams for subsystems can be found in the Appendix C.

3.4.5 Power Management

The power for the robot is managed and distributed entirely through a single veroboard placed centrally on the robot to optimize integration with all subsystems and components that require power.

Table 2: Component to Veroboard Pinouts

Veroboard Rails	Rail Pinout Connections
12V - 18V	Power Supply V+
	Voltage Regulator In V+
	L298N Motor Driver Vin
5V	Voltage Regulator Out V+
	L298N 5V In
	Ball Detecting IR Sensor VCC
	Reversing IR Sensor VCC

3.3V	Raspberry Pi 3.3V Out
	Left Motor Encoder V+
	Right Motor Encoder V+
Ground	Power Supply Ground
	Voltage Regulator In V-
	Voltage Regulator Out V-
	L298N Motor Driver Ground
	Raspberry Pi Ground
	Motor Left Encoder Ground
	Motor Right Encoder Ground
	Ball Detecting IR Sensor Ground
	Reversing IR Sensor Ground

The power supply used in our final design was a drill battery rated for 18V, which provided sufficient voltage and current for each component in the robot. During the testing process, the power supply was regularly substituted with a 12V power supply, and a 9V battery snap. In terms of functionality, any of the aforementioned power supplies can be used in the robot design, as long as the rated current is 5A.

To step down the input voltage (12V–18V) to 5V for powering the motor driver and infrared sensors, we use a DC-DC voltage regulator (part number: XL4015). This regulator allows for adjustments to both the output voltage and maximum current via onboard potentiometers.

All connections on the veroboard, including those for the voltage regulator and wires, were soldered using lead-free solder to comply with the RoHS directive, making the design environmentally friendly and suitable for long-term use and global distribution due to international electrical safety standards. To ensure neat wiring, components with multiple wires soldered to the veroboard are braided and bundled which reduced clutter on the robot and helped with debugging and assembly.

Table 3: Motor Currents

Component	Max Tested Current Draw
Motor left and right simultaneously	800 mA (Stalls at 7A)
Paddle Servo	120 mA (Stalls at 800 mA)
Bucket Servo	300 mA (Stalls at 3A)

Since the veroboard can carry up to 100mA without solder, and up to 1A with the tracks covered in solder, it's necessary to combine the tracks to emulate a thicker busbar capable of handling the current carried in the circuit. By measuring the current draw of each component on the robot, we can calculate the required width and thickness of the veroboard tracks.

Because the solder is primarily composed of tin, copper, and silver steel, and the veroboard strips are mostly copper, we'll use copper's material capacity factor of 1.2 in our calculations. This provides some margin for safety when applying these calculations in practice.

$$Material\ Carry\ Capacity\ Factor\ for\ Copper\ =\ 1.2$$

$$Current\ Limit\ =\ 5\ A$$

$$Current\ Limit\ =\ 1.2 \cdot Width \cdot Thickness$$

$$Width \cdot Thickness\ =\ 5\ /\ 1.2\ =\ 4.167\ mm$$

The wire connections from the veroboard also need to be soldered with the required connection type in order to interface smoothly with their respective components.

Table 4: Component Connection Type

Connection Type	Wire Connection / Components
Male Connection	Wheel Motor Connections from Veroboard
	Infrared Signals from Infrared Sensor to Raspberry Pi
	Bucket Servo Connections from Veroboard
Female Connection	Wires to Raspberry Pi from Veroboard
	Paddle Servo Connections from Veroboard
	Paddle and Bucket Servo Signals to Raspberry Pi
	Infrared Connections from Veroboard
Spade Connection	Power Supply V+ and Ground Connection from Veroboard
At Least 2 cm Exposed Wire	12V - 18V and Ground Wire Connection from Veroboard to Motor Driver

This power distribution circuit fulfills the needs and requirements of the robot by providing the required voltage and current to each component via the power supply, all of which is protected with circuitry on the voltage regulator that limits the current draw to a maximum of 5A, ensuring safe and reliable operation. The veroboard with exposed solder is placed flush with the base plate which prevents any short circuiting since the board can only be contacted on the sides and top. For future additions and adjustments, the veroboard provides extra rails and holes for new components and connections to be soldered if needed. With this power distribution veroboard, the robot can operate safely and efficiently to complete all tasks for the overarching design problem.

3.4.6 Software

The code that controls the robot is written in Python (v3.11) and is designed to run efficiently on a Raspberry Pi 4b running headless Ubuntu 22.04 as the operating system. Complex external libraries, such as ROS, were avoided because of the relatively simple navigation algorithm and to provide total control over code. The low-level `pigpiod` library was chosen to control all GPIO because of its high efficiency and ability to run accurate PWM signals to control both the drive motors and servo pulses[3]. The computer vision is supported by two libraries, the popular `opencv` and `ncnn`, a lightweight library that facilitates fast inference of images. All code can be found in Appendix B.

An Object-Oriented approach (Figure 4) was used to organise the code for a number of reasons:

- Abstraction - Complex behaviour is contained in low-level objects (eg. motor, camera), allowing the high-level objects (eg. robot) to solely contain the primary logic
- Code readability and maintainability - Each code block is logically located such that developers can easily debug and modify functionality
- Testing - Thorough unit test cases and routines are paired with each object to test both low-level and high-level controls. Having the ability to easily test individual components enables efficient troubleshooting and validation

A high-level file, `main.py`, contained all logic that allowed the robot to act autonomously. The main algorithm controlled the robot through the Robot class, which maintained the state of the robot as well as provided controls for all functions that the robot could perform. The World class maintains the current state of the environment which influences decisions in the main algorithm, for example World contains pre-defined boundaries of each quadrant allowing main to navigate safely within these limits.

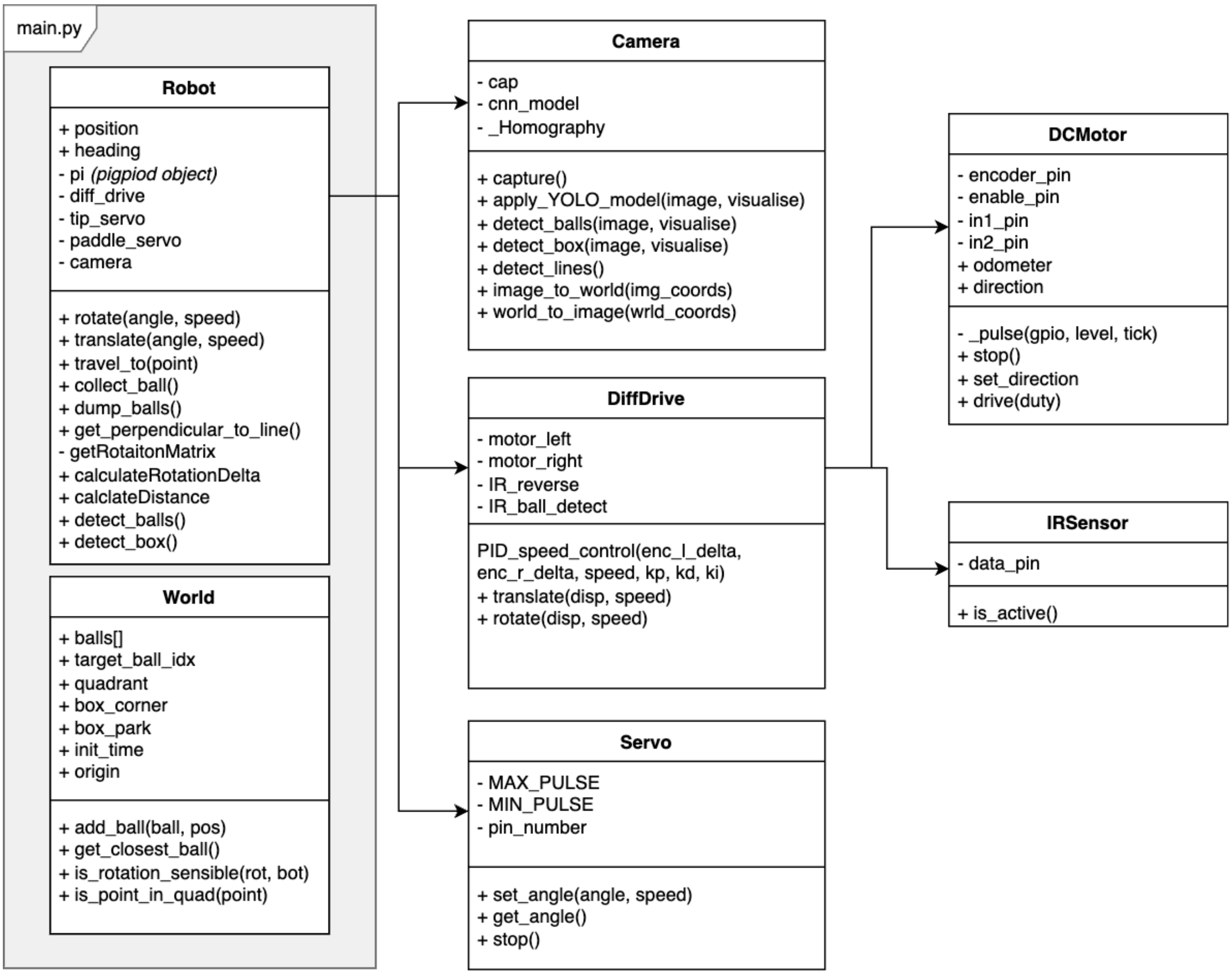


Figure 4: Class Diagram representing the structure of the code

A modern Convolutional Neural Network (CNN) approach was used to identify the distinct tennis ball and box objects. A pre-trained model, YOLO v11 nano, was chosen because of its speed, relatively small size, and modern compatibility/support structures. This was re-trained using a mixture of synthetic images (created with Blender) and diverse, real-world data gathered in varying conditions. A traditional CV approach was employed to detect lines due to the simple, predictable nature of the problem. This algorithm was designed to run extremely efficiently, analysing the gradients of just 12 single-pixel, parallel paths from the robot to a target. The identified peaks (representing blue-white edges) were combined into a single line using RANSAC. The validity of results was determined by a confidence score derived from the strength of the edge and the existence of two parallel, blue-white edges at a reasonable distance. Identified objects in the image plane were transformed to the tennis court plane using a pre-calculated homography matrix. The combined YOLO inference and line detection algorithm produced results in an average of 650 ms.

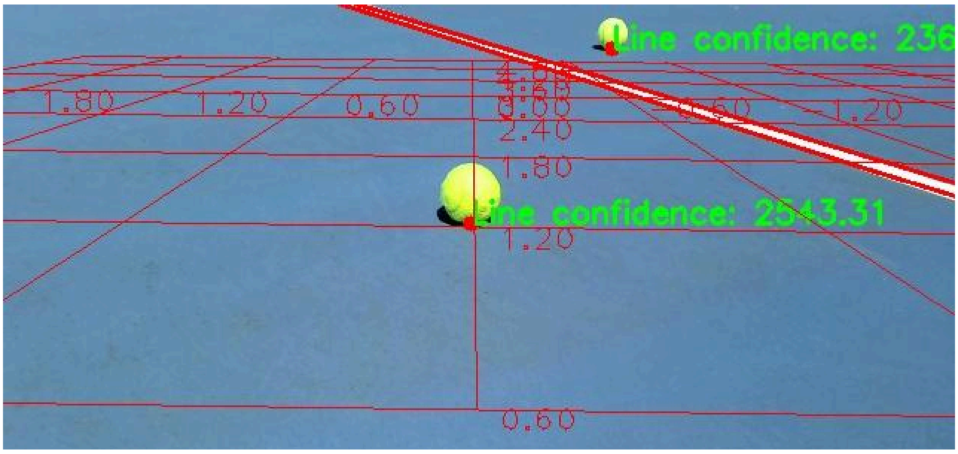


Figure 5: Output from computer vision algorithms with relative distance grid overlaid

3.4.7 Mechanical

The mechanical design philosophy is primarily based on the collection and depositing of multiple balls at once. This makes the ball storage compartment of the design the focal point of the design. The ball storage compartment is large enough to hold multiple balls at once. It is also positioned high enough to rotate and dump the balls in the box.

The holes in the bucket provide a general region for the balls to fall into and be loosely held still as they slightly sink into the gaps. This restriction of ball movement prevents the robot's movement and telemetry system from being affected by the ball's movement in an unpredictable manner. The mounting holes for the storage compartment needed to be placed accurately to ensure it was able to rotate around the correct central location. As a result of this, as well as the storage compartment's slanted angular face, it was determined that 3D printing the component was the ideal manufacturing method. This ensured that the part came out exactly as it was on CAD, and did not require any further manufacturing where other sources of error could be introduced. The storage compartment was secured to the robot by both the servo motor as well as a 3D printed structure to secure the rotational axis. The servo motor was selected to provide enough torque to rotate the storage compartment with up to 6 balls stored inside with a maximum torque of 20 kg/cm.

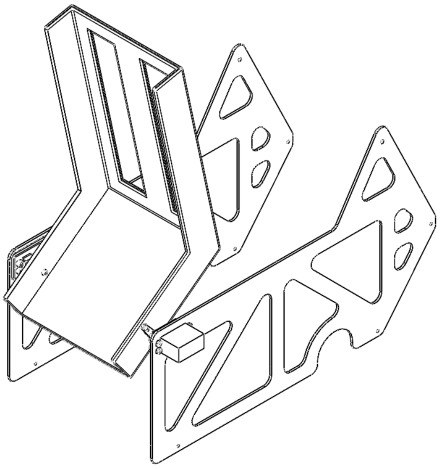


Figure 6: Mounted Bucket Mechanism - Engineering Drawing

The ramp and paddle system is designed to bring the balls up to the height of the storage compartment. The actuation of the paddle allows the balls to be pushed up to the height of the storage compartment. The motor selection for actuation of the paddle was specifically based on providing an amount of force on the tennis balls enough to push them up, but not propel them uncontrollably. Due to their complex shapes, the ramp and paddle were both manufactured using 3D printing. The paddle was rotated by the MG90-S servo, and was supported by a 3D print to ensure the paddle was secured across its rotational axis.

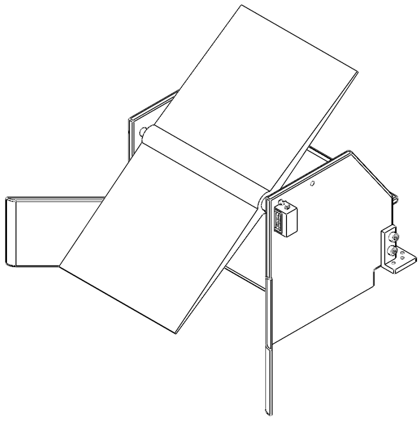


Figure 7: Drawing of Ramp and Paddle Sub-Assembly - Engineering Drawing

The frames for the robot were constructed around the position of the storage compartment, ramp and paddle systems, ensuring each component was provided adequate support for a rigid assembly. The frames were manufactured by laser cutting 3mm acrylic. Laser cutting acrylic supported the team's need for rapid prototyping, allowing new prototypes to be created within minutes. Laser cutting acrylic also allowed for precise and clean edges as well as a consistently strong and durable structure that wouldn't be prone to issues like layer separation or stress fractures that could have occurred if the frame was 3D printed. The frames were also supported by a camera mount at the front to increase the assembly's rigidity. The camera mount was designed to support the frames as well as secure the camera at a 65° from the ground as well as at a height of 30cm.

A larger base plate was manufactured by laser cutting acrylic. Due to the large number of mounting holes required, as well as the flat nature of the base plate, using acrylic was the ideal solution. A larger base plate was used to provide mounting points for all components. The large number of mounting holes were used to provide flexibility in the design during testing, allowing for structural components to be moved to optimise the robot's performance.

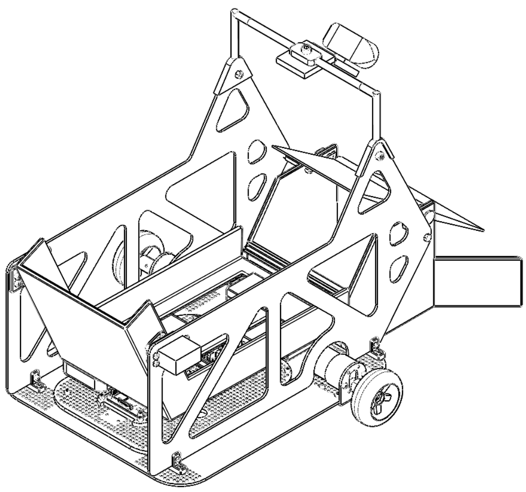


Figure 8: Fully assembled bot and its mechanisms - Engineering Drawing

A smaller base plate was used for mounting electrical components including the Raspberry Pi, L298N motor driver, both power supplies, voltage regulator and a veroboard. This smaller base plate was secured onto the larger base plate. The use of a smaller base plate allows for the further modularisation of the design, with the electrical components being mounted as a single combined assembly. This allows for easy removal of all electrical components for testing and maintenance work.

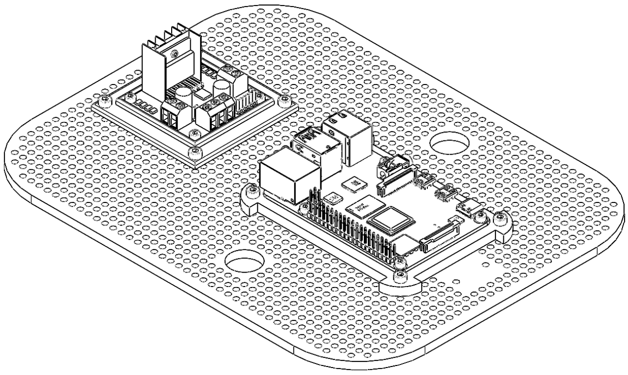


Figure 9: Drawing of electric mounting plate - Engineering Drawing

3.5 COMPETITION SPECIFIC STRATEGIES

Based on the robot demonstration brief, certain aspects of the robot were altered to ensure peak performance during the competition as well as support the robustness of the robot from both a software and hardware perspective.

The drive and the castor wheels were selected based on the location of the demonstration. The competition specifies that the robot will be demonstrated on a tennis hard-court. The wheels were selected to ensure that the robot does not slip on the surface.

From a software perspective, the robot's functional time was limited to the duration of the demonstration time. The robot was programmed to deposit all balls when nearing the end of the competition timer to ensure that all balls remaining on the robot were deposited before the end of the demonstration.

Furthermore, based on the predetermined dimensions of the competition space, the robot was pre-programmed with set boundary lengths. This allowed the robot to interpret its position on the court once detecting a line.

4.0 SYSTEM TESTING

4.1 Test Environments

Functional requirements and performance characteristics were tested initially in a controlled, indoor environment with light grey linoleum floors before outdoor testing on blue, concrete tennis courts. All testing environments were flat. The weather conditions when testing outside varied between three testing sessions:

- Test 1 (Sunday 6th October 5PM - 6PM) - Sunny, slightly windy (20-25 km/h)
- Test 2 (Monday 7th October 10AM - 12PM) - Mostly cloudy, windy (25-30 km/h)
- Test 3 (Wednesday 9th October 11AM - 1PM) - Sunny, slight breeze (10-12 km/h)

Interface Requirements were thoroughly tested in an Electrical Laboratory with professional, certified equipment. Safety and Regulatory requirements were monitored and followed across all previously mentioned test environments.

4.2 Test Methods

Collecting balls [F.1] - Balls of varying size (6.0 cm-8.0 cm) were fed to the collection mechanism and filming was used to accurately measure the total time for the collection process, starting when the paddle first actuates and ending when the ball entered the container. The state of the balls was analysed before and after the collection process to ensure no damage occurred.

Unloading balls [F.2] - The tipping mechanism was tested with all possible quantities of balls (1 to 5) to ensure it behaved predictably. Two target boxes were used, one with a height of 18 cm and the other with a height of 12 cm. After having to replace the stronger servo (20 kg cm torque) with the 5 kg cm torque servo that was available, tests were repeated.

Autonomous exploration [F.3] - The robot's exploration algorithm was tested in all 4 quadrants of the tennis court with tests running until 100% coverage of the quadrant had been achieved. The simple case of zero balls in the environment was tested 1 times per quadrant. The more complex case with balls outside of the quadrant was tested 2 times per quadrant.

Computer vision [F.4] - The YOLO Object Detection Model was tested by arranging a varying number of tennis balls and the cardboard in front of the robot (at varying distances from 0.5-6.0 m) and reviewing an output image with bounding boxes drawn around the identified objects. Both precision and recall were noted to evaluate the effectiveness of the model. Line detection was also monitored by occasionally placing objects outside of the quadrant boundaries.

Continual operation [F.5] - All three testing sessions demonstrated the battery life and cooling capabilities of the robot for well over the required 30 minutes. Drive motors were stress tested at 70% duty cycle on battery power to test continual power output.

Physical characteristics [N.2, N.3, N.5] - Weight, dimensions and cost was constantly measured throughout the design process and final results were determined after complete construction and testing of the robot.

Modularity and Maintenance [N.1, N.3] - An experienced team-member was timed disassembling and assembling the robot with the required hand tools.

Performance [N.6] - The speed and acceleration characteristics were measured in the well-defined indoor testing environment using Tracker - Video Analysis and Modelling Tool. A test run involved accelerating to a requested speed from stationary, maintaining that speed for 1.0 sec before braking. The requested speed of the motor was gradually increased until the requirement had been succeeded by 20%. Testing faster speeds was not performed due to maintaining a low risk level.

Component Connections [I.1] - The consistent approach of connecting components with braided, appropriately coloured wires with header pins was monitored throughout the design process.

Battery voltage [I.2] - Battery voltage was measured with a multimeter before and after testing sessions.

Raspberry Pi GPIO Current Draw [I.3] - Total current draw for the Pi was measured before and after connecting GPIO and averaged over 1 minute.

Servo Current Draw [I.4] - Total current draw for the 6.0 V voltage regulator powering the servos was measured from an 18 V DC power supply. Both the tipping mechanism (with 5 tennis balls in the bucket) and ball collection mechanism were tested.

Motor Current Draw [I.5] - Total current draw for the L298 motor driver powering the motors was measured from an 18 V DC power supply. Both single motor and dual motor operation were tested.

Surface Impact [S.1] - The quality of the testing floor was closely monitored throughout testing with before and after pictures taken to ensure the robot was not damaging the surface.

Collision avoidance [S.2] - Obstacle detection was first tested by planting a backpack in-front of the moving robot. After safe functionality had been demonstrated, human trials were performed. The infrared sensors were unreliable in full sunlight so these were deactivated for outdoor testing.

Standard Practice [S.1] - Relevant guidelines were followed from *IEEE Recommended Practice for Electronic Power Subsystems: Parameters, Interfaces, Elements, and Performance 1573-2021*. Both undervoltage and overvoltage protection was tested for the power circuitry with a DC power supply. Electromagnetic Compatibility (EMC) was monitored between components, with no interference observed between subsystems of the external environment.

5.0 PROJECT MANAGEMENT RETROSPECTIVE

5.1 Team Roles and Distribution of Tasks:

Joseph and Sadman were pivotal in perfecting the vision system. Starting with simple computer vision techniques, they advanced to training YOLO models, achieving over 90% accuracy in tennis ball detection.
Isabelle and Rohit ensured the integration and optimization of subsystems including sensors and actuators, contributing to the system's robustness.
Aryaman and Edric excelled in the mechanical design, focusing on the CAD modelling and assembly of the robot's chassis.

5.2 Key Accomplishments

The project culminated in a highly functional robot capable of autonomously navigating and performing tasks within a tennis court environment. The vision system was particularly successful, reliably detecting tennis balls and navigating the court. The soldering of components enhanced system robustness, reducing potential failure points. The introduction of new motors were both strong and reliable. The iterative redesign of the paddle mechanism and thorough servo testing culminated in a system that consistently collected tennis balls.
A notable failure occurred when a voltage regulator malfunctioned, damaging the metal gear servo responsible for the ball tipping mechanism. This incident not only caused a temporary setback but also highlighted the need for spare critical components. Although an alternative servo was used, its performance under full load was suboptimal, underscoring the importance of component compatibility and redundancy in design.

5.3 Critical Path to Completion

The integration of the advanced vision system with the robot's mechanical and navigation systems was critical. This integration was central to the robot's ability to operate autonomously and adaptively in a dynamic environment. Regular team meetings and agile response to challenges ensured that setbacks were quickly addressed keeping the project on track towards its successful completion. The project's success was a testament to the team's resilience and collective problem-solving capabilities. Each setback was met with a determined response leading to valuable improvements in the robot's design and functionality. This experience has not only refined our technical abilities but also strengthened our collaborative and adaptive skills essential for future engineering challenges.

6.0 FINAL BILL OF MATERIALS

Table 5: Bill of Materials

Component	Functionality enabled	Unit Cost	Quantity	Cost
Computing				
Raspberry Pi 4b	This computer is the sole on-board processing unit, used for computer vision, navigation and actuator control.	$73.80	1	$73.80
SanDisk Ultra 16GB MicroSD Card	Used to store the operating system and instructions for the Raspberry Pi. Provides ample storage for the OS and additional software needed for the robot's operation.	$6.95	1	$6.95
Heatsink for Raspberry Pi	Aluminium heatsink for cooling the Raspberry Pi.	$1.62	1	$1.62
Vision and Sensors				
HP 320 Full HD Webcam	The only vision system on the robot, used to detect tennis balls, court lines, the collection box and obstacles on the court.	$69	1	$69
Infrared Line/Object Presence Sensor	Detects presence of ball in the paddle's collection area, and the presence of the box when reversing (mainly for low-light environments).	$1.50	2	$3.00
Motors and Actuators (Including Mobility Accessories)				
12V DC Motor 251 RPM with encoder	These high torque motors are used for all movement functions of the robot.	$27.64	2	$55.28
Wear-Resistant Wheel and Tyre	Translate the rotational motion of the motors to lateral movement of the robot, while providing ample traction.	$5.945	2	$11.89
Caster Ball	These caster wheels are chosen so that they keep the robot level with the ground under all loads and speeds, while minimising rolling resistance.	$3.95	2	$7.90
MG90S 9g Metal Gear Servo Motor	This servo rotates the paddle mechanism that collects the tennis balls.	$5.15	1	$5.15
FT5320M Aluminium Servo	This servo drives the collection bucket, allowing collected balls to be tipped into the box.	$26.95	1	$26.95
Power				
12V 1.3 Ah Battery	This sealed lead acid (SLA) battery powers the actuators of the robot (motors and servos).	$26.95	1	$26.95
Portable Power Bank	The powerbank supplies power exclusively to the Raspberry Pi.	$13	1	$13
USB A to Type C Cable	Cable to connect the power bank to the Raspberry Pi.	$4.61	1	$4.61
DC-DC Adjustable Step-down Module 5A 75W	The voltage regulator is used to step down the voltage supplied to the servos from 18 V to 6.0V.	$3.55	1	$3.55

L298N Motor Driver	The driver to assist controlling the DC motors responsible for robot movement.	$5.95	1	$5.95
Toggle Switch	Switch for powering the robot on and off.	$2.90	1	$2.90
Veroboard	Provides the base for circuit components to be soldered to.	$4.90	1	$4.90
Structural				
Chassis Components (Acrylic)	The chassis components provide the physical structure and framework on which the robots subsystems are mounted.		1	
Paddle	The paddle which collects the tennis balls. 3D printed.		1	
Ramp	The ramp on which the tennis balls travel on transit to the collection bucket from the paddle. 3D printed.		1	
Collection Bucket	The storage facility where the collected tennis balls are stored before they are deposited into the box. 3D printed.		1	
Fasteners				
Fastener Kit	Used for Securing all components to robot	$27.99	1	$30
M6 Bolts	Mounting Webcam		1	
M5 Bolts	Mounting Camera Mount		2	
M5 Nuts	Mounting Camera Mount		2	
M4 Bolts	Mounting Miscellaneous Components		12	
M4 Nuts	Mounting Miscellaneous Components		12	
M3 Bolts	Mounting Miscellaneous Components		51	
M3 Nuts	Mounting Miscellaneous Components		51	
M2.5 Bolts	Mounting Raspberry Pi & Motor Driver		8	
M2.5 Nuts	Mounting Raspberry Pi & Motor Driver		8	
M2 Bolts	Mounting Paddle Servo		2	
M2 Nuts	Mounting Paddle Servo		2	
TOTAL COST		$351.39		

The revised bill of materials above details the materials used to construct the robot, along with a link to purchase the items. The total cost of construction of the robot was $351.39 This represents a cost saving of $17.85 compared to the estimated cost outlined in the proposal ($369.24). Despite the cost saving achieved, this robot utilises a more powerful computing unit than the initial design, as well as a more robust structural frame with more custom components. Some of the sources of cost savings are outlined below:

- Re-ordering the supply chain to procure materials from reliable suppliers at lower prices
- 3D printing and laser cutting structural materials to reduce costs whilst increasing customizability and durability
- Utilising larger motors with similar torque specifications, but with a far lower cost

6.0 RECOMMENDATIONS AND FUTURE WORK

Despite the mostly successful performance of our robot, there are a few areas where further improvement or additional features can be sought. Some of these improvement areas include:

- The implementation of an easily reachable emergency stop button to shut off the power supplies. This would be a useful feature if the robot is interrupted by pets or small children.
- The addition of a laser sensor for obstacle avoidance. The tennis court can be a busy place, and there is potential to drive into players, spectators or even other ball collection robots. Although our team has mostly written code for this very purpose, implementation was not covered.
- The IR sensors can be replaced with ones that offer greater adjustability and higher distance range to work in bright conditions. This adds in a layer of redundancy, so that any translation errors stemming from the motor or vision subsystem affect the ball collection and depositing mechanism to a far lesser degree.
- A stronger servo motor for the tipping mechanism can be used, so that it can comfortably tip >6 balls.
- A printed circuit board can be developed to improve the reliability, durability, compactness and weight of the circuitry.
- Additional structural reinforcement and fairings could be added to the robot, to improve rigidity and aesthetics.

Although these design improvements will increase the production cost of the robot, they will also greatly improve the functionality, aesthetics and consequently marketability of the robot, which make them a worthwhile consideration.

7.0 CONCLUSIONS

The project proved to be a mostly successful venture, achieving the majority of its intended objectives. The robot demonstrated its ability to navigate the tennis court environment efficiently and robustly, collecting tennis balls scattered across the area. This core functionality met the majority of user requirements and showcased the viability of using robotics for automating tasks in sports and recreational settings.

However, improvements must be made before this is considered a polished product. While the robot performed well in collecting tennis balls, it was unreliable in delivering large payloads to the cardboard box due. Additionally, improving the robot's safety should be prioritised, especially in outdoor environments, through upgrading sensors. Despite these minor limitations, the project can be considered a success, particularly given the tight time frame and challenging conditions the team faced. The iterative design process provided a valuable learning experience for all team members, ensuring continuous improvement of the robot and progressively aligning its performance with the project requirements.

8.0 REFERENCES

[1] "Tennibot," TenniBot. https://www.tennibot.com/

[2] "IEEE Recommended Practice for Electronic Power Subsystems: Parameters, Interfaces, Elements, and Performance," IEEE, Dec. 2021, doi: https://doi.org/10.1109/ieeestd.2022.9775772.

[3] "pigpio library," abyz.me.uk. https://abyz.me.uk/rpi/pigpio/python.html

9.0 APPENDICES

9.1 APPENDIX A (Engineering Drawings)

Engineering Drawing Drive

9.2 APPENDIX B (Code)

Code Drive

9.3 APPENDIX C (Circuit Diagrams)

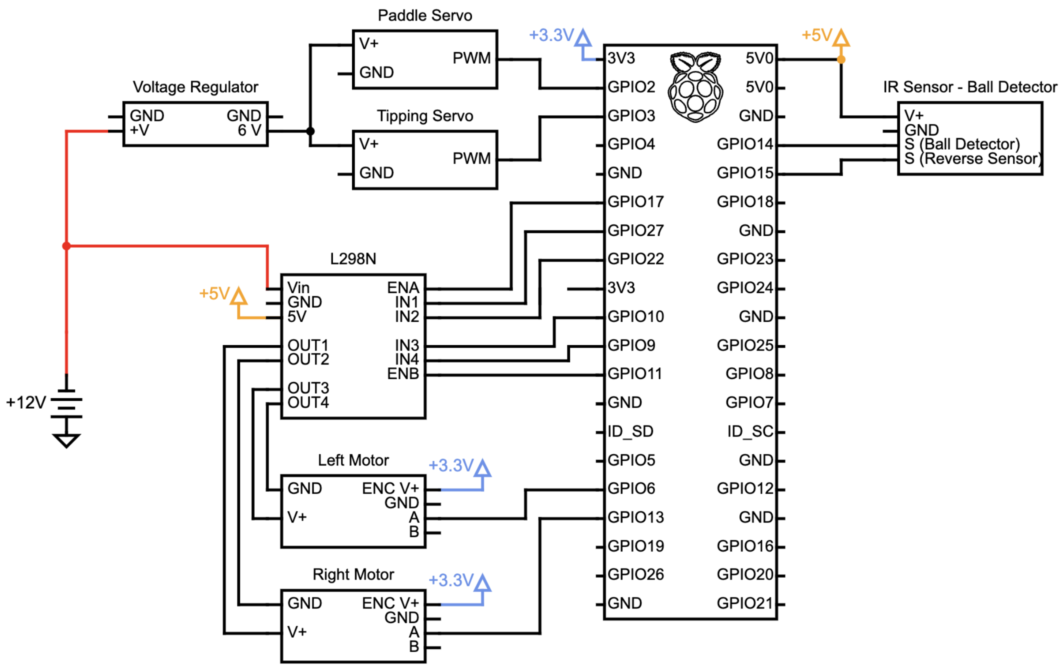


Figure 10 - Sensors and Actuators (note all component GND Pins are connected to a common ground)

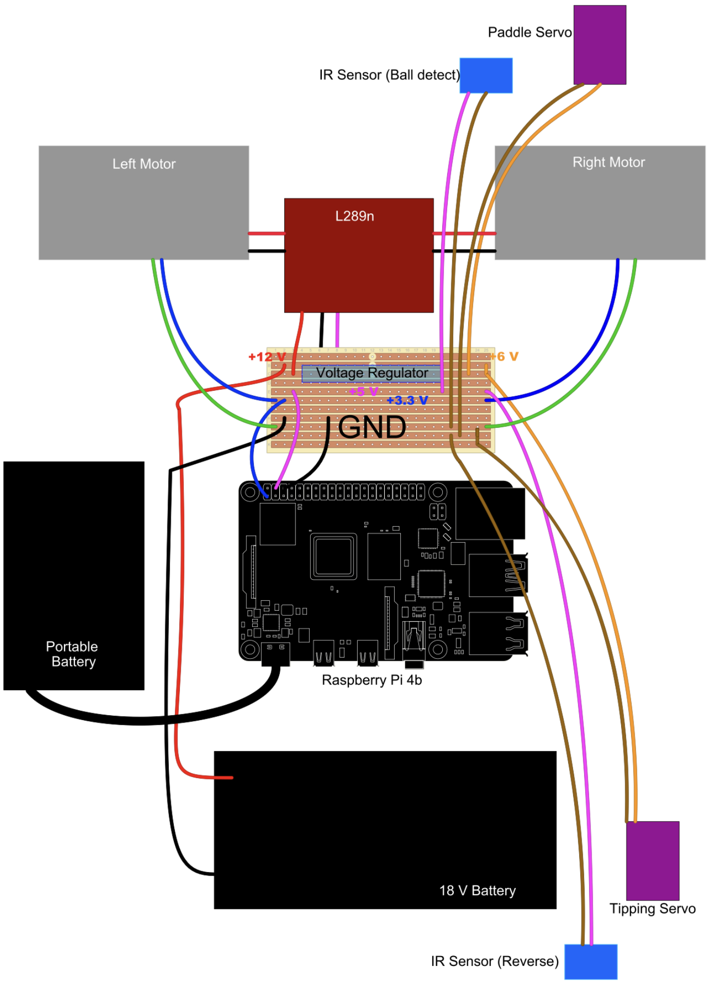


Figure 11 - Power Distribution