

## 1. Introduction

Weeds are an obstacle for agricultural productivity, reducing crop output and quality. According to Agriculture Ministry data, India lost agricultural products worth more than \$10,000 million - more than the Centre's budgetary allocation for agriculture for 2017-18 - to weeds in 10 important crops in different districts of 18 States between 2003 and 2014. This equates to an annual loss of over \$ 910 million in agriculture crops due to weeds across the country. Weed management and control are thus essential to the production of high yielding and high-quality crops. Below pie chart shows the statistical data of percentage loss to agriculture from different factors is obtained from the study conducted by Tamil Naidu Agricultural University .[1]

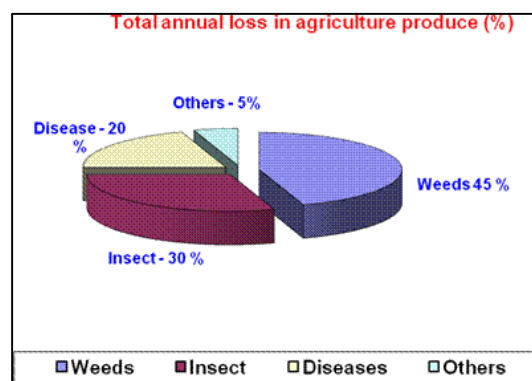


Figure 1.1: [1] Pie chart presenting annual loss in agriculture due to various factors

AWR will be a self-controlled weeding robot which will be used for identifying and removing the harmful weeds grown in the farms without harming the ecosystem by using various technologies like RTK (Real Time Kinematics) based GPS system, Robot Operating System (ROS), image processing and many more.

### 1.1. Need of Project

The field of agriculture faces numerous challenges, including labor shortages, rising labor costs, and increasing pressure to reduce chemical herbicide usage. The need for autonomous weeding robots in agriculture and highlight their potential benefits.

#### 1. Labor Shortages and Rising Costs:

Traditional weeding methods in agriculture heavily rely on manual labor, which is becoming scarce and expensive in many regions. Farmers struggle to find an adequate

workforce for manual weeding tasks, leading to delays in operations and compromised crop health. Autonomous weeding robots can alleviate this issue by reducing the dependency on human labor and ensuring consistent and efficient weed control.

## 2. Chemical Herbicide Reduction:

Excessive use of chemical herbicides poses significant environmental and health risks. There is a growing concern regarding the accumulation of chemical residues in crops and soil, as well as their impact on biodiversity. Autonomous weeding robots provide a viable alternative to chemical herbicides by precisely targeting and removing weeds without the need for harmful chemicals, promoting sustainable farming practices.

## 3. Precision and Efficiency:

Autonomous weeding robots are equipped with advanced technologies such as computer vision, machine learning, and GPS guidance systems. These technologies enable them to accurately identify weeds, distinguish them from crops, and precisely apply the necessary mechanical or thermal weed control methods. This level of precision ensures minimal damage to crops, increases weed removal efficiency, and maximizes yield potential.

## 4. Time and Resource Optimization:

Weeding is a time-consuming task that demands significant resources. Autonomous weeding robots can operate 24/7, eliminating the constraints of human labor availability and allowing farmers to optimize their time and resources. These robots can efficiently cover large areas, work in challenging terrains, and adapt to various crop types, ensuring timely and effective weed management.

## 5. Data-Driven Insights:

Autonomous weeding robots generate valuable data during their operations, including weed density, distribution patterns, and crop health indicators. This data can be utilized to gain insights into weed growth patterns, optimize weed control strategies, and make informed decisions regarding crop management practices. By harnessing the power of data analytics, farmers can enhance overall productivity and profitability.

## **1.2. Problem Statement**

Development of an autonomous ground vehicle or AGV using various technologies for removing weeds which majorly harm the agricultural sector.

The technologies such as RTK (Real Time Kinematics) GPS system and LIDAR sensor to capture the images from the crop fields

## **1.3. Objectives**

1. To provide an eco-friendly alternative for toxic herbicides used for removing weeds.
2. To solve the problem of human labor scarcity in rural areas.
3. To develop an efficient autonomous robotic drive having capability of detecting and removing weeds.

## **1.4. Scope**

1. To reduce the total cost of the product.
2. There are many Fuels Engine machines available in market for Weeding purpose but they are very costly and also required more running cost, apart from that main disadvantage of this machine is that they consume more fuel. Due to more usage of conventional fuels more Air pollution occurs which is most serious concern. So, by using agricultural robot we are trying to overcome the problems of fuel engine machines.
3. The developed Agribot will consists of an adaptable robotic drive which can be used to perform various agricultural processes using different mechanisms.
4. It provides an ecofriendly alternative for toxic weedicides used for removing weeds so the fertility of soil doesn't get affected.
5. It will reduce the require amount of labor power which makes it cost effective.

## 1.5 Methodology Planned

Methodology planned and followed during the project is as follows:-

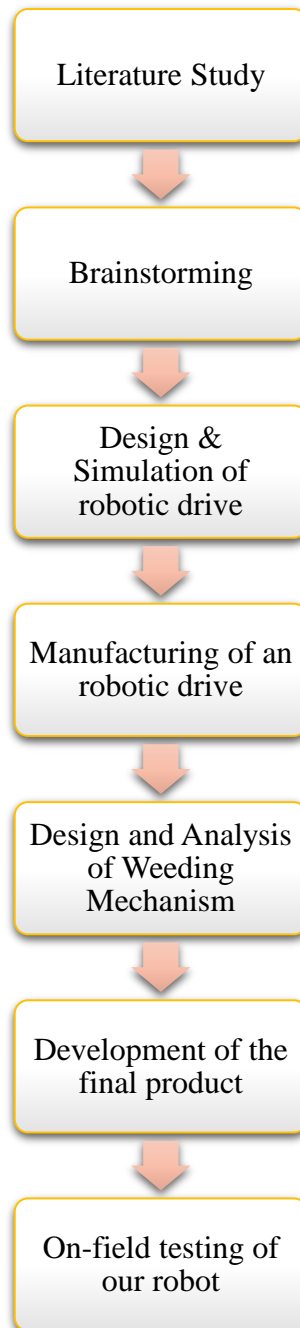


Figure 1.2: Methodology followed while working on the project

## 2. Literature survey

### **“Agricultural Robotic Platform with Four Wheel Steering for Weed Detection”**

**(Thomas Bak, Hans Jacobson, 2004)**

A robotic platform for mapping weed populations in fields was used to illustrate intelligent concepts for autonomous cars in agriculture, which might lead to a new sustainable model for developed agriculture in the future. The vehicle seen here is designed to work in 0.25 and 0.5m row crops and is outfitted with cameras for row navigation and weed identification. A modular method is used, with four identical wheel modules, enabling for four wheel steering and platform propulsion. As a result, the vehicle's mobility improves, allowing parallel displacement during turns by decoupling position modifications from orientation adjustments. The platform is controlled by a vehicle electronics and control system that is based on embedded controllers and conventional communication protocols. The programme employs a hybrid intentional software design that enables for the operation's hierarchical deconstruction. The most basic level employs a reactive feedback control technique based on an extension of simple control for car-like vehicles to the four-wheel case. The controller's architecture requires the vehicle's front and rear to follow a predetermined course while allowing the vehicle to keep a fixed orientation relative to the path. The controller reasoning is described, as well as field experiment results.

### **“On the design of a low-cost, light-weight, and highly versatile agricultural robot”**

**(Grimstad, Pham, Phan, & From, 2015)**

This article discusses the development and key factors for building the Thorvald platform, a multipurpose agricultural robot. The major goal is to create a robot that can do everything from sowing to weeding to harvesting. This necessitates the robot's ability to transport a large range of tools. Furthermore, the robot must be lightweight in order to function during rainy seasons without becoming trapped or injuring the soil structure. We focus on keeping the overall expenses of the robot low enough to make it economically feasible in comparison to traditional methods. We do this by designing

the frame to flex, which minimises complexity and lowers the cost of construction while ensuring that all wheels are in touch with the ground. We also detail the creation of tools that will be mounted to the platform and explore the effects of the flexible design on the robot and tool control.

### **“A Design of an Autonomous Agricultural Robot to Navigate between Rows”**

**(Celen, Onler, & Kilic, 2015)**

A row guiding system based on ultrasonic distance measuring is described to guide a robot platform that is created independently to drive across row crops in a field. The offset of the robot platform is detected in real-time to guide the robot inside the crop row and to automatically pivot at the end of the rows to the subsequent row. In the rose field, preliminary row guiding trials were carried out. The experimental findings reveal that row guiding and headland turn algorithms are based on the factors measured and analysed, such as the offset for row guidance and the difference between the robot's motion trajectory and the predicted trajectory. At a speed of 1 m/s, row guidance accuracy is 70mm. Tractacus took third place in the free style category of the Field ROBOT Event 2012 with a smart spraying application that begins spraying only when it detects the crop canopy.

### **“A Vision System for Autonomous Weed Detection Robot”**

**(Asif, et al., 2010)**

A visual guiding system for an autonomous weed detecting robot is provided in this research. The proposed vision system employs a number of image processing algorithms to identify the inter-row space between crops and then uses the Hough transform to compute the current posture and orientation. The dynamic model is used to forecast changes in position and orientation from frame to frame and to predict the development of values over time. The vision system is built and simulated in Matlab, and the created system effectively detects and calculates crop boundary pose and direction on both real and synthetic pictures.

---

### **“Automatic Model Based Dataset Generation for Fast and Accurate Crop and Weeds Detection”**

**(Cicco, Potena, Grisetti, & Pretto, 2017)**

One of the most difficult tasks in farm robots is selective weeding. A farm robot should be able to recognise plants properly and discriminate between crop and weeds in order to do its mission. The majority of the promising cutting-edge technologies rely on appearance-based models trained on massive annotated datasets. Unfortunately, producing big agricultural datasets with pixel-level annotations is a time-consuming job, making data-driven solutions less appealing. In this research, we address this issue by offering an innovative and successful technique that attempts to significantly reduce the amount of human involvement required to train detection and classification algorithms. The objective is to build huge synthetic training datasets procedurally by randomising essential aspects of the target environment (for example, crop and weed species, soil type, and light conditions). More precisely, by fine-tuning these model parameters and utilising a few real-world textures, a huge number of realistic views of an artificial agricultural environment may be rendered with minimal effort. The data obtained can be used directly to train the model or to supplement real-world photos. We validate the suggested technique by employing a current deep learning-based picture segmentation architecture as a testbed. We examine the classification results derived from actual and synthetic photos used as training data. The obtained findings validate the efficacy and potential of our technique.

### **“Real-Time Weed Detection using Machine Learning and Stereo-Vision”**

**(Siddhesh Badhan, 2021)**

Weeds are a nuisance because they compete with desirable crops for water, nutrients, and space. Weeds can also become caught in machinery, preventing effective harvesting. As a result, weed eradication systems are required. The development of a good weed removal system requires accurate identification of the undesirable plants. The research offers a real-time weed identification system that identifies weeds in fields using machine learning and stereo-vision for 3D crop reconstruction. The structure from motion approach is used on a farm video to produce a 3D point cloud. The machine learning model is trained on two manually constructed cucumber and onion crop

datasets. The machine learning models are trained using Convolutional Neural Networks (CNN) and the ResNet-50 methods. The ResNet-50 model outperforms the Convolutional Neural Networks model. The ResNet-50 model has an overall accuracy of 84.6% for the cucumber dataset and 90% for the onion crop dataset.

### **“A critical review on agricultural robots”**

**(Reddy, Reddy, Pranavadithya, & Kumar, 2016)**

Agribot, or agricultural robot, is a robot used for agricultural reasons. The introduction of agricultural robots has enhanced agricultural production and output in various nations. Furthermore, the use of agricultural robots lowered agricultural operational expenses and lead time. The current research examines the success stories of robotic agriculture in several agricultural fields. The research also sheds insight on the future potential of robotic agriculture, particularly in underdeveloped nations.

### **“Real-Time Localization and Mapping Utilizing Multi-Sensor Fusion and Visual-IMU-Wheel Odometry for Agricultural Robots in Unstructured, Dynamic and GPS-Denied Greenhouse Environments”**

**(Yan, Zhang, Zhou, Zhang, & Liu, 2022)**

A unique real-time localization and mapping strategy for unstructured, dynamic, and GPS-denied greenhouse situations is proposed in this work. The proposed method integrates measurements from wheel odometry, an inertial measurement unit, and visual-inertial odometry into a loosely coupled multi-sensor fusion framework based on the Extended Kalman Filter to compensate for the defects of a single sensor in order to achieve accurate and robust pose estimation of the robot in complex greenhouses. The updated ORB-SLAM2 technique is used to build a dense 3D point cloud map of the surroundings while also performing localization. Agricultural robots can achieve more precise re-localization and create the foundation for future autonomous navigation by integrating the results of accurate localization with dense 3D point cloud maps.



### 3. Mechanical system design of AWR

#### 3.1 Introduction to AWR

AWR's prime focus is the weed monitoring through autonomously and tele- operation of robot. For designing such a robot, we will be using the ROS frame work, deep learning and Machine Vision algorithms for crop-weed classification. These factors must be considered when applying weed control to crops in the field. The road on which the car will be driven is quite uneven. The wheels should be tough enough to withstand driving in these circumstances. If necessary, the driving system should be able to perform a zero radius turn. The vehicle's track length may vary. The ground clearance of such a robot should be at least 600 mm so that the on-board modules and crops are not harmed. The robot must have an on- board power supply and communication. It must be capable of handling real-time data and making decisions based on it in a real-time setting. Precise monitoring and identification of weeds is more vital so that farmers may easily care for themselves afterwards. The primary processor should be able to handle several data streams from various sources at varying rates. Below diagram represents entire system in diagrametical form.[11]

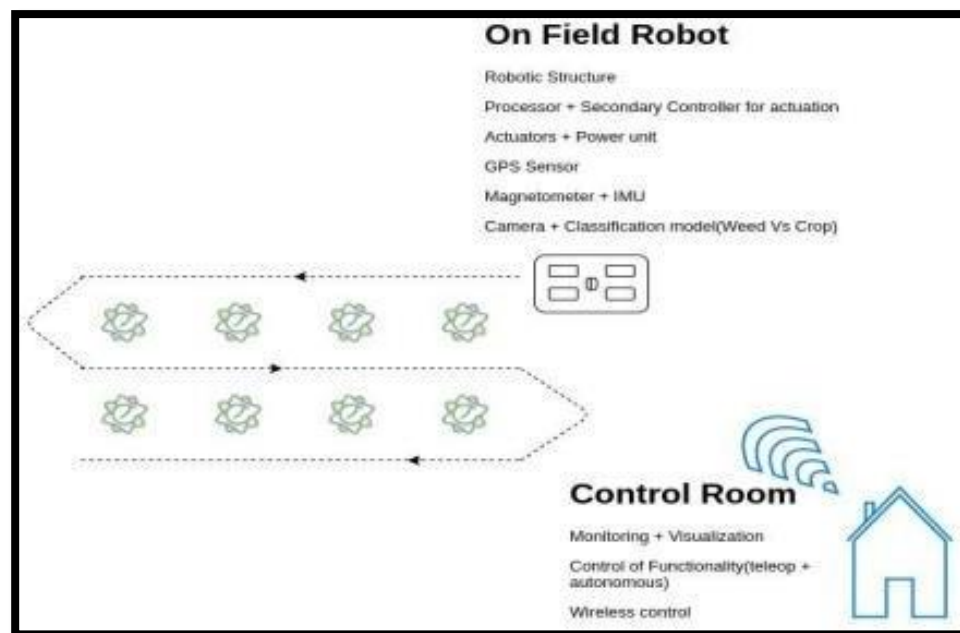


Figure 3.1: [11] Overview of the AWR System

### 3.2. CAD Model

We design and drafted the CAD Model of AWR in SolidWorks software. After doing the calculation we first design and drafted the drive of AWR then designed mechanism for removing weeds.

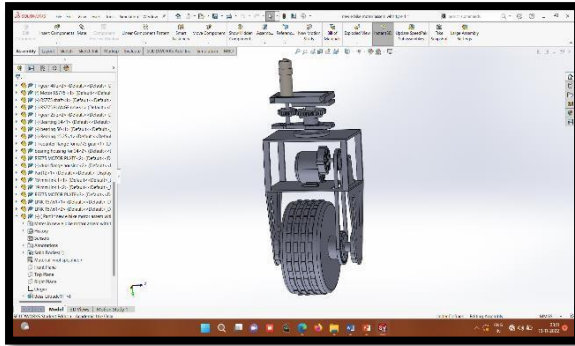


Figure 3.2: Single Wheel Assembly

Main spec of complete assembly of drive  
 L x B x H: 200 X 250 X 750 mm  
 Weight: 17 kg  
 Material Used :- Plain Carbon steel, Nylon 66, ABS

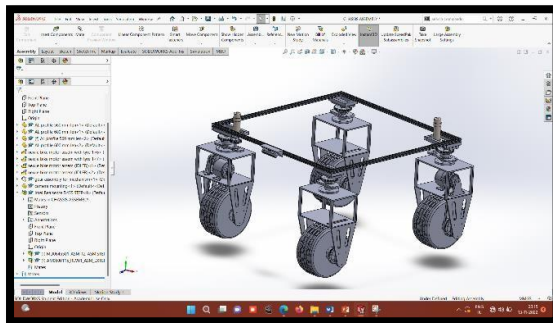


Figure 3.3: Overall wheel assembly with chassis

Drive consists of swerve mechanism for its holonomic 360-degree movement. Main spec of complete assembly of drive  
 L x B x H: 1000 X 1000 X 800 mm  
 Weight: 48.7 kg  
 Chassis Material: - Plain Carbon

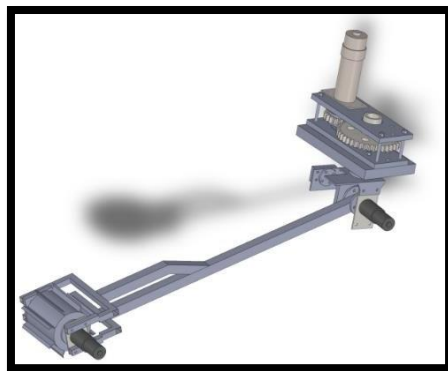


Figure 3.4: Weeding mechanism assembly

Mechanism consists of motor driving rotor for removing weeds which will actuate after weed detection.

Main spec of mechanism  
 Length of arm: - 500 mm  
 Diameter of rotor: - 80 mm  
 Length of rotor: - 150 mm

### 3.2.1. Complete cad model of AWR

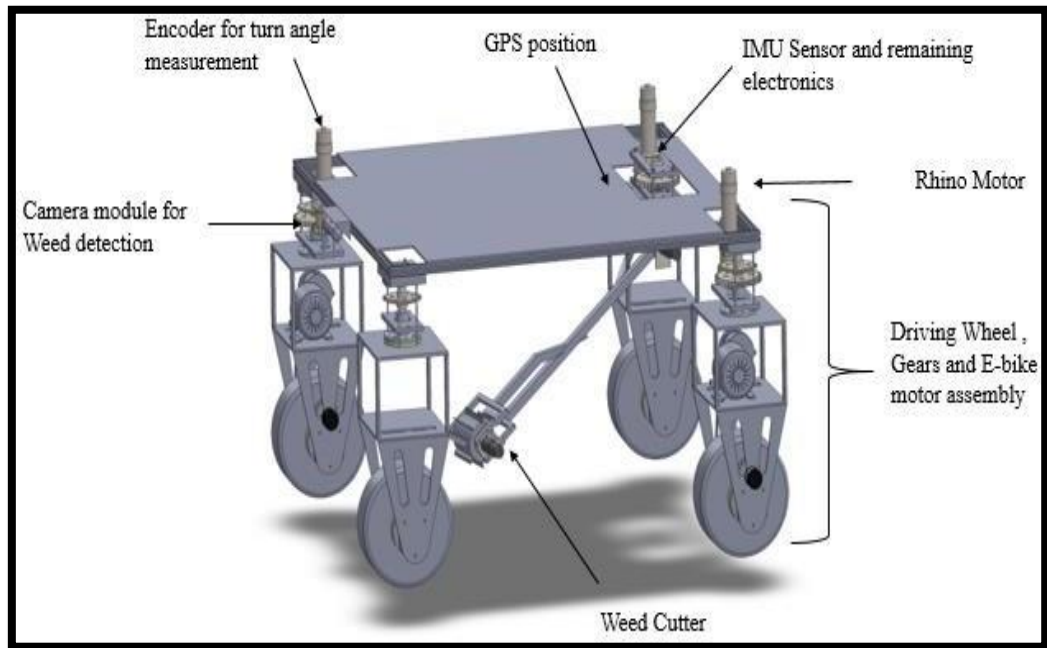


Figure 3.5: Complete Assembly of AWR

### 3.3 Chassis Calculations

#### 3.3.1. Specifications of Material

Table 3.3.1.: Material Specifications

Property	Aluminium 6061	Plain Carbon Steel	Stainless Steel
Mass density (g/cm <sup>3</sup> )	2.7	7.8	8
Yield Strength (Mpa)	55.14	220.59	206.80
Ultimate tensile strength(Mpa)	124.084	399.82	517.017
Poisons ratio	0.33	0.28	0.29
Shear Modulus(Gpa)	26	79	75
Young's Modulus(Gpa)	69	210	190

Material – Plain Carbon Steel

$$E = 2.10 \times 10^5 \text{ N/mm}^2$$

$$\text{Poisson ratio} = 0.28$$

$$\text{Capacity of Bot} = 15 \text{ Kg} = 147.5 \text{ N} \quad \text{Capacity of Bot with 1.25 \%} = 183.9375 \text{ N}$$

$$\begin{aligned} \text{Weight of body with motor assembly} &= 100 \text{ Kg} = 981 \text{ N} \\ \text{Total Load Acting on Chassis} &= 981 + 185 = 1166 \text{ N} \end{aligned}$$

Chassis has four beams so load acting on each beam is one/fourth of total load acting on the beam

$$\therefore \text{Load acting on the single Frame} = 1166/4 = 291.5 \text{ N/Beam}$$

### 3.3.2 Calculations for Reaction

Beam is considered as simply supported beam supported at A and B with uniformly distributed load.

Load acting on,

$$\text{Entire span of beam} = 291.5 \text{ N} \quad \text{Length of beam} = 1000 \text{ mm}$$

$$\therefore \text{Uniformly Distributed load} = 291.5/1000 = 0.2915 \text{ N/mm} \quad \text{Moment at point A is 0.}$$

$$\text{Therefore, } (0.291 \times 1000 \times (1000/2)) - R_B \times 1000 = 0 \quad \text{Hence, } R_B = 145.5 \text{ N}$$

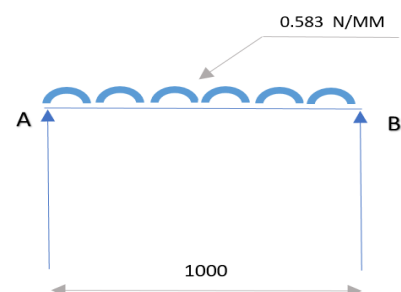
$$\therefore \text{Total load acting on the beam} = 0.2915 \times 1000 = 291.5 \text{ N}$$

$$\therefore R_A + R_B = 291.5$$

$$\therefore R_A + 145.5 = 291.5$$

$$\therefore R_A = 291.5 - 145.5$$

$$\text{Therefore } R_A = 145.5$$



FBD

### 3.3.3. Calculation of Shear Force and Bending Moment

Shear force Calculations:

$$\therefore F_b = -145.5 + 291 = 145.5 \text{ N}$$

$$\therefore F_a = -145.5 \text{ N}$$

Bending Moment Calculations:

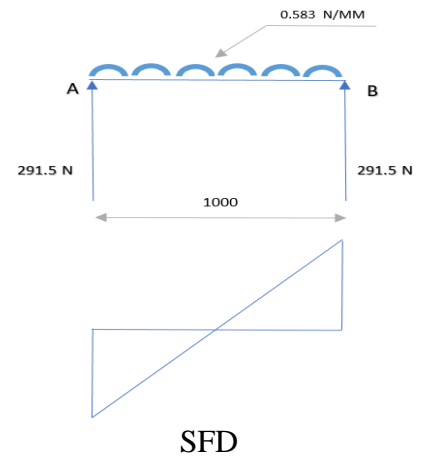
$$\therefore M_a = 0$$

$$\therefore M_b = (0.291 \times 1000 \times 500) - (145.5 \times 1000)$$

$$\therefore M_b = 0$$

$$\therefore M_{\max} = WL^2 \div 8 = (0.291 \times 1000^2) / 8$$

$$\therefore M_{\max} = 36437.5 \text{ N.mm}$$



### 3.3.4. Bending stress Calculations:

$$\therefore (M/I) = (\sigma/y) = (E/R)$$

$$\text{Therefore } I_{xx} = [(bh^3/12) - (b_1h_1^3/12)] = 2112.08 \text{ mm}^4$$

$$\therefore M_{\max} = 36437.5 \text{ N.mm}$$

$$\therefore \sigma = 36437.5 \times 9.25 / 2112.08$$

$$\therefore \sigma = 159.58 \text{ N/mm}^2$$

$$\therefore \sigma(\text{allowable}) = S_{ut} / 3 = 800 / 3 = 266.67 \text{ N/mm}^2$$

$$\therefore \sigma < \sigma(\text{allowable})$$

Therefore, Design is Safe

### 3.4 Mechanism Calculations:

Motor Selection for lifting rotor assembly and torque required to lift four bar links and rotor assembly.

$$T = \text{weight} \times \text{perpendicular distance}$$

$$= 12 \times 3.5$$

$$\therefore T = 42 \text{ kg.cm}$$

Taking Factor of Safety = 1.5

$$\therefore T = 42 \times 1.5 = 63$$

$\therefore$  We select 70 kg.cm RSMotor Selection for rotating mechanism  $T = \text{weight} \times \text{perpendicular distance}$

$$= 15 \times 5 = 75 \text{ kg.cm Taking Factor of safety} = 1.5$$

$\therefore$  So, we used gear pair of ratios 2:1 RS775 motor

#### 3.4.1 Gear Calculations:

Torque Required

The weight of wheel assembly = 15 kg (approx.) = 150 N Centre Distance between gears = 50 mm (Assume)

To achieve required torque as per chosen motor torque we assume 50 mm Distance Therefore Torque of motor = 70 Kg.cm

$$\text{Calculated Torque} = 150 \times 150 = 7500 \text{ N.mm}$$

$$= 7500 \times 0.010 \text{ kg.cm} = 75 \text{ Kg.cm Therefore } 40\text{C8} = \text{Ultimate tensile strength} = 580 \text{ N/mm}^2$$

Hardness = (BHN) 217 We have selected same material for both gears

Therefore, we consider 20° Full-depth involute gear tooth system so according to this, no. of teeth on pinion is 20 and no. of teeth on spur gear is 40 (according to std. reduction ratio for single stage gear drive)

To Calculate transmitting Power (P);  $P = (2\pi NT)/60$

$N_p = 500$  rpm (Motor rpm)  $T = 7500$  Nmm = (7.5 Nm)

Therefore  $P = (2\pi \times 500 \times 7.5)/60$

$P = 392.69$  W  $Z_p = 20$ ,  $Z_g = 40$

$P = 392.69$  W,  $N_p = 500$  rpm

$S_{ut} = 580$  N/mm<sup>2</sup> BHN = 217

Service Factor = 1.5 (K), Factor of safety = 1.75 Beam Strength:  $\sigma_b = S_{ut}/3 = 580/3 = 193.33$  N/mm<sup>2</sup>

Therefore  $[F_b = \sigma_b \cdot b \cdot m \cdot Y_p]$

$Y_p$  (form factor) =  $0.484 - (2.87/Z_p) = 0.484 - (2.87/20) = 0.3405$

Assume  $b$ (face width) = 10m

Therefore  $F_b = 193.33 \times 10m \times m \times 0.3405 = 658.28m^2$  N

Wear Strength:

$F_w = d_p \cdot b \cdot Q \cdot k$

$d_p$  (PCD of pinion) =  $m \cdot Z_p = 20m$

$Q$  (Ratio Factor) =  $(2Z_g)/(Z_g + Z_p) = (2 \times 40)/(40+20) = 1.33$   $F_w = 20m \times 10m \times 1.33 \times 1.5 = 400m^2$  N

$\therefore$  Effective Load =

$\therefore V$  (Pitch line Velocity) =  $(\pi d_p N_p)/(60 \times 1000) = (\pi \times 20m \times 500)/(60 \times 103) = 0.5236$ mm/s

$\therefore F_t$  (Tangential Force) =  $P/V = 392.69/0.5236\text{m} = 749.98/\text{m}$  N Assuming the gear pair is manufactured by hobbing

$$\therefore K_v = 6/(6+V) = 6/(6 + (0.5236\text{m}))$$

$$\therefore F_{eff} = (K_a.K_m.F_t)/K_v \quad \dots(K_m = \text{load concentration factor} = 1)$$

$$= (187.495(6+0.5236\text{m}))/\text{m} \quad \text{N}$$

Estimation of module:

In order to avoid pitting failure:

$$\therefore [F_w = N_f \cdot F_{eff}]$$

$$\therefore 400\text{m}^2 = 1.75 \times (187.495/\text{m})(6+0.5236\text{m})$$

$$\therefore 400\text{m}^3 = 1968.6975 + 171.8016\text{m}$$

$$\therefore \text{m}^3 = 4.9 + 0.4295\text{m}$$

$$\therefore \text{m}^3 - 0.4295 - 4.922 = 0$$

$$\therefore \text{m} = 2.44 \text{ mm} \approx 3\text{mm}$$

### 3.4.2 Shaft for pinion:

$$\tau_{max} = 90.67 \text{ N/mm}^2 \quad \sigma_{max} = 181.332 \text{ N/mm}^2 \quad T(\text{shaft}) = 7.5 \text{ Nm}$$

$$N(\text{shaft}) = 500 \text{ rpm}$$

$$\text{Tangential force on gear } F_t = 2T/D \text{ (PCD of pinio)} = 60\text{mm}$$

$$F_t = (2 \times 7.5)/0.06 = 280 \text{ N}$$

Normal load acting on pinion:

$$w = \frac{F_t}{\cos a} \quad (a = \text{pressure angle} = 20^\circ)$$

$$w = 266.04 \text{ N}$$



$$\text{Bending moment} = \frac{W_{ab}}{L} = \frac{266.04 * 0.045 * 0.035}{0.08} = 5.2376 \text{ N.m}$$

Twisting moment subjected to shaft = 7500 N.m Bending moment = 5237.66 N.mm

According to Maximum Shear Stress Theory of failure, Equivalent Twisting Moment

$$T_e = \sqrt{(M^2 + T^2)}$$

$$T_e = \sqrt{(5237.66)^2 + (7500)^2} = 9147.85 \text{ N.mm}$$

According to,  $J = 16T_e / (\pi d^3)$   $d = 8.009 \text{ mm} = 10 \text{ mm}$

According to maximum bending / principle shear stress theory of failure, Equivalent

Bending Moment  $T_c = 1(M + \sqrt{(M^2 + T^2)})/2 = 7192.755 \text{ N.mm}$

$$\delta = (32 M_e) / (\pi d^3)$$

$$d = \sqrt[3]{\frac{32 * 7192.755}{\pi * 181.332}} = 7.39 \sim 10 \text{ mm}$$

### 3.4.3 Shaft for Spur Gear:

Design of shaft for spur gear:

(T) Torque transmitted through shaft =  $14.99 * 10^3 \text{ N.mm}$

(N) Speed of shaft = 250 rpm

Permissible  $\tau_{max} = 90.67 \text{ N/mm}^2$

Permissible  $\sigma_{max} = 181.332 \text{ N/mm}^2$

Tangential force on gear  $F_t = \frac{2T}{D} = 249.99 \text{ N}$

Normal load acting on gear = 266.03 N

Bending moment (M) =  $W_{ab}/L = 7729.64 \text{ N.mm}$  According to Maximum shear stress theory of failure,

Equivalent Twisting Moment  $T_c = \sqrt{(M^2 + T^2)} = 16885.57 \text{ N.mm}$  Permissible

$$\rho_{max} = 16 T_e / (\pi * d^3)$$

$$d=16 \text{ mm}$$

According to Maximum shear stress theory of failure,  $M_e = 12297.605 \text{ Nmm}$

$$\text{Permissible } \sigma_{\max} = (32 M_e) / (\pi d^3) \quad d = 16 \text{ mm}$$

### 3.5 Finite Element Analysis of main components of AWR

Finite Element Analysis of chassis performed using Ansys software. Analysis consist of following steps:-

1. Import Geometry :- Cad model is imported from SolidWorks software in igs file
2. Material Selection :- According to calculations We selected Plain Carbon Steel material for chassis an applied for further analysis.
3. Meshing :- Fine meshing is done with selecting triangular shape nodes.
4. Boundary Conditions :- In boundry condition lower side of chassis is fixed and 1.2 KN force applied from upper side on all four sides of chassis.
5. Solving and results :- Further solving the boundary condition we take result of Maximum Principle stress and Total Deformation.

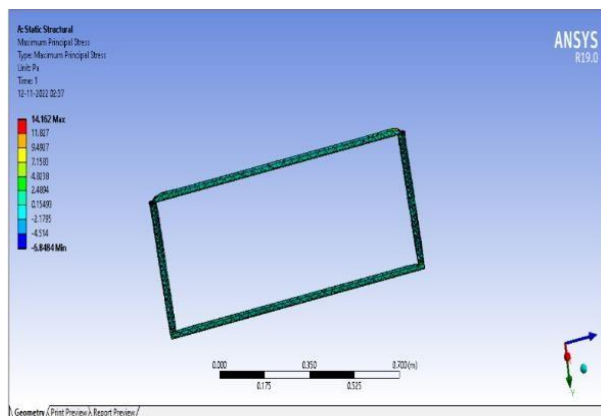


Figure 3.6 Maximum Principal Stress

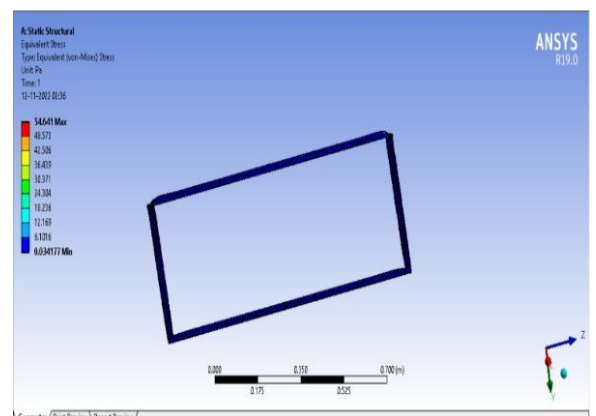


Figure 3.7: Equivalent stress

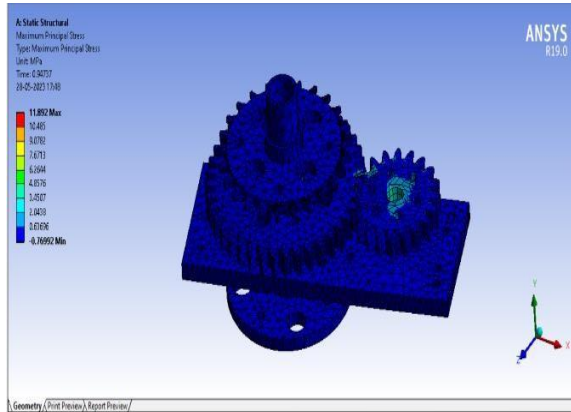


Figure 3.8 Maximum principle stress

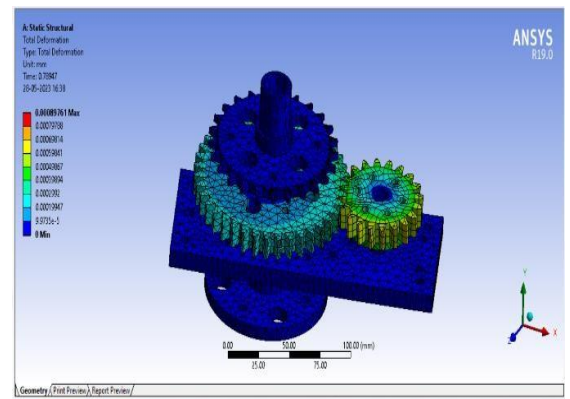


Figure 3.9 Total Deformation in gears

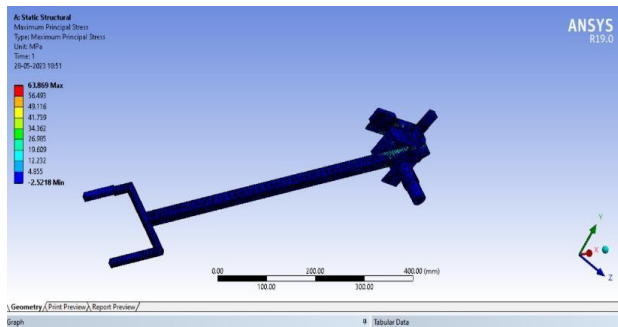


Figure 3.10 Maximum Principal Stress

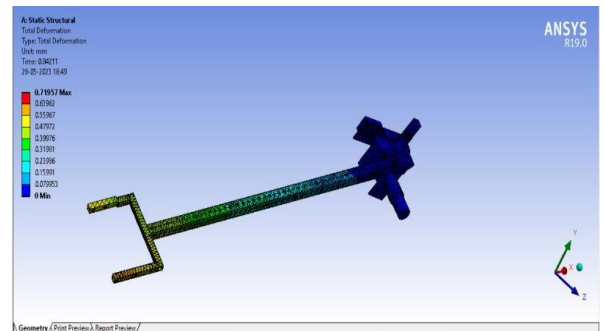


Figure 3.11: Total Deformation

Table 3.3.2: Results of FEA

Component Name	Parameter	Maximum Value
<b>Chassis</b>	$\sigma_{\text{principal}}$	14.162 Pa
	$\sigma_{\text{equivalent}}$	5.41 Pa
<b>Gear Assembly</b>	$\sigma_{\text{principal}}$	11.89 Mpa
	$\delta_{\text{total}}$	0.089 mm
<b>Mechanism</b>	$\sigma_{\text{principal}}$	6.3 Mpa
	$\delta_{\text{total}}$	0.71 mm

### 3.6. Chain Drive Calculations

#### 3.6.1 Chain Drive Calculations for E-Bike motor and wheel.

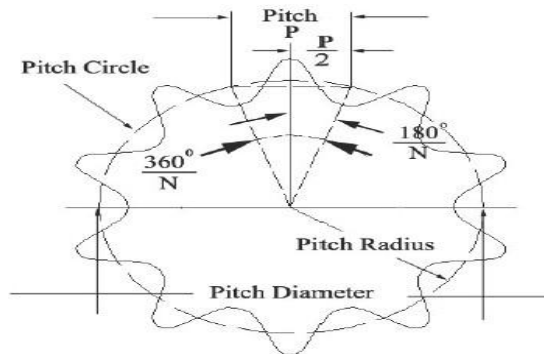


Figure 3.12 Gear and Chain meshing

$P = \text{Pitch}$

$D = \text{Diameter}$

As our E-Bike motor is already connected with one sprocket. Therefore, The sprocket attach to the motor has the given data as,  
Teeth on sprocket = 9

Pitch = 12.7 mm

From the Pitch and number of teeth we can find Diameter of sprocket.

$$P = 2 \left[ \frac{D}{2} \sin \frac{\theta}{2} \right]$$

Therefore,  $P = D \sin(\theta/2)$

Here,  $\theta = 360/z$  ;  $z = \text{No. of teeth}$

$$P = D \sin \left[ \frac{180}{z} \right]$$

Putting the values in above formula , we get

$$12.7 = D \sin (180/9)$$

$$12.7 = D \sin(20)$$

$$12.7 = D \times 0.342$$

Therefore,  $D = 12.7/0.342$  Hence,  $D = 37.13 \text{ m}$

The Diameter of sprocket attach to motor is 37.13 mm

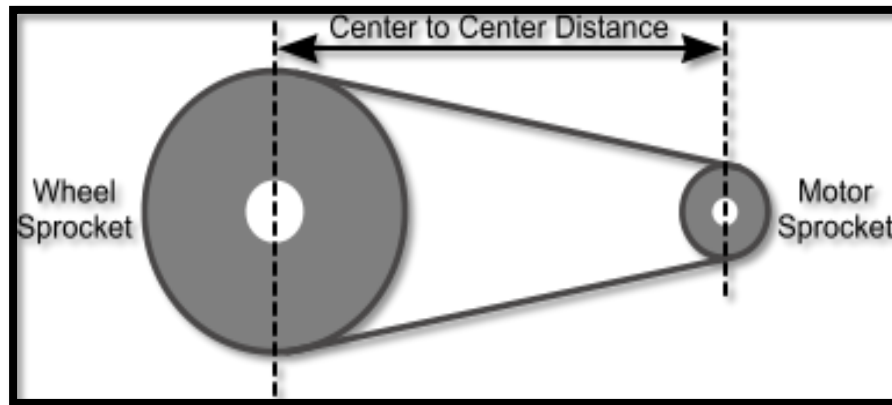


Figure 3.13 : Center to center distance between wheel and motor sprocket

The Sprocket 1 is attached with the motor shaft and sprocket 2 is attach with wheel shaft Therefore The speed of sprocket 1 will be equal to motor rpm.

Now, Calculating the speed of sprocket 2 which is connected to the wheel. As our gear ratio between the two sprocket is 3: 1

The number of teeth on sprocket 2 is 27 and number of teeth on sprocket 1 is 9 (Given)

And the rpm of motor is given as 324 rpm

$$N_1 = 324 \quad ; \quad Z_1 = 9 \text{ or } T_1 = 9 \quad N_2 = ? \quad ; \quad Z_2 = 27 \text{ or } T_2 = 27$$

By Using Formula:

$$N_1.T_1 = N_2.T_2$$

$$\ast N_2 = (N_1.T_1) / T_2$$

$$N_2 = (324 \times 9) / 27$$

$$\ast N_2 = 108 \text{ rpm}$$

Therefore, the speed of sprocket 2 will be 108 rpm.

The velocity ratio (i) of the chain drive is given by :

$$i = \frac{\text{Number of teeth on sprocket 2}}{\text{Number of teeth on sprocket 1}}$$

$$i = 27/9 \text{ Therefore } i = 3$$

Now Finding the length of chain.

For Finding the length of chain we require the Centre distance between two sprockets.

Centre Distance (C) = 300 mm (Given)

$$L = 2C + \frac{T_2 + T_1}{2} + \frac{(T_2 - T_1)^2}{4\pi^2 C}$$

Here , T2 = number of teeth on big sprocket.

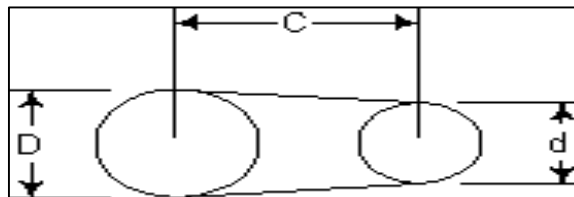
T1 = number of teeth on small sprocket.

$$L = 2(300) + (36/2) + (324/4\pi^2 \cdot 300)$$

$$L = 618.0859 \text{ mm}$$

Therefore the Length of the chain we get is 618.0859 mm

### 3.6.2 Chain drive calculations for turning the wheels:



N1 = Speed of sprocket 1 N2 = Speed of sprocket 2

T1 = Number of teeth on sprocket 1 T2 = Number of teeth on sprocket 2

As sprocket 1 is connected with the motor shaft to get the power for rotation, And the rpm for the motor is 500 rpm

The speed of the sprocket 1 is 500 rpm

$$N1 = 500 \text{ rpm}$$

And the teeth of the sprocket are known

$$N1.T1 = N2.T2$$

$$N2 = N1.T1 / T2$$

$$N2 = (500 \times 30) / 30$$

$$N2 = 500 \text{ rpm}$$

Therefore, the speed of the sprocket 2 will be 500 rpm

Now Calculating the Diameter of the Sprocket:

P = Pitch

D = Diameter

Teeth on sprocket = 30 (Given) Pitch = 12.7 mm (Given)

From the Pitch and number of teeth we can Find Diameter of the Sprocket.

$$P = 2 \left[ \frac{D}{2} \cdot \sin \frac{\theta}{2} \right]$$

Therefore,  $P = D \cdot \sin(\theta/2)$

Here  $\theta = 360 / z$

Z = Number of Teeth

$$P = D \cdot \sin(180/z)$$

Putting the values in above formula, we get  $12.7 = D \cdot \sin(180/30)$

$$12.7 = D \cdot \sin(6) \quad 12.7 = D \times 0.104$$

$$D = 12.7 / 0.104$$

Hence,  $D = 122.115 \text{ mm}$

The Diameter of sprocket attach to motor is 122.115 mm.

Now Finding the length of chain:

For Finding the length of chain we required the centre distance between the two sprockets.

Centre Distance (C) = 900 mm (Given)

Here, T2 = Number of teeth on sprocket 2 T1 = Number of teeth on sprocket 1

$$L = 2(900) + (60/2) + 0$$

$$L = 1800 + 30$$

$$L = 1830 \text{ mm}$$

Therefore the length of chain we get is 1830 mm for turning the wheels

### 3.7 Manufacturing and Testing Details:

Manufacturing of AWR starts with the development of Mechanical drive which consist of swerve mechanism for 360 degree movement. Mechanical drive consist of Driving wheel assembly and idler wheel assembly in which gears, sprocket and chains are used for transmitting the motion.

Mechanism for removing weeds is consist of rotor used for cutting the weeds from ground which is rotate by using motor. Mechanism consist of three motors, one for turning the mechanis, second for rotating the rotor arm and third for the rotating the rotor. 2:1 Gear ratio is used for turning the mechanism at require torque.

#### 1] Mechaical drive

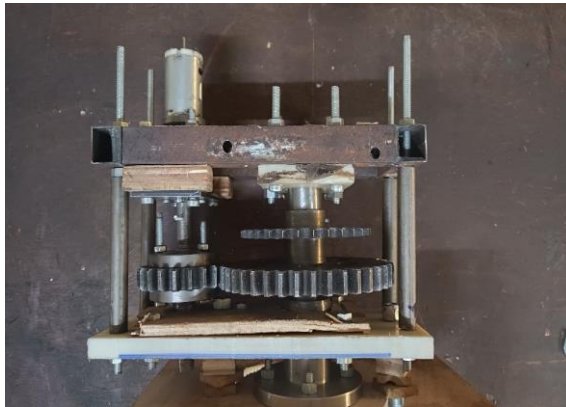


Figure 3.14 : Gear assembly for rotating driving wheel assembly



Figure 3.15 : Gear assembly for rotating idler wheel assembly



- Wheel assembly



Figure 3.16 : Wheel Assembly

Wheel assembly consist of gear assembly with sprocket, E-bike motor mounting for driving wheels and wheel mounting Gear assembly for turning driving wheels consist of one driving gear and driven gear with sprocket on same shaft. Chain drive is used for transmitting the motion from one wheel assembly to another parallel wheel assembly(idler). Wheel assembly consist of following components with their manufacturing and material details:

1] Flanges :

Material:- Mild Steel

Diameter:- 50,80 mm      Length:- 30,45,55,70 mm

Process used for manufacturing: Lathe turning, facing, Grinding

2] Bearing Housings

Material:- Nylon

Length:- 30,45,55,70 mm      Width:- 50,60 mm

Process used for manufacturing: VMC Milling, drilling, finishing.

3] Gears

Material:- Mild steel, N8

Gear ration: 2:1      Module: 3

Process used for manufacturing: Lathe turning, facing, Hobbing, drilling, finishing, Hardening.

### 3] Sprockets

Material:- Mild steel

ratio: 1:1    Module: 3

Process used for manufacturing: Lathe turning, facing, Hobbing, drilling, finishing, Hardening.

### 4] Laser cut parts

Laser cut parts includes motor plates, Wheel structure plates, encoder plates, motor base plates

Material:- Mild steel

Dimension according to CAD

Process used for manufacturing: Laser cutting

### 5] Chain

Material:- Alloy steel

3/8" chain for turning wheels and 08A chain for rotating wheels

Process used for manufacturing: Laser cutting.



Figure 3.17 : Mechanical drive

## 1] Mechanism

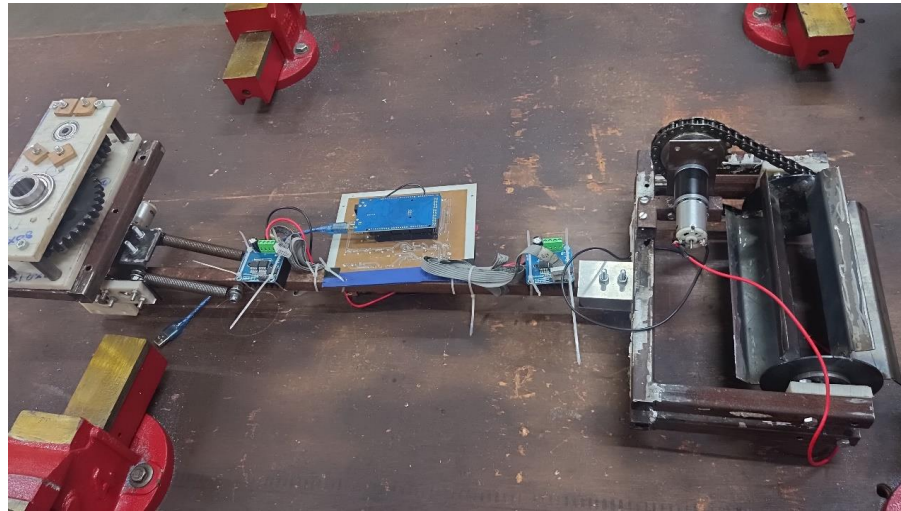


Figure 3.18 : Mechanism

Mechanism used for removing weeds is consist of rotor with 2 mm thick 6 blades for cutting weeds arm length is taken 550 mm from reference of surface with chassis for phase 1 prototyping the material taken for arm is Mild steel then after for final arm and other link material shifted to aluminium with 6061 grade. Mechanism consist of following components with their manufacturing and material details:

### 1] Bearing Housings

Material:- Nylon

Length:-210, 60, 70 mm      Width:- 80, 60 mm

Process used for manufacturing: VMC Milling, drilling, finishing.

### 2] Gears

Material:- Mild steel, N8

Gear ration: 2:1      Module: 3

Process used for manufacturing: Lathe turing, facing, Hobbing, drilling, finishing, Hardening, Milling, drilling, finishing.

## 4. Electronics hardware

### 4.1 Introduction

The electronic hardware plays a crucial role in the development and operation of autonomous weeding robots. It encompasses a wide range of components and systems that enable the robot to sense its environment, make intelligent decisions, and perform precise actions. In this section, we provide an overview of the electronic hardware used in autonomous weeding robots, highlighting its importance and key functionalities.

These components include sensors, actuators, microcontrollers or processors, power management systems, communication modules, and navigation systems. The sensors gather information about the environment, while actuators perform physical actions. Microcontrollers or processors process data and make decisions, and power management systems optimize energy usage. Communication modules enable data exchange, and navigation systems aid in precise positioning. The integration of these components is crucial for the successful development and operation of autonomous weeding robots in the agricultural industry. Below block diagram represents the integration of various electronic modules for the efficient control of AWR.

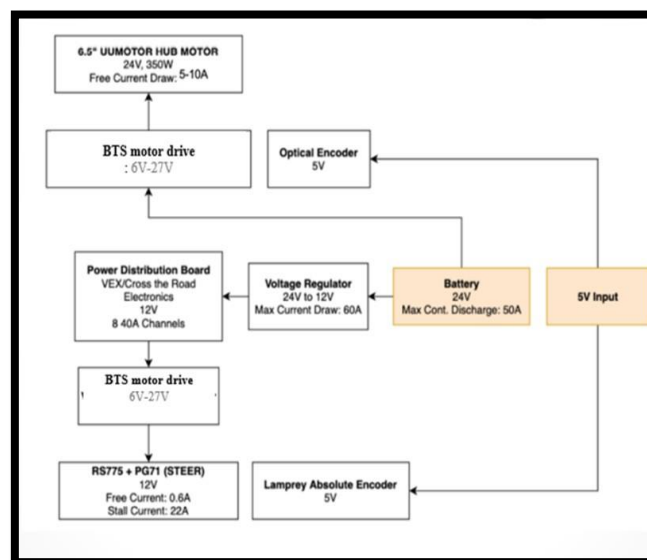


Figure 4.1 Block diagram representing the electronics architecture of AWR motor control

## 4.2 Component Details

### 4.2.1 Intel i3 processor:

The main processing of Robot Operating System (ROS) is done on the system equipped with 8 GB RAM, Intel i3 processor using Ubuntu 18.0 operating system. Here are some general specifications which are observed for using an Intel Core i3-6006U processor with ROS (Melodic):

#### Specifications:

1. Processing speed: 2.00 GHz
2. Cores: 2
3. Logical processors: 4



Figure 4.2: Intel i3 processor

\*(Note: It is not recommended to use system with above specifications but due to the cost and time constraints tests are performed on this system.)

### 4.2.2 Arduino Mega

The Arduino Mega's specifications make it suitable for controlling various aspects of an autonomous weeding robot, including sensor integration, actuator control, communication with other modules, and data processing. It provides a versatile and flexible platform for implementing the control system of the robot and integrating with other hardware components.



Figure 4.3: Arduino mega micro-controller

#### Specifications:

1. Microcontroller: ATmega2560

2. Operating Voltage: 5V
3. Input Voltage (limits): 6-20V
4. Digital I/O Pins: 54 (of which 15 provide PWM output)
5. Analog Input Pins: 16
6. DC Current per I/O Pin: 40 mA
7. DC Current for 3.3V Pin: 50 mA
8. Flash Memory: 256 KB of which 8 KB used by bootloader
9. SRAM: 8 KB
10. EEPROM: 4 KB
11. Clock Speed: 16 MHz

#### 4.2.3 BTS7960 Motor Driver

By integrating BTS7960 motor driver into the control system of an autonomous weeding robot, precise control over the weed removing mechanism can be achieved. The coordination between image processing and motor control allows the robot to accurately target and remove weeds, improving the efficiency and effectiveness of the weeding process in agricultural applications.

##### Specifications:

1. Input voltage: 6V-27V
2. Model: IBT-2
3. Maximum current: 43A
4. Input level: 3.3-5V
5. Control mode: PWM or level
6. Duty cycle: 0-100%



Figure 4.4: BTS7960 Motor driver

#### 4.2.4 Ebike Motor controller

With the help of an ebike motor controller for MY1016Z2 DC geared motor, the autonomous weeding robot can effectively control and regulate the movements of its E-bike motors. This allows for efficient navigation, precise control over speed and



torque, and the implementation of regenerative braking for improved energy efficiency (It is the future scope of the project and not utilised in current version of AWR).

**Specifications:**

1. Rated voltage: 24 v DC
2. Current limit: 33A
3. Rated power: 350 w
4. Matching motor: DC brushed motor
5. Under Voltage Protection: 20V



Figure 4.5: E-bike Motor Controller

**4.2.5 E-bike MY1016Z2 DC Geared Motor.**

According to the calculations the use of the MY1016Z2 DC Geared Motor in an autonomous weeding robot can provide a suitable propulsion system for the robot's movement. The MY1016Z2 is a commonly used electric motor in various applications, including electric bikes and small electric vehicles. Here are some key features of MY1016Z2 motor in the AWR.



Figure 4.6: E-bikeMotor

**Specifications:**

1. Output Power: 250 Watt.
2. Rated Voltage: 24V
3. Rated Speed: 360 RPM

4. Full load Current:  $\leq 13.4A$
5. Torque: 21 N.m

**4.2.6 RS775 DC Motor:** By utilizing the RS775 DC motor in a weed removing mechanism, the autonomous agricultural robot can effectively remove weeds from the field. The motor's power, torque, and speed control capabilities enable efficient and precise weed removal, contributing to the overall performance and effectiveness of the autonomous weeding robot.

**Specifications:**

1. Voltage: 12V DC.
2. Speed: 7000 RPM.
3. Torque: 5.29 N-cm.
4. Output Power: 13.20w.
5. Rated current (mA): 2000
6. Shaft diameter: 5 mm.
7. Shaft length: 16 mm.
8. No-load current: 0. 80A.



Figure 4.7: RS775 Motor

**4.2.7 9 DOF IMU Sensor:**

The Inertial Measurement Unit (IMU) sensor plays a crucial role in the Autonomous Weeding Robot by providing orientation estimation, motion tracking, stability control, and the potential for gesture-based control. By integrating the IMU sensor with other sensors (like GPS) and employing sensor fusion techniques, the AWR can achieve more accurate and reliable navigation, leading to improved performance and efficiency in weed detection and removal tasks.

**Specifications:**

1. Operating Voltage range: 3.0V – 5.0V (on-board low dropout regulator)
2. Module size 14.3mm \* 20.5mm (More compact than GY-80)
3. Resolution: 16bit AD converter, 16-bit data output
4. Gyroscopes range:  $\pm 250, 500, 1000, 2000^\circ / s$

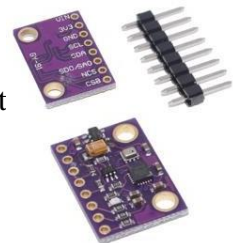


Figure 4.8 9Dof IMU



#### 4.2.8 NEO-M8N high precision GPS sensor

The NEO-M8N module provides high-precision positioning information with a high level of accuracy. It utilizes multiple satellite systems, such as GPS, GLONASS, Galileo, and BeiDou, to obtain precise location coordinates. This accuracy is crucial for navigation tasks where precise positioning is required. The main processing is provided by U-BLOX which comes in integrated form with receiver antenna module.



Figure 4.9 NEO M8N GPS Module

#### Specifications:

1. Operating Temperature Range (°C): 45 to 105
2. Tracking Sensitivity (dBm): 161 dBm
3. Co-Ordinate System: WGS-84Receiver
4. Type : 72-channel Ublox M8 engine.
5. Input Supply Voltage (VDC) : 3 ~ 5
6. Main Chip : Ublox NEO-M8N
7. Sensitivity (dBm) : 156
8. Position Accuracy (Meter):2,2.5 m

#### 4.2.9 Camera Module

The use of camera sensors in autonomous weeding robots enables accurate weed detection, real-time monitoring, environmental assessment, and integration with image processing and AI algorithms from our calculations and literature study we have selected the 1080P 2.1 Megapixel 30 FPS USB webcam as a vision sensor for gaining the high resolution inputs.



Figure 4.2 1080P 2.1 Megapixel 30 FPS USB webcam

**Specifications:**

1. Series : ZQ-1080
2. Frame rate: 30FPS
3. Colour: Black
4. Product Dimensions: 9.5 x 8 x 7 cm; 200 Grams
5. Video Capture Resolution: 1080p
6. Minimum Focal Length: 28
7. Lens Type: Wide-Angle

## **5. SOFTWARE**

### **5.1 Introduction**

The design of the robot's software is discussed in detail, focusing on key considerations such as sensor integration, path planning, obstacle avoidance, and control algorithms. The Robot Operating System (ROS) framework was used to create the fundamental software architecture. In addition, the system had a number of modules that were in charge of various functions such as localization, obstacle avoidance, and weed detection. The localization module combined data from the GPS and IMU sensors using sensor fusion techniques to precisely determine the robot's location and orientation. The obstacle avoidance module used LiDAR data to detect and avoid obstacles in the field. The weed detection module used image processing techniques to identify weeds in the field, and the weeding tool was activated to remove them. The modules were integrated and tested using simulation on Gazebo simulator along with the experimental evaluation.

### **5.2 Robot Operating System (ROS)**

ROS is an open-source framework that provides a set of software libraries and tools for creating and developing robotic systems. It was created at Willow Garage, a robotics research facility, and is presently maintained by the Open Source Robotics Foundation (OSRF). ROS offers a flexible and modular architecture that simplifies the development of complex robotic systems by providing a standardized platform for communication, sensor integration, perception, control, and more. To use ROS in agriculture automation, current agricultural machinery may be made ROS-compatible, or ROS-compatible robots can be introduced into agriculture.

For this project ROS Melodic Morenia is used. It is the 12th official release of the Robot Operating System (ROS) and is designed to work specifically with Ubuntu 18.04 (Bionic). It was released in May 2018 and is known for its long-term support (LTS) status, ensuring stability and compatibility for a significant period.

### 5.3 YOLO

YOLO is a popular deep learning algorithm for real-time object detection. It stands out for its speed and accuracy, making it suitable for applications that require quick and efficient object detection in images or video streams. YOLO breaks a picture into grids and immediately predicts bounding boxes and class probabilities. It has the ability to identify many things in a single pass, hence the term "You Only Look Once."

It processes the input picture using a combination of convolutional layers, pooling layers, and fully connected layers to predict the existence, position, and class of objects within the image. It runs on a fixed-size grid and executes convolutional operations over grid cells, directly predicting bounding boxes and class probabilities.

### 5.4 OpenCV

OpenCV (Open Source Computer Vision Library) is a popular open-source library for image processing and computer vision tasks. It provides a set of methods and algorithms that allow developers to efficiently process and alter photos and videos. Image processing methods supported by OpenCV include image filtering, feature extraction, picture segmentation, and object detection. The integration of YOLO and OpenCV allows the AWR to perform real-time weed detection and classification, enabling efficient and targeted weed management. It provides the ability to autonomously identify and respond to weed presence, allowing for effective and precise weed control in agricultural environments.

## **6. NAVIGATION AND CONTROL OF AWR**

### **6.1 Basic introduction to navigation:**

Navigation and control of AWR is a critical aspect of its overall operation. In order to accurately and reliably navigate through the fields, the robot must integrate information from multiple sensors, including GPS, wheel encoders, and IMU (Inertial Measurement Units).

GPS provides a global position estimate, but its accuracy can be limited by factors such as atmospheric conditions, signal blockage, and multipath interference. The IMU records the robot's acceleration and angular velocity, but its accuracy declines over time owing to drift. Encoders assess the distance traversed by the robot's wheels, however they are susceptible to slippage, wheel slide, and uneven terrain. To cope with these conditions, sensor fusion methods are utilised in conjunction with ROS.

### **6.2 Use of Robot Operating System:**

1. The basic prerequisite for any autonomous ground vehicle or autonomous vehicle is that it comprehend or receive information about its surrounding environment and interpret that information to locate itself in that world.
2. In our project we are using ROS (Robot Operating System) as a main processing unit for all the data we gathered by using various sensors.
3. The following processes are involved in making an agricultural equipment ROS- compatible:
  - Building its description model,
  - ROS sensor drivers, and ROS controller interfaces.
  - Developing custom ROS packages to enable the robot to perform the necessary agriculture tasks
4. The first stage/basic requirement for ROS implementation is to develop a simulated model or system to test our ROS packages. To do so, we must first build the (Unified Robot Description Format) URDF file, which holds the information created by the robot's CAD (for example, dimensions, size, appearance, and so on) in order to spawn or represent that robot model in the gazebo simulator.

5. Gazebo is a free and open source 3D robotics simulator. The ODE physics engine, OpenGL visuals, and sensor simulation and actuator control support code are all part of it. It supports a number of high-performance physics engines, including ODE, Bullet, and others (with ODE serving as the default). It creates realistic environments with high-quality lighting, shadows, and textures. It may mimic
6. sensors that can "see" the simulated environment, such as laser range finders, cameras (including wide-angle), Kinect-style sensors, and so on. It makes use of the OGRE engine for 3D rendering.
7. For getting the environmental data we are using the GPS sensors to localize the AWR in the field. It gives the latitude, longitude, and altitude of the robot body based on the parameters provided.
8. According to the observations and literature study we found that using single GPS sensor/receiver for the localization is not a good idea as its data gets affected by number of factors one of it is the reception impediment and noise which results in serious inaccuracy of positioning.

### **6.3 Sensor Fusion**

Sensor fusion of GPS, IMU, and encoders is done using the Extended Kalman Filter (EKF) in ROS. The EKF is a state estimation method that estimates the state of the system using a nonlinear model and data. In this scenario, the state comprises the robot's location, orientation, and velocity. The EKF technique employs a mathematical model that takes into consideration the robot's kinematics, the motion model of the sensors, and the noise characteristics of the sensors to fuse the sensor data. The model is updated with measurements from the sensors at regular intervals, producing an estimate of the robot's state. By combining the data from multiple sensors, the EKF can compensate for the limitations and errors of individual sensors, providing an accurate and reliable estimate of the robot's state. Also it can handle non-linear and time-varying systems.

The above method is performed with the help of robot\_localization package of ROS as it provides a framework for sensor fusion of GPS, IMU, and encoders using EKF. The package includes pre-built nodes for GPS, IMU, and encoder sensors, as well as a node for the EKF algorithm. The package also includes a configuration file for setting up the

sensor fusion algorithm, which allows the user to specify the sensor types, their noise characteristics, and other parameters.

In combination with GPS, IMU, and wheel encoders, a LiDAR is also used to improve the robot's obstacle avoidance capabilities. If an obstacle is detected in the robot's path, the robot can either stop and replan its path or adjust its trajectory to avoid the obstacle. This is achieved using a variety of obstacle avoidance algorithms such as potential fields, trajectory optimization, or model predictive control.

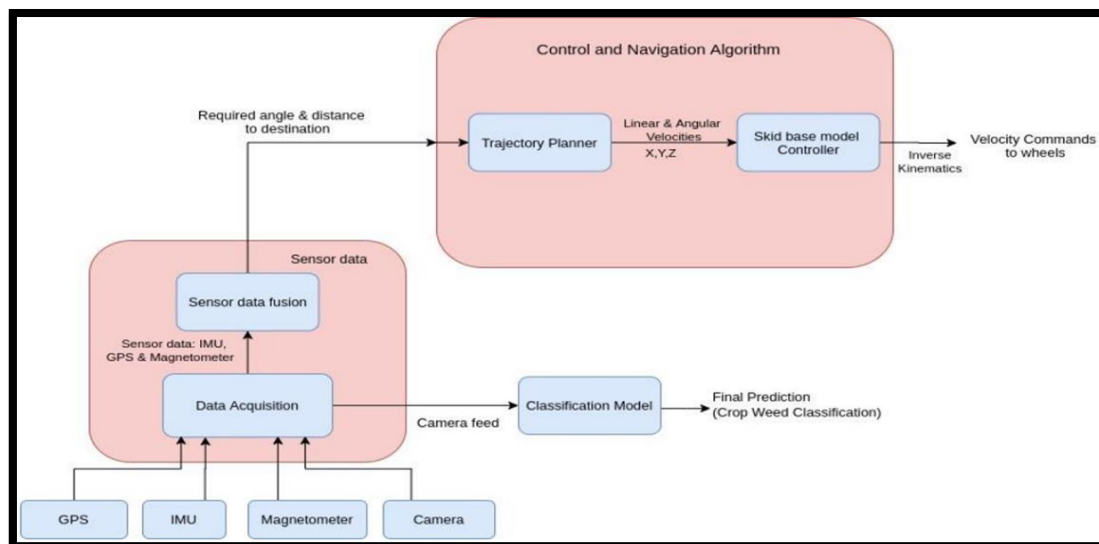


Figure 6.1 Block diagram representing the sensor data fusion and path planning

#### 6.4 Working of complete navigation system

The control and navigation algorithm in ROS for autonomous navigation of an Autonomous Weeding Robot (AWR) using sensor fusion of GPS data and IMU sensor typically involves the following steps:

1. **Sensor Data Acquisition:** Collect GPS and IMU sensor data from their respective sensors attached to the robot. The GPS sensor provides location coordinates (latitude and longitude), while the IMU sensor provides orientation (roll, pitch, and yaw) and acceleration data.
2. **Sensor Fusion:** Sensor fusion methods, such as the Extended Kalman Filter (EKF) or Unscented Kalman Filter (UKF), can be used to combine GPS and

IMU data. These filters aid in the integration of readings from both sensors, resulting in a more precise and trustworthy estimation of the robot's location and orientation.

3. **Localization:** Use the fused sensor data to estimate the robot's current position in the environment. This can be done by comparing the sensor data to a pre-built map or by employing SLAM (Simultaneous Localization and Mapping) techniques to create a map on-the-fly. ROS provides packages like `robot_localization` or `gmapping` for localization and mapping tasks.
4. **Path Planning:** Once the robot's position is determined, utilize ROS navigation stack, specifically the `move_base` package, for path planning. Define the robot's starting position and the desired goal location. The path planner generates a collision-free path based on the robot's current position, the goal, and the environment map.
5. **Obstacle Avoidance:** Incorporate obstacle avoidance algorithms to ensure that the planned path avoids any obstacles detected by sensors such as lidar or cameras. ROS provides packages like `costmap_2d` and `obstacle_layer` to assist in obstacle detection and avoidance.
6. **Motion Control:** Implement a motion control system using ROS controllers. Translate the desired path from the path planner into control commands for the robot's actuators, such as motors or servos. This allows the robot to execute the planned path accurately and smoothly.
7. **Feedback and Adaptation:** Continuously monitor the location and orientation of robot using the GPS and IMU data. Compare the actual position to the desired path and make necessary adjustments using feedback control mechanisms, such as PID controllers or model predictive control. This helps the robot adapt to any deviations or errors in its navigation.

## **6.5 Outline of path planning algorithm for AWR**

1. **Initialization:**
  - Initialize the robot's pose and set the initial GPS and IMU sensor readings.



- Load the map of the environment or perform SLAM (Simultaneous Localization and Mapping) to create a map.
2. Global Path Planning:
    - Use GPS data to determine the present position of the robot in the world coordinate system.
    - Based on the application requirements, determine the desired objective position.
    - Generate a high-level path from the present position to the target using a global planner, such as the move\_base package.
    - The global planner takes into account the map, obstacles, and any cost functions to find an optimal or near-optimal path.
  3. Local Path Planning:
    - Based on the high-level path generated by the global planner, the local planner generates a more detailed path that considers the immediate surroundings of the robot.
    - Utilize the IMU sensor data to estimate the robot's current orientation and incorporate it into the local planning process.
    - Use a local planner, such as the teb\_local\_planner or sbpl\_lattice\_planner, to generate a smooth and collision-free path that takes into account dynamic obstacles and terrain variations.
  4. Obstacle Detection and Avoidance:
    - Utilize sensor data, such as LiDAR or camera images, to detect and identify obstacles in the robot's vicinity.
    - Update the planned paths in real-time to avoid obstacles and ensure safe navigation.
    - Incorporate obstacle detection and avoidance algorithms into the path planning process to dynamically adapt to the environment.
  5. Motion Control:
    - Implement a motion control system that translates the planned paths into appropriate control commands for the robot's actuators, such as motors or servos.

- Use feedback from the GPS and IMU sensors to monitor the robot's position and orientation and adjust the control commands accordingly.
6. Execute and Iterate:
    - Execute the planned paths and continuously monitor the robot's navigation performance.
    - Evaluate the effectiveness and efficiency of the navigation system and iterate on the parameters and algorithms as needed.
    - Incorporate feedback and sensor fusion techniques to improve the accuracy and robustness of the navigation system.
  7. The needed speed information is obtained by inverse kinematics in the form of linear and angular velocities and, in accordance with the intended path, the controller produces driving speed for each wheel. These outputs serve as starting points for the necessary ROS controllers, which govern the position, velocity, and effort on each joint.

## 7. WEED DETECTION AND CLASSIFICATIONS:

Previously, robots lacked the capacity to recognise and classify items in real time. As image systems technology advanced, robots were outfitted with cameras. Plant segmentation was accomplished using colour cameras, whereas plant categorization was performed by spectral cameras. With the growth of deep learning and other image processing algorithms, it is now feasible to categorise many weed species and crops under outdoor illumination. Due to the rise in the development of CNNs, Encoder-Decoder segmentation architecture is used for training and classifying the vegetation mask. This mask along the bounding boxes is trained on a VGG 16 Network to classify the blobs extracted from the bounding box image into crops and weeds. This bounding box based segmentation and classification process is shown in the below figure.

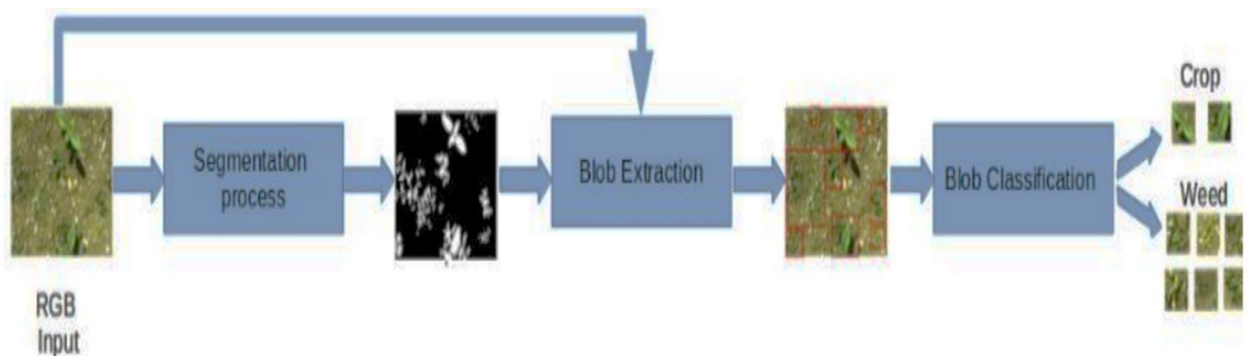


Figure 7.1 Block diagram representing the image processing steps

Weed detection using image processing is a key component of an AWR, as it allows the robot to accurately identify and target weeds for removal. The process of weed detection using image processing typically involves the following steps:

1. **Image acquisition:** The first step is to capture images of the agricultural field using a camera mounted on the robot.
2. **Preprocessing:** Once the images have been acquired, they need to be preprocessed to remove noise and enhance the features of interest. This may involve techniques such as image filtering, thresholding, and morphological operations.
3. **Feature extraction:** The next step is to extract the features of interest from the preprocessed images. In the case of weed detection, these features may include the color, texture, and shape of the weeds.

4.      **Classification:** Once the features have been extracted, they need to be classified as either weeds or non-weeds. This is possible through the use of machine learning techniques such as support vector machines, decision trees, and neural networks.
5.      **Post processing:** Finally, the results of the weed detection algorithm need to be post processed to remove false positives and refine the weed boundaries. This involves techniques such as edge detection, segmentation, and region growing

## **7.1      Details of weed detection and classification**

For classification of weed images, we have used Convolutional Neural Network (CNN) named YOLO for Feature extraction and dense layer with Random Forest for classification. Along with that the dataset used for the project is Open Sprayer Images dataset from Kaggle. It has the dataset with some data augmentation for the purpose of weeds classification, preprocessing, and also it includes images of docks with wide leaves and picture of the land without broad leaved docks. The Open Sprayer Project's authors have made it public on Kaggle. Open sprayer is an autonomous land drone project that seeks to self-drive over an agricultural field spraying for weeds using high accuracy GPS and deep learning.

### **7.1.1    Pre-processing:**

The dataset is uneven since the number of samples in one class is much lower than in the other. Data Augmentation is a prominent solution to this problem. Data Augmentation is a technique for creating extra data from existing data by utilising transformations such as scale, translation, rotation, and others. After treating the imbalance of data, we are ready to plug them into our feature extractor a CNN. Further pre-processing consists of cropping/resizing and normalizing images. Normalization of images helped in controlling the gradients during backpropagation.



Figure 7.2 Comparison of original image (on left) and preprocessed image (on right)

### 7.1.2 Feature Extraction:

Convolutional Neural Networks are an effective tool for obtaining picture characteristics. They have outperformed classic feature extraction approaches like HOG, Wavelet Transform, FFT, and SIFT, to mention a few. However, training CNNs from scratch may be highly computationally costly. Thus, for our purposes, we will employ a technique known as Transfer Learning.

Transfer Learning uses pre-trained CNNs to extract features, which are then fed into our classifier of choice to train on any dataset. These pre-trained CNNs are often taught on the massive ImageNet picture database, making them more powerful than a CNN trained from scratch in terms of extracting relevant features. We used InceptionV3 CNN among the numerous available pre-trained CNNs.

To utilize InceptionV3 as a feature extractor, we delete the fully linked layers at the end that include ImageNet-trained weights. As a result, the final layer is the Global\_average\_pooling\_2d layer, which produces picture features in the shape of [batch size, 2048]. These characteristics may now be classified using any classifier of our choice.

### 7.1.3 Classifier

In machine learning, a classifier is a model that learns to categorize data into one or more classes. In this task, we compared four different classifiers using a combination of dense layer and various other classification algorithms such as Softmax, SVM, Random Forest, and Decision Tree.

1. Classifier using Dense Layer and Softmax : The Softmax classifier is widely used in deep learning. It's a multi-class classifier that generates a probability distribution across classes. We will utilise a dense layer and the Softmax activation function to construct the classifier. The dense layer will extract a collection of features from the input data, which will then be sent to the Softmax algorithm, which will generate the class probabilities.
2. Classifier using Dense layer and SVM: SVM is a linear classifier that separates data into two groups by calculating the optimum hyperplane that maximises the margin between the two classes. We will use a dense layer and the SVM technique to construct the classifier. The dense layer will extract a collection of features from the input data, which will be delivered to the SVM algorithm for classification.
3. Classifier using Dense Layer and Random Forest: Random Forest is a decision tree-based method that predicts using an ensemble of decision trees. It is a strong classifier that can deal with both numerical and categorical input. We will use a dense layer and the Random Forest technique to construct the classifier. The dense layer will extract a collection of features from the input data, which will then be fed into the Random Forest algorithm, which will do the classification.
4. Classifier based on Dense Layer and Decision Tree: The Decision Tree method is a simple yet effective technique that works by recursively separating the data into smaller groups depending on the most discriminative characteristics. We will use a dense layer and the Decision Tree approach to construct the classifier. The dense layer will extract a collection of features from the input data, which will then be sent into the Decision Tree algorithm, which will conduct the classification. The accuracy, f1 score, and AUC score of each model are

calculated and the classifier combination providing highest accuracy and scores is selected for the weed detection and classification of weeds in AWR.

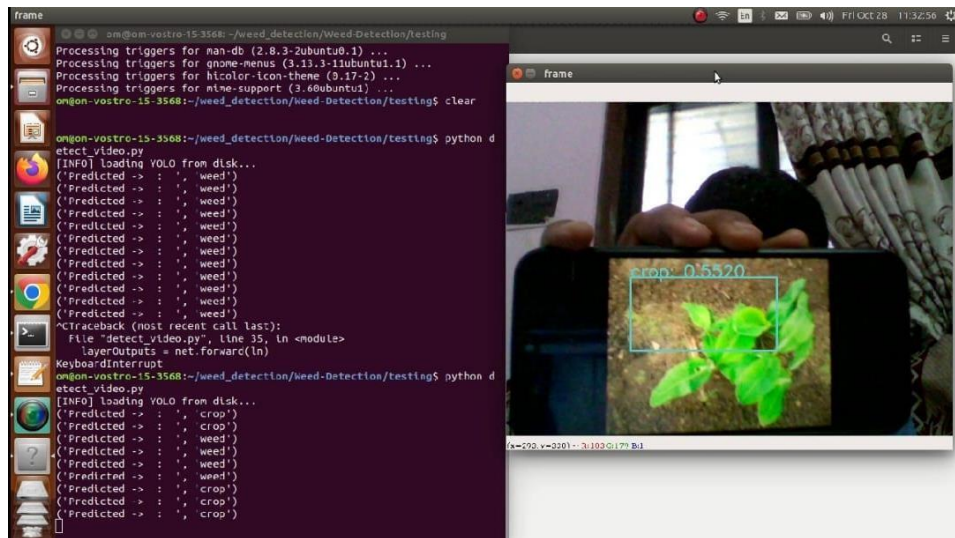


Figure 7.3 Image representing the testing of the CNN model on the images obtained from the internet

## 8. SIMULATION AND EXPERIMENTAL SETUP

### 8.1 Simulation Setup

The complete AWR is simulated using the Gazebo simulation environment, which is supported by ROS. The Gazebo simulator provided dynamics simulation by taking gravity, friction, and contact forces into account, as given by the integrated physics engines Open Dynamics Engine - ODE. To construct control interfaces and sensor systems for AWR, a number of Gazebo plugins have been enabled. The following components make up the complete system

#### 8.1.1 Robot model description

The robot model is a crucial component of the simulation setup for an autonomous weeding robot in Gazebo. It is important to create or choose a robot model that accurately represents the physical robot being simulated. It includes all the physical components of the robot, such as the body, wheels, arms, and any other attachments. Along with that it also includes details such as the weight, dimensions, and center of gravity of the robot, as these can affect the robot's performance in the simulation. A robot, as a dynamic entity in the environment, is made up of links that are connected together by joints. A link in Gazebo represents the kinematic and dynamic properties of a physical link in the form of visual/collision geometry and inertia information. A joint model represents the kinematic and dynamic properties of a joint, such as joint type, motion axes, and joint safety constraints. All of this information is stored in the Universal Robotic Description Format

- URDF file format, which allows us to sketch out design restrictions. It is an XML-based file format used to describe the physical and kinematic properties of a robot. The modelled AWR consists of ten links including map, odom, 1base\_link and its 7 child links which includes each gps,imu,camera sensor and 4 agricultural wheels. Along with that it also consists of the 9 joints describing relation between base\_link and its child\_links. The overall mechanical design and the kinematics of AWR is discussed in section 2 of the paper. Some ROS third-party applications, such as navigation, need a virtual connection called base\_footprint, which is put on the ground and has the same size as that of AWR's footprint.



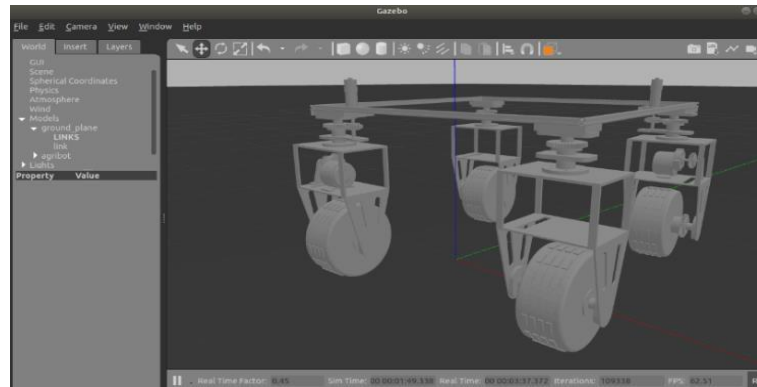


Figure 8.1 Spawned AWR in Gazebo Simulator according to the generated URDF

### 8.1.2 Environment description:

Creating a realistic and accurate simulation environment is an essential aspect of developing and testing autonomous robots. In the case of the autonomous weeding robot, the simulation environment must include a variety of features that replicate the conditions and challenges of real-world agricultural environments. Some key aspects of the environment created for AWR are,

**Terrain:** The simulated environment in Gazebo includes realistic terrain features that mimic real-world agricultural fields. This includes variations in soil type, elevation, and roughness, which can affect the performance of the robot's wheel encoders and overall movement.

**Vegetation:** The environment includes realistic vegetation models that replicate the appearance and behaviour of different types of crops and weeds. This allows researchers to test the robot's ability to distinguish between crops and weeds, as well as its ability to navigate through dense vegetation.

**Lighting:** Realistic lighting conditions are critical for accurate simulation of the robot's camera sensor. The Gazebo environment includes different lighting conditions, including various times of day and weather conditions, to test the robot's performance under different lighting scenarios.

**Obstacles:** The environment includes obstacles such as rocks, trees, and other obstacles that the robot must navigate around. This tests the robot's obstacle avoidance and navigation capabilities.

All these models are custom made using the blender software and spawned in the gazebo simulation environment.

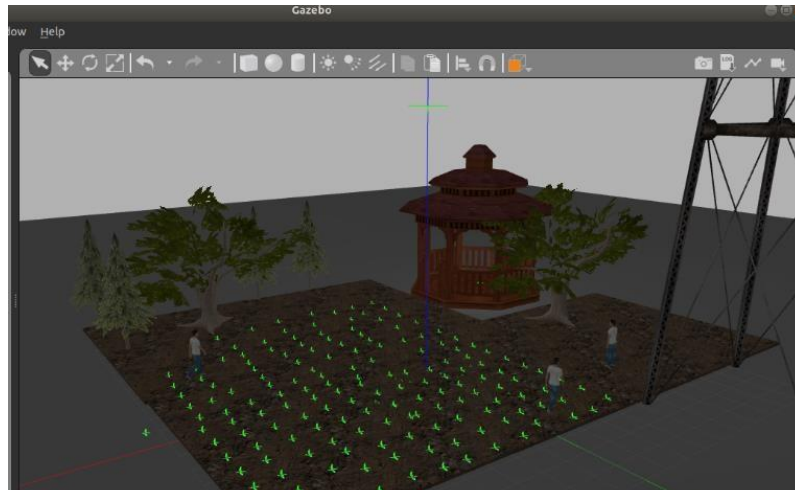


Figure 8.2 Virtual Environment created and simulated on gazebo simulator

### 8.1.3 Addition of sensors:

We can simulate actual sensor data and assess the robot's performance in a range of circumstances by adding GPS, IMU, and camera sensors to the robot model in Gazebo simulator. As a connection, several Gazebo plugins for each sensor are physically coupled to the robot model. The URDF file contains these plugins. The following approach is used to add sensors in a simulated environment:

1. Adding sensor plugins to the URDF: Sensor plugins is added to the robot's URDF model at the links where the sensors are placed. These plugins imitates sensor output and post sensor data to ROS topics.
2. Configure the sensor plugins: These sensor plugins are configured to define the sensor type, update rate, noise characteristics, and other attributes. The plugin type is adjusted to "gps" or "imu" for GPS and IMU sensors, respectively, and the sensor noise settings are changed to imitate real-world sensor failures.
3. Launch Gazebo with ROS: To run the simulation in Gazebo, the robot model can be launched with ROS. This will start the Gazebo simulator and create ROS topics for the sensor data.

4. Subscribe to sensor data topics: Nodes in ROS can subscribe to sensor data topics in order to receive and process sensor data. For example, GPS data may be used to locate the robot in the environment, IMU data can be used to estimate attitude, and camera data can be used to identify and track objects.

5. Real-time sensor data visualisation in RViz: RViz is used to visualise sensor data in real-time. For example, the GPS data is shown as a point cloud, the IMU data as a 3D orientation arrow, and the camera data as an image stream.

After creating all these parameters we successfully created the simulation of autonomous weeding robot in gazebo simulator. The figure 8.3 represents the entire environment and the robot model spawned at the start position or origin with all the sensors mounted on it.

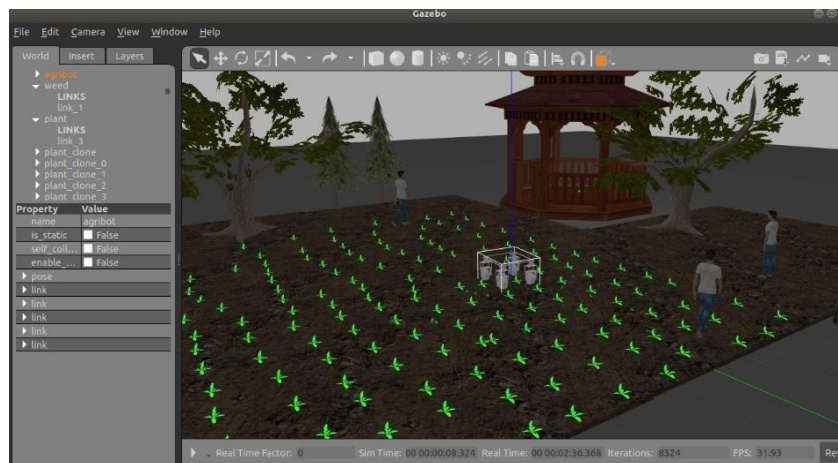


Figure 8.3 Entire simulated environment and robot model

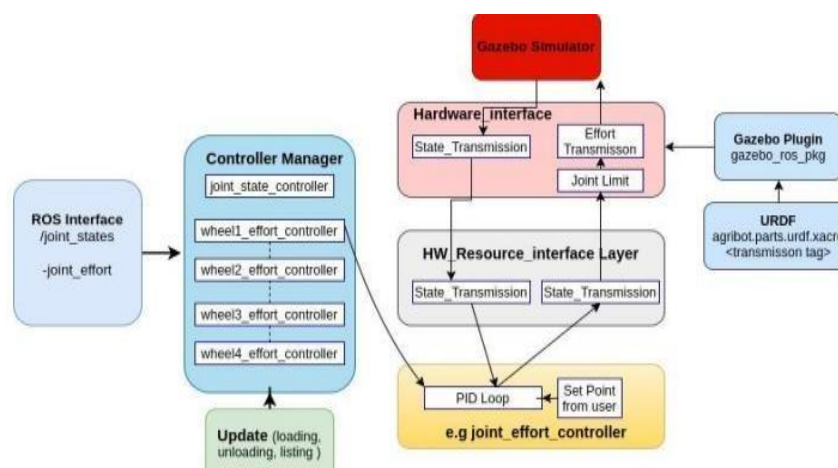


Figure 8.4 Block diagram architecture of the entire simulated environment in gazebo

## 8.2 Experimental/Physical Setup

An experimental setup with different types of crops and weeds is created for the testing of the physical model of the autonomous agribot's navigation, weed classification and removal. Cameras are mounted at different angles for capturing the movement below figure shows the entire setup created for the testing of AWR.






Figure 8.5 Experimental setup for testing of AWR

Different types of weeds and crops are gathered from the agricultural area for proper testing and validation of the CNN Model below table shows the list of crops and weeds used while testing the AWR.



Table 8.1 Table showing various types weeds used in experimental setup

Type of plant	Description
Weed	
	<p>Name : Common Ragweed (<i>Ambrosia artemisiifolia</i>)</p> <p>This weed can cause significant yield losses in crops such as corn and soybeans due to competition for resources and allelopathic effects.</p>
	<p>Name: Palmer Amaranth (<i>Amaranthus palmeri</i>)</p> <p>It is known for its rapid growth and resistance to herbicides, Palmer amaranth is a highly invasive weed that can outcompete crops and reduce yields.</p>
	<p>Name: Canada Thistle (<i>Cirsium arvense</i>)</p> <p>This perennial weed can quickly colonize agricultural fields and reduce crop productivity. Its extensive root system makes it challenging to eradicate.</p>

## ACTUAL PHOTOGRAPHS OF AWR FROM INITIAL STAGE TO CURRENT STAGE



## 9. RESULTS

### 9.1 Image processing outputs

Before applying the image processing model to the AWR system, four types of image processing models are compared according to the various parameters like accuracy, f1 score, AUC score, etc. Below table shows the comparison between each models on different parameters.

Table 9.1 : Comparison between each models based on different parameters.

	<b>CNN + SOFTMAX</b>	<b>CNN + SVM</b>	<b>CNN + RANDOM FOREST</b>	<b>CNN + DESECCION TREE</b>
<b>TRAIN</b>	0.948	0.976	0.976	1.0
<b>ACCURACY</b>				
<b>VALIDATION</b>	0.894	0.928	0.934	0.916
<b>ACCURACY</b>				
<b>F1 SCORE</b>	0.932	0.956	0.960	0.948
<b>AUC SCORE</b>	0.929	0.946	0.938	0.866
<b>AVERAGE</b>	0.967	0.981	0.973	0.941
<b>PRECISION</b>				
<b>SCORE</b>				

In addition, a ROC (receiver operating characteristic) curve is shown to analyse classifier performance. A ROC graph is a graph that shows the false positive rate on the X axis and the true positive rate on the Y axis. In many cases, a classifier contains a parameter that may be changed to improve TP at the price of raising FP or to improve TP at the expense of reducing FP. Each parameter setting produces a (FP, TP) pair, and a series of such pairings may be used to produce a ROC curve. According to the studies, the ROC curve or point is independent of class distribution or mistake costs. A ROC graph provides all of the information included in the confusion matrix since FN is the complement of TP and TN is the complement of FP. These curves are a visual tool for evaluating the tradeoff between a classifier's ability to correctly identify positive instances and the number of negative cases that are incorrectly classified.

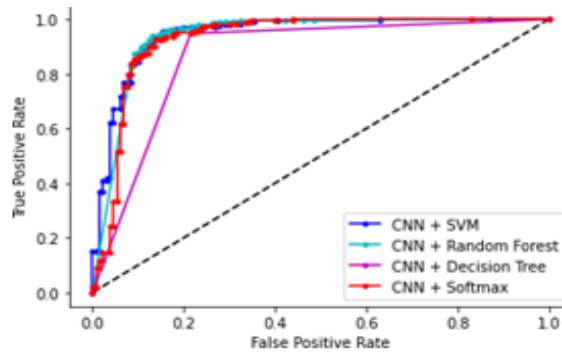


Figure 9.1 ROC graph showing the performance of classifiers

All of the four models are trained on total 1176 docks with wide leaves folder and 4851 images of land without docks with wide leaves folder of the Open Sprayer Project dataset. Along with that it is validated on total 670 images obtained from the same dataset. After following these processes the combination of CNN and Random forest is considered as a prior classification model among all four models.

So after selecting the classifier model an on field tests are performed for checking its accuracy on the local field conditions. From our tests we got around 70% accuracy while classifying and detecting the weeds from the 150 images acquired from cabbage, wheat, & sunflower, fields.

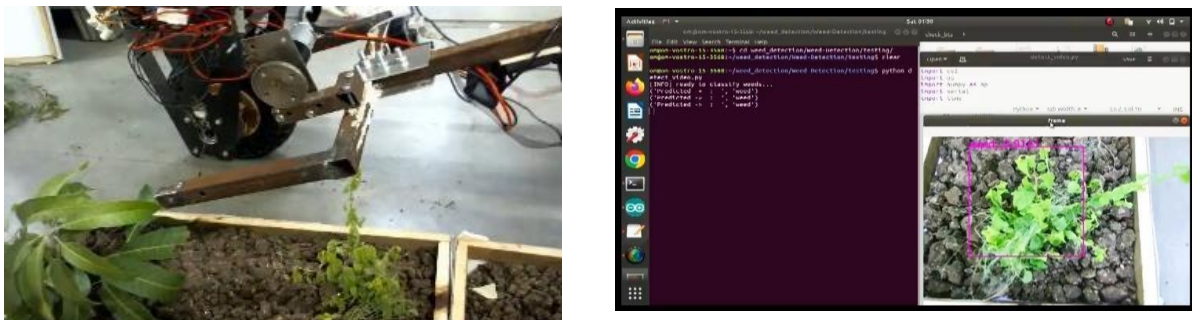


Figure 9.2 Output showing Classification of weed and crop using CN

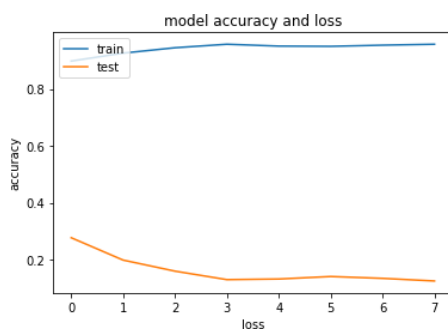


Figure 9.3 Accuracy verses loss graph showing how well the model fits the data



## 9.2 Navigation Results

### 9.2.1 Comparison of navigation based GPS and fusion of sensor data

During the navigation of an Autonomous Weeding Robot (AWR), GPS and IMU data are continuously obtained to provide essential information for position estimation, orientation tracking, and navigation control. Here is a description of GPS data and IMU data obtained during the navigation of an AWR:

#### 1. GPS Data:

- **Position:** GPS data includes latitude, longitude, and sometimes altitude information. These values represent the AWR's current global position on the Earth's surface.
- **Timestamp:** Each GPS data point is typically associated with a timestamp, indicating the time at which the position measurement was obtained.
- **Accuracy:** GPS data may also provide accuracy estimates, such as horizontal and vertical dilution of precision (HDOP, VDOP), which indicate the quality and precision of the GPS position measurement.

#### 2. IMU Data:

- **Acceleration:** IMU data includes measurements of linear accelerations about all three coordinate axes. These measurements represent the AWR's acceleration in three-dimensional space.
- **Angular Velocity:** IMU data also includes measurements of angular velocities or rotational rates around the x, y, and z axes. These measurements provide information about the AWR's orientation changes.
- **Orientation:** In our case, IMU data such as Euler angles (roll, pitch, and yaw) or quaternions, which describe the AWR's orientation in space are also taken into account.

GPS measurements alone may be affected by factors such as signal interference, multipath effects, or signal blockage in certain environments, leading to inaccuracies. So to avoid such inaccuracy its data is fused with IMU sensor. Below graph shows the navigation accuracy verses time step curves.

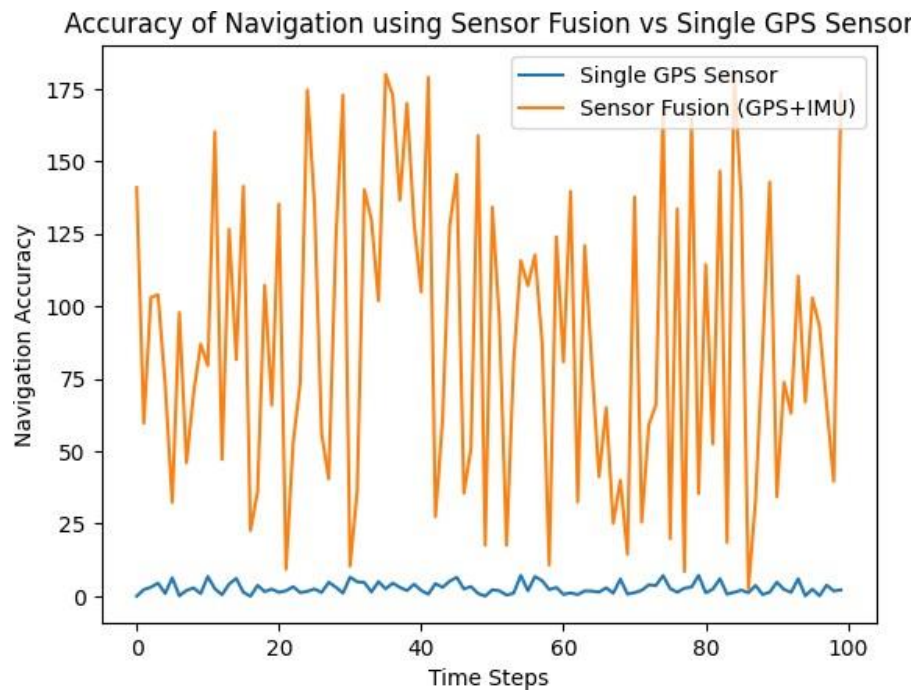


Figure 9.4 Graph of Navigation accuracy w.r.t Time between Single GPS sensor and Fusion of GPS and IMU sensor

So from the above graph we found that by combining the GPS and IMU data, the AWR can obtain more accurate and robust navigation information, enabling it to autonomously navigate its environment, detect and avoid obstacles, and perform precise weed removal tasks after successful detection.

### 9.2.2 Autonomous navigation in simulated environment:

Navigation in a simulated environment based on GPS and IMU sensor plugins in Gazebo involves integrating virtual representations of these sensors into the simulation to mimic their behaviour and provide navigation information. The results obtained from the simulated environment are as follows:

1. At first the teleoperation is performed to map the obstacle present in the path and along with that the GPS waypoints are set and stored relative to a global coordinate system or coordinate system of global map. The waypoints are

modelled as nodes in a graph, each with a route to the next. The shortest path that ensures all GPS waypoints and obstacles are met Dijkstra's algorithm. Prior to the robot travelling through the course, the sequence in which the waypoints will be navigated is decided. Once the GPS coordinates are put into the system, the route is quickly designed.

2. In the figure 9.5 blue line (in left image) shows the plotted trajectory (or designed route) according to the waypoints.

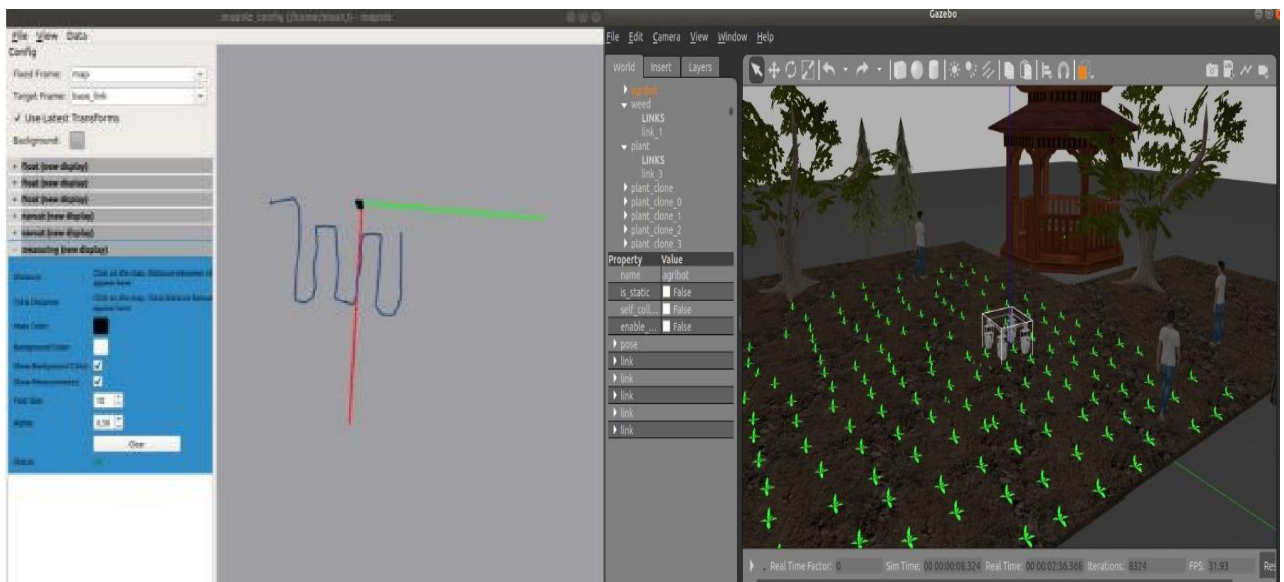


Figure 9.5 Field traversing in simulated environment (on right side) with visualization of path using mapviz (on left side)

3. After the path is set the trajectory planner algorithm plans the local path between each waypoint. Below figure shows the mapviz visualization between each waypoint in two cases. In first case only one motion of the AWR i.e. either linear or rotational motion is controlled. And in second case both linear and rotational movement is controlled by the controller according to the planner.

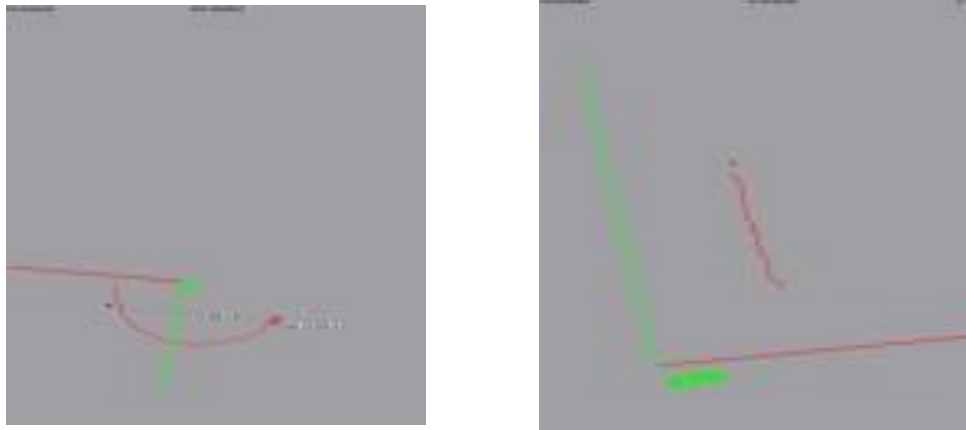


Figure 9.6 Mapviz visualization of first case on left and second case on right of the plots

4. From the above figure 9.6 we found that in first case the resultant trajectory may fail sometimes to reach the destination point and the traversed trajectory is a curve not a straight line which increases the total navigation time. While in the second case the simultaneous control of both results in successfully reaching the destination point by following the shortest path between each waypoint along with the obstacle avoidance.
5. Final traced trajectory by AWR is shown in below figure 9.7 where blue dotted line shows the path traced by AWR according to the global coordinates or waypoints provided in initial phase

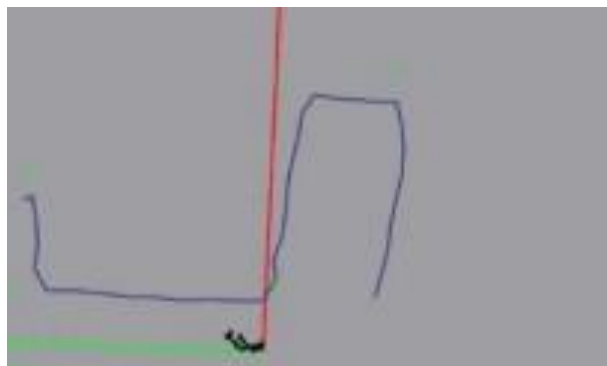


Figure 9.7 Final traced path by AWR by following the second case.

## 10. CONCLUSIONS

From the observations, literature survey and experimental results obtained through conducted tests, it is found that the development of autonomous weeding robot represents a significant advancement in the field of agricultural robotics. By following the planned methodology, a project development concludes in providing following effective outcomes:

1. After performing several tests of weed removing mechanism on the experimental setup we found that the developed CNN model works perfectly in positioning plant with the selected classifier which classifies the weeds from the crops while navigating in the setup.
2. Out of 12 runs of entire AWR carried on the experimental setup with different configuration of plants and weeds each time. AWR is successful in detecting, positioning, classifying and removing the weeds in 9 runs or tests.
3. According to the observations done on the experimental setup the developed mechanical drive is stable and provides good support to different equipments like sensors, controllers, laptop ,weed removing mechanism,etc mounted on it with a net weight of 48 Kg. Along with that the drive mechanism made using chain drive also provides the smooth movement to AWR.
4. The integration of GPS and IMU sensor with acceleration range  $\pm 8g$  allowed more accurate localization and positioning of the AWR. The graph shown in figure clearly presents that the sensor fusion approach (GPS+IMU) provides much better or centimetre level navigation accuracy than that of the single GPS sensor-based approach which provide the errors in meters as compared to with respect to each time step.
5. Along with that the graph (figure ) also demonstrates, how the accuracy of navigation varies over time. It provides insights into the performance of the sensor fusion algorithm and the single GPS sensor-based approach throughout the operation of the autonomous system. Also by analysing the graph, we can identify any patterns or trends in the accuracy and make further improvements to the navigation system if needed.

6. The use of Lidar in obstacle detection contributes to improved safety and reliability of the AWR's navigation system. According to the test carried on experimental setup the A1 LiDAR enables the robot to identify obstacles in real-time, avoiding collisions and potential damage to the robot itself and the surrounding environment.

So in summary the development of the autonomous weeding robot showcases the potential for robotics and automation to revolutionize the agricultural sector. The successful implementation of various technologies, algorithms, and hardware components demonstrates the feasibility and effectiveness of the robot in addressing the challenges of weed management. As further advancements are made in this field, autonomous weeding robots have the potential to improve agricultural productivity, reduce labor costs, and promote sustainable farming practices.

## 11. FUTURE SCOPE

The autonomous weeding robot is an emerging technology that has the potential to revolutionize the agriculture industry. As the technology evolves, there are many possible future directions for its development and application. Some future scopes which are decided by us for autonomous weeding robots are

1. Multi-spectral imaging: Integrating multi-spectral imaging with the weeding robot can help in identifying and distinguishing crops from weeds with greater accuracy.
2. Increased coverage: The size of the weeding robot can be increased to cover larger areas of land in a single pass. This will help to reduce the time and effort required for weeding
3. Integration with other technologies: The weeding robot can be integrated with other technologies such as drones, soil and weather monitoring sensors to provide a comprehensive solution for precision agriculture.
4. Solar powered: The integration of solar power in AWRs presents an exciting avenue for sustainable and energy-efficient farming practices. The advancements in solar panel technology, coupled with efficient energy management systems, can pave the way for more environmentally friendly and economically viable solutions in the future.
5. Cost reduction: As the technology advances and becomes more widespread, the cost of the weeding robot is likely to decrease, making it more accessible to farmers.
6. Customization: The weeding robot can be customized according to the needs of individual farmers and crops.

## 12. BUDGET

	Consumables	Budget
<b>Mechanical</b>	Material(Al,Ms,SS,Polymer,etc)	5000/-
	Motors	20000/-
	Tyres	10000/-
	Fabrication	10000/-
<b>Electronics</b>	Sensors(RTK-GPS, LIDAR, Optical sensor)	26000/-
	Power Distribution	1500/-
	Motor drivers	4800/-
	Microcontroller	800/-
	PCB's	1600/-
	Connectors	1000/-
<b>Power Source</b>	Solar Panel	20000/-
	Battery	25000/-
	<b>Contingency</b>	<b>Approx. 5000/-</b>
	<b>Total</b>	<b>1,30,700/-</b>



### 13. References

1. Statistical study conducted by Tamil Naidu Agricultural University on weed management, (Source [https://agritech.tnau.ac.in/agriculture/agri\\_weedmgt\\_culturalmethod.html](https://agritech.tnau.ac.in/agriculture/agri_weedmgt_culturalmethod.html))
2. Abhishek Singh, et al, "Structural Analysis of Ladder Chassis for Higher Strength", International Journal of Emerging Technology and Advanced Engineering, ISSN: 2250-2459, Volume 4, Issue 2, February 2014.
3. Patel Vijaykumar, et al, "Structural Analysis of Automotive Chassis Frame and Design Modification for Weight Reduction", International Journal of Engineering Research & Technology, ISSN: 2278-0181, Volume 1, Issue 3, May 2012.
4. Vishal Francis, et al, "Structural Analysis of Ladder Chassis Frame for Jeep Using Ansys", International Journal of Modern Engineering Research, ISSN: 2249-6645, Volume 4, Issue 4, April 2014.
5. Monika S. Agarwal, et al, "Finite Element Analysis of Truck Chassis", International Journal of Engineering Sciences & Research, ISSN: 2277-9655, December 2013
6. M. Aitkenhead, I. Dalgetty, C. Mullins, A. McDonald, and N. Strachan, "Weed and crop discrimination using image analysis and artificial intelligence methods," Computers and Electronics in Agriculture, vol. 39, no. 3, pp. 157–171, 2003.
7. K. A. B. Vijay and R. Cipolla, "Segnet: A deep convolutional encoder-decoder architecture for image segmentation," arXiv preprint arXiv:1511.00561, 2015.
8. T. F. Burks, S. A. Shearer, R. S. Gates, and K. D. Donohue, "Backpropagation neural network design and evaluation for classifying weed species using color image texture," Transactions of the ASAE, vol. 43, no. 4, pp. 1029–1037, 2000.
9. F. Feyaerts and L. V. Gool, "Multi-spectral vision system for weed detection," Pattern Recognition Letters, vol. 22, no. 6-7, pp. 667–674, 2001.
10. S. N. G. Stefan, P. Jeremy and L. Geoff, "Real-time procedural generation of 'pseudo infinite' cities," in Proc. of the 1st international conference on Computer graphics and interactive techniques in Australasia and South East Asia, 2003.
11. H. Hattori, Y. V. N. Boddeti, K. M. Kitani, and T. Kanade, "Learning scene-specific pedestrian detectors without real data," in Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), 2015.