“TEABOT” - TEA PLANTATION PRESERVATION USING AN INTELLIGENT ROBOT : A RESEARCH

2023-044

Final Report

Illeperumarachchi Imalka Erandith Gunawardana

B.Sc. (Hons) Degree in Information Technology

Specialized in Software Engineering

Department of Computer Science and Software Engineering

Sri Lanka Institute of Information Technology

Sri Lanka

September 2023

“TEABOT” - TEA PLANTATION PRESERVATION USING AN INTELLIGENT ROBOT : A RESEARCH

2023-044

Final Report

Illeperumarachchi Imalka Erandith Gunawardana

IT19973470

B.Sc. (Hons) Degree in Information Technology

Specialized in Software Engineering

Department of Computer Science and Software Engineering

Sri Lanka Institute of Information Technology

Sri Lanka

September 2023

DECLARATION

ABSTRACT

Tea cultivation stands as one of Sri Lanka's most significant export industries, playing a vital role in the nation's GDP. Ceylon Tea, with its renowned history and global recognition, is a prized brand worldwide, holding a prominent position in international markets. Sri Lanka's favorable climate and fertile soil contribute to the distinctive taste of Ceylon Tea, cherished by consumers worldwide. Historically, the country boasted an abundant labor force capable of tending to expansive tea estates. However, as the industrial revolution reshaped the landscape, people gradually shifted toward diverse professions. Unlike many crops, tea demands meticulous care and entails substantial costs. The dwindling labor force left tea estate owners struggling to maintain their plantations, resulting in decreased yields. Some estate owners turned to cultivating alternative crops with lower maintenance requirements, significantly impacting the export market. In response to this challenge, the TeaBot emerged as an advanced autopilot robot, offering a compelling solution to replace human labor in watering and fertilizing large-scale tea estates. Among its robotic peers, TeaBot stands out for its specialized design, engineered to operate seamlessly in off-road conditions without requiring meticulously arranged paths for navigation. The robot's primary tasks revolve around efficiently watering and fertilizing tea plants, essential for sustaining continuous hydration and providing the necessary nutrients to maximize crop yield. By doing so, TeaBot enhances efficiency while reducing water and fertilizer wastage. TeaBot's impressive capabilities encompass path identification, enabling it to navigate effectively, and the ability to identify the end of tea plant stems, ensuring precise and efficient watering.

Keywords – ROS, Tea, Robot, Liquid Fertilization, PID, Motor Controlling, Raspberry Pi, I2C, Arduino

TABLE OF CONTENTS

DECLARATION i
ABSTRACT i
LIST OF FIGURES iii
LIST OF TABLES iv
LIST OF ABBREVIATIONS v
1 INTRODUCTION 1
1.1 Background and Literature Survey 1
1.2 Research Gap 2
1.3 Research Problem 3
2 OBJECTIVES 4
2.1 Main Objective 4
2.2 Specific Objectives 5
2.2.1 Development of the robot controller (Hardware) 5
2.2.2 Development of the robot controller (Autopilot) 7
2.2.3 Development of the robot controller (Manual) 8
2.2.4 Hazard Identification .... **Error! Bookmark not defined.**
3 METHODOLOGY 9
3.1 Requirement Analysis 9
3.2 Feasibility Study 9
3.3 Implementation 10
3.4 Work Breakdown Structure 33
3.5 Testing 33
3.6 Gannt Chart 34
4 DESCRIPTION OF PERSONNEL & FACILITIES 36
5 BUDGET & BUDGET JUSTIFICATION 37
6 REFERENCES 38
7 APPENDICES 40
7.1 Plagiarism Report 40

LIST OF FIGURES

Figure 2-1: Development of robot chassis (Side View).. 5
Figure 2-2: Development of robot chassis (Top View) 5
Figure 2-3: Robot Controller Overview... 6
Figure 2-4: Controller autonomous and manual 6
Figure 2-5: System Overview Diagram 7
Figure 3-1: Work Breakdown Structure... 33
Figure 3-2: Gantt Chart... 34

LIST OF TABLES

Table 1-1: Robots Comparison 3
Table 5-1: Estimated Budget.. 37

LIST OF ABBREVIATIONS

ROS	Robotics Operating System
2WD	Two wheel drive
4WD	Four wheel drive
2WS	Two wheel steer
4WS	Four wheel steer
LKR	Lankan Rupees
PID	Proportional–Integral–Derivative
DC	Direct Current
GDP	Gross Domestic Product

1 INTRODUCTION

1.1 Background and Literature Survey

Tea is a vital export product that significantly contributes to Sri Lanka's GDP. Ceylon Tea, with its rich history, is renowned worldwide as a well-known beverage. However, the industrial revolution led to a decline in laborers available for tea plantations, causing challenges for estate owners in maintaining their tea crops. To address this issue, the TeaBot, an autopilot robot, was introduced to replace human resources in the large-scale tea estates. Unlike other robots, TeaBot specializes in operating in off-road conditions, eliminating the need for clear paths for navigation. Its primary functions include watering and fertilizing tea plants, ensuring they receive consistent hydration and nutrition while improving efficiency and reducing water and fertilizer wastage.

The navigation system is a critical aspect of TeaBot. Unlike other robots, it can identify tracks and navigate between tea plants in challenging off-road conditions without the need for highly arranged, leveled paths. The robot employs a PID controller for precise navigation adjustments and sensors to measure angles and speed. The robot's controller operates on top of ROS using a Raspberry Pi minicomputer and incorporates hazard identification to enable emergency stops, safeguarding the robot's hardware components and notifying administrators.

Agricultural robots come in various sizes and configurations, such as four-wheel drive mobile robots with in-wheel motors, Arduino-powered robots, and those designed for specific tasks like strawberry harvesting and crop plucking. Some are suitable for flat terrain, while others are designed for rough terrain or greenhouse applications. Many of these robots employ PID control and advanced technology, such as Intel NUC or Raspberry Pi, to enhance their performance and efficiency in agricultural tasks.

In addition to TeaBot, various other agricultural robots are available with unique capabilities. TurtleBot3 employs LiDAR sensors and a created map for navigation, using ROS for obstacle avoidance[1]. DC motors, controlled by a Raspberry Pi, are equipped with soil moisture sensors for automatic irrigation, PIR sensors, and rain sensors[2]. Autonomous Mobile Robots operate on rails, running on ROS[3]. Four-wheel and front-wheel steering robots utilize GPS and have a weight of 150kg[4]. Trajectory tracking systems with skid-steering robots employ PID control, Raspberry Pi, and Wi-Fi connectivity[5]. BoniRob navigates row-based cultures using sensors, LiDAR, Wi-Fi, and an Intel Atom embedded PC[6]. RagriBot specializes in row-type seed sowing, powered by solar energy[7]. A sizable agricultural robot designed for autonomous row-type agricultural tasks[8]. This four-wheel-drive agricultural mobile robot features in-wheel motors, non-gear mechanisms, and four-wheel steering[9]. AgriBot, equipped with four-wheel steering (4WS), is controlled by an Arduino and boasts a large size, with a slightly elevated center of gravity[10]. The Thorvald II is a four-wheel steering robot designed specifically for strawberry harvesting in agricultural settings[11]. A four-wheel steering robot with center arms dedicated to efficiently plucking crops[12]. This four-wheel agricultural robot is GPS-enabled, enhancing its navigation capabilities[13]. An autonomous seed-sowing agricultural robot, utilizing Arduino technology, suitable for flat road applications[14]. A four-wheel-drive mobile robotic platform designed for weed monitoring, featuring front-wheel steering, and particularly well-suited for greenhouse environments[15]. An agricultural robot suitable for both flat and rough terrain, It can operate autonomously in greenhouses with front-wheel steering (FWS) and rear-wheel steering (RWS), utilizing a four-wheel steering (4WS) configuration[16]. A PID-controlled four-wheel-drive greenhouse agricultural robot The Agri robot Crop Phenotyping, powered by an Intel NUC, specializes in crop analysis and classification[18]. A solar-powered autonomous agricultural robot designed for crop monitoring and material handling[19]. Agricultural robots employing Arduino Uno and Raspberry Pi for various applications in agriculture[20].

The summarization of prior studies.

1) **Computer :** Raspberry Pi, Arduino, Intel NUC
2) **Wheels Mechanism :** Skid steering, Four wheel drive (4W), two wheel drive (2W), four wheel steering (4WS) and two wheel steering (2WS)
3) **Wheels :** 6 to 10 inches flat and rough terrain rubber wheels
4) **Programming Languages :** Python, C++
5) **Power Supply :** Solar, engine driven, battery powered

1.2 Research Gap

Previous studies have shown that robotic controllers vary depending on their intended purposes. Many robots excel in greenhouse environments, where power management and wheel operation are optimized for flat terrain. On the other hand, 2W drive systems are ill-suited for rough terrain due to their increased power requirements for off-road use. For navigating challenging landscapes, the 4W drive configuration proves to be the most compatible choice. When maneuvering in confined spaces, 2WS (two-wheel steering) systems can be somewhat challenging. In contrast, 4WS (four-wheel steering) requires additional motor power and a mechanism to rotate all wheels but provides better control and maneuverability. On sloped terrains, the 4WS mechanism ensures stable horizontal movement without sliding. Alternatively, robots equipped with wide V-shaped wheels can navigate slopes effectively without horizontal slippage. For simplicity in turning, skid steering is an attractive option, eliminating the need for complex mechanical designs. Regarding power sources, solar panels are most suitable for flat terrains with light power demands. However, they are not practical for rough terrain or areas with significant tree cover. In terms of computing capabilities, Arduino is limited to motor control, sensor reading, and basic data retrieval from a cloud. In contrast, using ROS (Robot Operating System) with Python enables robots to perform more advanced processing, including image analysis. A complete robot system can be integrated as a package, allowing it to perform a wide range of tasks efficiently using a Raspberry Pi for processing.

Table 1-1: Robots Comparison

Robot	Wheels Mechanism	Power Supply	Computer	Scale	Eligibility for tea fields
Thorvald	4WD, 4WS	12V Li battery (High power)	Dedicated robot control unit	Extra Large (200 kg)	✗
Ragribot	4WD, Skid steering	Solar Power	Raspberry Pi + Arduino	Small	✗
Agribot	4WD, 4WS	12V (High power)	Dedicated robot control unit	Extra Large (150 kg)	✗
BoniRob	4WD, 4WS	12V (High power)	Intel Atom	Extra Large	✗
TeaBot	4WD, Skid steering	12V (Low power)	Raspberry Pi (Python / ROS)	Fair (2 x 2 feet) (30 kg)	✓

The proposed solution for TeaBot involves equipping it with the capability to navigate across all types of terrain. To achieve this, the robot will be fitted with robust, hard-threaded and wide wheels. These wheels will be part of a 4W drive system that also incorporates skid steering for precise control. Powering the four wheels will be four motors, and the gear system will utilize a sprocket chain gear mechanism to maximize torque. To fine-tune its movements, TeaBot will rely on the PID control method. For its operations, TeaBot will be equipped with a 12V, 45A battery designed for power efficiency. Power-saving methods will be employed to maximize the robot's operational duration. The computational processing will be handled by ROS (Robot Operating System) using Python. To keep costs low and ensure sufficient processing power while minimizing energy consumption, a Raspberry Pi will serve as the computer. Overall, TeaBot is being designed with affordability in mind and will be well-suited for all-terrain navigation with the aforementioned mechanisms in place.

1.3 Research Problem

Tea growers and researchers have undertaken numerous experiments in an effort to create machinery capable of managing tea estates with fewer human workers. However, many of these tools are designed for single tasks only. In contrast, in some other countries, large-scale robots have been developed to handle tasks such as watering, fertilizing, and crop harvesting. The challenge lies in the fact that these robots often struggle to adapt to various soil and land conditions, requiring expensive modifications for their operation. To address this issue, this research project adopts a specific methodology. The central research problem at hand is how to enhance the TeaBot's ability to navigate efficiently in off-road environments and optimize its power usage. This research problem comprises several distinct subsections, each focusing on a particular aspect of the solution..

1. How to navigate the TeaBot in the correct path without a well-organized track. For an instance, the tea planters no need to make fully leveled tracks or add any additional track to navigate the robot.
2. How to control left and right wheels by measuring angles and the speed to maintain a constant speed and navigate smoothly.
3. How to stop the robot at the path end.
4. How to read coordinates which provided by the computer vision to navigate the robot in the track.
5. How to design the robot chassis to easily move and consume a low power to increase the power saving.

2 OBJECTIVES

2.1 Main Objective

The TeaBot is a specialized robot designed to address challenges within the tea industry, particularly the scarcity of human resources. Its primary mission is to efficiently perform various tasks required for tea plant maintenance, ensuring optimal crop yields. To achieve this, the robot employs computer vision-based path identification to autonomously navigate among tea plants. The maintenance procedure involves precisely delivering liquid fertilizer to the base of each plant stem, ensuring proper hydration and NPK saturation. TeaBot accomplishes this task using pressurized water nozzles, which target the stem's end. By integrating computer vision technology, the robot determines the stem's coordinates, allowing the robot controller to adjust the water nozzles in real-time to achieve precise fertilizer application. TeaBot continuously strives to maintain a consistent speed to ensure even fertilizer distribution. TeaBot offers two navigation methods. The first is automated navigation, where computer vision provides coordinates to guide the robot. The robot controller processes these coordinates, incorporating sensor data on speed and driving angles to maintain a steady pace. This automated process propels the robot between rows of tea plants. TeaBot also incorporates hazard identification and path end detection systems to safeguard its operation. For instance, if the water hose becomes stuck or the robot encounters an obstacle, it promptly halts and alerts the administrator. The second navigation method is manual and remote-controlled. Administrators can use an app-based remote controller that includes live camera views to guide the robot manually, allowing them to steer it out of predefined tracks or address specific operational needs.

2.2 Specific Objectives

These four objectives must be met to fulfil the above-mentioned main goal.

2.2.1 Development of the robot controller (Hardware)

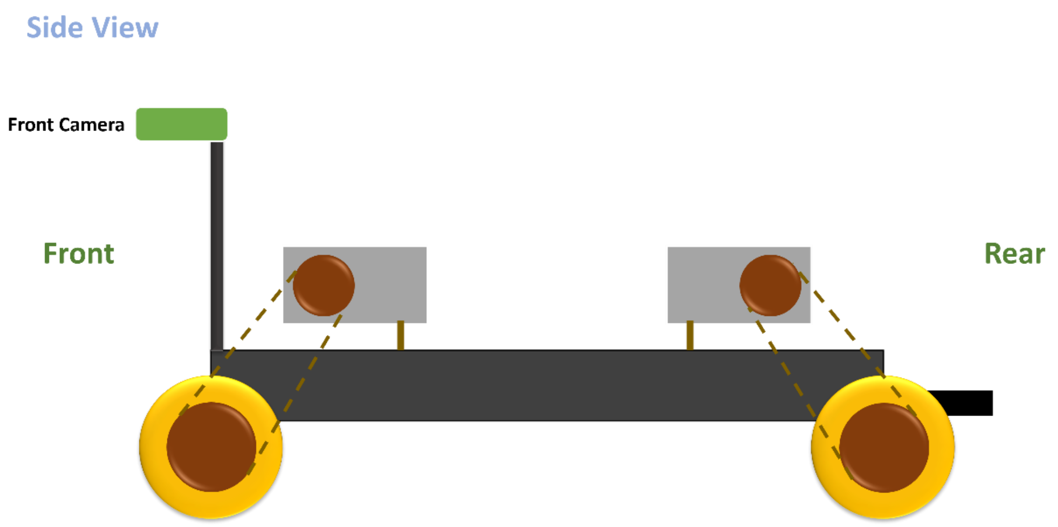


Figure 2-1: Development of robot chassis (Side View)

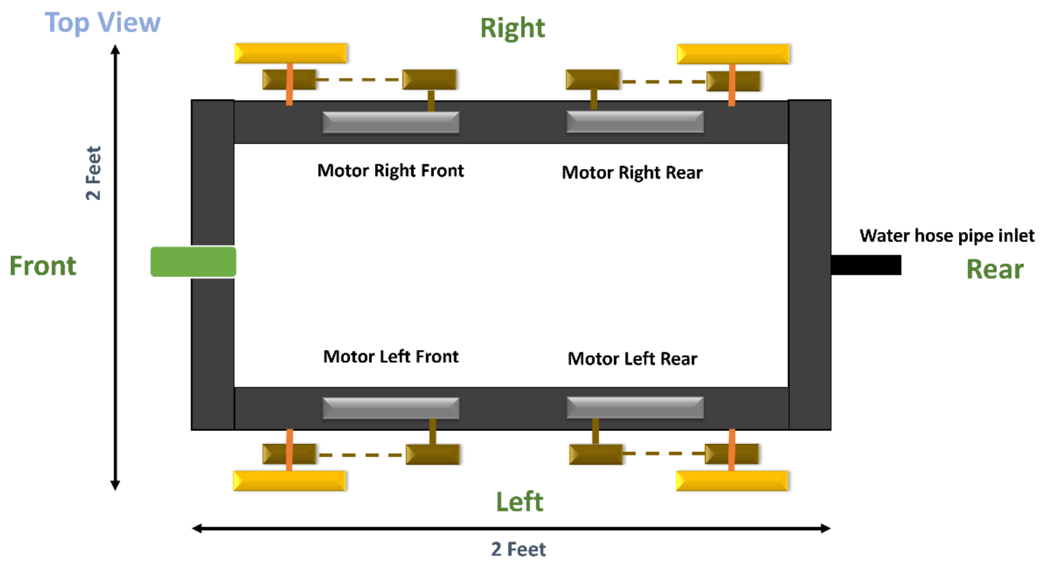


Figure 2-2: Development of robot chassis (Top View)

The robot is constructed with an iron frame that incorporates four eight-inch wheels, ensuring a stable and rigid structure with consistent ground clearance. All four wheels are equipped with motorized power, with the front and rear wheels arranged in parallel, both on the left and right sides. This configuration of all four wheels being motor-powered significantly enhances the robot's torque and overall power, making it well-suited for navigating off-road terrain. The rear section of the robot is dedicated to carrying the water hose. To efficiently transfer power to the wheels, a sprocket and chain system is employed, ensuring an appropriate gear ratio that enhances torque for effective movement.

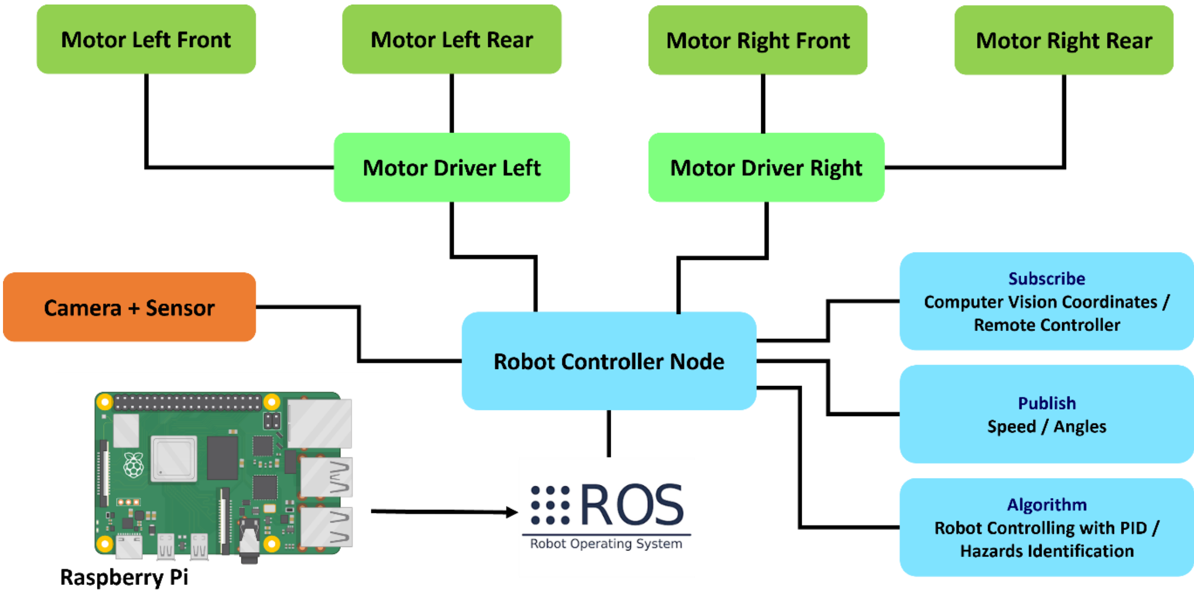


Figure 2-3: Robot Controller Overview

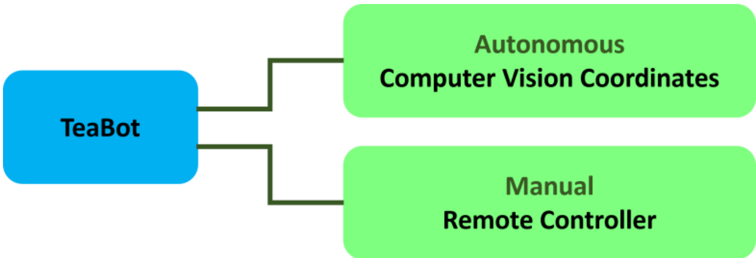


Figure 2-4: Controller autonomous and manual

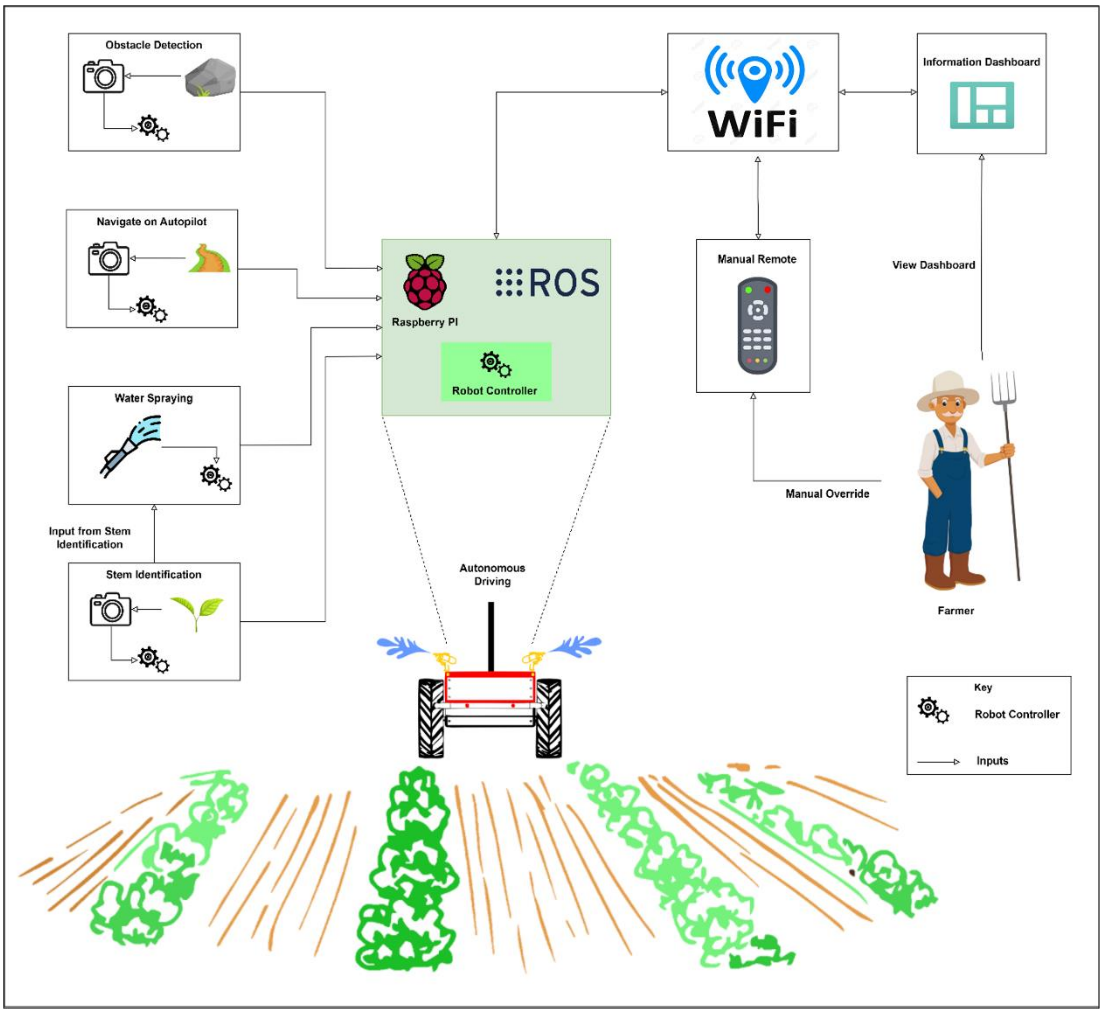


Figure 2-5: System Overview Diagram

2.2.2 Development of the robot controller (Autopilot)

The robot's wheel movements are directed by the provided coordinates from computer vision data, as well as sensor inputs regarding the robot's speed and driving angle. Utilizing its computer vision capabilities, the robot can recognize the pathway between rows of tea plants, enabling it to autonomously navigate. It seamlessly transitions to new paths, continuing to self-drive effectively (Figure 2-5). To ensure precise navigation adjustments. The robot can identify the actual path and the four motors are operating to navigate the robot to correct its path. ******* . The path navigation provides the actual path (red line), current path (black line) and the difference of the x direction coordinates of both lines (deviation). Then the robot navigates to a proper angle to coincide the two lines to correct its navigation. Another option is the red line following method. That method is to follow a red colored stick by the computer vision. There are pros and cons on both features.

Tea Row Path Navigation

Advantages	Disadvantages
• Best solution for large scale tea estates. • No need any special arrangements to the paths.	• Should reduce the background noises. • Should maintain the path more clear.

Red Stick Navigation

Advantages	Disadvantages
• Best solution for medium scale tea estates. • No need any special arrangements to the paths.	• Need to mark every end of the path. • Difficult to mark in curved paths.

If the tea rows are not clear and the distance of paths are less than 200m. The red stick navigation option is the better option. This is a low cost solution. The user only needs to fix a red colored stick at the end of the path. The robot can follow the path and the computer vision real time check the width of the red stick to identify the end of the path. If the path is greater than 200m, the width and the height of the stick should be increased.

2.2.3 Development of the robot controller (Manual)

For a self-driving robot, it's essential to incorporate a manual control method for navigation. This allows the administrator to take control when necessary, whether to guide the robot to a different section of tea plants or respond to an emergency situation. The TeaBot is equipped with a software application-driven remote controller, complete with camera views, enabling manual operation of the robot. (Figure 2-5). The remote controller can be switched the robot to either manual or auto mode by the user. The user can set a speed and start the robot to navigate.

2.2.4 Path end detection

The robot is dragging the water hose and drive forward. If the hose is getting stuck on somewhere the robot should immediately stop and notify the hazard to the administrator. If the robot is facing to an end of a path, the robot controller will stop the motor controller. (Figure 2-5).

3 METHODOLOGY

TeaBot will be developed using the following methods to provide an effective product to the tea industry.

3.1 Requirement Analysis

Through a meticulous process involving interviews and on-site visits, we diligently gathered valuable insights into the prevailing challenges within tea plantations across the country. Our dedicated efforts unveiled a range of unresolved issues, with one predominant concern taking center stage: the scarcity of labor resources and the inefficient utilization of fertilizer and water resources. In our pursuit of comprehensive solutions, we embarked on a rigorous examination of tea plant requirements. This investigation encompassed diverse factors, including land conditions, soil quality, the ongoing growth status of crops, and the potential yield achievable through optimal maintenance practices. Furthermore, we conducted an extensive survey to assess land conditions, seeking cost-effective strategies to establish pathways conducive to the robot's mobility. Our research also delved deeply into the liquid fertilization process, taking into account soil conditions during both dry and wet seasons. The data meticulously collected during these comprehensive surveys played a pivotal role in shaping the development of the robot controller. This enabled us to tailor its functionalities precisely to effectively address the identified challenges.

3.2 Feasibility Study

Many tea estates span vast expanses of land, often surpassing a single acre in size. In specific regions, network connectivity proves to be unreliable, presenting challenges for the TeaBot's real-time computational operations. Regardless of whether the network quality is stellar or less than ideal, it is essential for the TeaBot to possess the capability to perform these operations autonomously on its onboard computer. In areas with deficient network coverage, data transmission and reception may require an alternative method, such as radio waves. This approach becomes particularly vital in cases where establishing a dependable network infrastructure is economically unfeasible due to associated costs. Furthermore, in regions with limited connectivity, the quality of available roads for the robot's navigation often falls below the desired standard. Fully rectifying these road conditions to perfection is frequently cost-prohibitive. Consequently, the robot's design must prioritize effective operation on these uneven and less-than-ideal road surfaces.

3.3 Implementation

The robot design is a combination of hardware and software.

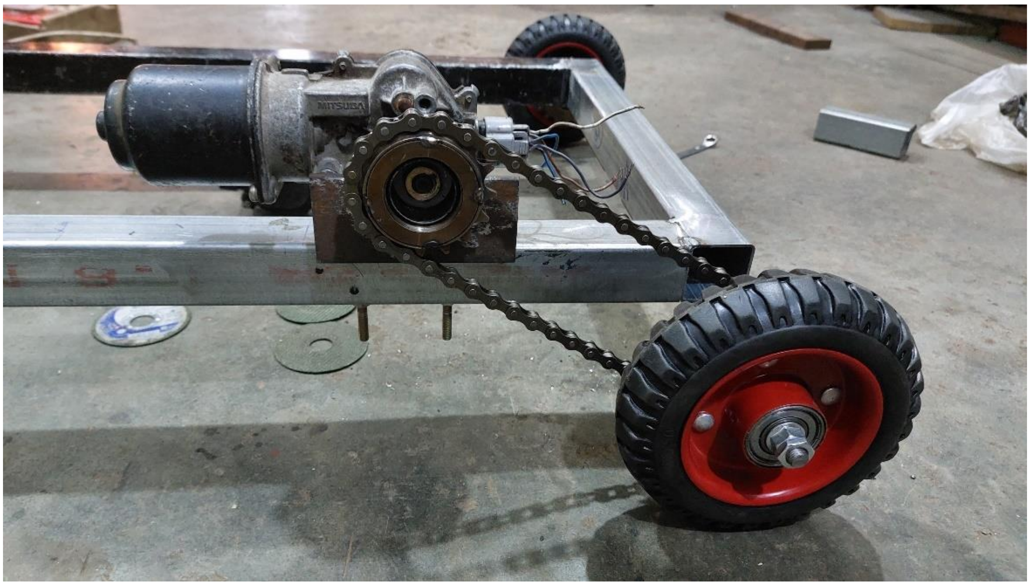
3.3.1 Hardware

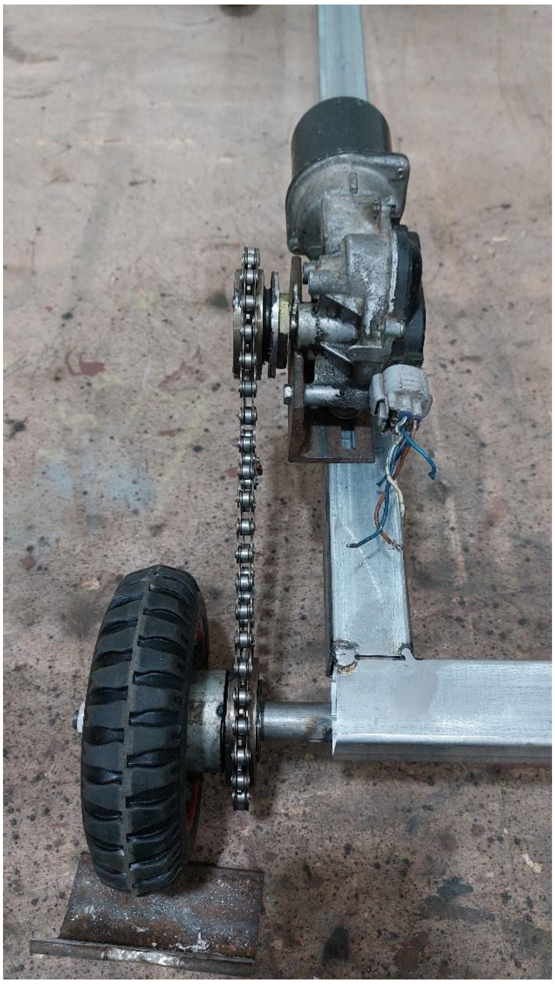
The robot's chassis demands a resilient and durable mechanical structure, a crucial requirement given its operation in demanding off-road conditions. Its chassis measures 21 x 28 inches, providing a stable base for the robot's integral components. Constructed from iron, the chassis incorporates four 8-inch wheels with versatile functionality. These wheels not only maintain a consistent body height for the robot but also ensure sufficient ground clearance. Their robust treads are instrumental in minimizing wheel slippage on diverse soil types. The chassis itself has a total weight of 20kg, carefully selected to maximize friction and enhance the robot's traction on challenging terrain. Each of the four wheels is independently powered by 12V motors, with the front and rear wheels aligned in parallel on both the left and right sides. This arrangement of all four motor-powered wheels substantially bolsters the robot's torque and overall power, equipping it for navigating the most demanding off-road environments. At the rear section of the robot, there is a dedicated platform for carrying the water hose, ensuring that essential equipment for plant maintenance remains easily accessible during operation. To efficiently transfer power to the wheels, a sprocket and chain system is implemented, maintaining an optimal gear ratio that further amplifies torque for effective movement.

The front camera is located at a best place to capture the front view without capturing the objects of the robot chassis.

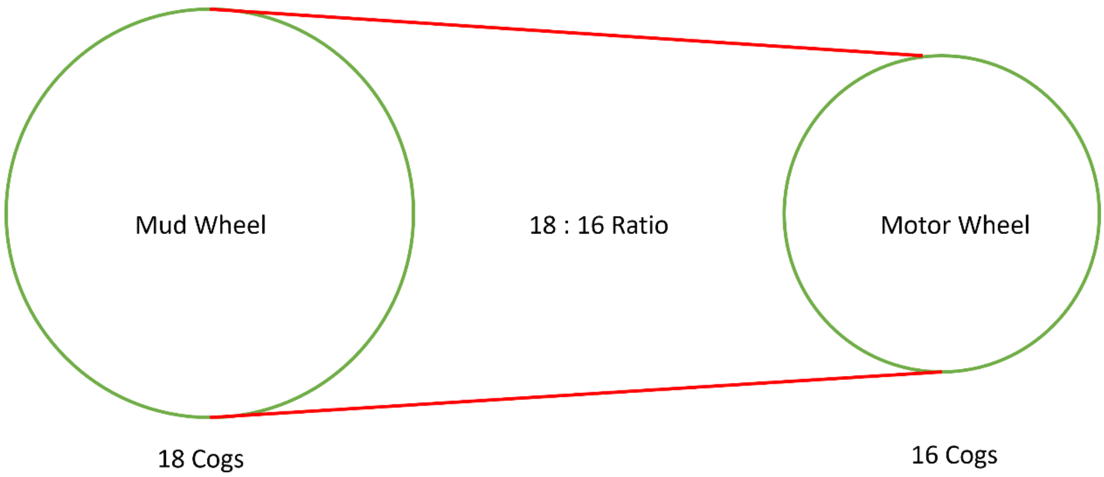
The chassis is 2.5 X 2 feet box bar structure. The wheels are grooved wheels. The motors are located at the four corners of the chassis. The power is transmitted through the chain and sprocket system. The electronic components box is located in front the robot, the water tank is located in the middle of the robot and the battery units are located at back of the robot. The cameras a re located on top of the robot, since the cameras should capture freely. The camera mount is adjustable to adjust the height of the cameras. The water sprays are located at the latter part of left and right sides of the chassis to protect the other components with the water.

The chain sprocket system





The wheels and the motors are having two sprockets. The wheel is having 18 cogs and the motor wheel is having 16 cogs for the power transmission.



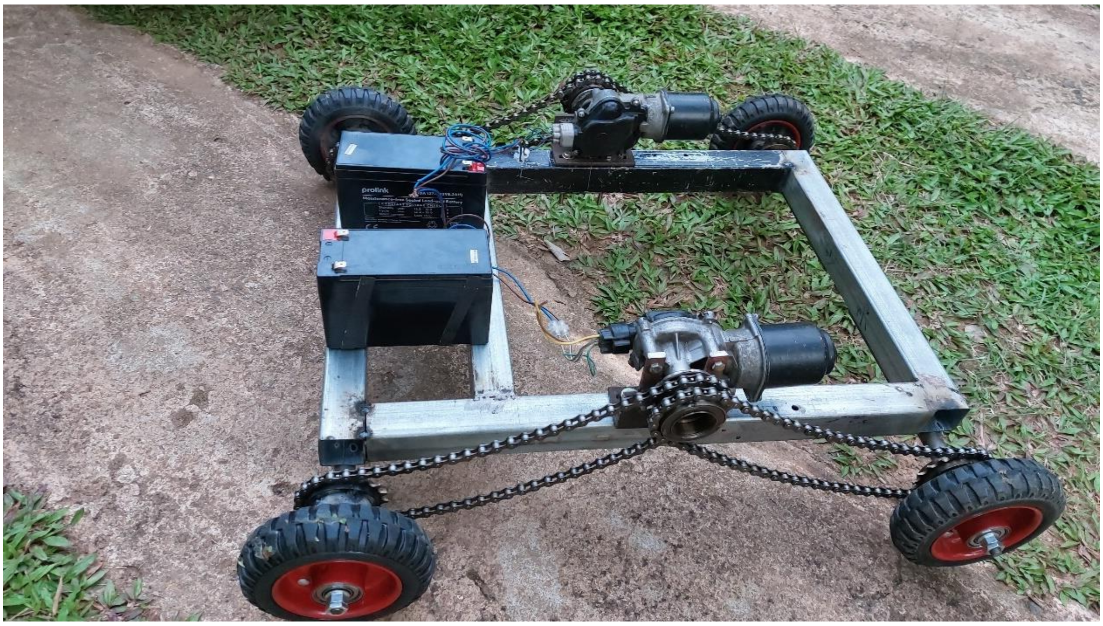
This gear ratio is used to carry a heavy weight even all terrains without getting stuck the motors. Increasing the gear ratio is helps to relax the load to motors during the power transmission. This setup no needs a gear box since the robot is driving in a constant speed. If gear box is used the mechanical design may be complexed. But the chain and the sprocket system are the most ideal for this setup since it is light weight, and the mechanical design is not complex. The belt and sprocket system are also compatible for this version, but the problem is, the belt can slip on wheels when taking a higher torque. But the chain does not slip on wheels since chain is locked with the cogs in sprockets.

The motors are fuse protected, since the motors are not locking because of their high power. But the motors should protect if any overload. The motors are protected with fuses for overload protection. The wheels can be changed to rubber or mud wheels easily by simply removing a nut. This hardware setup can drive more than 3 hours continously with a 12V 35Ah battery.

Water tank

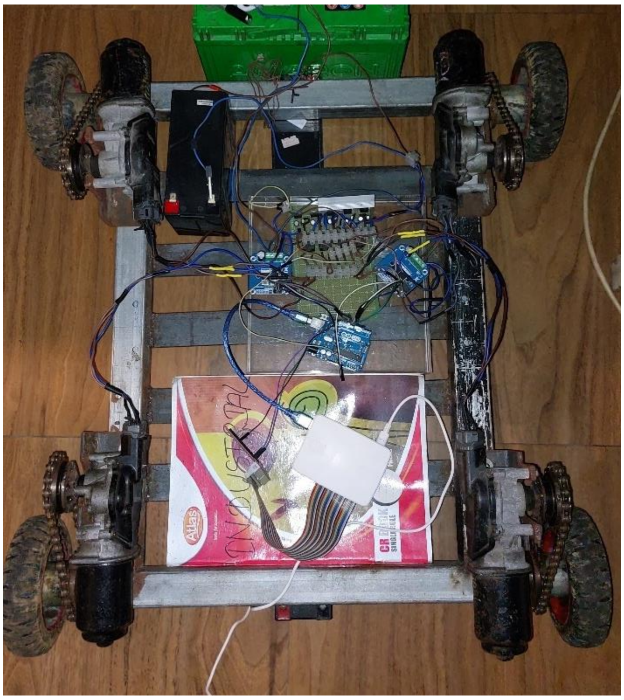
The water tank is used to maintain the water pumps pressure consistent. If the water supply to the robot is not constant, the pumps cannot maintain a constant pressure.

Two motors powered transmission.



The previous power transmission used only two motors to drive the robot. But power was not sufficient to drive the robot. Another problem was, the chains cannot adjust with that setup.

Four motors powered transmission.



All four powered motors can drive the wheels without any power issue. The motors can easily adjust and can quickly transmit the power to the wheels. Individual motors are responsible for respective wheels. One motor driver is used to power two motors and during the testing, the robot could able drive in a hard terrain.

3.3.2 Software

The motors are under the control of dedicated motor drivers, each motor demanding a power supply of 10 amperes. These motor drivers possess the capability to manage power loads of up to 43 amperes and are directly linked to the Raspberry Pi. To drive the motors, a 45 ampere battery is employed, while the Raspberry Pi operates autonomously with its 12V battery and a power regulator to maintain consistent performance. In the realm of software operations, the Robotics Operating System (ROS) is utilized, and all programming is executed using Python. ROS operates on a modular architecture, employing separate nodes that adhere to the publisher-subscriber model, facilitating seamless data exchange among various nodes through topics. For motor control, the Raspberry Pi communicates with an Arduino via Serial. The robot encompasses a range of components, including motors, cameras, pressure pumps, and water nozzles, all centrally coordinated by the Raspberry Pi. Exploiting its computational prowess, the Raspberry Pi is capable of executing Python libraries to perform complex computer vision tasks. This comprehensive software package functions as an integrated system, streamlining the deployment process for new robots.

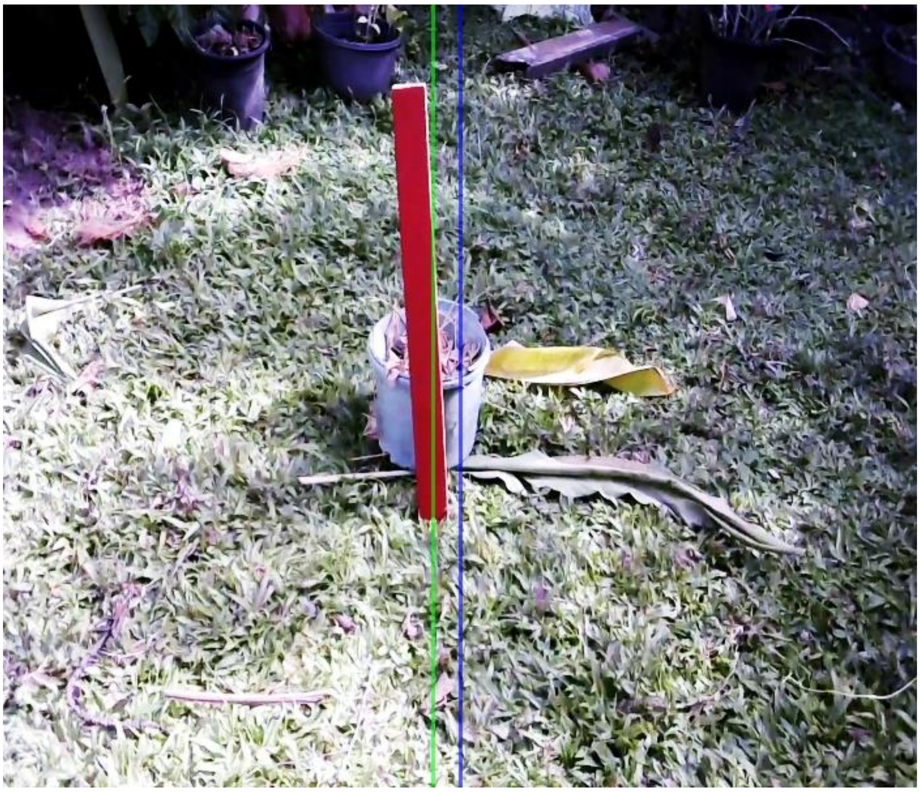
3.3.3 Path Navigation (Tea Row)

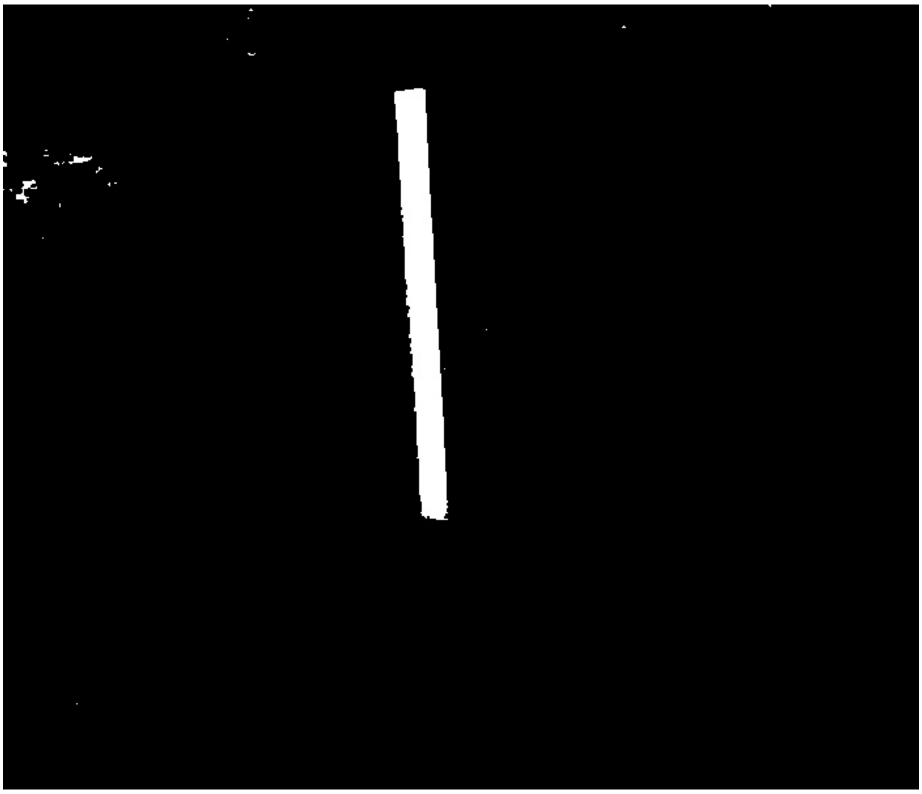
In the tea row navigation. The computer vision can identify the actual path and it is marked as a red line on the picture.

The blue line is denoted as the current path of the robot and the red line is the actual path of the tea row. Then the algorithm is computing the difference of the deviation and coincide the blue and red line by controlling the speeds of the left and right wheels of the robot.

3.3.3 Path navigation and end of the path (Red Stick)

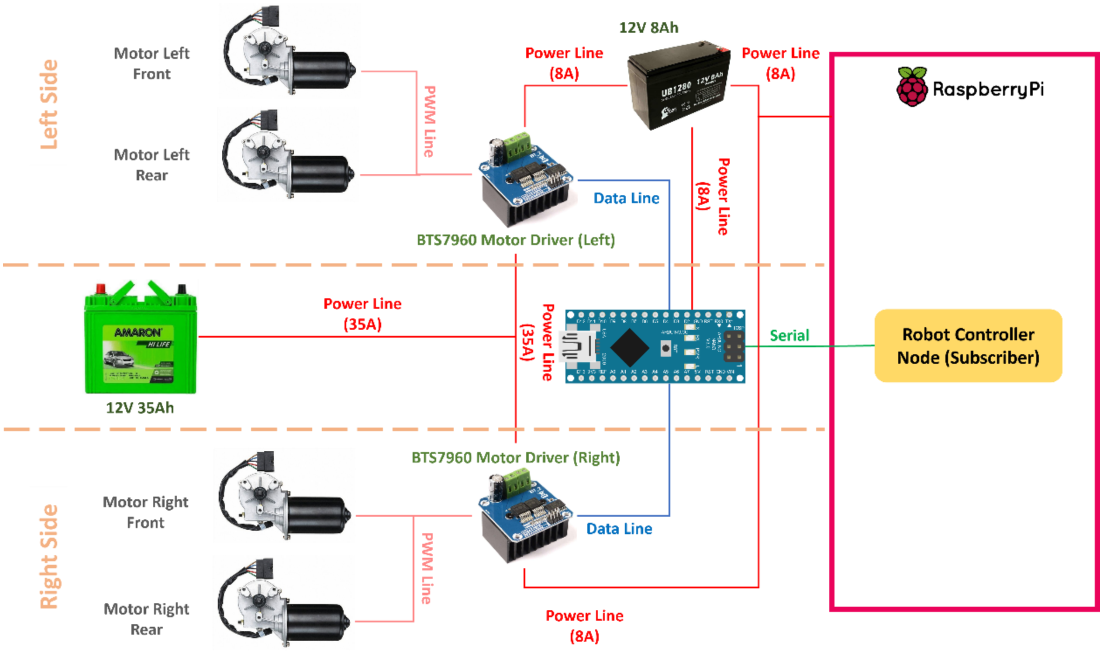
For end detection and path navigation, a red colored stick located at the end of the path, then using computer vision, the robot can stop at the end of the path by calculating the red color intensity of the stick.





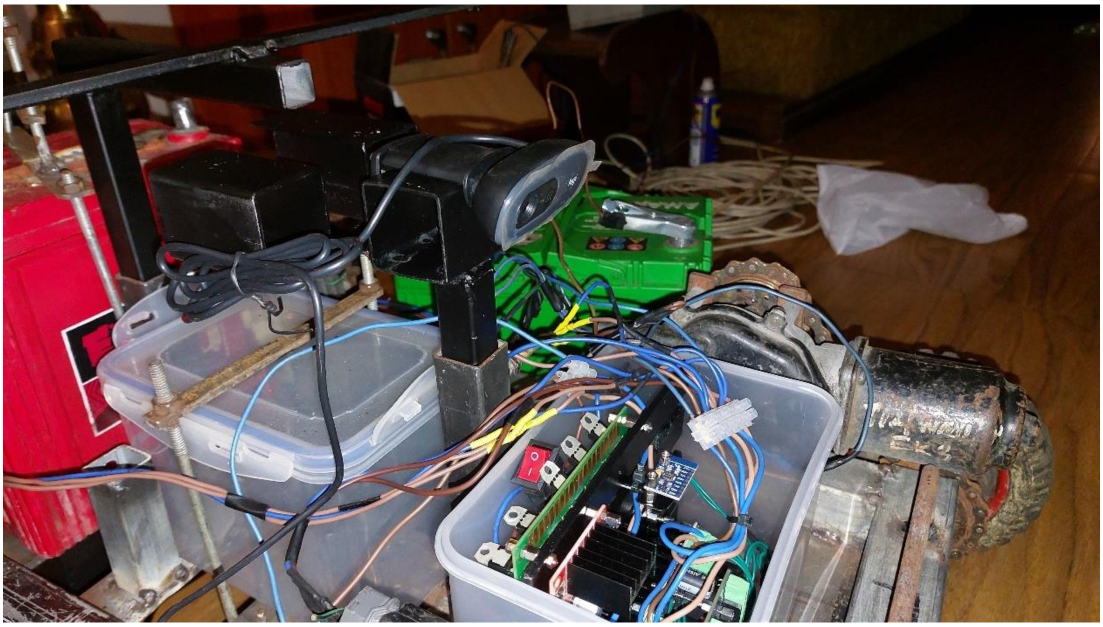
The specialty of selecting a red colored stick. Tea row is saturated with green and light brown colors. The red is a unique color for that environment. The robot can easily identify the red stick uniquely without the background noises. When robot coming towards the stick. The width of the stick is getting increase. Then robot can stop when the width of the stick coming to a limited amount. The algorithm is computing the difference of the deviation and coincide the blue and green line by controlling the speeds of the left and right wheels of the robot. Then calculate the width of the red stick in this scenario. In a highly green saturated environment, the robot can see the red stick clearer. This helps to make readings more clearer and the motors can operate more efficiently.

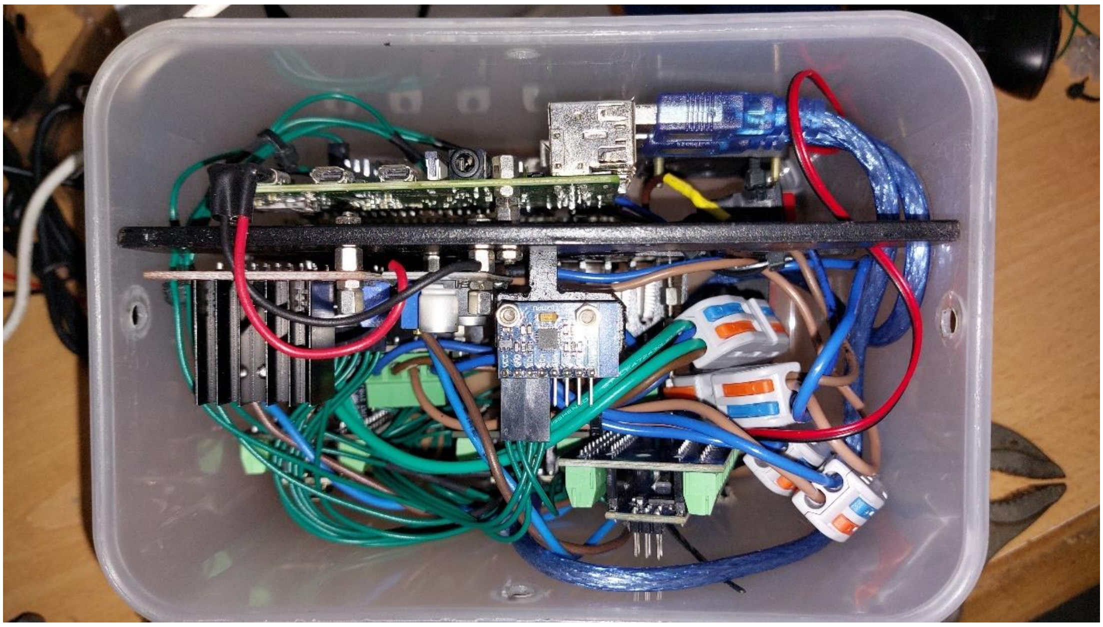
3.3.3 Motor Controller Diagram

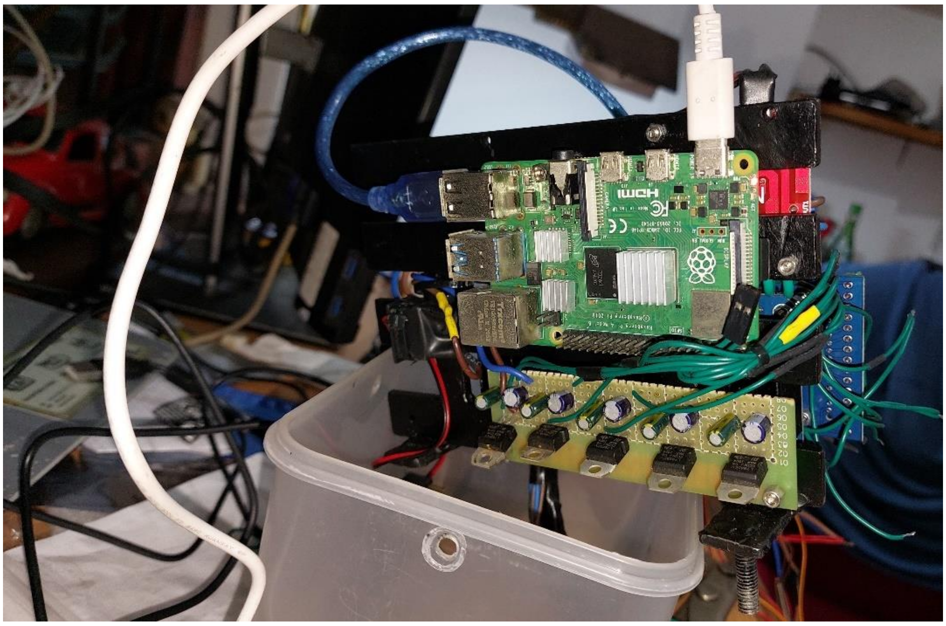


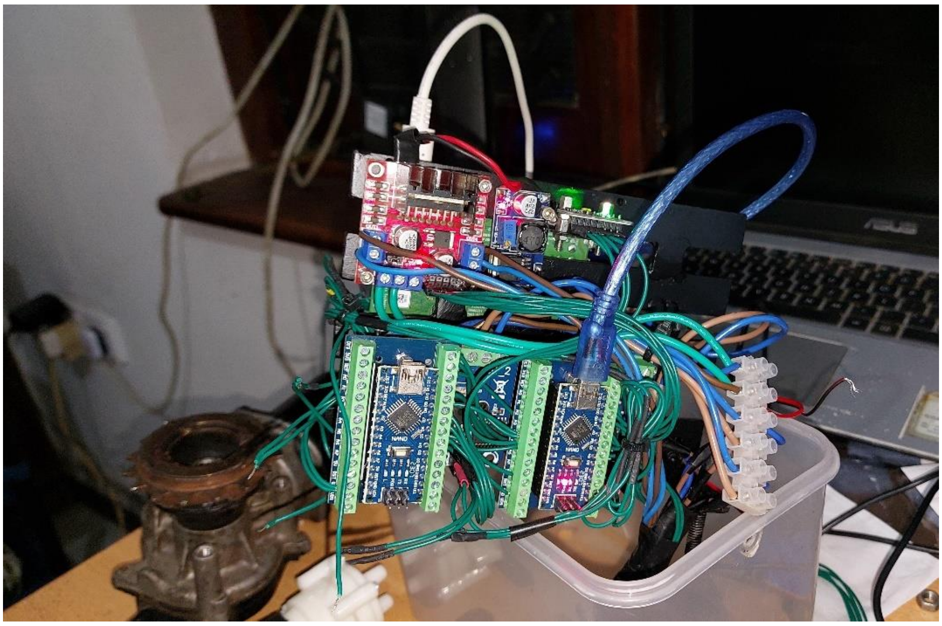
The motor controller consists of four wiper motors, since the motors can operate a high torque. The total weight of the robot structure is 40kg and the motors can operate that weight without any problem. The PWM signals are controlled by two motor drivers with a maximum capacity of 45A. Each motor controller responsible for two motors respectively for the left and right. Each motor controller is controlled by an Arduino Nano. The specialty of the Nano is, all the required pins are available and the size is small. So, it helps to arrange the items more compactly. The Nano contains a single threaded motor controlling algorithm to drive the left side and the right side of the motors parallelly.

Two batteries are used to power the motors and the controller separately. The high capacity (35Ah) battery is used to power the motors since it needs more power. The low capacity (8Ah) battery is used to power the motor controllers. When the motors are taking a high power, sometimes the controllers are having low power for a short period. To overcome the problem. There is a low-capacity battery to maintain a constant power supply to the Arduino Nano and the Raspberry Pi.

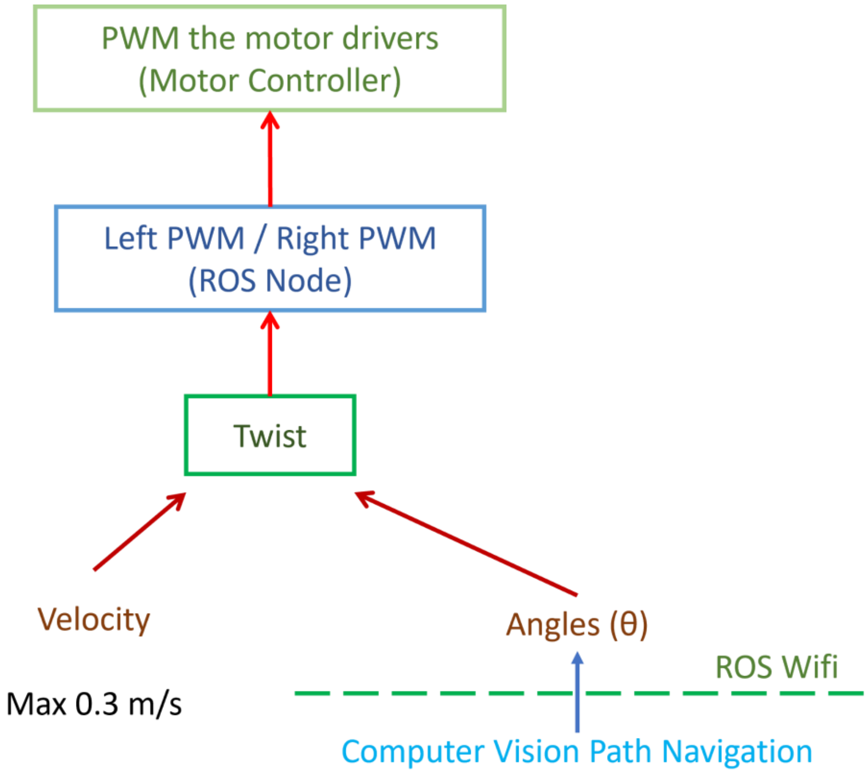








The electronic parts consist with Raspberry Pi, two Arduino nanos, Motor controller for water pumps and two 45A motor controllers. All the components are fixed to an iron frame and store the unit in a water proof plastic cabin. The temperature of components are controlled by a mini exhaust fan.



The Raspberry Pi contains an Ubuntu server to run Robot Operating System (ROS). It contains a Robot Controller Node as a Subscriber. The subscriber is listening for velocity and angle. The computer vision path navigation publisher is publishing those values to the subscriber. The robot controller is having a separate algorithm to translate the velocity and the angles to left and right PWM values.

3.3.4 Motor Controlling algorithm

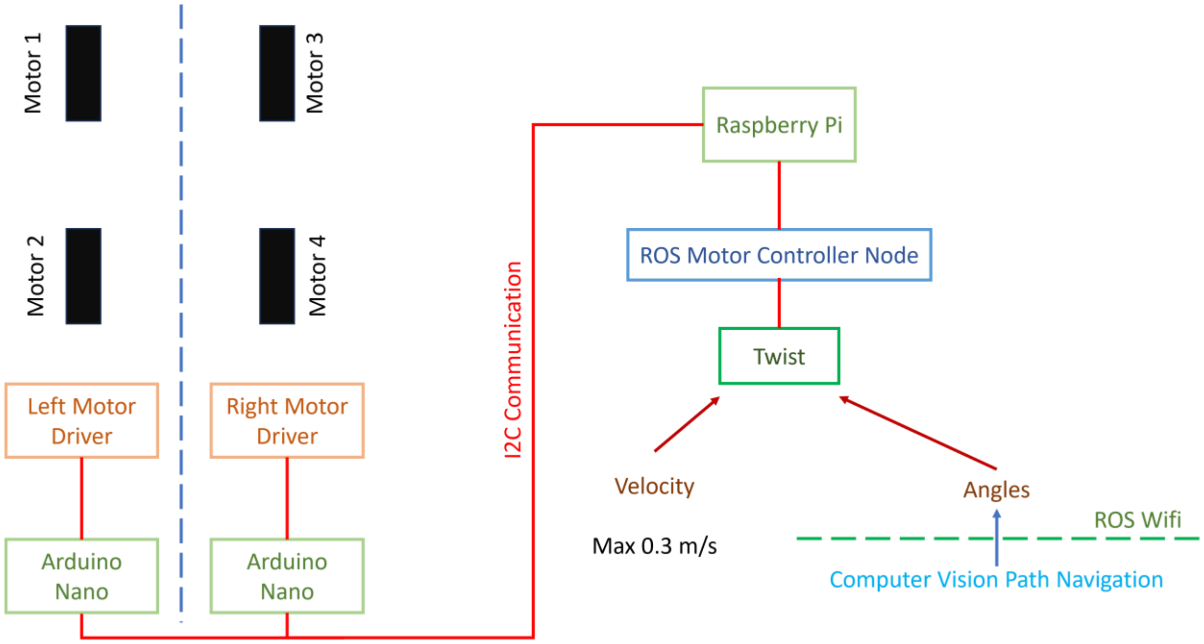
PWM linear = (velocity / max velocity) * 255

PWM angle = 255 – ((angle – 15) / 15)

The velocity is converted to the PWM values first. The PWM is varying from 0 to 255. The challenge is the program should always maintain this range for every input values. The PWM linear is responsible for drive the motors for a selected velocity. The given speed is converted to 0 to 255 range value. Then the PWM angle is calculating the next PWM value to make a PWM difference among both sides and turn the robot.

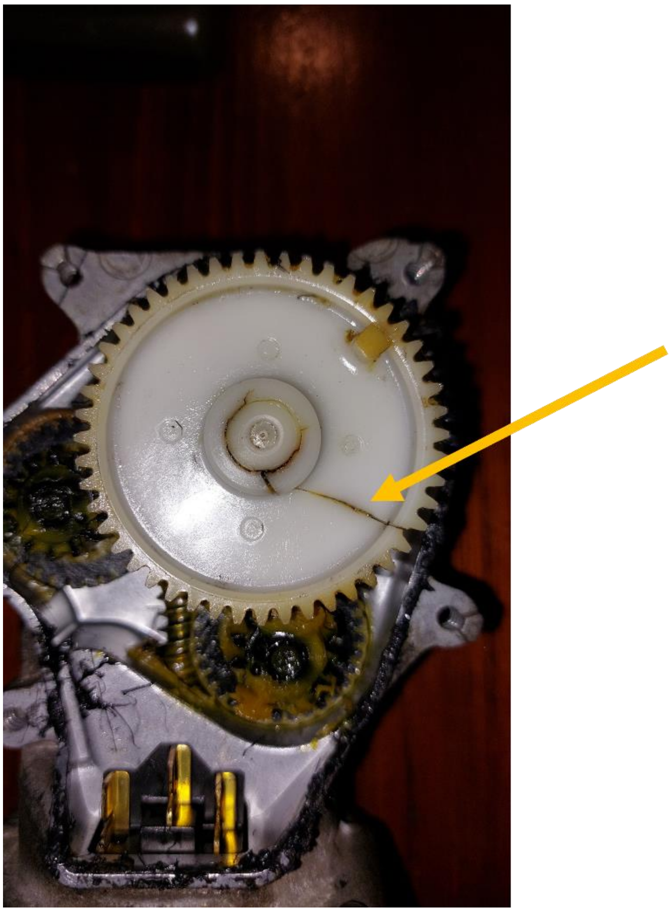
3.3.4 Development of the algorithm

Approach 1 – Two Arduino Nanos with I2C communication.



Rather than using Raspberry Pi, the Arduino Nano has less features. In this scenario, the left side and the right side of the motors should operate independently. But Nano is running in single thread. So the individual motor drivers are responsible with one Arduino Nano. But when the raspberry pi communicates with Arduino using I2C communication. But the I2c cannot communicate with two Arduinos simultaneously. Since the I2C can send only one signal at a time. If the Arduinos has no delays, the communication speed latency can ignore, since those two signals are passing within 1ms.

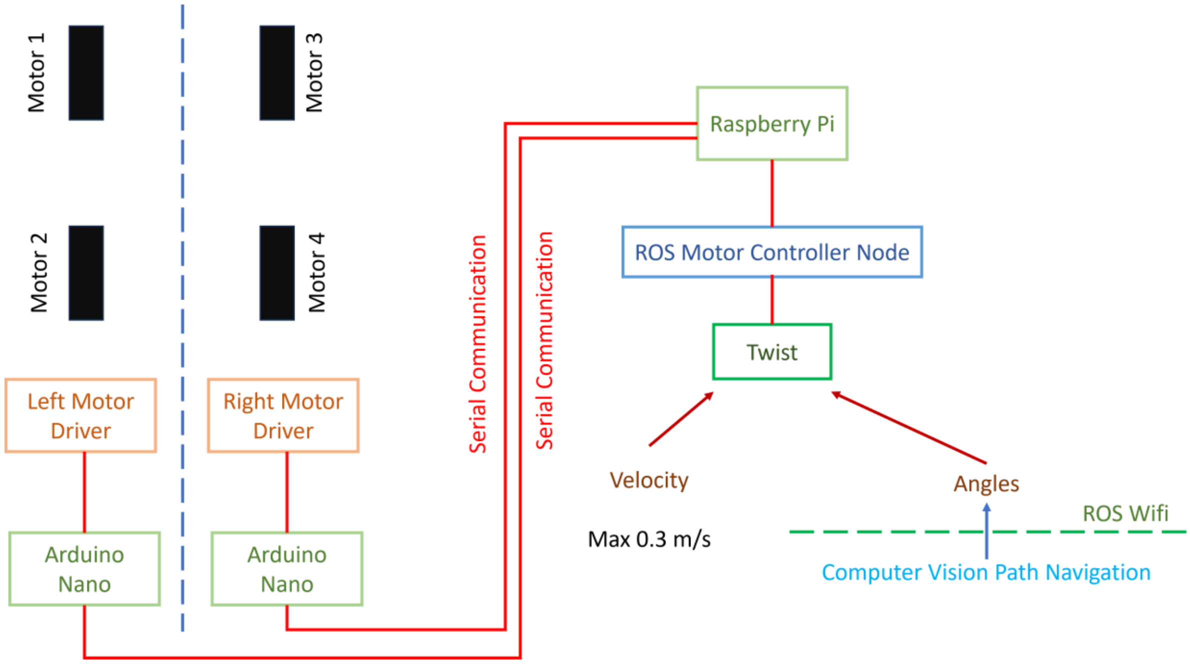
But the robot chassis weighed 40kg. the motors should be protected by sudden starts and stops. If motors are suddenly start or stop, the wheels can be damaged because of the heavy weight with the higher torque of the motors.



One motor wheel was cracked due to sudden start.

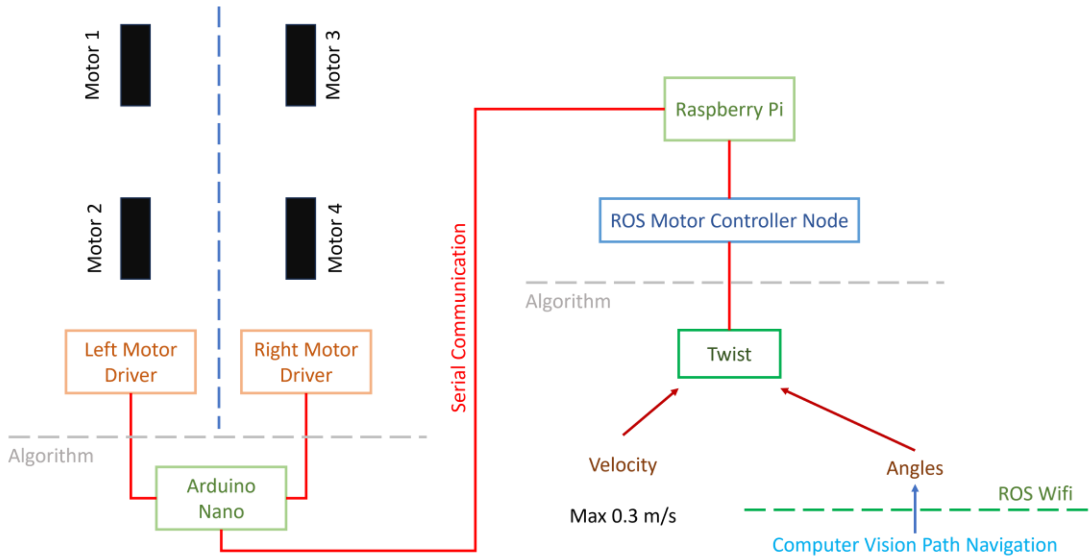
The solution is to gradually start and gradually stop the motors. Since the slow motions helps to handle higher weights without jerking. For that approach the Arduinos should handle with delays. But that approach will direct the motor driving wrong way.

Approach 2 – Two Arduino Nanos with Serial communication.

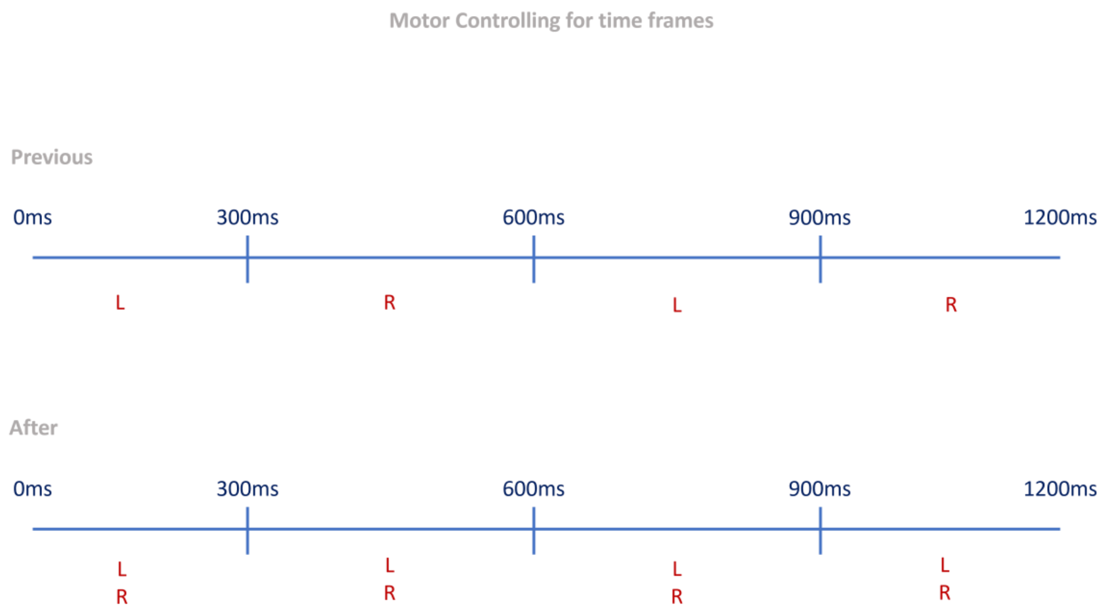


In this approach, serial communication done by USB. For two Arduinos, it uses two serial communications. This approach can operate the motor drivers simultaneously. But this approach is wasting the resources, since the raspberry pi should reserve two USB ports and lots of digital pins are not using in Arduinos. The size cannot compact more to reduce the size of the control box since the serial communication wires are bit larger size.

Approach 3 – One Arduino Nano with Serial communication



This approach is doing the motor controlling job in a single threaded environment and manage the resources minimum. the motor controllers are driving the left and right independent. In programming level, this scenario is done in a multi-threaded environment. But the motor controlling algorithm is developed to run that approach in a single threaded environment. The newly developed motor controlling algorithm can manage the motor driver simultaneous. This helps to reduce the electronic components, increase the power efficiency and make the components more compact.



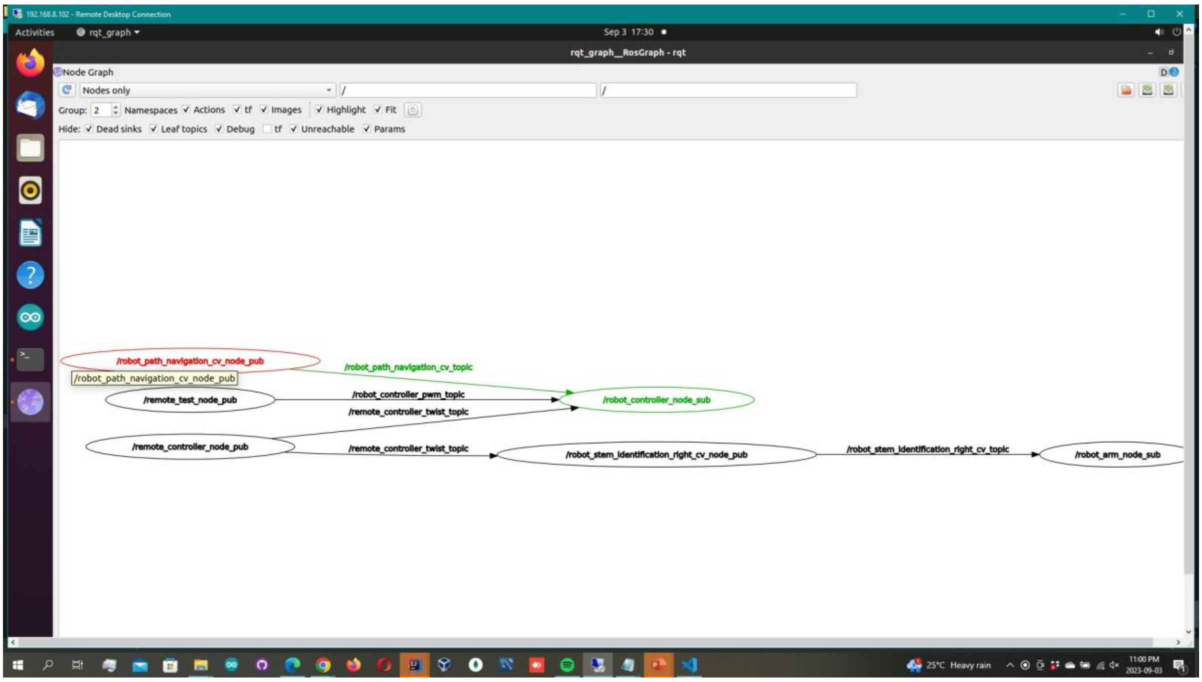
The previous approach operates the motors one after another. But now the motors can operate simultaneously with time delays to protect the motors. If the motors are operating one after the another, the robot cannot make turns more precisely. To operate the motors more precisely, the wheels operation should do simultaneously.

The computer vision provides the coordinates per 500 milli seconds. The motors should quickly response to the coordinates before the next 500 milli seconds. The problem is the heavy motors are not optimized for quickly changing the speed. But the motors are tuned to change the speed smoothly within 300 milli seconds. The motors are gradually changing the speed without jerking the wheels within 300ms when speeding up and speeding down. When the motors are simultaneously operating, the efficiency of the wheel controlling can make optimize.

The heavy weight of the robot helps to make more friction force with the soil without slipping. The chain sprocket method can make a gear ratio carry the heavy weight. These all aspects are used to drive the robot in all-purpose terrain.

Robot Operating System (ROS)

Robot operating system is used to communicate the components. The robot controller is having robot controller node and it is listening to the computer vision node and the computer vision node is the publisher and the robot controller is the subscriber.



The green color is robot controller and the red color is the publisher and a topic is used to communicate among the publisher and the subscriber. The nodes are running in an ubuntu environment. The Raspberry Pi is running an ubuntu server and running the robot controller subscriber node. All the algorithms are running in python and the processed data is transmitted to the Arduinos through serial communication. The specialty of using the ROS is, it is optimized for robotic operations and can make a convention to integrate other robotics based libraries easily.

The ROS can communicate among nodes through Wi-Fi. The computer vision coordinates are transmitting through Wi-Fi, since those operations are done using a another computer.

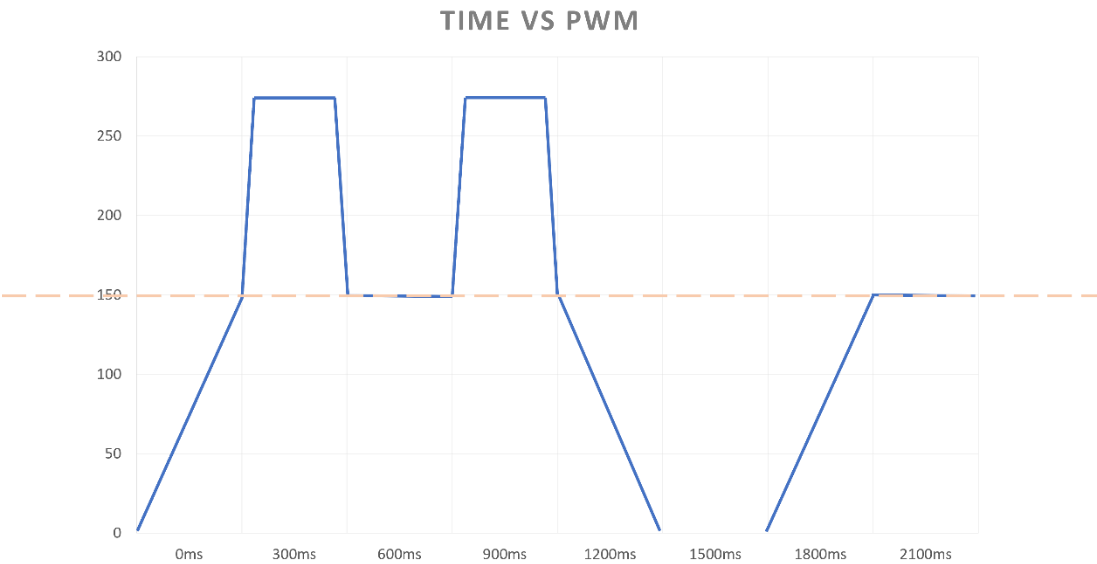
The ROS starts automatically when computers are booted up.

3.3.5 Remote Controller

The remote controller consists with manual and auto drive features. Every autonomous item should have a manual operation for any emergency. The manual controller can operate the robot forward backward and steer left or right. The auto mode is commanding to the robot to drive forward automatic path navigation with liquid fertilizing. The computer of the robot is having a MQTT service to communicate the robot and the remote controller. The controller is a web based application. Every mobile device can get the remote controller and control the robot. The remote controller can set the robot to auto drive in both modes.

3.3.6 Results

The graph explains the previous approach. Suddenly start and stop the motors. This led to damage the motors. With carrying a heavy weight in a hard terrain. The motors should smoothly run without jerking and should maintain the power more balanced to protect the motors, it will extend the lifetime of the motors.



In the next approach the motors are protected by the motor controlling algorithm now. With the time vs PWM graph. The motors are gradually start and stop until the 150 PWM value and slightly take to next level without jerking the motors. The 150 PWM value is selected by huge number of testing in a hard terrain.

3.4 Work Breakdown Structure

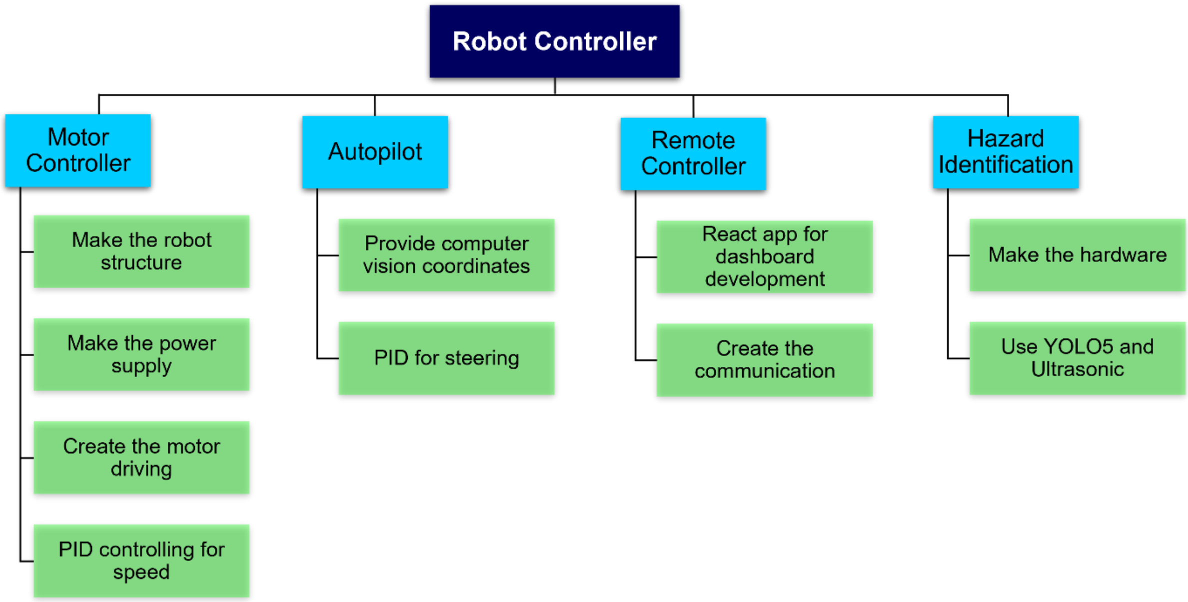


Figure 3-1: Work Breakdown Structure

3.5 Testing

Testing will be done at multiple stages.

1. Conduct tests to assess the frame and motorized wheels' performance across various terrains.
2. Evaluate the robot's forward and backward movements, as well as its skid steering capabilities, on diverse terrain types.
3. Validate the effectiveness of the PID control system and fine-tune it to ensure straight-line movement.
4. Verify the power management system and assess the Python coding through comprehensive unit tests, including speed control assessments.
5. Integrate all components and execute field tests of the robot in a tea plantation setting.

3.6 Gannt Chart

This proposing research is planned to carry out as follows to meet the required deadlines without any conflicts. Project implementation will be started in late March and expected to complete by mid-October while testing will be carried out from mid-October to end of November. The proposing research project will be completed by December 2023.

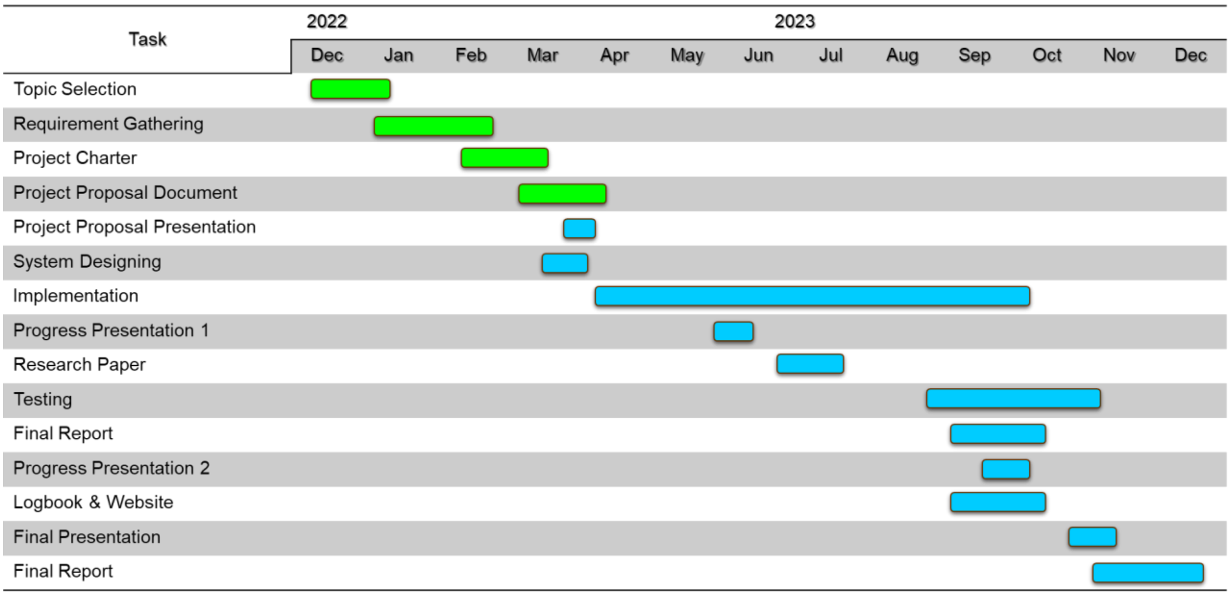


Figure 3-2: Gantt Chart

Future Work

The robot will carry a water hose and continously refill the water tank automatically. The water tank contains a water level sensor and maintain the water level by turning on and off an electronic valve. Then the water pumps can maintain a constant pressure. The wheels are to be extend to mud wheels to increase the friction and decrease from slipping on multi terrains. Then the weight of the robot structure will decrease for easily carrying.

The batteries are charging automatically using solar panel and maintain the power consistent. A proper battery management system will be installed for that purpose.

4 DESCRIPTION OF PERSONNEL & FACILITIES

Ms. Shashika Lokuliyana is the supervisor for this study. She conducts Information Security, Computer Systems and Networking, and Computer Systems Engineering as a Senior Lecturer. She is now employed with the Faculty of Computing, Sri Lanka Institute of Information Technology (SLIIT), Malabe, Sri Lanka, in the Information Systems Engineering Department.

Ms. Narmada Gamage is co-supervising for this project. She is an Assistant Lecturer in the Department of Information Systems Engineering, Faculty of Computing, Sri Lanka Institute of Information Technology (SLIIT), Malabe, Sri Lanka.

Mr. Rajitha de Silva external supervisor for this project. He is a PhD scholar of University of Lincoln, England, UK.

This research will be conducted by the following 4 members as shown below.

Gunawardana I.I.E – He is responsible for the research component of designing the robot chassis and developing the algorithm for the robot controller to navigate the robot.

Perera P.V.Y – She is responsible for the research component of developing the algorithm for the path detection, end of the path detection and provide the precised coordinates to the robot controller.

Premathilake H.T.M - She is responsible for the research component of developing the algorithm for the end of the tea stem detection and provide the precised coordinates to the liquid nozzles controller.

Bamunusinghe G.P - He is responsible for the research component of developing the algorithm for the liquid nozzles controller and navigate the nozzles more precise with the relative movement of the robot.

5 BUDGET & BUDGET JUSTIFICATION

Table 5-1: Estimated Budget

Item	Quantity	Amount(LKR)
Rubber wheels	4	4,000
Sprocket, chain and wheel (gear system)	4	10,800
Iron frame	1	15,500
Motors	4	14,000
43A Motor drivers	4	5,400
Raspberry Pi zero w	1	5,600
12V 45A battery	1	30,000
Camera	3	15,000
Liquid Nozzles	2	3,000
Servo motors	8	9,600
Total Estimated		**112,900.00**

6 REFERENCES

[1] Z. Al-Mashhadani, M. Mainampati, and B. Chandrasekaran, “Autonomous Exploring Map and Navigation for an Agricultural Robot,” *2020 3rd Int. Conf. Control Robot. ICCR 2020*, pp. 73–78, 2020, doi: 10.1109/ICCR51572.2020.9344404.

[2] M. Arivalagan, “Fertilizing and Vigilance for Crops,” no. Icadee, pp. 2020–2022, 2020.

[3] E. T. Baek and D. Y. Im, “ROS-Based Unmanned Mobile Robot Platform for Agriculture,” *Appl. Sci.*, vol. 12, no. 9, 2022, doi: 10.3390/app12094335.

[4] T. Bak and H. Jakobsen, “Agricultural Robotic Platform with Four Wheel Steering for Weed Detection,” *Biosyst. Eng.*, vol. 87, no. 2, pp. 125–136, 2004, doi: 10.1016/j.biosystemseng.2003.10.009.

[5] O. Barrero, S. Tilaguy, and Y. M. Nova, “Outdoors Trajectory Tracking Control for a Four Wheel Skid-Steering Vehicle∗,” *2018 IEEE 2nd Colomb. Conf. Robot. Autom. CCRA 2018*, pp. 1–6, 2018, doi: 10.1109/CCRA.2018.8588153.

[6] P. Biber, U. Weiss, M. Dorna, and A. Albert, “Navigation System of the Autonomous Agricultural Robot ‘BoniRob,’” *Work. Agric. Robot. Enabling Safe, Effic. Afford. Robot. Food Prod.*, pp. 1–7, 2012.

[7] Cihan YURTSEVER, Y. ERTAS, O. SUSTAM, and C. ULU, “Development of an Agriculture Robot for Row-type Seed Sowing Applications,” *J. Selcuk. Spec. Issue 2020 (ICAT’20*, vol. 2020, pp. 228–242, 2020.

[8] A. English, P. Ross, D. Ball, and P. Corke, “Vision based guidance for robot navigation in agriculture,” *Proc. - IEEE Int. Conf. Robot. Autom.*, pp. 1693–1698, 2014, doi: 10.1109/ICRA.2014.6907079.

[9] Z. Fan, Q. Qiu, and Z. Meng, “Implementation of a four-wheel drive agricultural mobile robot for crop/soil information collection on the open field,” *Proc. - 2017 32nd Youth Acad. Annu. Conf. Chinese Assoc. Autom. YAC 2017*, pp. 408–412, 2017, doi: 10.1109/YAC.2017.7967443.

[10] E. P. Godoy, G. T. Tangerino, R. A. Tabile, R. Y. Inamasu, and A. J. V. Porto, “Networked control system for the guidance of a four-wheel steering agricultural robotic platform,” *J. Control Sci. Eng.*, vol. 2012, 2012, doi: 10.1155/2012/368503.

[11] L. Grimstad and P. J. From, “Thorvald II - a Modular and Re-configurable Agricultural Robot,” *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 4588–4593, 2017, doi: 10.1016/j.ifacol.2017.08.1005.

[12] J. B. Han, K. M. Yang, D. H. Kim, and K. H. Seo, “A Modeling and Simulation based on the Multibody Dynamics for an Autonomous Agricultural Robot,” *2019 IEEE 7th Int. Conf. Control. Mechatronics Autom. ICCMA*

2019, pp. 137–143, 2019, doi: 10.1109/ICCMA46720.2019.8988607.

[13] B. He, “Precise navigation for a 4WS mobile robot,” *J. Zhejiang Univ. Sci.*, vol. 7, no. 2, pp. 185–193, 2006, doi: 10.1631/jzus.2006.A0185.

[14] P. V. S. Jayakrishna, M. S. Reddy, N. J. Sai, N. Susheel, and K. P. Peeyush, “Autonomous Seed Sowing Agricultural Robot,” *2018 Int. Conf. Adv. Comput. Commun. Informatics, ICACCI 2018*, pp. 2332–2336, 2018, doi: 10.1109/ICACCI.2018.8554622.

[15] M. H. Ko, K. C. Kim, B. S. Ryuh, A. Suprem, and N. P. Mahalik, “Development of autonomous traveling for agricultural robot drive platform by using a single camera,” *Lect. Notes Eng. Comput. Sci.*, vol. 1, pp. 361–365, 2013.

[16] M. H. Ko, B. Ryuh, K. C. Kim, A. Suprem, and N. P. Mahalik, “Strategies From System Integration Perspective : Review and Application,” *Ieee/Asme Trans. Mechatron.*, vol. 20, no. 4, pp. 1705–1716, 2015.

[17] Y. B. Koca, Y. Aslan, and B. Gokce, “Speed Control Based PID Configuration of a DC Motor for An Unmanned Agricultural Vehicle,” *2021 8th Int. Conf. Electr. Electron. Eng. ICEEE 2021*, pp. 117–120, 2021, doi: 10.1109/ICEEE52452.2021.9415908.

[18] T. Mueller-Sim, M. Jenkins, J. Abel, and G. Kantor, “The Robotanist: A ground-based agricultural robot for high-throughput crop phenotyping,” *Proc. - IEEE Int. Conf. Robot. Autom.*, pp. 3634–3639, 2017, doi: 10.1109/ICRA.2017.7989418.

[19] B. V. Naik and R. R. Goud, “Design and fabrication of a solar powered autonomous agricultural robot,” *AIP Conf. Proc.*, vol. 2358, pp. 1–8, 2021, doi: 10.1063/5.0057867.

[20] S. E. Oltean, “Mobile Robot Platform with Arduino Uno and Raspberry Pi for Autonomous Navigation,” *Procedia Manuf.*, vol. 32, pp. 572–577, 2019, doi: 10.1016/j.promfg.2019.02.254.

7 APPENDICES

7.1 Plagiarism Report