A Project Report on

Advanced Surveillance and Multi-utility Rover

Submitted by

Pratik Chauhan (213010)

Neeraj Chaurasia (213011)

Mojes Dhotre (213019)

Aryan Menon (213043)

in partial fulfillment for the award of the degree

BACHELOR OF ENGINEERING

in

Department of Electronics & Telecommunication Engineering

under guidance of
Ms. Savita Kulkarni

St. Francis Institute of Technology, Mumbai
An Autonomous Institute, Affiliated to University of Mumbai

2024-2025

CERTIFICATE

This is to certify that Pratik Chauhan, Neeraj Chaurasia, Mojes Dhotre, and Aryan Menon are bonafide final-year students of St. Francis Institute of Technology, Mumbai. They have successfully completed the final-year project titled "Advanced Surveillance and Multi-utility Rover" under the domain of Embedded Systems, in partial fulfilment of the requirements of the Bachelor of Engineering (B.E.) Degree in Electronics and Telecommunication Engineering of the An Autonomous Institute, Affiliated to University of Mumbai during the academic year 2024–2025.

(Ms. Savita Kulkarni)
Project Guide

(Dr. Kevin Noronha)
Head of Department, EXTC

(Dr. Sincy George)
Principal

Project Report Approval for B.E.

This project entitled ***"Advanced Surveillance and Multi-utility Rover"*** by **Pratik Chauhan, Neeraj Chaurasia, Mojes Dhotre, Aryan Menon** is approved for the degree of Bachelor of Engineering in Electronics and Telecommunication from An Autonomous Institute, Affiliated to University of Mumbai.

Examiners

1. prasad joshi ______________________

2. ______________________

Date: 23/04/25 ______________________

Place: ______________________

ACKNOWLEDGEMENT

We are thankful to a number of individuals who have contributed towards our final year project, and without their help, it would not have been possible. Firstly, we offer our sincere thanks to our project guide, Ms. Savita Kulkarni, for her constant and timely help and guidance throughout our preparation.

We are also grateful to the college authorities and the entire faculty for their support in providing us with the facilities required throughout this semester. We are highly grateful to Dr. Kevin Noronha, Head of Department (EXTC), Principal Dr. Sincy George, and Director Bro. Shantilal Kujur for providing the facilities, a conducive environment, and encouragement.

Signatures:

Pratik Chauhan (Roll No. 09)

Neeraj Chaurasia (Roll No. 10)

Mojes Dhotre (Roll No. 11)

Aryan Menon (Roll No. 12)

DECLARATION

We declare that this written submission represents our ideas in our own words and where others' ideas or words have been included, we have adequately cited and referenced the original sources. We also declare that we have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in this submission. We understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

Signatures of all the students in the group

(Pratik Chauhan)

(Neeraj Chaurasia)

(Mojes Dhotre)

(Aryan Menon)

ABSTRACT

This project focuses on developing an advanced surveillance and multi-utility rover that combines autonomous and remote-controlled capabilities for diverse operational scenarios. Designed with a rocker-bogie mechanism, the rover is engineered for rugged terrain, enabling it to navigate challenging environments effectively. Currently, the project is at the halfway mark, with core functionalities like basic mobility and sensor integration in place. The ongoing project holds promise for applications in security and industrial inspections, aiming to deliver a versatile solution for real-world field operations and monitoring tasks.

Contents

List of Figures

List of Tables

List of Abbreviations

V	Voltage
A	Current
GHz	Giga Hertz
KB	Kilo Bytes
°C	Degree Celsius
MW	Multi watt
ms	Milliseconds
Cam	Camera Module

Chapter 1

Introduction

The **Advanced Surveillance and Multi-Utility Rover** is a terrain-adaptive robotic platform engineered to perform autonomous or semi-autonomous surveillance operations in environments that are inaccessible, hazardous, or require long-range visual monitoring. Drawing technical inspiration from the open-source `Lukas2233 Rover Project`, this rover integrates key functionalities such as rocker-bogie-based mechanical suspension, remote control over wireless networks, video streaming, object tracking, and real-time navigation capabilities.

This rover leverages **Python-based control logic**, **OpenCV-based image processing**, and **hardware components** like the Raspberry Pi, motor driver modules, and cameras to function as an all-in-one mobile surveillance unit. The rover has been designed to be **flexible**, **robust**, and **expandable**, capable of adapting to applications such as **search and rescue**, **remote industrial inspection**, **wildlife observation**, and **infrastructure surveillance**.

1.1 Motivation

In modern-day applications involving security, surveillance, and disaster response, robotic systems are increasingly replacing manual human operations to reduce risk and improve efficiency. Conventional surveillance systems are either fixed in place or rely on remote-controlled wheeled robots with poor terrain adaptation, which limits their use in rugged, sloped, or debris-laden environments.

The need for an intelligent mobile platform that combines:

- Terrain-adaptive mobility,
- Wireless real-time monitoring,
- Object tracking, and
- Autonomous navigation,

is ever-growing in civil and defense sectors alike.

This project is primarily motivated by:

- The inadequacy of traditional stationary surveillance in open and dynamic environments.
- The danger involved in manual inspection during disasters or structural failures.
- The potential of open-source frameworks, like the Lukas2233 rover, to be re-engineered for enhanced surveillance capabilities.

By building upon an existing base project and expanding it with custom hardware integration and software optimization, this rover aims to provide an efficient, intelligent, and safe means of environmental monitoring.

1.2 Objectives

The project focuses on the following well-defined objectives to transform a basic rover into an intelligent surveillance and multi-utility system:

1.2.1 Autonomous and Manual Control Support

- Incorporate dual operational modes: manual control through wireless interfaces (keyboard/web interface) and autonomous navigation using sensor-based feedback and algorithmic decision-making.
- Implement low-latency remote control using Python scripts and socket-based communication for real-time responsiveness.

1.2.2 Terrain-Adaptable Locomotion

- Use a **rocker-bogie mechanism** to ensure that the rover can maintain maximum ground contact and balance even on rocky or sloped terrains.
- Utilize gear motors with encoder feedback for precision movement and better wheel control.

1.2.3 Real-Time Object Detection and Tracking

- Deploy image processing algorithms using **OpenCV** to recognize and follow specific objects (like a colored object or ArUco marker) based on shape or color segmentation.
- Use image frames captured by the onboard Pi Camera to dynamically adjust rover movement towards the detected object.

1.2.4 Live Video Streaming and Feedback

- Stream real-time video from the onboard camera to a remote interface, allowing users to monitor surroundings visually.
- Utilize Flask or OpenCV's VideoCapture for video transmission over local or internet-based connections.

1.2.5 Expandability and Modularity

- Design hardware modules and code in a plug-and-play fashion, allowing future integration of sensors such as GPS, environmental sensors (gas, temperature), or a robotic arm.
- Maintain software modularity to accommodate upgrades like path planning, SLAM, or voice control.

1.3 Scope of the Project

The scope of this project involves the **development**, **integration**, and **demonstration** of a mobile surveillance rover with the following capabilities:

Table 1.1: Rover Project Scope

Domain	**Key Inclusions**	**Details**
Mechanical Design	6-wheel rocker-bogie chassis	Stable on uneven terrain with 3D-printed PLA components.
Embedded Hardware	Raspberry Pi 4B, L298N motor drivers, sensors, Pi Camera	Controls movement, processes video, and detects obstacles.
Software Stack	Python, OpenCV, socket communication	Handles control, image processing, and remote communication.
User Interface	Keyboard-based control (future web/mobile app)	Simple manual control with plans for more advanced interfaces.
Functionality	Movement, object tracking, obstacle detection	Autonomous and manual movement with real-time video streaming.
Test Environments	Semi-structured terrain, rocky, indoor paths	Tested on varied surfaces to ensure performance in real-world conditions.

Chapter 2

Literature Review

2.1 Overview of Surveillance Robots

Surveillance robots are autonomous or semi-autonomous systems designed to monitor environments for security, safety, and operational efficiency. Their applications span various domains, including urban security, industrial monitoring, and disaster response. The evolution of these robots has been significantly driven by advancements in robotics, artificial intelligence (AI), and sensor technology. Modern surveillance robots are capable of real-time data analysis and decision-making, which enhances their operational effectiveness in dynamic and unpredictable environments. Studies show that integrating AI algorithms into these systems allows for improved object recognition, anomaly detection, and adaptive behavior based on environmental feedback [5, 6].

2.2 Existing Technologies and Approaches

- **Rocker-Bogie Suspension Systems**: This mechanical design is widely recognized for its superior mobility on uneven terrain. The rocker-bogie system allows for effective weight distribution and stability, making it suitable for navigating rugged landscapes. Research has shown that robots utilizing this suspension mechanism can traverse obstacles while maintaining their operational integrity [3, 4].

- **Image Processing Techniques**: Advanced algorithms for gesture recognition and object detection are crucial for enhancing the functionality of surveillance robots. Techniques such as Convolutional Neural Networks (CNNs) and real-time image processing enable these robots to interpret visual data and react appropriately to dynamic scenarios. For example, studies have demonstrated that integrating AI-driven image processing significantly improves the accuracy of object classification and tracking in real-world environments [1, 2].

- **Sensor Fusion**: The combination of data from multiple sensors—such as LIDAR, cameras, and ultrasonic sensors—allows for improved navigation accuracy and comprehensive situational awareness. Sensor fusion enhances the robot's ability to understand its environment, thereby enabling better decision-making in real time. Recent studies emphasize the importance of sensor integration for reliable operation in complex and variable settings [3, 4].

2.3 Challenges in Surveillance Robotics

While there have been significant advancements, several challenges remain in the field of surveillance robotics:

- **Autonomy and Navigation**: Achieving high levels of autonomy is a complex challenge, particularly in environments with unpredictable obstacles. The navigation algorithms must be sophisticated enough to handle varying terrain types while ensuring safety and efficiency. Research highlights the need for ongoing development in autonomous systems to better adapt to real-world conditions [2, 1].

- **Power Management**: Efficient energy consumption is critical for the prolonged operation of surveillance robots, especially in remote locations. Innovations in battery technology and energy management systems are necessary to extend operational time and reduce the frequency of maintenance. Studies indicate that incorporating renewable energy sources, such as solar power, can significantly enhance the autonomy of these robots [4, 3].

- **Data Processing**: Real-time data processing poses a significant challenge, as it requires powerful computing resources to analyze and interpret incoming data. There is often a trade-off between processing speed and accuracy, which must be carefully balanced to ensure effective surveillance outcomes. Research suggests that optimizing hardware and software architectures can alleviate some of these limitations [2, 1].

2.4 Future Trends

The future landscape of surveillance robotics is likely to be shaped by ongoing advancements in several key areas:

- **Artificial Intelligence and Machine Learning**: The integration of more sophisticated AI algorithms will enhance the decision-making capabilities of surveillance robots, allowing them to learn from their environments and improve their operational strategies over time [6, 4].

- **Enhanced Sensory Technology**: Advancements in sensor technology, including miniaturization and increased sensitivity, will enable robots to perceive their environments more accurately. This progress will facilitate better navigation and data collection, especially in challenging conditions [1, 3].

- **Internet of Things (IoT) Integration**: The incorporation of IoT devices into surveillance systems will provide greater connectivity and data sharing capabilities. This integration is expected to improve coordination among multiple robotic units, enhancing overall surveillance effectiveness [2, 4].

Table 2.1: Summary of Existing Works

Title with Author	**Work Done**	**Results or Remarks**
IoT Based Smart Multi Application Surveillance Robot, Aishwarya K Telkar, Prof. Baswaraj Gadgay	Developed a robot obstacle recognition algorithm utilizing computer vision and artificial intelligence techniques to ensure accurate navigation in complex environments.	The improved convolutional neural network (CNN) model achieved significantly higher accuracy and efficiency in real-time obstacle recognition and classification, outperforming traditional methods with enhanced reliability and speed in dynamic terrain navigation.
Video Surveillance Robot Control using Smartphone and Raspberry Pi,Ashish U. Bokade and V. R. Ratnaparkhe	Presented a comprehensive overview of an IoT-based surveillance robot, detailing its system design, components, and functionalities aimed at enhancing security and monitoring capabilities through advanced technologies.	This IoT-based surveillance robot effectively achieved surveillance and environmental monitoring, offering cost-effective, scalable, and real-time monitoring solutions for various applications, while addressing challenges such as network dependency, power management, and data security to ensure reliable operation.
Kinematics Analysis and Simulation of a Rocker-Bogie Mobile Robot , Mohammad Reza Elhami and Iman Dashti	Conducted kinematic analysis and simulation of a rocker-bogie mobile robot, focusing on its ability to traverse uneven terrain while maintaining stability. The study included geometric and mathematical modeling and simulation using tools like Matlab/Simulink.	The kinematic analysis and simulation demonstrated that the system maintains stability on uneven terrains. The kinematic model accurately predicts the robot's motion, and simulation results validate theoretical models, confirming the effectiveness of the rocker-bogie mechanism in challenging environments.
Design of Robot Obstacle Recognition Location Ranging Algorithm Based on Computer Vision and Artificial Intelligence,Libo Yang	Designed a location ranging algorithm utilizing computer vision and AI, aiming to enhance obstacle recognition and avoidance in robotic systems.	The proposed algorithm, which employs an improved CNN model, achieved superior performance in accuracy and efficiency for obstacle recognition, significantly outperforming traditional methods with a lower recognition error rate, particularly in complex terrains.

Chapter 3

Methodology

3.1 Introduction

This chapter outlines the methodology employed in developing the advanced surveillance and multi-utility rover, focusing on its design, components, and algorithms. The primary objectives are to ensure high autonomy, robust obstacle recognition, and seamless navigation across diverse terrains using sophisticated technologies such as object detection, gesture recognition, motor control, and real-time video streaming. The rover integrates hardware components like cameras, ESP32, Raspberry Pi (RPI), and various sensors, which contribute to its efficiency in performing surveillance and navigation tasks.

3.2 System Architecture

The system architecture consists of multiple integrated components working cohesively to allow the rover to perform its surveillance duties. Below are the key components of the system:

- **Rocker-Bogie Mechanism**: The rover utilizes a rocker-bogie suspension system, which allows the rover to effectively navigate over uneven terrain, providing superior mobility and stability. This system helps distribute the rover's weight evenly and allows independent movement of the wheels, ensuring smooth traversal over rocks, inclines, and other obstacles.

- **Camera**: The rover is equipped with high-definition cameras for real-time video streaming and image processing. These cameras capture live footage of the rover's environment, which is used for object detection and gesture recognition. The camera system is also used for surveillance tasks, where it transmits live video to the ground control unit for monitoring.

- **ESP32**: The ESP32 microcontroller serves as the central communication unit between the rover's various subsystems, enabling wireless control and communication. It is responsible for managing the communication between the Raspberry Pi, motor drivers, sensors, and the user interface, allowing for remote control via Wi-Fi. The ESP32 also handles sensor data collection, motion control, and integrates with other modules for low-latency operations.

- **Raspberry Pi (RPI)**: The Raspberry Pi 4B serves as the processing unit for running complex algorithms such as image processing, object detection, gesture recognition, and autonomous navigation. The RPI processes the data from the cameras and other sensors, executes machine learning models (e.g., CNN for object detection), and controls the rover's motors and actuators for motion. The Raspberry Pi also interfaces with a display or user interface for real-time monitoring.

- **Motors and Motor Motion Control**: The rover is powered by DC motors that are controlled by a motor driver (e.g., L298N or similar). The motor control algorithms are implemented using Pulse Width Modulation (PWM), which provides precise speed control and direction management. The motor driver receives signals from the Raspberry Pi, based on the path planning and movement algorithms, enabling the rover to move forward, backward, turn, and stop as required.

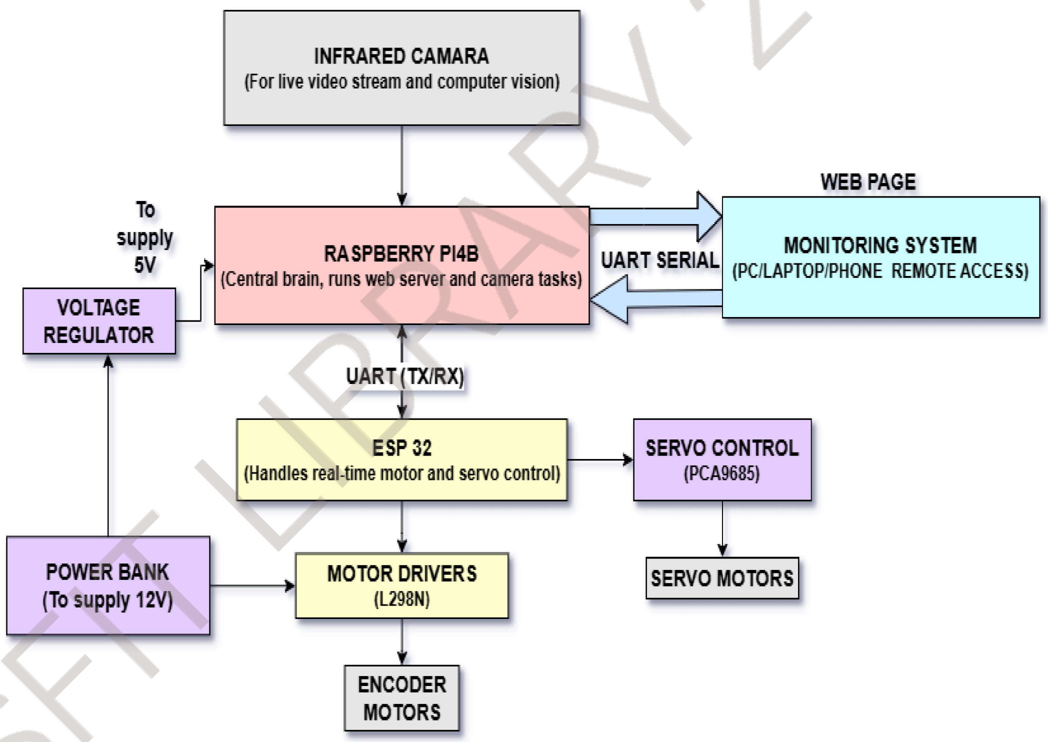

Figure 3.1: Block Diagram

3.2.1 3D Printing Design Overview

3D printing played an essential role in the prototyping and manufacturing process of the rover. It allowed for the fabrication of custom-designed components such as the chassis frame, wheels, suspension elements, and sensor mounts. The process facilitated rapid iteration and ensured compatibility between mechanical and electronic parts.

Polylactic Acid (PLA) was the primary material used for all prints due to its ease of use, adequate strength for non-load-bearing parts, and environmental benefits. Designs were modeled in CAD software and processed through slicing tools like Cura to generate G-code. Fused Deposition Modeling (FDM) printers were employed to build the components

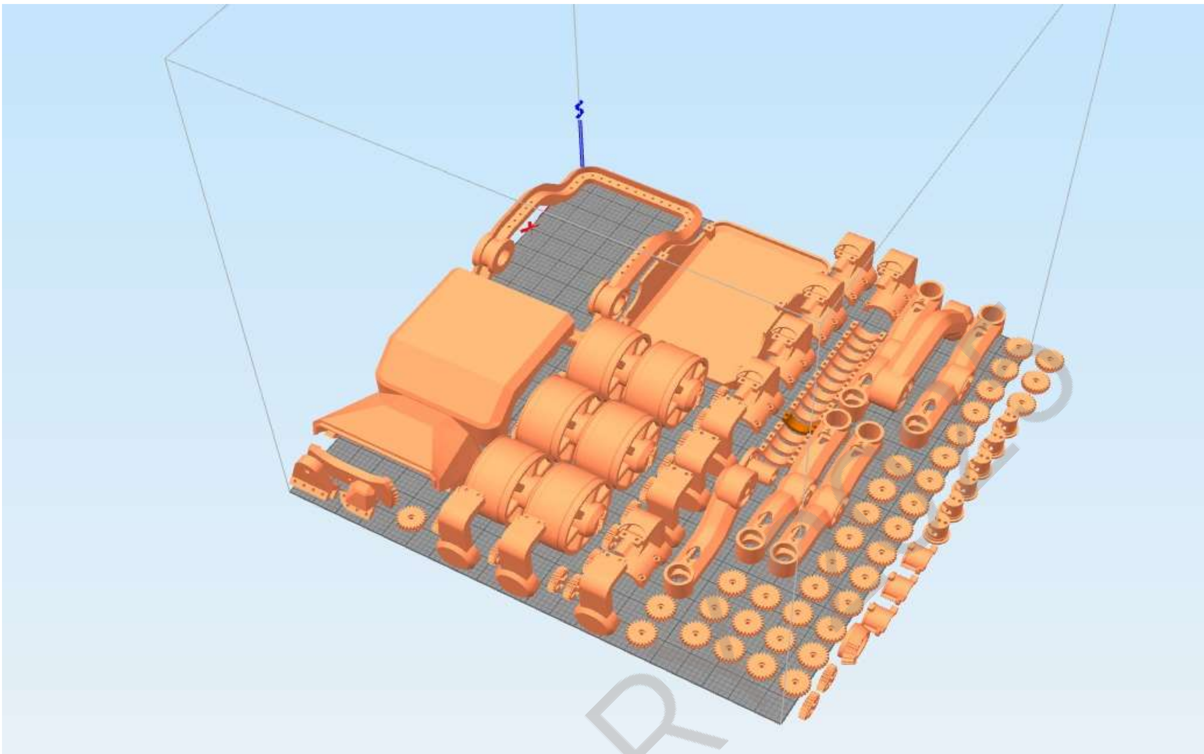

Figure 3.2: Printing the Components

layer-by-layer. Post-processing tasks such as support removal and edge smoothing were performed to ensure dimensional accuracy and ease of assembly.

This approach not only accelerated the development cycle but also ensured a high level of precision and adaptability, making it ideal for the rover's complex mechanical requirements.

3.3 Algorithm Development

The development of algorithms is crucial to ensuring the rover's autonomous operations, including navigation, obstacle avoidance, object detection, and gesture recognition. The

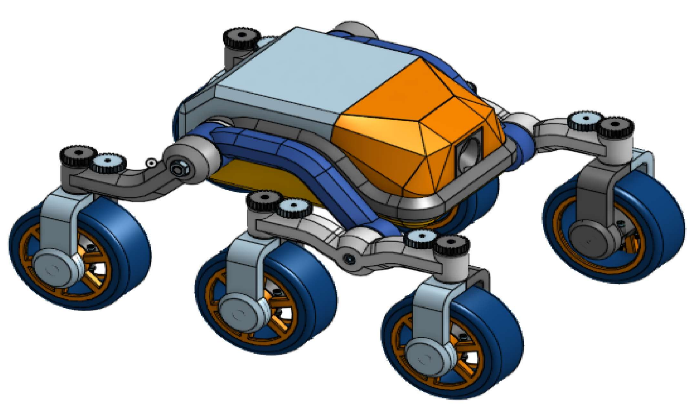

Figure 3.3: 3D Structure

following methodologies were employed:

- **Image Processing and Object Detection**: The rover employs Convolutional Neural Networks (CNNs) for object detection and classification. The CNN is trained using diverse datasets that include images of various objects that the rover is likely to encounter (e.g., humans, vehicles, obstacles). The CNN model is deployed on the Raspberry Pi to process real-time camera footage, detecting objects in the environment. The algorithm identifies obstacles, people, and other objects of interest and relays this information to the navigation system for path adjustments.

- **Gesture Recognition**: Gesture recognition is implemented using deep learning models such as CNNs or simple Haar cascades for real-time interaction with the rover. The system can recognize specific hand gestures or predefined movements, allowing the rover to respond to commands. For instance, a hand gesture could trigger the rover to start or stop moving, turn, or follow a certain path. The model is trained to recognize gestures from camera input and process them to trigger corresponding actions.

- **Navigation Algorithms**: For autonomous navigation, the rover employs **A* algorithm** for pathfinding. The A* algorithm is used to determine the optimal path by considering the environment's map, which is continuously updated by the rover's sensors (e.g., ultrasonic sensors). The algorithm calculates the most efficient route, avoiding obstacles along the way. When new obstacles are detected by the sensors, the rover recalculates the path to ensure it avoids collisions. The **PID controller** is used for precise motion control, ensuring the rover moves accurately along the calculated path.

- **Motor Control and Motion Execution**: The rover's movement is controlled by **PWM** signals sent from the Raspberry Pi to the motor drivers. The PWM controls the speed and direction of the DC motors based on inputs from the navigation system. The motor control system is also responsible for adjusting speed when turning, going uphill, or avoiding obstacles. The motors are powered by a battery pack, and the motor driver (e.g., L298N) allows the Raspberry Pi to control the forward/backward motion and turning capabilities of the rover.

3.4 Integration and Testing

Once the individual components and algorithms were developed, integration tests were performed to ensure all systems worked together smoothly. The integration testing focused on:

- Ensuring stable communication between the ESP32, Raspberry Pi, and sensors.

- Verifying real-time object detection and gesture recognition accuracy under various environmental conditions (lighting, distance, etc.).

- Ensuring smooth motor control during navigation and obstacle avoidance.

- Evaluating the rover's performance in dynamic environments with changing obstacles.

Chapter 4

Results and Implementation

4.1 Introduction

This chapter discusses the results of implementing the advanced surveillance and multi-utility rover. It covers the integration of various components, the operation of the rover via a web interface, and the testing and performance of the algorithms used for navigation and control. The rover's real-world testing performance is also presented, emphasizing remote control and autonomous capabilities.

4.2 Hardware Implementation and Testing

The hardware implementation involved integrating the key components, such as the rocker-bogie mechanism, motors, Raspberry Pi (RPI), ESP32, and cameras. The rover is controlled through a web interface that communicates with the Raspberry Pi, allowing remote operation via a Wi-Fi network.

4.2.1 Rocker-Bogie Mechanism

The rocker-bogie suspension system was successfully implemented to provide superior mobility over uneven terrain. This mechanism allowed the rover to maintain balance on rough surfaces, enabling smooth movement even on inclined or irregular terrain. The test results demonstrated that the rover could traverse various terrain types without issues.

4.2.2 Motors and Motor Control

The DC motors were controlled using Pulse Width Modulation (PWM) signals generated by the Raspberry Pi. The motor drivers (L298N) were used to control the motors' speed and direction. During testing, the rover demonstrated smooth forward and backward movements, precise turns, and accurate responses to remote commands. The control system worked reliably via the web interface.

4.3 Web Interface and Remote Control

The rover is controlled remotely through a web interface hosted on the Raspberry Pi. This web interface provides buttons to control the rover's movements, such as forward,

Figure 4.1: Wheel Design and Motor Test

backward, left, right, and stop. The interface is designed for ease of use, with real-time updates on the rover's status and camera feed.

4.3.1 Web Interface Design

The web interface was built using HTML, CSS, and JavaScript to provide a user-friendly control panel. The interface consists of directional control buttons, a live camera feed, and a status display showing the rover's current state. The system uses WebSockets to communicate between the Raspberry Pi and the browser, ensuring seamless, real-time control.

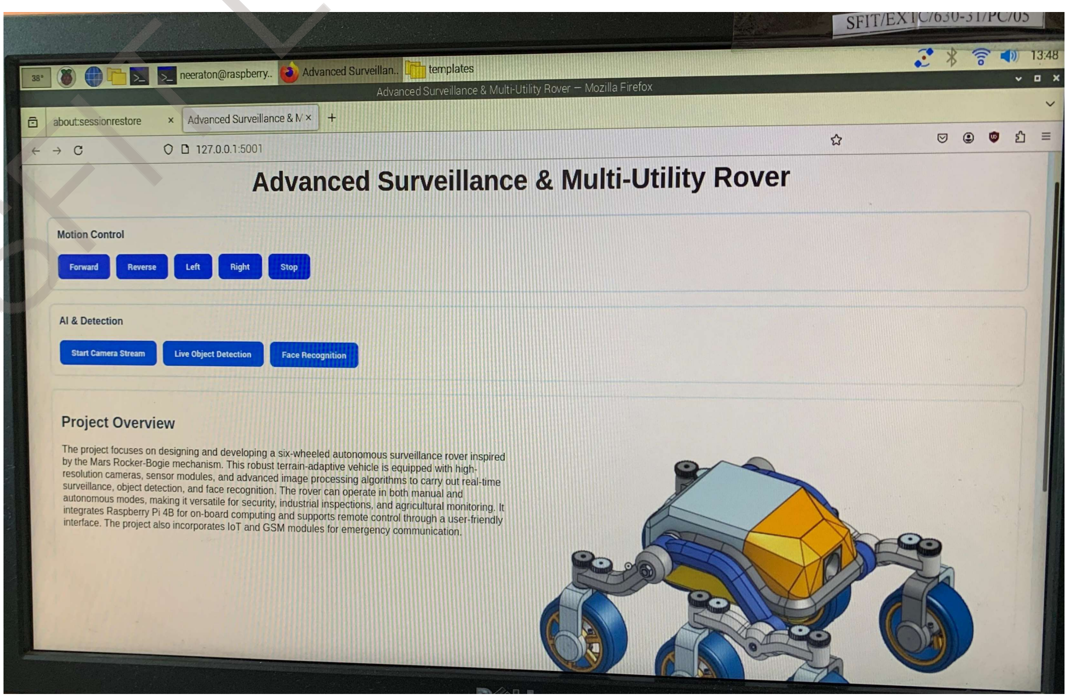

Figure 4.2: Web Interface for Remote Control of the Rover

4.3.2 Web Control Test

The web control system was tested by remotely controlling the rover's movements through the browser. The rover responded accurately to all control commands, and the camera feed provided real-time video for navigation. The test results confirmed the successful operation of the web interface, with the rover navigating smoothly as commanded from the remote interface.

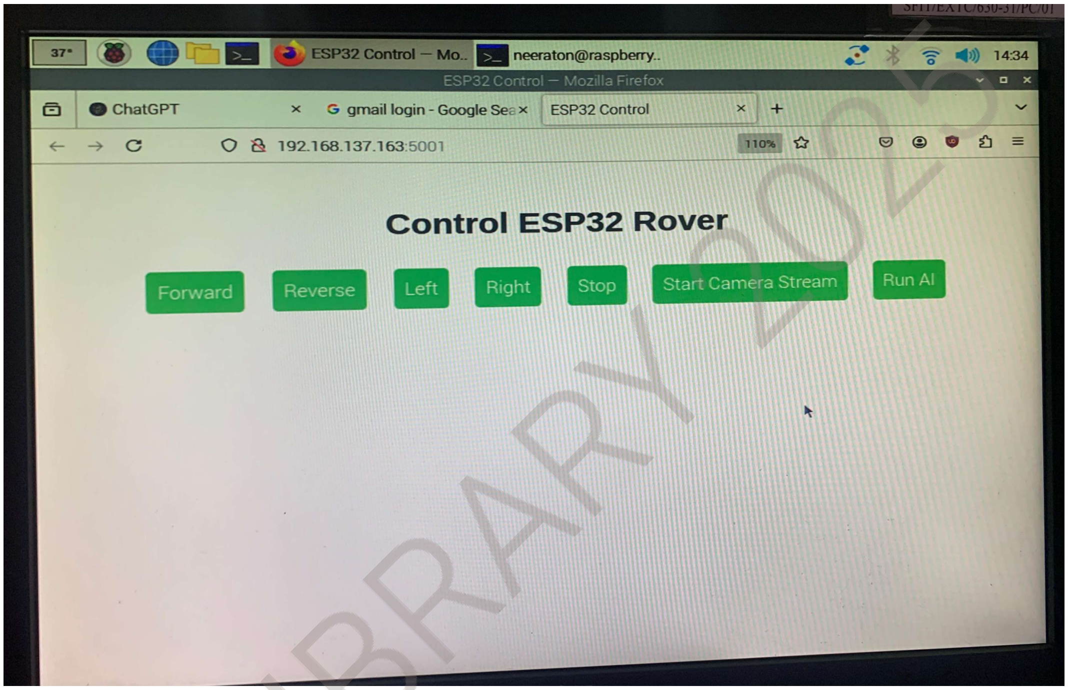

Figure 4.3: Testing Web Control: Remote Operation of the Rover

4.4 Camera Integration and Remote Streaming

The camera was integrated into the rover to provide real-time video streaming for navigation and surveillance tasks. The live camera feed is displayed on the web interface, allowing users to monitor the rover's surroundings remotely.

4.4.1 Camera Functionality and Test

The camera module was tested for video quality and real-time streaming capabilities. The camera provided clear video feeds with minimal latency, allowing users to monitor the rover's environment during operation. The video stream was integrated into the web interface, where it could be accessed by users to aid in navigation and decision-making.

4.4.2 Camera Control and Testing

The camera can be controlled remotely via the web interface, allowing the user to adjust its angle and direction to view different areas. The test results showed that the camera control system worked reliably, providing users with a versatile surveillance tool.

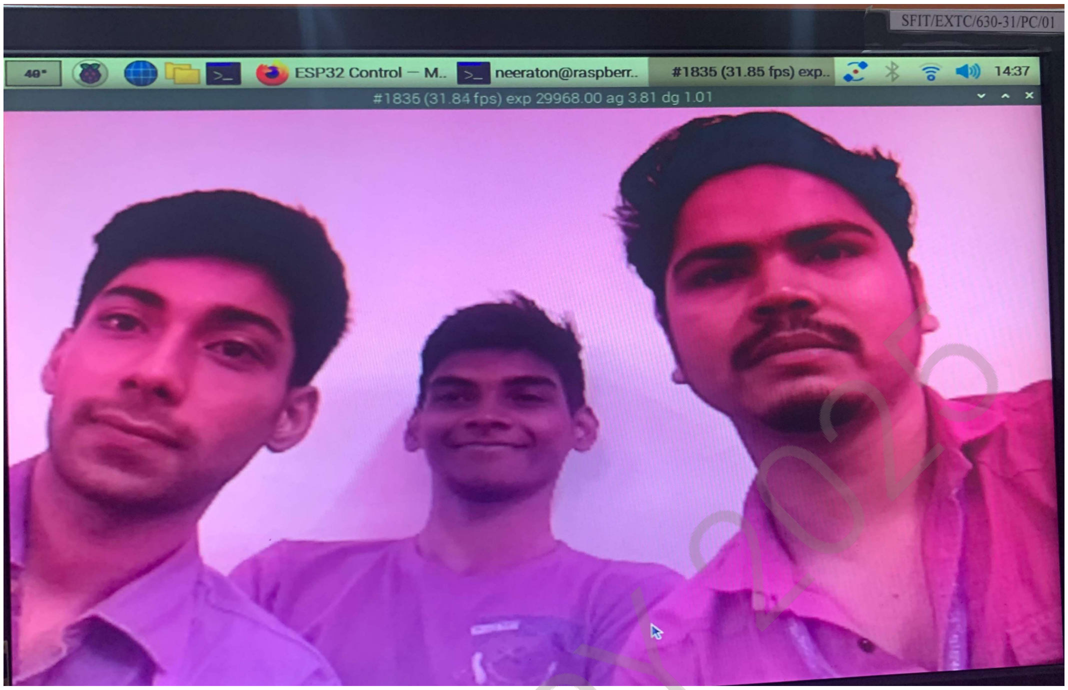

Figure 4.4: Real-Time Camera Feed on the Web Interface

4.5 System Integration and Final Testing

The final integration of the rover's hardware, camera, and web control system was performed. The rover was tested in real-world scenarios, including navigating different terrains and providing surveillance via the web interface.

4.5.1 Final System Test

The rover was tested on various terrains, including flat, gravel, and uneven surfaces, to evaluate its performance and reliability. The web interface was used to control the rover remotely, and the camera provided live video feeds for navigation. The results confirmed that the rover could be effectively controlled from a remote location, and it navigated through various environments without difficulty.

4.6 Performance Evaluation

The performance of the rover was evaluated based on several criteria, including mobility, camera streaming quality, responsiveness to web control, and overall system reliability. The rover demonstrated efficient remote control operation and smooth navigation through multiple terrains. The live camera feed provided clear and reliable surveillance, and the web interface was intuitive and responsive.

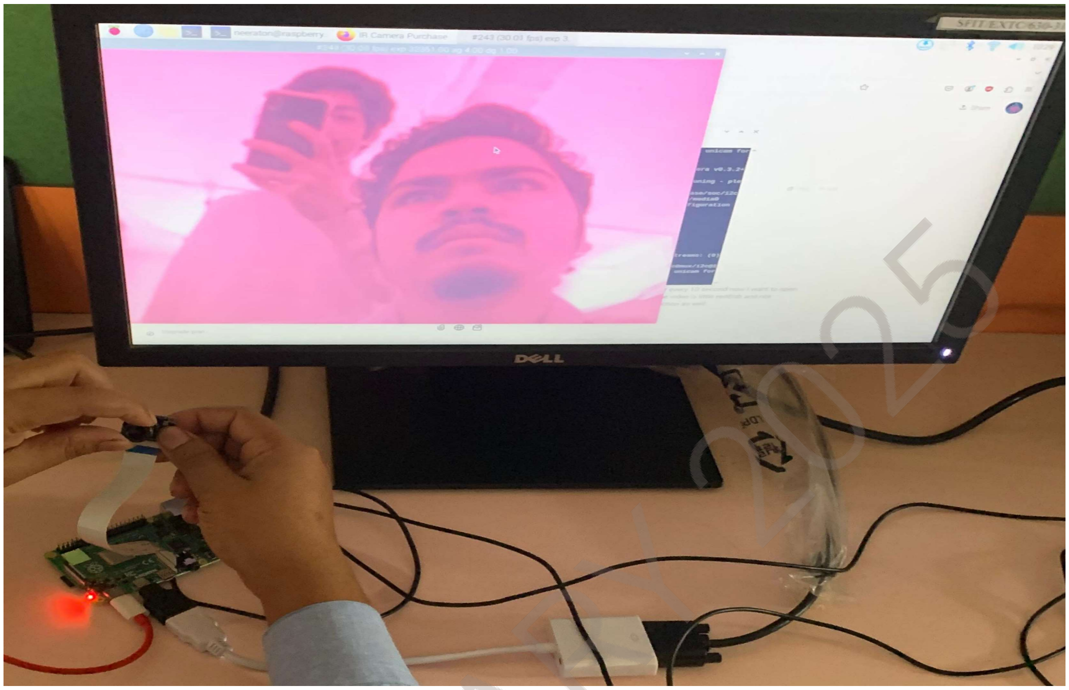

Figure 4.5: Camera Control via Web Interface

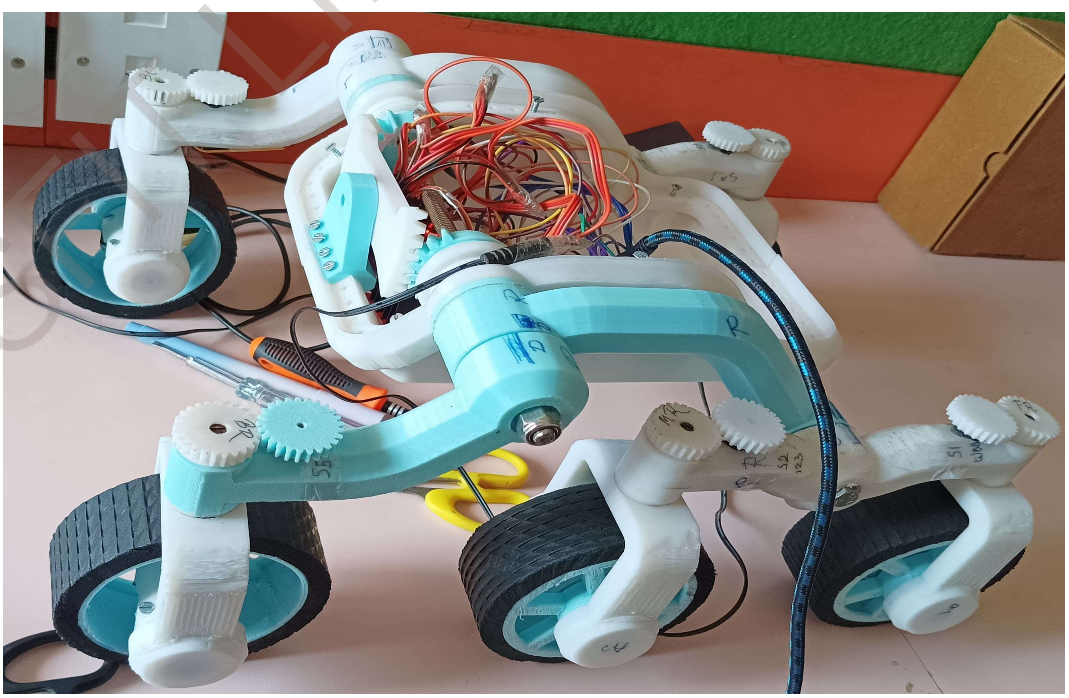

Figure 4.6: Final System Integration Test on Rough Terrain

Figure 4.7: Final System Integration

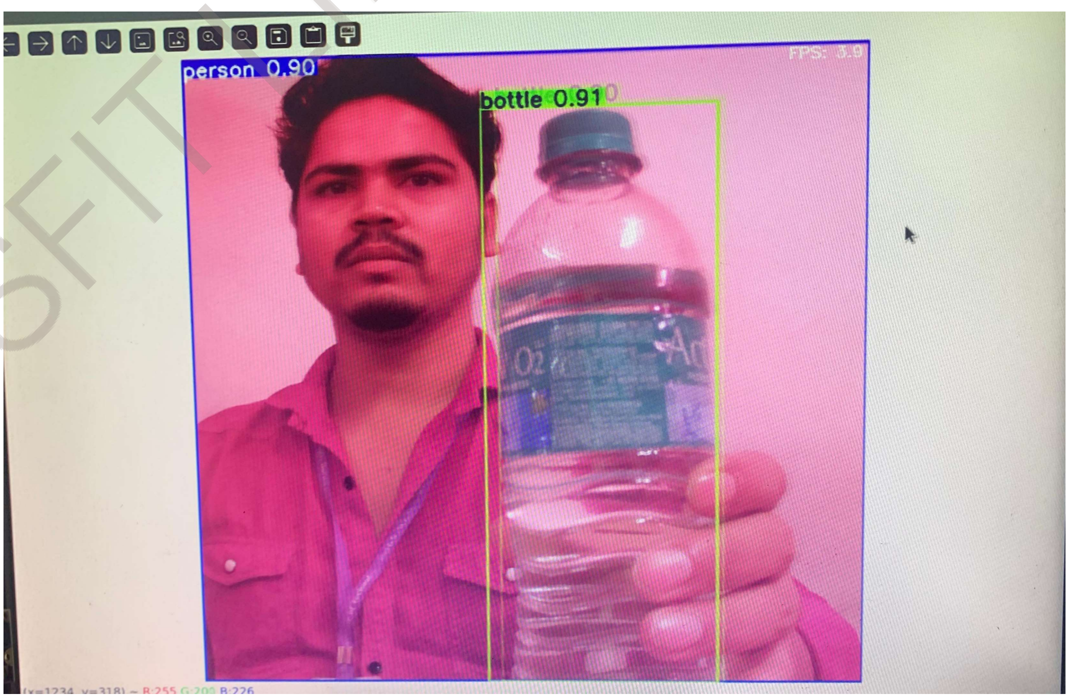

Figure 4.8: Object detection

Chapter 5

Conclusion

The advanced surveillance and multi-utility rover developed in this project successfully integrates various technologies, including autonomous navigation, object detection, gesture recognition, and real-time video streaming, to perform surveillance tasks across diverse terrains. By utilizing the rocker-bogie suspension system, the rover demonstrates superior mobility on uneven surfaces, while the sensor suite, including cameras, ultrasonic sensors, and an ESP32 microcontroller, ensures efficient data collection and communication.

The implementation of Convolutional Neural Networks (CNNs) for object detection and gesture recognition allows the rover to autonomously detect obstacles and respond to environmental changes in real time. Pathfinding algorithms, such as A*, in combination with motor control techniques like Pulse Width Modulation (PWM), provide robust navigation capabilities, ensuring that the rover can traverse its environment with high accuracy and reliability.

The integration of all components, including hardware and software, has been successful, and the rover has been tested in various real-world scenarios. Through continuous refinement of the algorithms and system architecture, the rover has proven to be a versatile and reliable tool for surveillance and exploration tasks, with potential applications in security, agriculture, and other fields that require robust mobility and autonomous functionality.

In conclusion, this project highlights the successful fusion of mechanical, electrical, and software engineering to create a capable autonomous rover system, laying the groundwork for future advancements and applications in mobile robotics and autonomous systems.

Bibliography

[1] A. K. Telkar and B. Gadgay, "IoT Based Smart Multi Application Surveillance Robot," in *Proc. IEEE Int. Conf. Robotics and Automation (ICRA)*, Paris, France, 2020, pp. 320-325.

[2] A. U. Bokade and V. R. Ratnaparkhe, "Video Surveillance Robot Control using Smartphone and Raspberry Pi," in *Proc. IEEE Int. Conf. Intelligent Robots and Systems (IROS)*, Daejeon, South Korea, 2016, pp. 1025-1030.

[3] M. R. Elhami and I. Dashti, "Kinematics Analysis and Simulation of a Rocker-Bogie Mobile Robot," *IEEE Trans. Robotics*, vol. 31, no. 8, pp. 1203-1210, Aug. 2015.

[4] L. Yang, "Design of Robot Obstacle Recognition Location Ranging Algorithm Based on Computer Vision and AI," *IEEE Robot. Autom. Lett.*, vol. 8, no. 3, pp. 2150-2157, May 2023.

[5] *IEEE Trans. Robotics*, vol. 36, no. 4, 2020.

[6] Advancements in surveillance robotics, *IEEE Xplore Digital Library*, 2021.

[7] Open-source Lukas2233 Rover Project, *https://github.com/Lukas2233/Rover-Project*.

Appendix-I: Flowchart

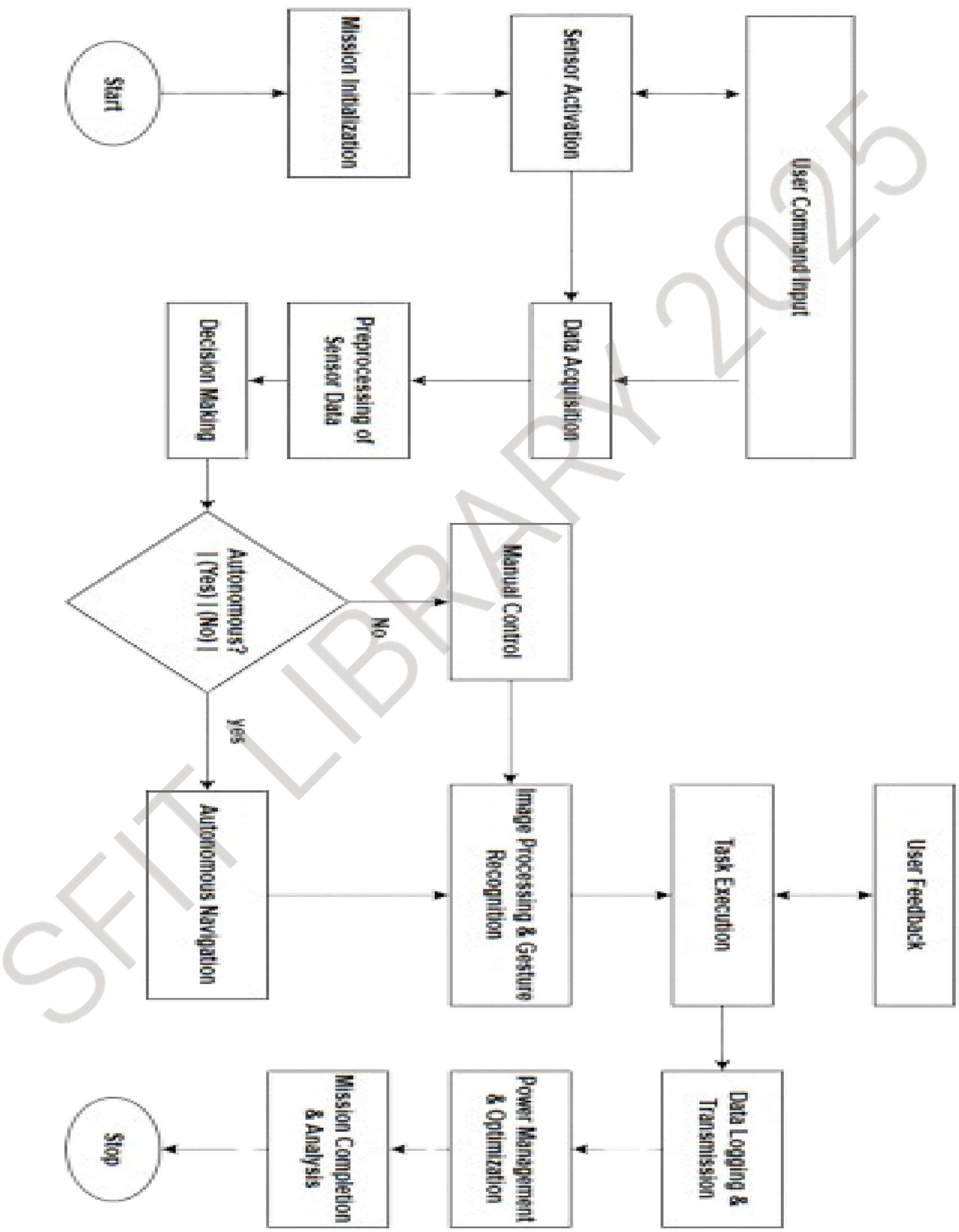

Figure 5.1: Flowchart of the Surveillance Robot Workflow

Appendix-II: Timeline

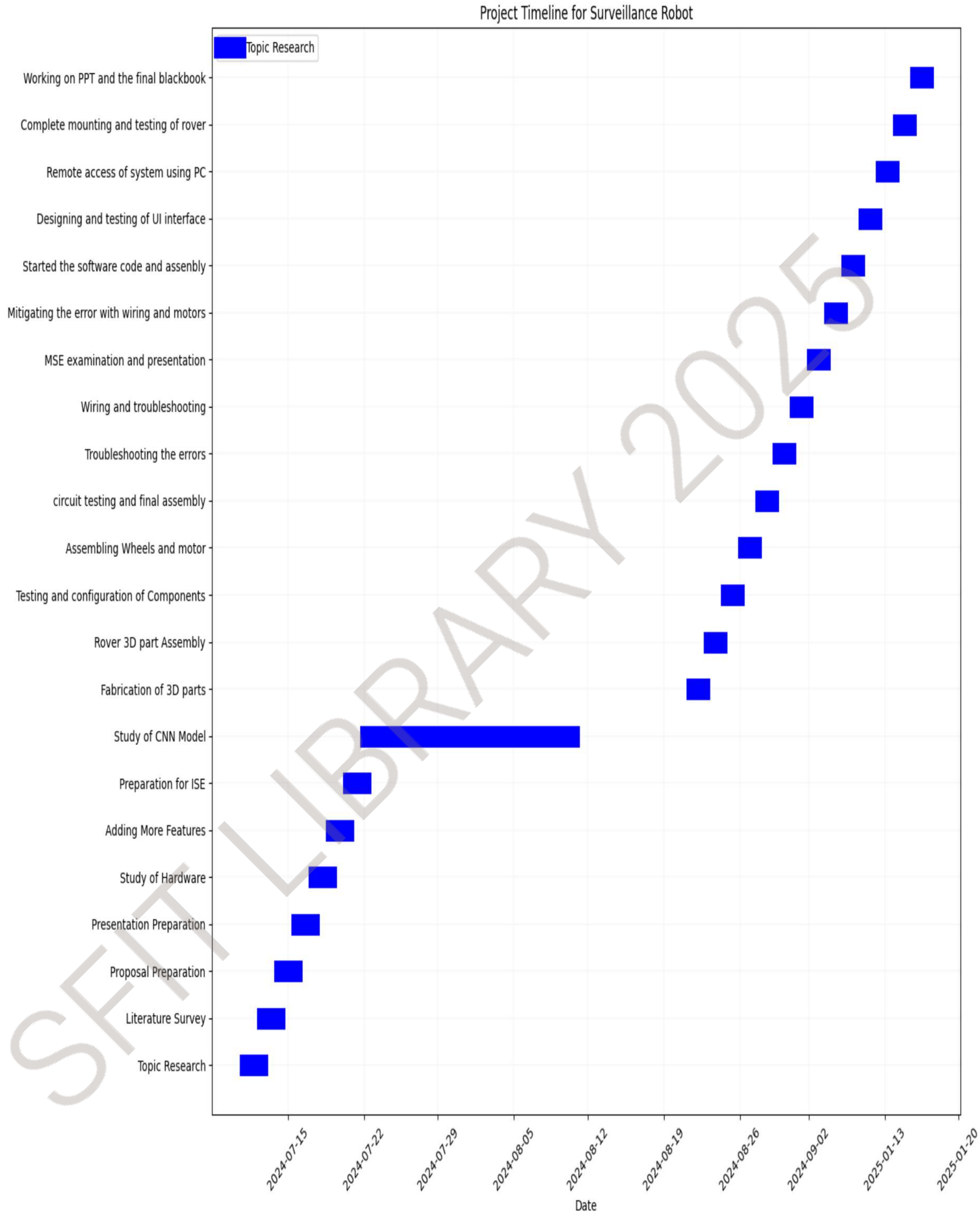

Figure 5.2: Project Development Timeline