IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

RESCUE BOT WITH ROTATING LEG MECHANISM

Sanket Singh, Kartik D, Nirendra Patel, Armaan Ali

Dept: Robotics and Automation College: Parul institute of technology, Vadodara

Email: sanketspsingh@gmail.com

Guided: Dr. Prabodh Kumar Sahoo (Associate professor)

Keyword: Rescue people from earthquake, building collapse, gather information of victims.

ABSTRACT

In disaster-stricken environments, such as earthquake-affected zones or collapsed structures, conventional wheeled robots often face mobility challenges due to uneven and obstructed terrain. To address this, we propose a **Rescue Bot** featuring a **rotating leg mechanism**, designed for enhanced traversal over obstacles.

The robot employs a **quadrupedal wheeled-leg system**, where each leg consists of a rotating mechanism inspired by RHex-style motion. This design enables adaptive movement, allowing the bot to maneuver efficiently over debris, climb small obstacles, and maintain stability on rugged terrain. The bot's **lightweight yet robust frame** provides durability while ensuring ease of deployment.

Key functionalities include:

- **Rotational leg-based locomotion** for seamless movement across uneven surfaces.
- **Compact and modular design** to navigate confined spaces in rescue missions.
- **Scalability for autonomous or remote-controlled operations**, making it suitable for real-world disaster relief scenarios.

This design aims to contribute to the advancement of robotic rescue operations, offering improved terrain adaptability and accessibility in critical rescue missions. Future iterations may include AI-driven path planning and real-time feedback mechanisms for optimized performance

CHAPTER I

INTRODUCTION

1.1 Overview

Natural disasters and hazardous environments often present extreme challenges for human-led rescue operations. Traditional wheeled and tracked robots struggle to navigate through uneven, debris-filled terrain, limiting their effectiveness in critical missions. To address these challenges, we present a **Rescue Bot** equipped with a **rotating leg mechanism**, designed to enhance mobility in disaster-stricken areas.

This robotic system integrates a **quadrupedal wheeled-leg mechanism**, enabling it to traverse rough surfaces, climb over obstacles, and maneuver through confined spaces where conventional robots may fail. The design draws inspiration from bio-inspired robotics, incorporating an innovative rotating leg system that provides both stability and adaptability.

1.2 Problem Definition

Disaster response operations in environments such as **earthquake-affected zones, collapsed structures, and rugged terrains** present significant mobility challenges for both human rescuers and traditional robotic systems. Conventional **wheeled and tracked robots** often struggle with **unstable surfaces, debris accumulation, and confined spaces**, limiting their effectiveness in reaching victims quickly and safely. These limitations highlight a critical gap in current robotic rescue technologies, necessitating the development of a **more adaptable and terrain-agile robotic system**.

Key Problems:

Terrain Navigation: Traditional wheeled robots struggle to maneuver over uneven,unstable, or debris-laden surfaces, limiting their effectiveness in rescue operations.

Victim Detection: Identifying and locating victims in challenging environments, especially in low-light or obstructed conditions, is often inefficient and time-consuming with current technologies.

Data Transmission: Real-time communication of vital information back to rescue teams is often hindered by the remote locations and challenging conditions typical of disaster sites.

Operator Safety: Human rescuers face significant risks when navigating dangerous environments. There is a need for robotic solutions that can enter hazardous areas,minimizing risk to human life.

Integration of Tools: Existing robotic solutions may lack the ability to integrate various tools needed for specific rescue tasks, such as debris removal or medical assistance.

By leveraging these features, the proposed robotic system **enhances mobility, stability, and operational efficiency**, making it an ideal solution for disaster relief and search-and-rescue missions.

1.3 Objective

The **Rescue Bot with Rotating Leg Mechanism** is designed to enhance mobility and adaptability in disaster-stricken environments. Its primary objective is to traverse **uneven terrains, debris-filled areas, and confined spaces** using a **hybrid leg-wheel system**, ensuring smooth movement on flat surfaces while effectively climbing over obstacles. The robot's **compact and lightweight design** allows for easy deployment and maneuverability in emergency scenarios, making it a reliable tool for rescue missions.

To improve **autonomous navigation and remote-control capabilities**, the bot will incorporate **sensor-based decision-making** using **LiDAR, thermal imaging, and ultrasonic sensors** for real-time obstacle detection and victim localization. The system will also focus on **energy efficiency**, optimizing power consumption for prolonged operation in disaster zones.

Furthermore, the bot will feature **real-time monitoring and communication** through **wireless data transmission, camera feeds**, enabling rescue teams to receive live updates during operations. Lastly, ensuring **safety, stability, and reliability** in harsh environments is a crucial objective, with robust fail-safe mechanisms to prevent tipping or mechanical failures. By fulfilling these objectives, the Rescue Bot will serve as an efficient and life-saving solution for search-and-rescue teams operating in hazardous conditions.

1.4 Purpose & Importance:

Purpose

Navigability: Rescue can traverse difficult terrains—such as rubble, uneven ground, or debris—thanks to their multiple legs, which provide stability and adaptability.

Payload Capacity: They can carry essential equipment, such as medical supplies or communication devices, to victims in hard-to-reach areas.

Surveillance and Reconnaissance: Rescue can be equipped with cameras and sensors to gather real-time data, helping rescuers assess the situation before human intervention.

Remote Operation: Many Rescue can be controlled remotely, allowing operators to maintain a safe distance from hazardous environments.

Importance

Increased Safety: By deploying Rescue, human rescuers can avoid dangerous environments, minimizing the risk to their lives during missions.

Efficiency: These robots can work quickly and tirelessly, often covering areas faster than human teams, which is crucial in time-sensitive rescue operations.

Enhanced Communication: Rescue can act as communication relays in disaster zones where traditional infrastructure is damaged or non-existent.

Versatility: They can be used in various scenarios, including natural disasters (earthquakes, landslides), search and rescue missions in challenging environments (mountains, forests), and urban search efforts.

Data Collection: The ability to collect environmental data can help inform rescue strategies and improve future disaster response planning.

1.5 System Requirements:

1	**MECHANICAL**	
1.1	Chassis(Aluminum)	
1.2	legs(Acrylic)	
1.3	screws,etc.	
2	**ELECTRICAL**	
	Components	**price**
	Raspberry pi 5	5800 Rs

	Arduino uno 3	1300 Rs
	Servo Controller(PCA9685*2)	230*2 Rs
	Raspberry pi Camera module	370 Rs
	Thermal Sensor / Camera(ML90641)	
	Gyroscope(MPU6050)	135 Rs
	Ultrasonic Sensor(HC SRO4)	60 Rs
	Temperature Sensor(DTH11)	60 Rs
	Microphone(MAX9814)	220 Rs
	LIPO Battery (3S,9V)	
	Servo Motor(MG995R*18)	310*18
	GMS Module(SIM800L)	475
3	**SOFTWARE**	
3.1	Arduino IDE	
3.2	Fusion 360	
3.3	Anaconda	
3.4	Ansys	
3.5	Tinkercad	

CHAPTER II
LITERATURE SURVEY

2.1 Existing Systems and Case Studies

RoboCup Rescue is an international robotics competition that challenges teams to develop autonomous robots for disaster response scenarios. Participants simulate search and rescue operations, focusing on locating victims in hazardous environments. The competition fosters innovation in robot design, navigation, and communication, ultimately advancing technologies applicable to real-world emergencies. It serves as a platform for collaboration between researchers and engineers, promoting the development of effective rescue solutions.

Case Study: Kamala Mills Fire, Mumbai (2017)

On December 29, 2017, a devastating fire broke out at the Kamala Mills compound in Mumbai,which housed several restaurants and bars. The blaze quickly spread due to flammable materials and inadequate fire safety measures, resulting in the tragic loss of 14 lives and injured over 50 people, many of whom were trapped inside. Firefighters responded promptly, but the narrow lanes and overcrowded structures complicated the rescue efforts. The rapid spread of smoke and flames, coupled with blocked exits, made It is challenging to reach victims on upper floors. This incident underscored the urgent need for enhanced rescue technologies, such as drones for aerial surveillance and hexapod robots capable of navigating debris and confined spaces. The Kamala Mills fire raised public awareness about fire safety and the importance of effective emergency response strategies, highlighting the potential benefits of integrating advanced robotics into disaster management to improve rescue operations and save lives in future emergencies.

2.2 Advantages, Disadvantages, and Limitations of Existing Systems

Rescue systems present several advantages, including exceptional mobility that allows them to navigate rough and uneven terrains, making them ideal for disaster scenarios where conventional vehicles may struggle. Their multiple legs provide enhanced stability, enabling them to traverse obstacles like rubble effectively. These robots can carry medical supplies, communication equipment, and sensors, supporting rescue operations while minimizing risks to human rescuers. They can be operated remotely, allowing operators to maintain safety while monitoring hazardous environments, and are often equipped with various sensors—such as cameras, thermal imaging.

However, there are notable disadvantages. The development and deployment of rescue robots can be costly, which may limit accessibility for smaller organizations or regions with fewer resources. The technology is complex, necessitating specialized knowledge for operation and maintenance. Additionally, many hexapod systems have limited battery life, restricting their operational time during critical rescue missions. Communication can be challenging in disaster-stricken areas where infrastructure is damaged, and harsh weather conditions may impair sensor functionality and mobility.manage user accounts, product listings, and transactions securely.

Despite their strengths, these systems also face limitations, such as restricted autonomy—many still require human intervention, which can slow down response times. Their payload capacity might not be sufficient for larger items needed in certain rescue scenarios, and navigating highly cluttered environments can pose challenges. Furthermore, the physical size of recue robots may limit access to very tight spaces crucial in urban search and rescue operations. Regular maintenance is necessary to ensure functionality, which can be logistically challenging in remote or disaster-affected areas.

2.3 Technology and Proposed System

The proposed system for hexapod rescue robots builds on existing technologies while introducing several enhancements to improve their effectiveness in disaster scenarios. Current rescue bot designs utilize advanced locomotion algorithms that enable versatile movement across various terrains, supported by an array of sensors, including cameras for visual data collection, and thermal sensors for locating survivors through heat signatures. However, the proposed system aims to enhance autonomy by incorporating advanced AI algorithms that allow the rescue to navigate complex environments without constant human intervention, utilizing machine learning for real-time obstacle recognition and decision-making.

Moreover, improved sensor fusion will integrate data from multiple sources to create a comprehensive understanding of the environment, aiding in victim detection. To address communication challenges in disaster areas, the system will implement robust mesh networking and satellite communication protocols, ensuring stable connections even when traditional infrastructure is compromised. A modular design will allow for easy swapping of components based on mission needs, such as different sensor packages for urban versus rural scenarios or increased payload capacity for carrying supplies.

CHAPTER -III

Methodology:

The methodology for developing rescue robots involves a structured approach that begins with a comprehensive literature review of existing technologies, robotic locomotion, sensor integration, and case studies to identify gaps in current systems. Following this, the design and specifications of the robot are defined, focusing on size, payload capacity, and sensor types while ensuring a modular design for customization based on mission needs. A prototype is then developed using CAD software and rapid prototyping techniques, such as 3D printing, to create physical models for initial testing.

3.1Proposed Methodology

The purpose of the methodology for developing rescue robots is to create a systematic framework that guides the entire design, development, and testing process. This approach aims to ensure that the resulting robotic system effectively addresses the challenges faced in real-world disaster scenarios, thereby enhancing search and rescue operations.

By beginning with a thorough literature review, the methodology seeks to build on existing knowledge and identify gaps that can be addressed through innovative design and technology integration. The structured phases of design, prototyping, integration, and testing are intended to facilitate the development of a reliable and adaptable robot capable of navigating complex environments, carrying payloads, and performing real-time victim detection.

Moreover, the methodology emphasizes iterative testing and optimization, allowing for continuous improvement based on performance metrics and user feedback. This adaptive process is essential for refining the robot's capabilities and ensuring it meets the evolving needs of emergency responders.

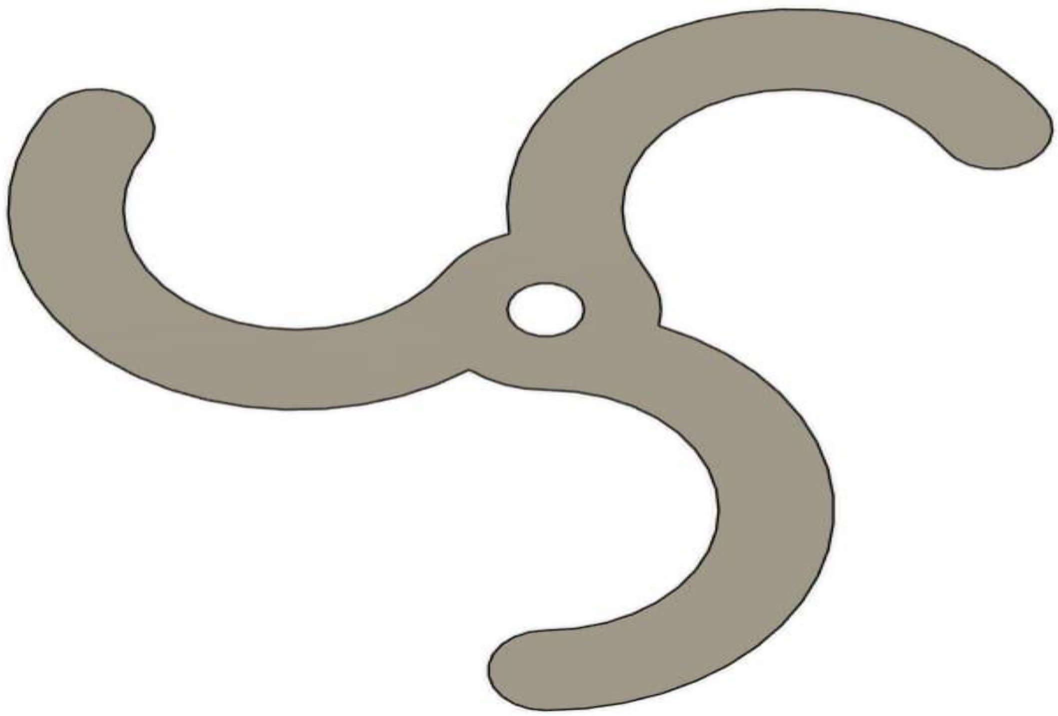
Ultimately, the purpose of this methodology is to produce a state-of-the-art rescue robot that not only enhances operational efficiency in emergency situations but also contributes to saving lives by enabling faster and more effective rescue efforts. By laying a clear and detailed path for development, the methodology ensures that the project remains focused on achieving its goals while accommodating advancements in technology and responding to the dynamic challenges of disaster response.

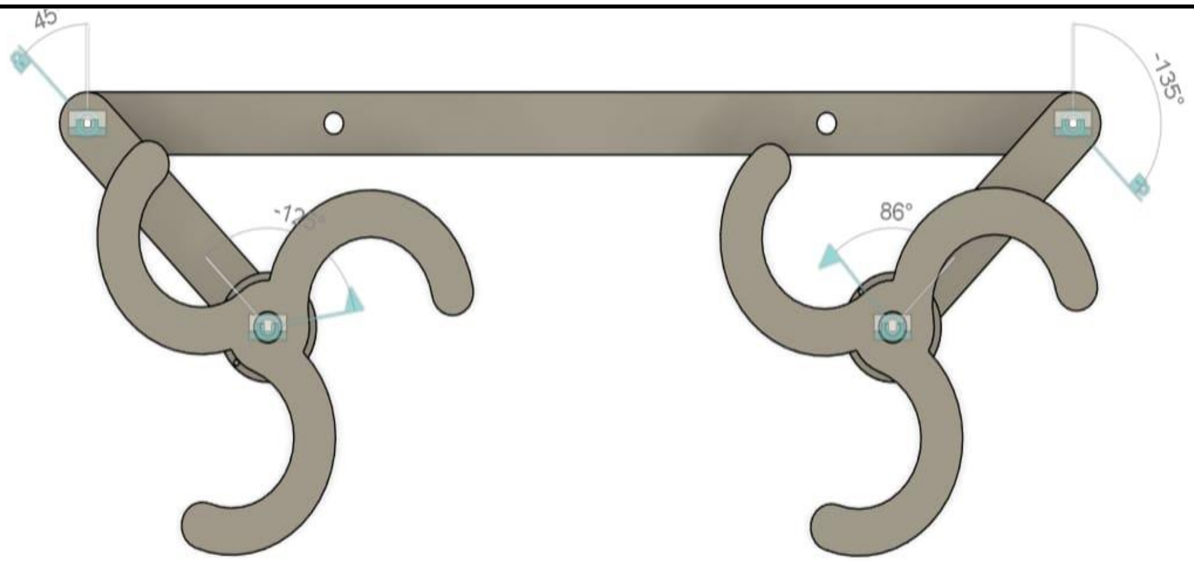
CHAPTER IV

DIAGRAM AND DESIGN

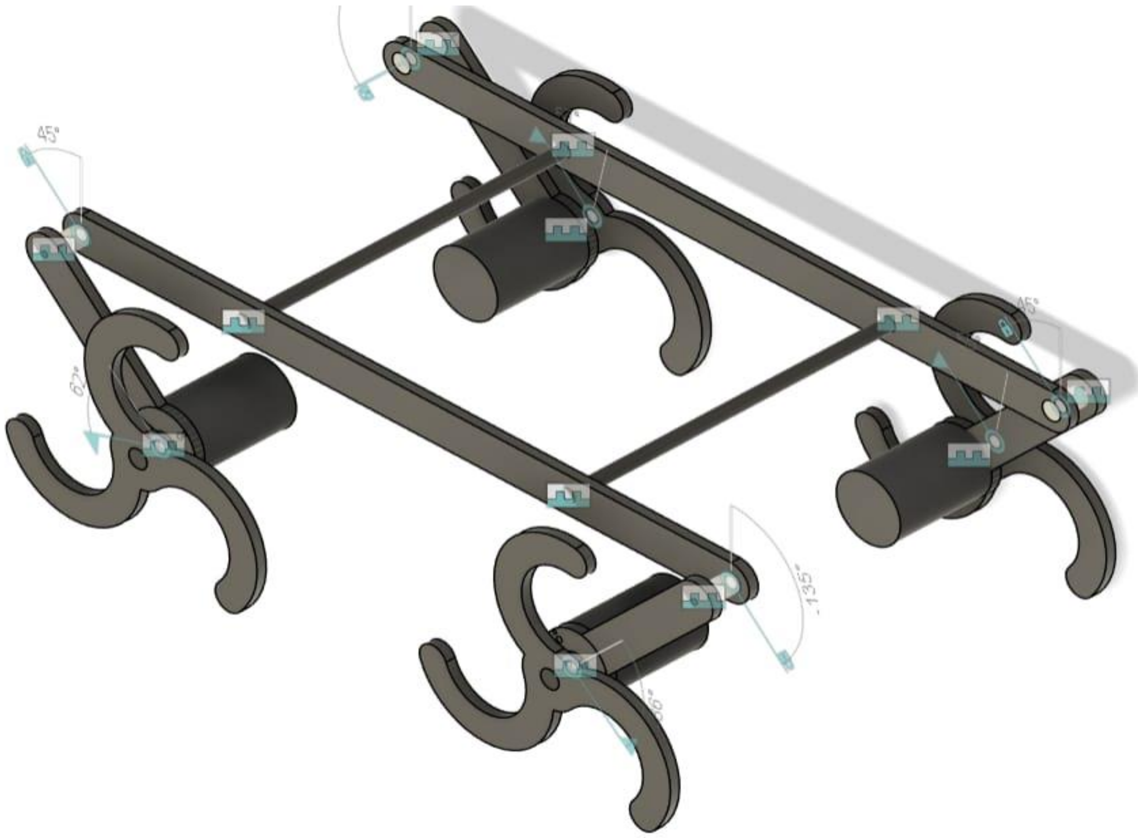
CAD DIAGRAM

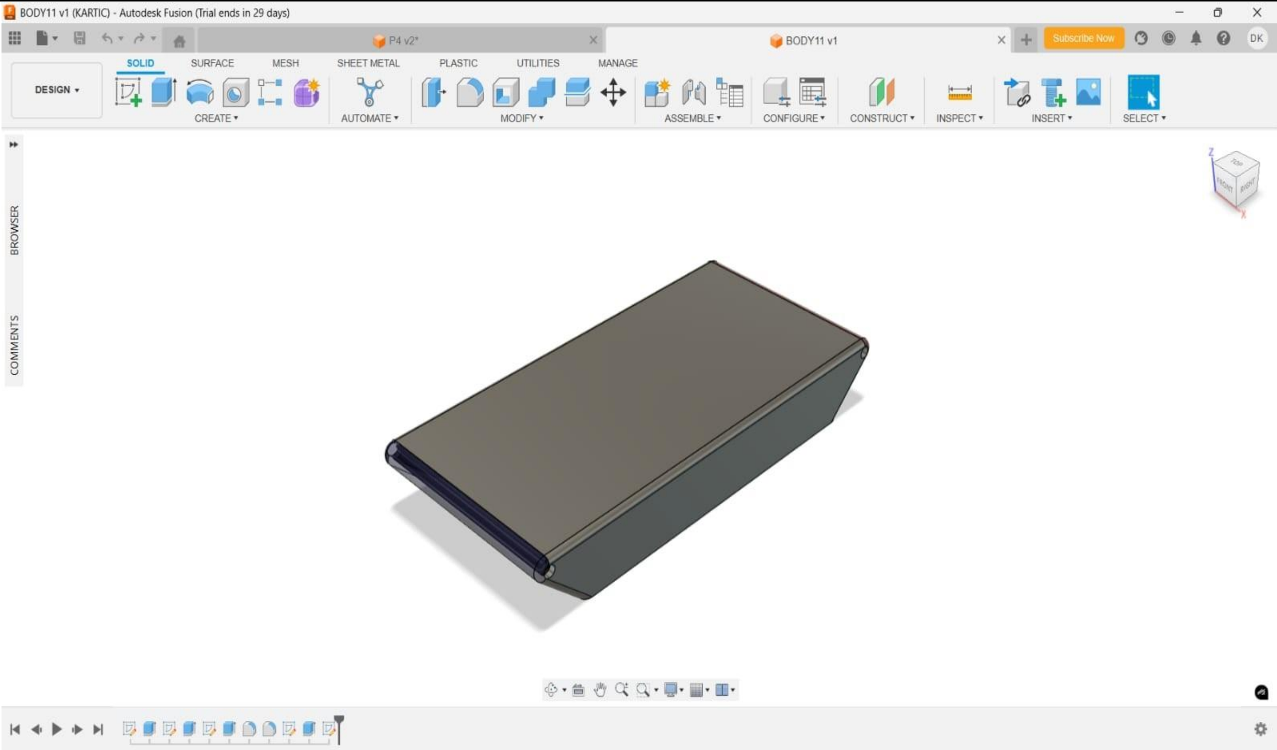
WHEEL OF BOT:



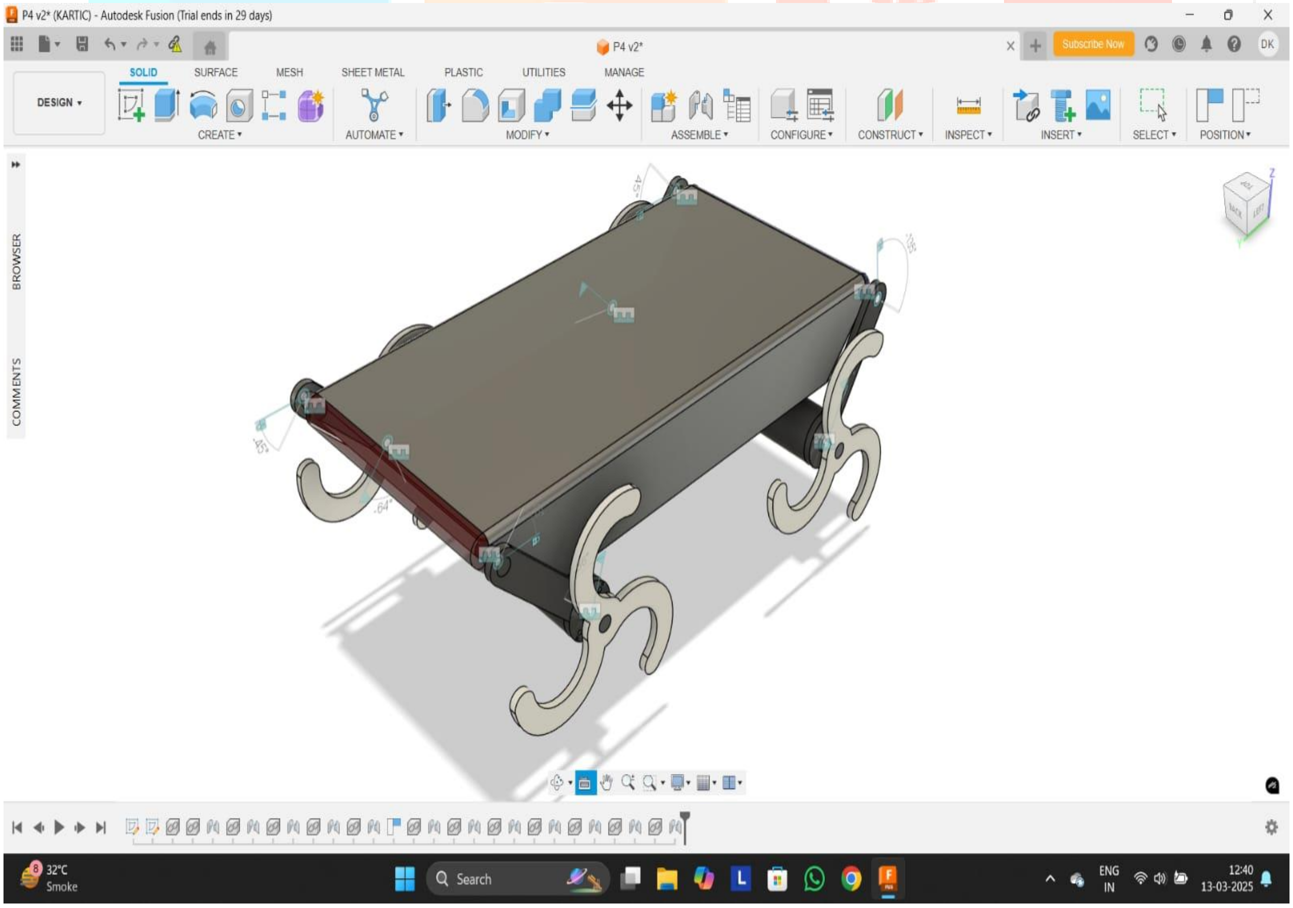


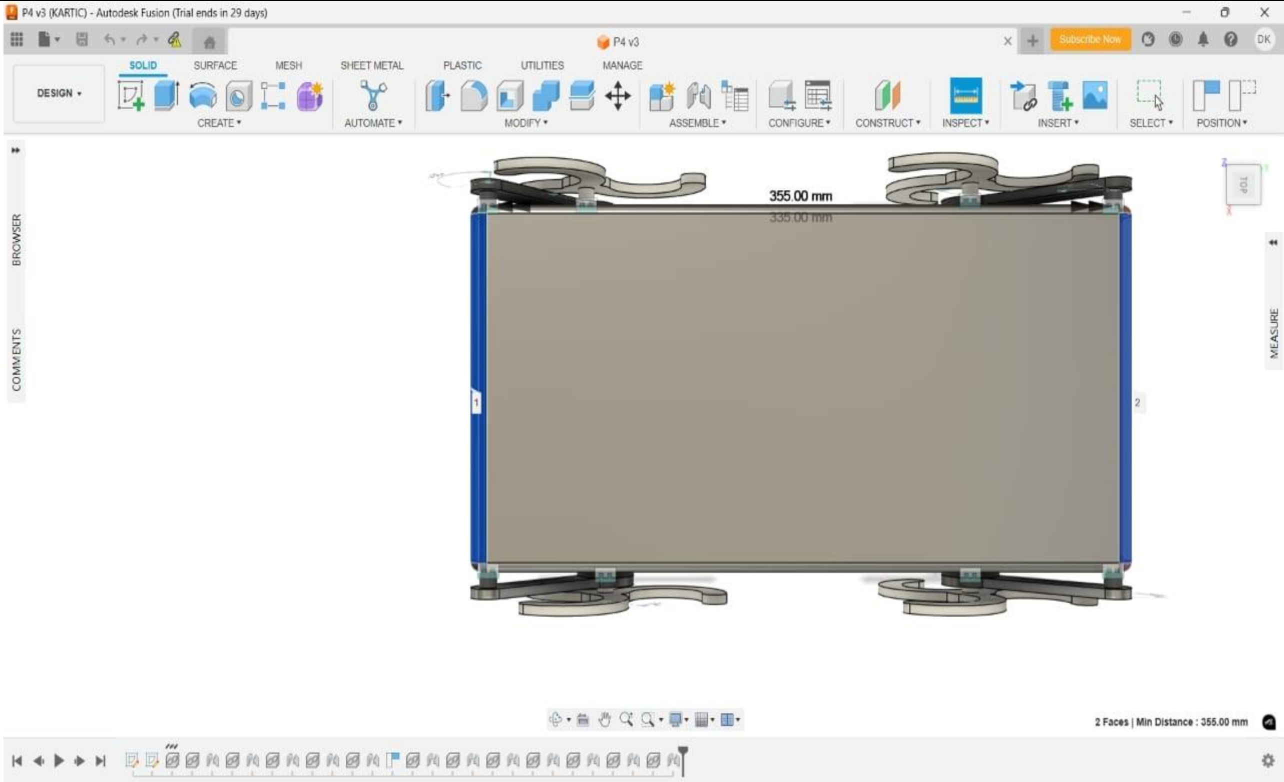
WIRE FRAME:

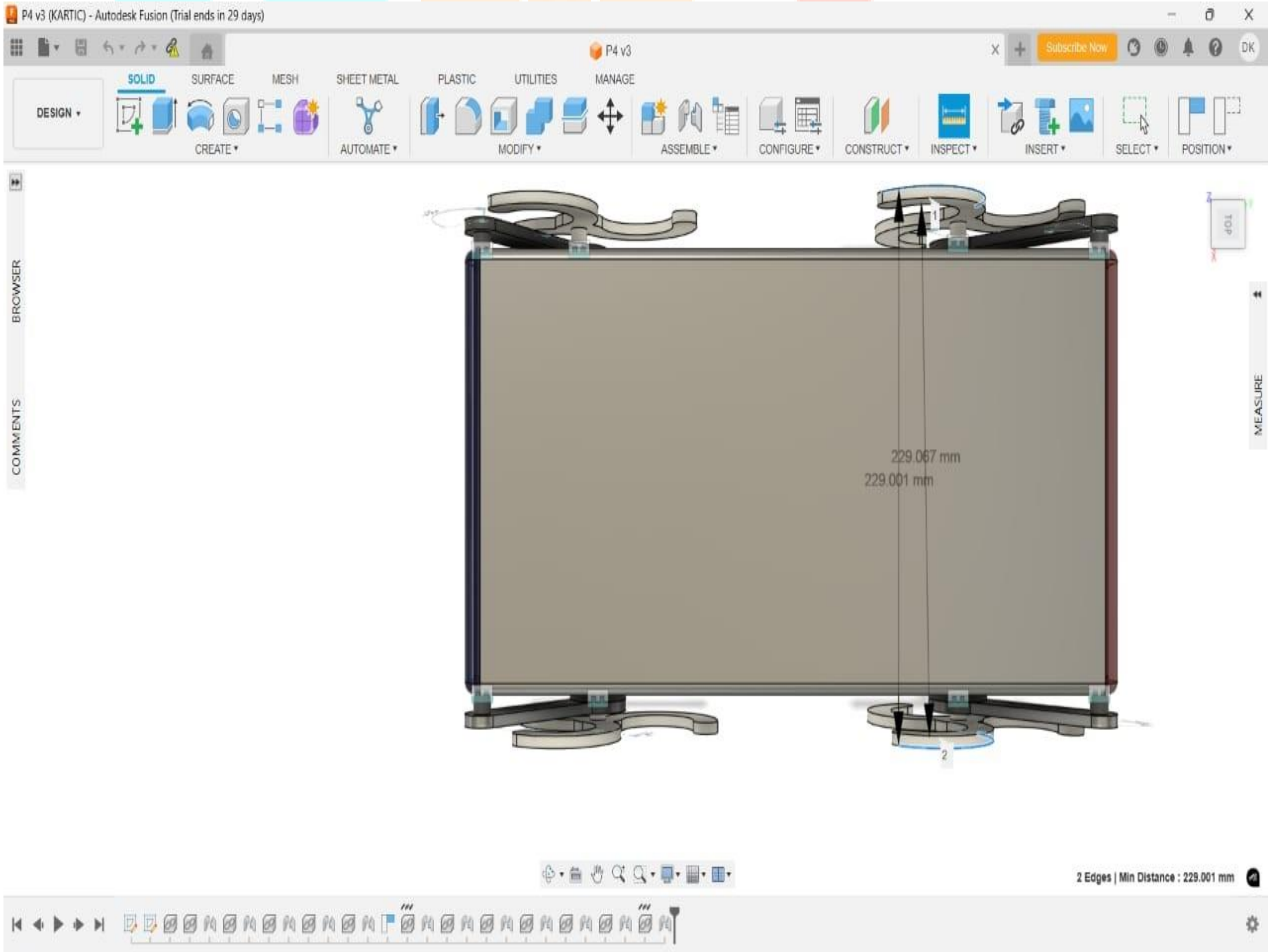



BODY11 v1 (KARTIC) - Autodesk Fusion (Trial ends in 29 days)
P4 v2*
BODY11 v1
Subscribe Now
DESIGN
SOLID
SURFACE
MESH
SHEET METAL
PLASTIC
UTILITIES
MANAGE
CREATE
AUTOMATE
MODIFY
ASSEMBLE
CONFIGURE
CONSTRUCT
INSPECT
INSERT
SELECT
BROWSER
COMMENTS

BOT BODY:


P4 v2* (KARTIC) - Autodesk Fusion (Trial ends in 29 days)
P4 v2*
Subscribe Now
DESIGN
SOLID
SURFACE
MESH
SHEET METAL
PLASTIC
UTILITIES
MANAGE
CREATE
AUTOMATE
MODIFY
ASSEMBLE
CONFIGURE
CONSTRUCT
INSPECT
INSERT
SELECT
POSITION
BROWSER
COMMENTS
32°C
Smoke
Search
ENG
IN
12:40
13-03-2025





ALUMINUM CHASSIS:

An **aluminum chassis** is a structural frame made from aluminum, commonly used in **robots, vehicles, and electronic enclosures**. It is **lightweight, strong, corrosion-resistant, and thermally conductive**, making it ideal for applications requiring durability and heat dissipation. Aluminum chassis are widely used in robotics for their balance of **strength and weight**, improving mobility and efficiency.

ACRYLIC LEGS:

Acrylic legs are structural components made from **acrylic (PMMA - polymethyl methacrylate)**, commonly used in **robots, prototypes, and DIY projects**. Acrylic is **lightweight, transparent, and easy to cut**, but it is also **brittle** and can crack under excessive stress. It is often used in robotic legs for small, lightweight bots where aesthetics and ease of fabrication are important. However, for high-load applications, materials like aluminum or carbon fiber are preferred.

SCREW:

A **screw** is a **fastener** used to join two or more objects together by threading into a material. It consists of a **head, shank, and threads** that help in securing components tightly. Screws come in various types, such as **machine screws, self-tapping screws, and wood screws**, each designed for specific applications. In robotics and mechanical projects, screws are essential for **assembling frames, mounting motors, and securing electronic components**.

SUSPENSION:

Suspension is a system used in **vehicles, robots, and machinery** to absorb shocks, maintain stability, and improve movement over uneven surfaces. It typically consists of **springs, dampers (shock absorbers), and linkages** that help in reducing vibrations and providing a smoother ride. In robotics, suspension systems are crucial for **off-road robots, legged robots, and drones**, ensuring better control and adaptability to differen terrains.

- Weight of the bot will around 2.3 kg.

Outline of Schematic A2: Engineering Data

	A	B	C	D	E
1	Contents of Engineering Data			Source	Description
2	Material				
3	Acrylic (PMMA)			Gra	Polymethylmethacrylate (PMMA) Sample materials data from Granta Design. Additional data and information available through the Granta website . Granta provides no warranty for the accuracy of the data.
4	Alumina			Gra	Aluminum oxide (Al2O3) Sample materials data from Granta Design. Additional data and information available through the Granta website . Granta provides no warranty for the accuracy of the data.
5	Structural Steel			Ger	Fatigue Data at zero mean stress comes from 1998 ASME BPV Code, Section 8, Div 2, Table 5-110.1
*	Click here to add a new material				

PROPERTIES OF STEEL

	A	B	C	D	E
1	Property	Value	Unit		
2	Material Field Variables	Table			
3	Density	7850	kg m^-3		
4	Isotropic Secant Coefficient of Thermal Expansion				
6	Isotropic Elasticity				
7	Derive from	Young's Modulus and Poisson...			
8	Young's Modulus	2E+11	Pa		
9	Poisson's Ratio	0.3			
10	Bulk Modulus	1.6667E+11	Pa		
11	Shear Modulus	7.6923E+10	Pa		
12	Strain-Life Parameters				
20	S-N Curve	Tabular			
24	Tensile Yield Strength	2.5E+08	Pa		
25	Compressive Yield Strength	2.5E+08	Pa		
26	Tensile Ultimate Strength	4.6E+08	Pa		
27	Compressive Ultimate Strength	0	Pa		

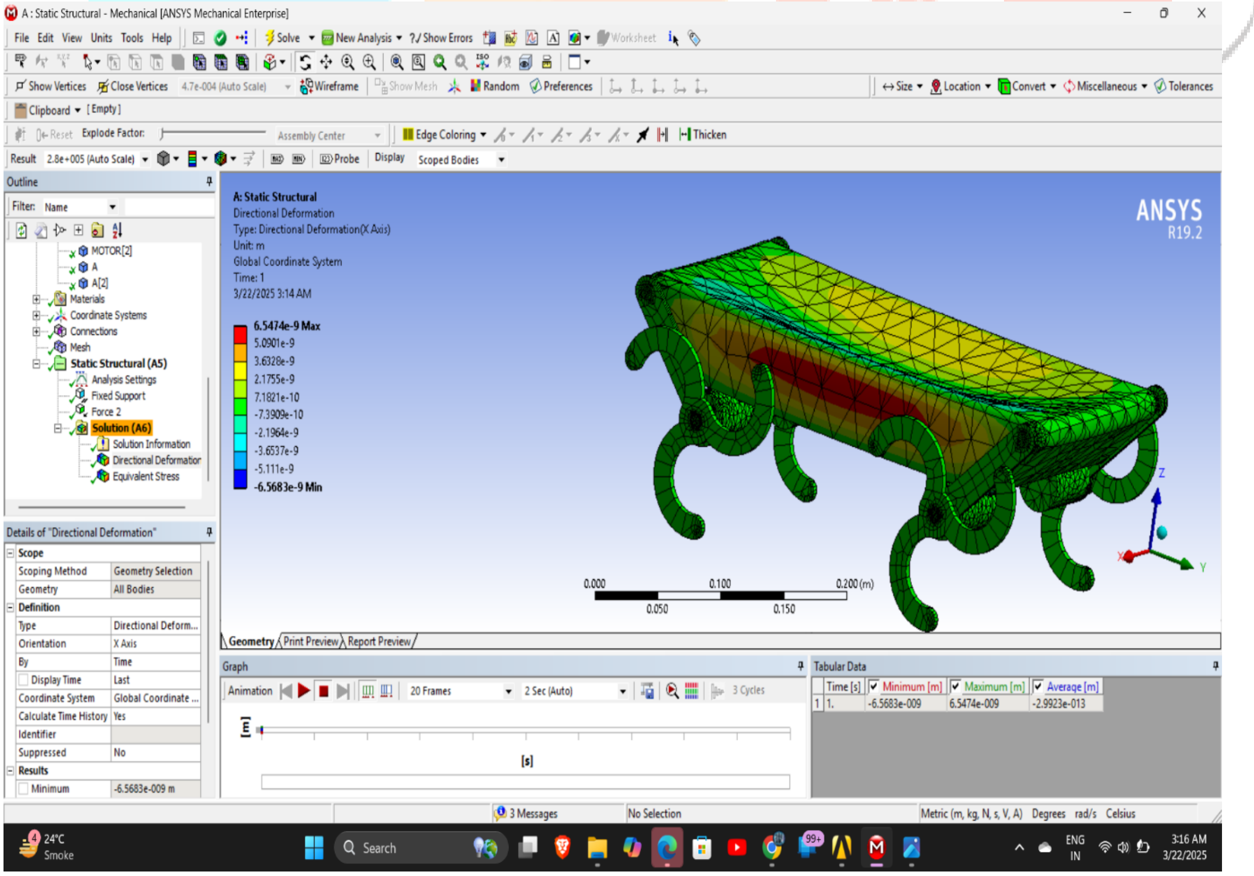
PROPERTIES OF ACRYLIC

	A	B	C	D	E
1	Property	Value	Unit		
2	Density	3700	kg m^-3		
3	Isotropic Secant Coefficient of Thermal Expansion				
5	Isotropic Elasticity				
6	Derive from	Young's Modulus and Poisson...			
7	Young's Modulus	3.3E+11	Pa		
8	Poisson's Ratio	0.238			
9	Bulk Modulus	2.0992E+11	Pa		
10	Shear Modulus	1.3328E+11	Pa		
11	Tensile Yield Strength	2E+08	Pa		
12	Tensile Ultimate Strength	2E+08	Pa		

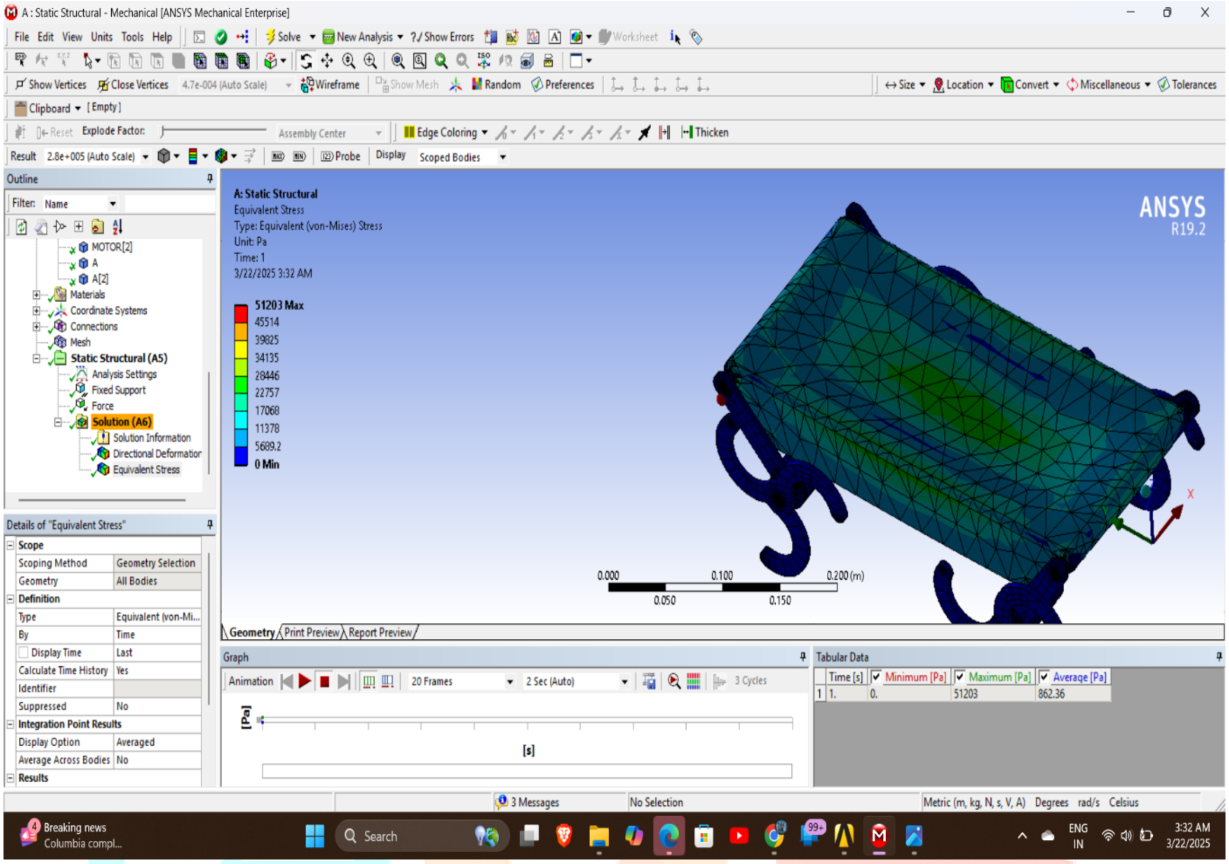
PORPERTIES OF ALUMINUM

	A	B	C	D	E
1	Property	Value	Unit		
2	Density	1180	kg m^-3		
3	Isotropic Secant Coefficient of Thermal Expansion				
5	Isotropic Elasticity				
6	Derive from	Young's Modulus and Poisson...			
7	Young's Modulus	2.69E+09	Pa		
8	Poisson's Ratio	0.395			
9	Bulk Modulus	4.2698E+09	Pa		
10	Shear Modulus	9.6416E+08	Pa		
11	Tensile Yield Strength	6.24E+07	Pa		
12	Tensile Ultimate Strength	6.71E+07	Pa		

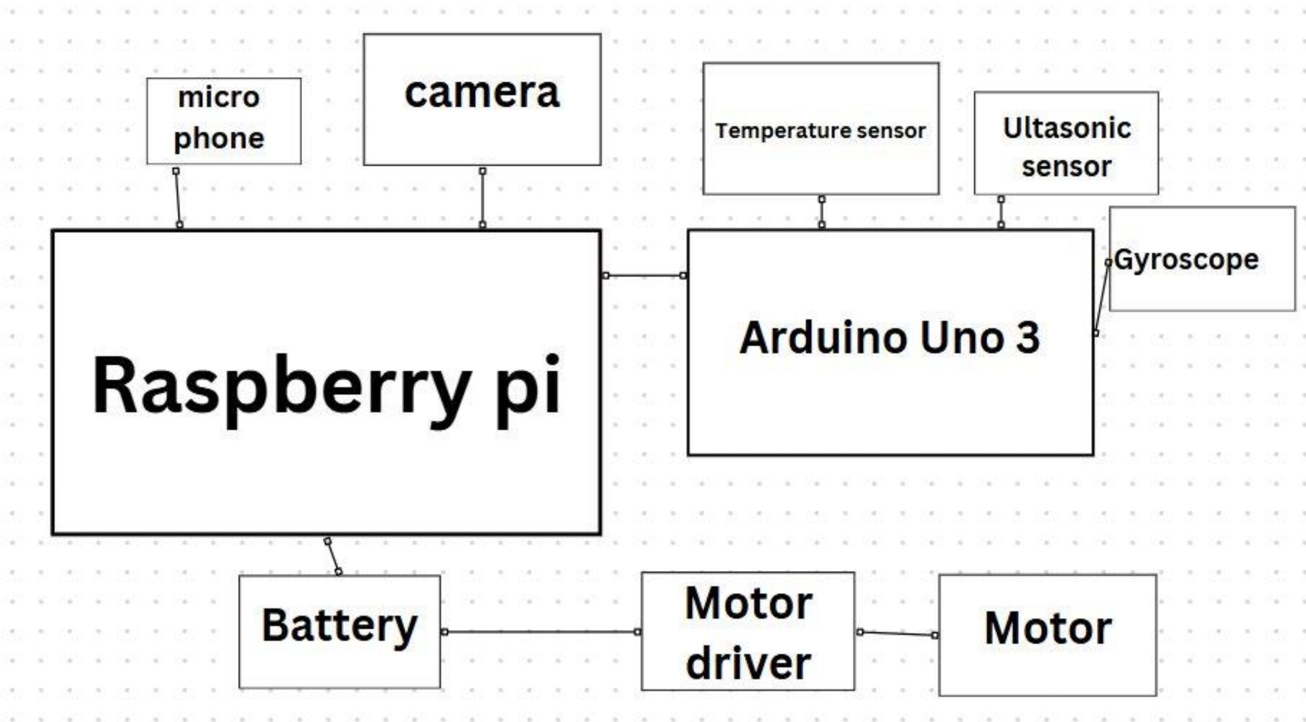
DEFORMATION ANALYSIS:

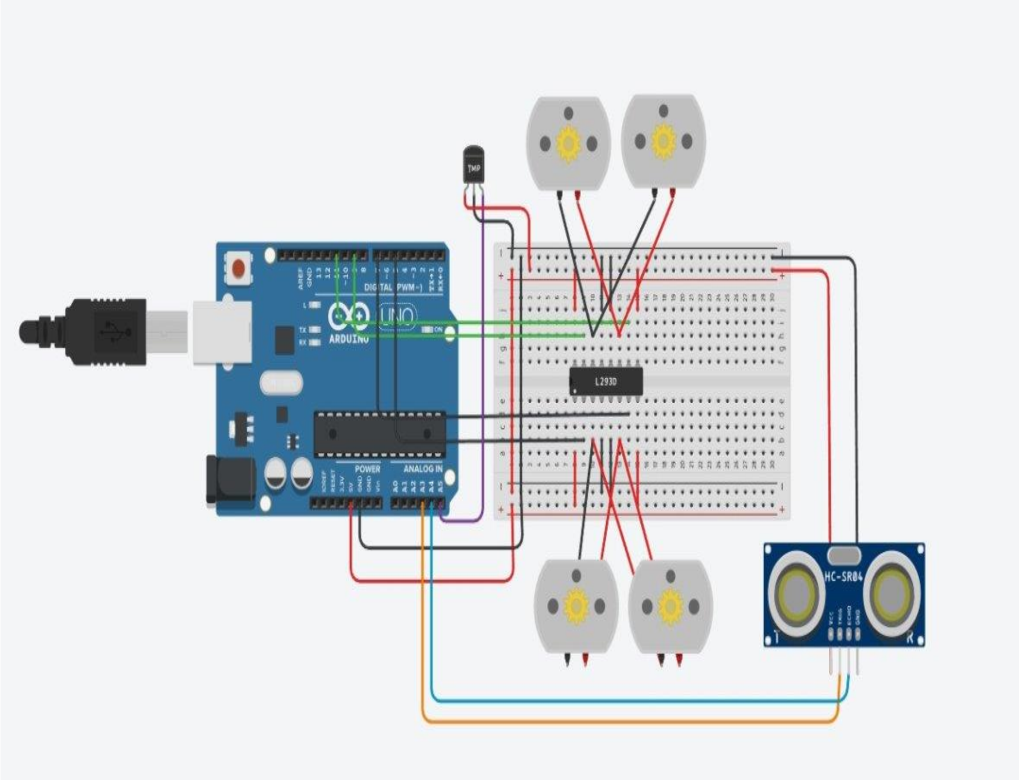


STRESS ANALISYS:



CIRCUIT DIAGRAM





Raspberry pi:

The **Raspberry Pi** is a **low-cost, credit-card-sized single-board computer** developed by the Raspberry Pi Foundation. It is widely used in **robotics, IoT, automation, and computer vision** projects due to its versatility and GPIO (General Purpose Input/Output) pins. It supports various operating systems like **Raspberry Pi OS, Ubuntu, and Windows IoT Core**, making it ideal for prototyping and embedded applications. With built-in **Wi-Fi, Bluetooth, HDMI, and USB ports**, it serves as a powerful yet compact computing platform for diverse applications.

ARDUINO UNO:

The **Arduino Uno** is an **open-source microcontroller board** based on the **ATmega328P** chip, widely used for electronics projects and prototyping. It features **14 digital input/output pins, 6 analog inputs, a USB connection, a power jack, and a reset button**, making it versatile for various applications. The board can be programmed using the **Arduino IDE**, supports multiple sensors and actuators, and is ideal for beginners and advanced users in robotics, automation, and IoT. Its ease of use, affordability, and strong community support make it one of the most popular microcontrollers for embedded systems development.

CAMERA:

A **camera** is an optical device used to capture images or videos by converting light into an electronic signal. In robotics and computer vision applications, cameras such as **RGB, infrared, LiDAR, and depth cameras** are used for **object detection, navigation, and environmental mapping**. Modern cameras integrate **CMOS or CCD sensors**, offering high resolution and real-time image processing for various applications. In rescue robots, cameras play a crucial role in **victim detection, obstacle avoidance, and remote monitoring**, enhancing autonomous and remote-controlled operations.

MICROPHONE:

A **microphone** is a device that converts **sound waves into electrical signals**, allowing for audio recording, communication, and sound processing. In robotics and IoT applications, microphones are used for **speech recognition, voice commands, environmental sound detection, and noise analysis**. They come in various types, such as **dynamic, condenser, and MEMS microphones**, each suited for different applications. In rescue robots, microphones can help in **detecting trapped victims by capturing distress calls or unusual sounds**, improving search-and-rescue efficiency in disaster scenarios.

TEMPERATURE SENSOR:

A **temperature sensor** is a device used to measure temperature in various environments and applications. It converts temperature changes into an electrical signal, which can be processed by microcontrollers like **Arduino or Raspberry Pi**. Common types include **thermistors, thermocouples, infrared (IR) sensors, and digital sensors like DHT11, DHT22, and LM35**. In rescue robotics, temperature sensors help in **detecting heat signatures of trapped victims, monitoring environmental conditions, and preventing overheating of robotic components**, making them crucial for search-and-rescue operations.

UTRASONIC SENSOR:

An **ultrasonic sensor** is a device that measures distance by emitting **high-frequency sound waves** and analyzing the echo reflected from objects. It typically consists of a **transmitter (which emits ultrasound) and a receiver (which detects the reflected signal)** to calculate the distance based on the time taken for the sound to return. Common models like the **HC-SR04** are widely used in robotics for **obstacle detection, collision avoidance, and autonomous navigation**. In rescue robots, ultrasonic sensors help in **mapping debris-filled environments, detecting obstacles in low-visibility conditions, and assisting in safe movement through confined spaces**.

COMMUNICATION METHODS:

The **NRF module** is a low-power, high-speed **wireless communication module** based on the **nRF24L01** transceiver. It operates in the **2.4 GHz ISM band**, offering **long-range, low-latency, and high-speed data transmission**. It supports **SPI communication**, making it compatible with microcontrollers like **Arduino, Raspberry Pi, and STM32**. In your rescue robot project, NRF modules will help enable **reliable, real-time communication** between robots or between the robot and a remote control station, ensuring smooth operation in disaster zones. Let me know if you need help with NRF module integration!

GYROSCOPE:

A **gyroscope** is a sensor that measures **angular velocity** and helps determine an object's **orientation and rotational movement**. It works based on the principles of **conservation of angular momentum** and is commonly used in **IMUs (Inertial Measurement Units)** alongside accelerometers for motion tracking. Popular gyroscope modules include the **MPU6050 and L3G4200D**, which provide precise data for **stabilization, navigation, and control** in robotics and drones. In rescue robots, gyroscopes assist in **maintaining balance on uneven terrain, stabilizing camera systems, and improving autonomous movement in complex environments**.

MOTOR DRIVER:

A **motor driver** is an electronic circuit or module that controls the **speed, direction, and power** of DC motors, stepper motors, or servo motors. It acts as an interface between a **microcontroller (like Arduino or Raspberry Pi)** and the motor, since microcontrollers cannot supply sufficient current to drive motors directly. Common motor drivers include **L298N, L293D, and TB6612FNG**, which allow bidirectional control and speed regulation using **PWM (Pulse Width Modulation)**. In rescue robots, motor drivers are essential for **precise movement control, obstacle avoidance, and terrain adaptability**, ensuring smooth and efficient mobility.

MOTOR:

A motor is a device that converts electrical energy into mechanical motion. It works by using electromagnetic forces to generate rotation or linear movement. Motors are commonly used in robotics, vehicles, appliances, and industrial machinery. There are different types, such as **DC motors, stepper motors, and servo motors**, each suited for specific applications.

BATTERY:

A **battery** is a device that stores chemical energy and converts it into electrical energy to power electronic devices. It consists of one or more cells, each containing electrodes (anode and cathode) and an electrolyte. Batteries can be **rechargeable** (e.g., lithium-ion, lead-acid) or **non-rechargeable** (e.g., alkaline, lithium primary).

SOFTWARE:

COMPUTER VISION:

Image detection is a **computer vision** technique used to identify and locate objects in an image or video. It involves processing visual data using **machine learning (ML) or deep learning models**, such as **YOLO (You Only Look Once), SSD (Single Shot Detector), and Faster R-CNN**.

Image detection is widely used in:

- **Robotics** (e.g., object tracking, navigation)
- **Security** (e.g., face recognition, surveillance)
- **Medical Imaging** (e.g., tumor detection)
- **Autonomous Vehicles** (e.g., obstacle detection

TENSORFLOW:

TensorFlow is an **open-source machine learning (ML) framework** developed by **Google** for building and deploying ML and deep learning models. It is widely used for **computer vision, natural language processing (NLP), robotics, and AI applications**.

- **Supports Deep Learning** – Works with neural networks, including CNNs and RNNs.
- **Hardware Acceleration** – Runs efficiently on **CPUs, GPUs, and TPUs**.
- **TensorFlow Lite** – Optimized for mobile and embedded devices.
- **TensorFlow.js** – Runs ML models in web browsers.
- **Kera's API** – Provides an easy-to-use interface for building deep learning models.

Import Necessary Libraries and Load a Pre-Trained Object Detection Model

```
In [1]: import tensorflow as tf
        import json
        import numpy as np
        from matplotlib import pyplot as plt

In [2]: images = tf.data.Dataset.list_files('D:\\data\\images\\*.jpg')

In [3]: images.as_numpy_iterator().next()

Out[3]: b'D:\\data\\images\\30.jpg'

In [4]: def load_image(x):
            byte_img = tf.io.read_file(x)
            img = tf.io.decode_jpeg(byte_img)
            return img

In [5]: images = images.map(load_image)
```

```
In [8]: image_generator = images.batch(4).as_numpy_iterator()
```

```
In [9]: plot_images = image_generator.next()
```

```
In [10]: fig, ax = plt.subplots(ncols=4, figsize=(20,20))
         for idx, image in enumerate(plot_images):
             ax[idx].imshow(image)
         plt.show()
```

Preprocess the Image

```
In [39]: train_images = tf.data.Dataset.list_files('D:\\agm_data\\train\\images\\*.j
         train_images = train_images.map(load_image)
         train_images = train_images.map(lambda x: tf.image.resize(x, (120,120)))
         train_images = train_images.map(lambda x: x/255)
```

```
In [40]: test_images = tf.data.Dataset.list_files('D:\\agm_data\\test\\images\\*.jpg
         test_images = test_images.map(load_image)
         test_images = test_images.map(lambda x: tf.image.resize(x, (120,120)))
         test_images = test_images.map(lambda x: x/255)
```

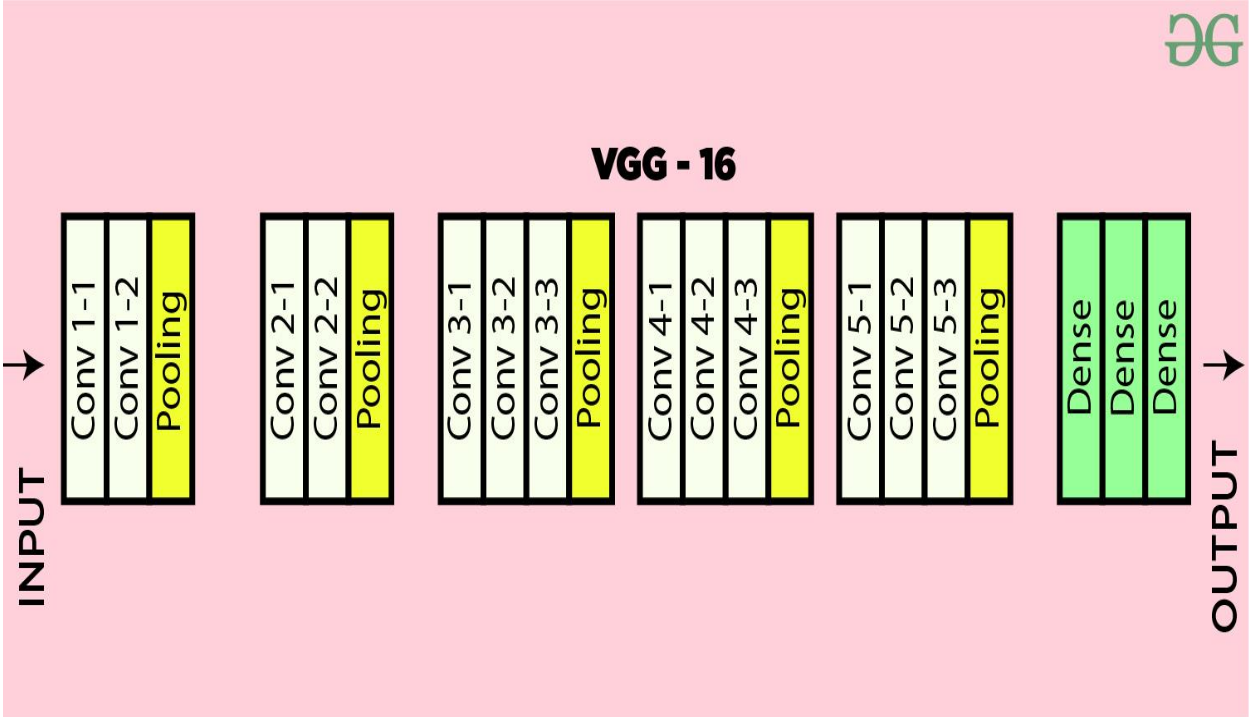
Train a Custom Object Detection Model (Optional)

```
In [61]: vgg.summary()

Model: "vgg16"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 input_1 (InputLayer)        [(None, None, None, 3)]   0
 block1_conv1 (Conv2D)       (None, None, None, 64)    1792
 block1_conv2 (Conv2D)       (None, None, None, 64)    36928
 block1_pool (MaxPooling2D)  (None, None, None, 64)    0
 block2_conv1 (Conv2D)       (None, None, None, 128)   73856
 block2_conv2 (Conv2D)       (None, None, None, 128)   147584
 block2_pool (MaxPooling2D)  (None, None, None, 128)   0
 block3_conv1 (Conv2D)       (None, None, None, 256)   295168
 block3_conv2 (Conv2D)       (None, None, None, 256)   590080
 block3_conv3 (Conv2D)       (None, None, None, 256)   590080
 block3_pool (MaxPooling2D)  (None, None, None, 256)   0
 block4_conv1 (Conv2D)       (None, None, None, 512)   1180160
 block4_conv2 (Conv2D)       (None, None, None, 512)   2359808
 block4_conv3 (Conv2D)       (None, None, None, 512)   2359808
 block4_pool (MaxPooling2D)  (None, None, None, 512)   0
 block5_conv1 (Conv2D)       (None, None, None, 512)   2359808
 block5_conv2 (Conv2D)       (None, None, None, 512)   2359808
```

VGG16: A Deep Learning Model for Image Classification

VGG16 is a **convolutional neural network (CNN)** architecture developed by the **Visual Geometry Group (VGG) at the University of Oxford**. It is widely used for **image classification, object detection, and feature extraction** in computer vision tasks.



Exporting the model:

```
In [93]: from tensorflow.keras.models import load_model

In [94]: balltracker.save('balltracker.h5')

WARNING:tensorflow:Compiled the loaded model, but the compiled metrics hav
e yet to be built. `model.compile_metrics` will be empty until you train o
r evaluate the model.

C:\Users\sanke\anaconda3\Lib\site-packages\keras\src\engine\training.py:30
00: UserWarning: You are saving your model as an HDF5 file via `model.save
()`. This file format is considered legacy. We recommend using instead the
native Keras format, e.g. `model.save('my_model.keras')`.
  saving_api.save_model(
```

Arduino IDE: An Overview

The **Arduino IDE (Integrated Development Environment)** is a software platform used to **write, compile, and upload code** to Arduino boards. It supports programming in **C/C++** and is widely used in **robotics, IoT, and embedded systems**.

Key Features of Arduino IDE

- **Simple Interface** – Easy-to-use for beginners and professionals.
- **Predefined Libraries** – Supports libraries for sensors, motors, displays, etc.
- **Supports Multiple Boards** – Works with **Arduino Uno, Mega, Nano, ESP32.**
- **Serial Monitor & Plotter** – Helps in debugging and real-time data visualization.
- **Cross-Platform** – Available on **Windows, macOS, and Linux**.

CONCLUSION & FUTURE SCOPE

1. Conclusion

The **Rescue Bot with Rotating Leg Mechanism** presents a **high-mobility robotic solution** for search-and-rescue operations in disaster-stricken environments. Its innovative **hybrid leg-wheel system** enables efficient movement across **rough, debris-filled terrains**, where traditional robots struggle. By integrating **autonomous navigation, remote control, and sensor-based obstacle detection**, the robot enhances rescue operations by **improving accessibility, stability, and efficiency**. With features like **real-time monitoring, victim detection, and energy efficiency**, this bot serves as a crucial tool for disaster response teams, reducing human risk and improving the chances of successful rescue missions.

The current design can be further **optimized and enhanced** with additional features to improve its functionality. Future developments may include:

- **Integration of advanced sensors**, such as **LiDAR, thermal imaging, and multi-spectral cameras**, for better victim detection and environmental mapping.
- **Improved power management**, such as **solar charging or energy-efficient batteries**, for extended operational time.
- **Water-resistant and fire-resistant design** for operations in extreme conditions, including floods and fire hazards.
- **Robotic arm integration** for victim assistance, debris removal, or medical supply delivery

By advancing these features, the **Rescue Bot** can become a **fully autonomous, highly efficient, and mission-ready solution** for search-and-rescue teams worldwide, significantly improving response time and effectiveness in life-saving operations.

CHAPTER VIII
REFERENCES

MECHANISM:

https://onlinelibrary.wiley.com/doi/10.1155/2017/5719381?utm_source

https://www.researchgate.net/publication/261354962_Dynamic_analysis_for_the_leg_mechanism_of_a_wheel-leg_hybrid_rescue_robot

https://www.researchgate.net/publication/320902565_Design_Analysis_and_Experiment_for_Rescue_Robot_with_Wheel-Legged_Structure

SOFTWARE :

https://arxiv.org/abs/1409.1556?utm_source=chatgpt.com

https://www.researchgate.net/publication/337105858_Transfer_learning_using_VGG-16_with_Deep_Convolutional_Neural_Network_for_Classifying_Images