International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 04 Issue: 04 April 2026
Page No: 2373-2385
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2026.0319

AI- Powered Autonomous Rover for Terrain Navigation and Obstacles Avoidance

Anusree KP[1], Deepek M[2], Aravindan CS[3], Mowlikadevi M[4],Vignesh S[5], Dr.C.Dhavamani[6]
[1,2,,3,4]UG- Aerospace Engineering, Mahendra Engineering College, Namakkal, Tamil Nadu,637503, India
[5]Assistant Professor, Aerospace Engineering, Mahendra Engineering College, Namakkal, Tamil Nadu,637503, India
[6]Head of department, Aeronautical Engineering,Mahendra Engineering College, Namakkal, Tamil Nadu,637503,India
EmailID:*anusreekpanu5@gmail.com[1],deepakmk1245@gmail.com[2],aravindavi504@gmail.com[3], mowlikathaarika@gmail.com[4]*

Abstract

Exploring unknown or dangerous environments without risking human life is becoming increasingly important, and this project focuses on developing a smart and affordable AI-powered rover capable of navigating outdoor terrains while detecting obstacles in real time. The rover uses a Raspberry Pi 5 as its brain, along with a camera and sensors that help it understand its surroundings. A machine learning model is used to recognize obstacles, while an ultrasonic sensor measures distance and an IMU maintains balance and stability. By combining all this information, the rover can make quick decisions, avoid collisions, and adjust its movement based on terrain conditions such as grass, soil, and gravel. A key feature of the system is its flexibility, as it can operate autonomously or be controlled manually through a simple web-based interface that provides live video streaming. This allows human intervention whenever required. The rover was tested in real outdoor conditions, where it demonstrated smooth movement, reliable obstacle detection, and adaptability to different surfaces. Overall, the project shows that an intelligent and effective autonomous system can be built using low-cost components, making it suitable for applications like surveillance, exploration, and research while reducing risks to human life.

Keywords: *Autonomous navigation; Embedded AI; Object detection; Sensor fusion; Terrain classification*

1. Introduction

In recent years, the development of autonomous ground vehicles (AGVs) has gained significant attention due to their ability to operate in hazardous and unstructured environments without direct human involvement. Advancements in artificial intelligence, embedded systems, and low-cost sensors have enabled the creation of compact and efficient robotic platforms capable of performing tasks such as surveillance, exploration, and terrain analysis. Machine learning techniques, particularly deep learning models, have greatly improved the capabilities of robots in object detection and environmental understanding. Traditional robotic systems relied heavily on manual control or basic rule-based algorithms, which often struggled in dynamic and complex outdoor environments. However, modern approaches using models like YOLO (You Only Look Once) have enabled real-time object detection with high accuracy, making them suitable for embedded applications (Chen, L et al., 2021; Zhang, Y et al., 2023). Additionally, terrain classification using machine learning has become an important aspect of autonomous navigation, allowing robots to adapt their movement based on surface conditions such as grass, gravel, or uneven terrain. Despite these advancements, many existing systems are either expensive or computationally demanding, limiting their practical deployment in cost-sensitive and real-time applications. There is a growing need for a low-cost, efficient, and intelligent system that can perform reliable navigation while maintaining real-time performance on embedded hardware. This work aims to design and develop an AI-powered autonomous rover capable of navigating mixed outdoor terrains using object detection, terrain classification, and sensor fusion. The proposed system integrates a lightweight deep learning model optimized for edge devices, along with ultrasonic and IMU sensors to enhance navigation accuracy and stability. The originality of this work lies in combining real-time AI inference, terrain-aware decision-making, and a hybrid control system (autonomous and manual) within a low-cost embedded platform. This approach provides a practical and scalable solution for applications such as surveillance, exploration, and research, bridging the gap between high-cost commercial systems and affordable academic prototypes.

1.1.Autonomous Navigation and Perception

Autonomous navigation is a key aspect of modern robotic systems, especially in applications involving unknown or unstructured environments. It refers to the ability of a machine to move from one location to another without continuous human intervention while safely avoiding obstacles. In recent years, this capability has improved significantly due to the integration of artificial intelligence and sensor technologies. Robotic systems use a combination of cameras, sensors, and algorithms to perceive their surroundings and make decisions. Vision-based techniques, particularly those using deep learning models, allow the system to detect and classify objects in real time. At the same time, sensors such as ultrasonic sensors and inertial measurement units (IMUs) provide additional information about distance, orientation, and motion. By combining these inputs through sensor fusion, the system can achieve better accuracy and reliability in navigation.

Furthermore, terrain awareness has become an important factor in outdoor robotics. Different surfaces require different movement strategies, and identifying terrain types helps improve stability and efficiency. Overall, the integration of intelligent perception and adaptive control plays a crucial role in enhancing the performance of autonomous rovers in real-world conditions.

1.2.Sensor Integration and Control System

The performance of an autonomous rover largely depends on how effectively it gathers and processes data from multiple sensors. Sensor integration, also known as sensor fusion, plays a crucial role in improving the accuracy and reliability of the system. In this project, different sensors such as ultrasonic sensors and an inertial measurement unit (IMU) are used to provide real-time information about the environment and the rover's motion. Ultrasonic sensors are primarily used for obstacle detection by measuring the distance between the rover and nearby objects. This helps in avoiding collisions and ensuring safe navigation. On the other hand, the IMU provides data related to orientation, acceleration, and angular velocity, which is essential for maintaining balance and stability, especially on uneven terrains. By combining these sensor inputs, the rover can make more informed decisions compared to relying on a single data source.

In addition to sensing, the control system is responsible for executing decisions based on the processed data. The rover operates in both autonomous and manual modes, allowing flexibility in operation. In autonomous mode, the system makes decisions independently, while in manual mode, the user can control the rover through a web-based interface. This hybrid control approach enhances usability and ensures better control in complex or uncertain environments.

2. Methodology

The methodology adopted in this work focuses on the design, development, and testing of an AI-powered autonomous rover capable of real-time navigation and obstacle detection. The system is built using a Raspberry Pi 5 as the central processing unit, integrated with a camera module, ultrasonic sensors, and an inertial measurement unit (IMU). The overall approach involves hardware integration, model deployment, and system-level testing. The object detection task is performed using a YOLOv8 deep learning model, which is trained and optimized for real-time inference on embedded hardware. The trained model is deployed using an efficient framework to ensure low latency and high performance. The rover captures live video through

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 04 Issue: 04 April 2026
Page No: 2373-2385
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2026.0319

the camera, processes each frame, and detects obstacles in real time. Based on the detected objects, navigation decisions are made to avoid collisions. Previously established object detection techniques and optimization methods are followed as described in earlier studies (Chen, L et al., 2021; Zhang, Y et al., 2023). For obstacle avoidance and stability, ultrasonic sensors and an IMU are used. The ultrasonic sensor continuously measures the distance to nearby objects, while the IMU provides orientation and motion data. These inputs are combined using sensor fusion techniques to improve the accuracy and reliability of navigation. The control system processes this data and adjusts the rover's movement accordingly. The rover operates in both autonomous and manual modes. A web-based interface is developed using FastAPI to enable live video streaming and remote control. This allows users to monitor and control the rover when necessary. The system is tested under real outdoor conditions to evaluate its performance in terms of obstacle detection, terrain adaptability, and response time.

2.1.System Architecture of AI Based Autonomous Rover

The figure illustrates the overall system architecture of the proposed AI-powered autonomous rover, highlighting the integration of hardware components and their interconnections. At the core of the system is the Raspberry Pi 5, which acts as the main processing unit responsible for executing AI algorithms, handling sensor data, and controlling the rover's movement. The power system supplies regulated 5V DC to the Raspberry Pi through a power regulator, ensuring stable operation. The camera module is connected via the CSI interface, enabling real-time video capture for object detection tasks. The input layer consists of sensors such as the ultrasonic sensor and the IMU (MPU6000). The ultrasonic sensor provides distance measurements for obstacle detection using trigger and echo signals, while the IMU supplies motion and orientation data for maintaining stability. The control signals from the Raspberry Pi are sent to the L298N motor driver, which acts as an interface between the low-power control unit and the high-power motors. The motor driver controls the speed and direction of the motors based on the processed data. Finally, the output/actuation layer represents the motors and telemetry system, which execute movement commands and provide feedback. The system also incorporates a FastAPI server back end for communication and remote control. Overall, this architecture demonstrates a well-integrated system combining AI, sensors, and control mechanisms for autonomous navigation figure 1 shows in.

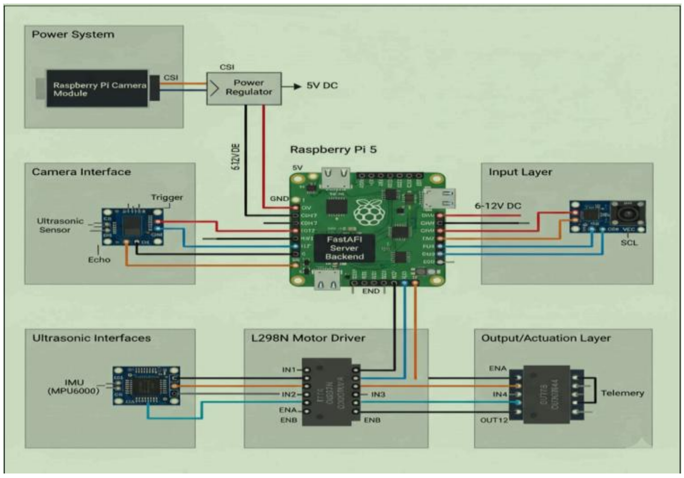

Figure 1 **System Architecture of AI Based Autonomous Rover**

2.2.Project focuses

Advancements in artificial intelligence, embedded computing, and low-cost sensor technologies have significantly transformed modern autonomous ground vehicle (AGV) systems. Raspberry Pi–based robotic platforms have become increasingly popular due to their affordability, portability, and ability to support real-time machine learning models. In outdoor and military environments, autonomous robots can play a crucial role in reconnaissance, surveillance, and the safe exploration of unknown or hazardous terrains. Terrain classification and object detection are essential capabilities for enabling autonomous navigation in mixed outdoor environments. Machine learning models such as YOLOv8 offer high accuracy in detecting obstacles and threats, and when optimized using frameworks like NCNN, they can achieve real-time inference on edge devices like the Raspberry Pi 5. Combined with sensor modules such as IMU, ultrasonic sensors, and vision systems, an intelligent rover can navigate autonomously while adapting to environmental challenges. This project focuses on developing an AI-enhanced autonomous rover capable of operating across general outdoor mixed terrain, integrating computer vision, sensor fusion, and motor control to provide reliable mobility and situational awareness.

2.3.Problem Statement

Military and security personnel often face the challenge of monitoring large, unstructured outdoor environments that may be dangerous or inaccessible. Traditional manual surveillance methods are labor-intensive, risky, and limited in coverage. Existing commercial autonomous robots are often expensive and lack modularity, making them unsuitable for cost-sensitive research or field deployment scenarios. There is a need for a compact, low-cost, AI-powered rover capable of classifying terrain, detecting obstacles, and navigating autonomously while also supporting manual control when mission demands require operator intervention.

2.4.General Objective

To design and develop an AI-powered autonomous rover capable of navigating mixed outdoor terrain using terrain classification, object detection, and sensor-based decision-making. Specific Objectives that follows to implement a custom-trained YOLOv8 model optimized with NCNN for real-time object detection on Raspberry Pi 5. Classify outdoor terrain types and adapt rover movement based on terrain conditions. And also to integrate ultrasonic and IMU sensors for stable, obstacle-aware navigation. To develop a FastAPI-based dashboard enabling manual control with live video streaming. Then create a modular hardware and software architecture suitable for research and military-inspired outdoor applications. This project focuses on the development of an autonomous rover designed for outdoor mixed-terrain navigation. The scope includes the Hardware integration of Raspberry Pi 5, motor driver, ultrasonic sensors, IMU, and camera module. Implementation of machine learning pipelines for object detection and terrain classification. Development of autonomous movement algorithms including obstacle avoidance and terrain-aware navigation. Deployment of a remote-control dashboard using FastAPI for manual operation. Field testing in general outdoor environments such as grass, soil, gravel, and light uneven surfaces.

2.5. Significance of the Project

The proposed system demonstrates the potential of low-cost embedded AI to enhance military-oriented reconnaissance and surveillance tasks. This rover reduces risks to soldiers by autonomously exploring uncertain or hazardous terrain and provides a research platform for autonomous navigation technologies, Helps validate the effectiveness of lightweight neural network deployment on edge hardware. Supports real-time decision-making through terrain and object awareness, It is modular and cost-efficient nature makes it valuable for academic

3. System Overview

The proposed autonomous rover system integrates computer vision, sensor fusion, motor control, and web-based teleoperation to achieve robust outdoor navigation. The methodology involves a combination of hardware and software subsystems operating synergistically on the Raspberry Pi 5 platform. The rover’s perception system uses a Raspberry Pi Camera Module for capturing real-time video, which is processed through a custom-trained YOLOv8 model optimized using NCNN. This enables efficient object detection directly on the edge device. Alongside vision-based detection, the system employs ultrasonic sensors and an IMU (MPU6050) for additional environmental awareness. Sensor fusion ensures improved accuracy in obstacle detection, orientation tracking, and movement stabilization. Motor control is implemented using an L298N motor driver, which receives speed and direction commands from the Raspberry Pi. The rover supports two operational modes:

- Autonomous Mode – where navigation decisions are made using AI-based detections and sensor feedback.

- Manual Control Mode – where the operator can control the rover remotely via a FastAPI dashboard with real-time video streaming.
- Power is supplied through a high-discharge Li-ion battery system with regulated output to ensure safe and stable operation. All components are integrated into a modular architecture enabling scalability and future enhancements.

3.1. Hardware Components

The hardware design includes sensing, processing, actuation, and power systems required for autonomous navigation in outdoor terrain. Each component is selected for its performance, power efficiency, and compatibility with the Raspberry Pi ecosystem.

3.1.1. Raspberry Pi 5

The Raspberry Pi 5 serves as the central processing unit of the rover. It features a quad-core ARM Cortex-A76 processor, improved GPU performance, and faster memory access, making it suitable for real-time machine learning inference. With support for high-speed interfaces, the Pi 5 can efficiently handle camera input, motor control commands, and sensor data processing.

Its key roles in the system include:

- Running the YOLOv8 NCNN inference engine.
- Executing navigation algorithms and decision-making processes.
- Communicating with peripheral sensors via I2C and GPIO.
- Hosting the FastAPI-based teleoperation dashboard.

The Raspberry Pi 5 offers a balanced trade-off between computational power and energy efficiency, making it ideal for lightweight autonomous vehicles.

3.1.2. Camera Module V2

The Official Raspberry Pi Camera Module V2 provides 8-megapixel still images and high-definition video streams. It uses a Sony IMX219 sensor known for its good low-light performance and compatibility with machine learning applications.

- In this project, the camera is primarily used for:
- Real-time video capture for YOLOv8 object detection.
- Terrain image acquisition for classification tasks.
- Streaming live video to the FastAPI dashboard for manual teleoperation.

The camera's CSI interface enables low-latency data transfer, which is essential for responsive AI-based control figure 2 shows.

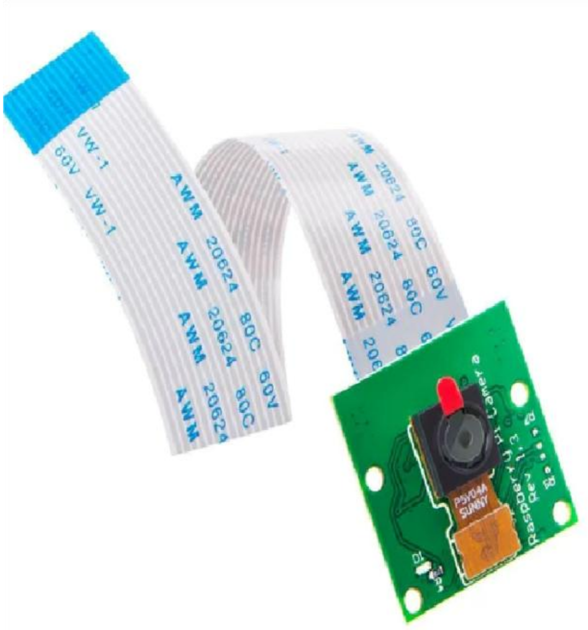

Figure2 Camera's CSI interface

3.1.3. L298N Motor Driver

The L298N motor driver module controls the rover's DC motors by regulating voltage and current. It supports up to 2A per channel, making it suitable for driving medium-load robotic chassis.

Functions of the L298N in the rover include:

- Controlling motor direction (forward, reverse, turning).
- Adjusting rotor speed through pulse-width modulation (PWM).
- Isolating the Raspberry Pi from high-current motor circuits.

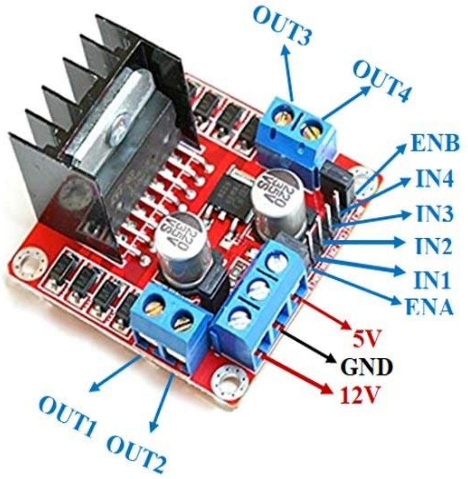

Figure3 Its robust design and ease of integration allow consistent motor performance across varying outdoor terrain conditions.

3.1.4. Ultrasonic Sensor

The ultrasonic sensor provides short-range distance measurements based on time-of-flight of sound waves. It helps the rover detect obstacles that may not be captured by the camera due to occlusion or environmental factors.

Key uses include:

- Measuring distances within 2–400 cm range.
- Supporting obstacle avoidance algorithms.
- Acting as a fail-safe system in case of AI detection limitations.
- Its low cost and reliability make it a valuable component in hybrid perception systems.

3.1.5. IMU (MPU6050)

The MPU6050 combines a 3-axis gyroscope and a 3-axis accelerometer, enabling the rover to track its orientation and detect motion dynamics.

Roles of the IMU include:

- Estimating rover tilt, rotation, and angular velocity.
- Providing drift-corrected orientation data through filtering.
- Enhancing navigation stability on uneven or sloped terrain.

IMU data is fused with other sensor inputs to support smoother motor control and more accurate environmental perception.

3.1.6. Battery and Power Regulation

The rover is powered by high-discharge Pro-Range 21700 P42A Li-ion batteries, which deliver sufficient current for motors and the Raspberry Pi system. A power regulation module ensures stable voltage output to prevent damage to sensitive electronics. Shows in figure 4.

Figure 4 Power system

Power system responsibilities include:

- Supplying regulated 5V to the Raspberry Pi and sensors.
- Providing 6–12V to the motor driver depending on motor requirements.
- Protecting the system against voltage drops during high-load operation.

The use of high-capacity Li-ion cells ensures long operational runtime, which is crucial for outdoor and field applications.

3.1.7. DC - DC Buck Boost Convertor

A DC–DC buck-boost converter is a power electronic circuit that converts a variable DC input voltage into a stable DC output voltage that can be higher (boost) or lower (buck) than the input. It is widely used in battery-powered systems where the supply voltage fluctuates during discharge. A buck-boost converter operates by rapidly switching an electronic switch (usually a MOSFET) on and off, storing energy temporarily in an inductor, and then releasing it at a controlled voltage level. The switching process is governed by a control circuit using Pulse Width Modulation (PWM).

When the input voltage is higher than the desired output, the converter operates in buck mode (step-down).

- When the input voltage is lower than the desired output, the converter operates in boost mode (step-up).
- When the input voltage equals the output, it transitions smoothly between modes.
- This dual capability ensures a constant output voltage even when the input voltage varies.

Figure5 LTC 3780 Buck Boost Convertor

3.2.Software Components

The software architecture of the proposed autonomous rover integrates machine learning models, sensor processing scripts, motor control logic, and a teleoperation dashboard. Python is used as the primary development language due to its compatibility with deep learning frameworks, hardware libraries, and web-based APIs.

3.2.1. Python Environment

The Python environment forms the backbone of the rover's software system. It includes essential libraries for machine learning inference, sensor interfacing, and web communication. Key components include:

- OpenCV for image acquisition and preprocessing
- NumPy for matrix and numerical computations
- RPi.GPIO / gpiozero for motor and sensor control
- smbus for I2C communication with the IMU
- FastAPI for backend server and teleoperation interface
- uvicorn for asynchronous API hosting
- ncnn Python bindings for running the YOLOv8 model efficiently on the Raspberry Pi

The environment is configured to optimize performance on the Raspberry Pi 5, ensuring low latency for both AI inference and control loops. YOLOv8 Custom Training A custom YOLOv8 model is trained to detect specific outdoor objects relevant to rover navigation, such as obstacles, humans, vehicles, and terrain features. The training process includes

- Dataset Preparation: Images are collected from outdoor environments and labeled using annotation tools like LabelImg or Roboflow.
- Model Selection: A lightweight YOLOv8 variant (e.g., YOLOv8n or YOLOv8s) is chosen for faster inference on edge devices.
- Training Parameters: The model is trained with tuned hyperparameters, such as batch size, learning rate, augmentation strategies, and epochs.
- Evaluation: Performance is assessed using metrics such as mAP (Mean Average Precision), precision–recall curves, and inference time.

Once trained, the model is exported for deployment in an optimized format.

3.2.2. Model Conversion to NCNN

To achieve real-time performance on Raspberry Pi 5, the YOLOv8 model is converted into the NCNN framework, which is designed for high-speed CPU inference.

The conversion steps include:

- Exporting YOLOv8 from PyTorch to ONNX format.

- Using NCNN tools (onnx2ncnn) to generate .param and .bin files.
- Testing the converted model on Pi using NCNN inference scripts.

This conversion significantly improves frame rates compared to running PyTorch or ONNX Runtime on Raspberry Pi.

3.2.3. Terrain Classification Pipeline

The terrain classification pipeline enables the rover to recognize surface types such as grass, gravel, sand, or mud. The pipeline consists of:

- Image Capture: Frame extraction from the Pi camera at regular intervals.
- Preprocessing: Resizing, normalization, and filtering to prepare input for the classification model.
- Model Inference: Lightweight CNN or YOLO-based terrain classifier processes each frame.
- Decision Layer: The rover adjusts speed and navigation strategy based on terrain results.

This pipeline ensures adaptive behavior across mixed terrain environments.

3.2.4. Object Detection Pipeline

The object detection pipeline utilizes the optimized YOLOv8-NCNN model to identify obstacles and relevant outdoor objects.

Steps:

- Capturing live video from the camera.
- Passing frames through NCNN-based YOLOv8 inference.
- Extracting bounding boxes, class IDs, and confidence scores.
- Filtering detections based on relevance and threshold values.
- Sending computed results to the navigation controller.

Detection results directly influence steering, stopping, or rerouting actions.

3.2.5. Real-Time Control using FastAPI Dashboard

The rover implements a FastAPI-based dashboard for remote monitoring and manual control. Key features include:

- Live Video Feed: MJPEG streaming from the Raspberry Pi camera.
- Manual Control Buttons: Forward, reverse, left, right, and stop.
- Telemetry Display: Sensor values (IMU angle, ultrasonic distance) and system status.
- Mode Switching: Ability to toggle between autonomous and manual operation.
- Asynchronous Processing: Ensures minimal latency and smooth control.

FastAPI's high-speed asynchronous engine provides a reliable interface for real-time teleoperation.

3.2.6. Navigation Algorithm

The navigation algorithm integrates terrain classification, object detection, and sensor feedback to make autonomous movement decisions. The algorithm runs continuously in a loop, evaluating the rover's environment frame-by-frame.

3.2.7. Terrain-Based Speed Adjustment

The rover adjusts its speed depending on the detected terrain type:

- Grass or soil: Normal speed
- Gravel or rocky terrain: Reduced speed for stability
- Wet or slippery surfaces: Minimum speed to avoid skidding
- Smooth terrain: Higher speed allowed

This adaptive speed control enhances stability, reduces motor strain, and improves energy efficiency.

3.2.8. Obstacle Avoidance Strategy

Obstacle avoidance combines object detection and ultrasonic sensor readings:

- Camera detects obstacles using YOLOv8.
- Ultrasonic sensor measures distance to confirm proximity.

Based on detection:

- If obstacle is close → Stop and avoid.
- If obstacle is partially visible → Steer away.
- If obstacle blocks path completely → Re-route using alternative direction.

Priority is given to sensor data when AI detection confidence is low.

3.2.9. Sensor Fusion for Stability

Sensor fusion integrates data from the IMU and ultrasonic sensor with AI outputs:

- IMU provides orientation, tilt, and angular velocity, allowing the rover to correct deviations or imbalance.
- Ultrasonic provides real-time range measurements for near-field obstacle detection.
- AI vision provides semantic understanding (object type, bounding boxes).

A fusion algorithm (e.g., complementary filtering) combines these inputs to improve navigation accuracy and reduce errors caused by noise or environmental conditions.

3.2.10. Autonomous vs Manual Mode Switching

The rover supports two operational modes:

- Autonomous Mode
- AI-based detection and classification control movement.
- Sensors guide obstacle avoidance and stability.
- Rover operates independently without human intervention.
- Manual Mode
- Operator controls movement via FastAPI dashboard.
- Live video feed assists remote navigation.
- Autonomous algorithm is paused to avoid conflict.
- Mode Switching Logic
- A command from the dashboard toggles system state in real time.
- Safety checks ensure motors stop before switching modes.
- Once in autonomous mode, the AI pipeline takes full control.

This hybrid configuration provides flexibility for both experimentation and field deployment.

3.3. Data Flow in the System

The data flow in the autonomous rover system outlines how information moves between sensors, processing modules, decision logic, and actuators. The system follows a cyclic process where real-time data is continuously collected, processed, and acted upon. The major stages include:

3.3.1. Input Acquisition

- The Camera Module V2 captures video frames.
- The Ultrasonic sensor gathers distance measurements.
- The IMU (MPU6050) outputs orientation and motion data.
- The FastAPI dashboard receives manual commands during teleoperation.

3.3.2. Preprocessing Layer

- Video frames are resized, normalized, and passed to the YOLOv8–NCNN inference engine.
- Sensor readings are filtered to reduce noise (e.g., smoothing ultrasonic distance, complementary filtering for IMU).

3.3.3. Inference & Analysis

- YOLOv8 NCNN detects objects and classifies visual features.
- Terrain classifier identifies terrain type.
- Sensor data is fused to enhance situational awareness.
- Decision-Making Module
- Navigation logic evaluates:
- Object detection results
- Terrain classification
- IMU and ultrasonic readings

Decisions include adjusting speed, avoiding obstacles, correcting orientation, or proceeding forward. In manual mode, decisions originate from the dashboard instead of autonomous logic.

3.3.4. Motor Control Output

- Control commands (PWM signals, direction instructions) are sent to the L298N motor driver.
- Motors execute the required movement.
- Feedback Loop
- New sensor values are collected and the cycle repeats.

- System continuously updates the dashboard in manual mode.

This cyclic data flow ensures real-time perception, decision-making, and actuation, supporting robust autonomous and manual operation.

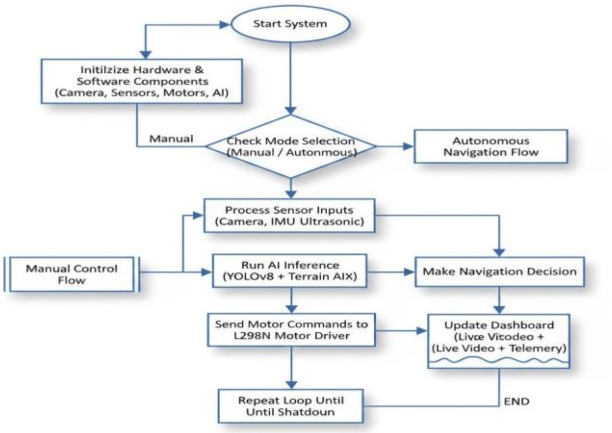

Figure 6 **cyclic data flow**

3.3.5. Block Diagram of the Proposed System

The block diagram represents the overall interaction between sensing, processing, decision-making, and actuation components of the autonomous rover. A block diagram is a high-level graphical representation of a system, illustrating major components and the flow of information between them. Description of the System Block Diagram

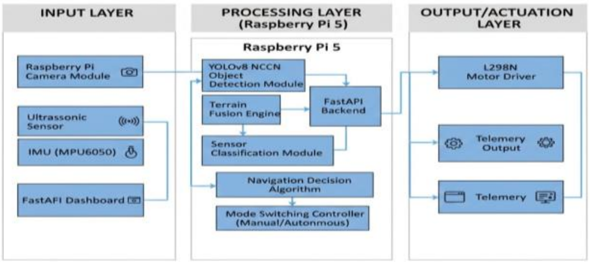

Figure7: System Block Diagram This layered design ensures a modular, scalable system suitable for both research and field applications.

3.3.6. Mechanical Design and Chassis Layout

The mechanical design focuses on creating a stable, robust chassis capable of operating across varied outdoor terrain. Mechanical design refers to the structural and physical arrangement of the rover's body, wheels, motors, and component housings. Key Features of the Chassis Layout Four-Wheel Drive (4WD) or Differential Drive Provides stable motion and maneuverability on uneven surfaces. Mounting Positions

- Camera mounted at the front for optimal field of view.
- Ultrasonic sensor placed forward-facing for obstacle detection.
- IMU centrally located to reduce vibration distortion.
- Battery compartment positioned at the bottom for low center of gravity.
- Raspberry Pi mounted on an internal platform for protection and airflow.
- Material Selection
- Aluminum or high-density plastic for durability.
- Shock absorbers or flexible joints (optional) for rough terrain.
- The mechanical layout ensures balanced weight distribution, structural integrity, and accessibility for maintenance.

3.3.7. Electrical Circuit Diagram

The electrical system interconnects sensors, actuators, and the main controller using regulated power and communication lines. An electrical circuit diagram visually represents the wiring connections, power flow, and electronic interactions in the system. Description of Circuit Connections

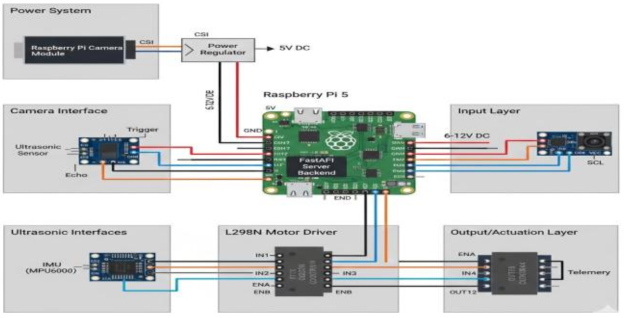

Figure 8: **This architecture ensures proper isolation between power-hungry components (motors) and sensitive electronics (Raspberry Pi and sensors).**

4. Software Architecture

The software architecture outlines the layered structure of modules that handle perception, control, data processing, and networking. Software architecture is the overall design of the software system, specifying how components interact to fulfill system functionality.

Layers of the Software Architecture:

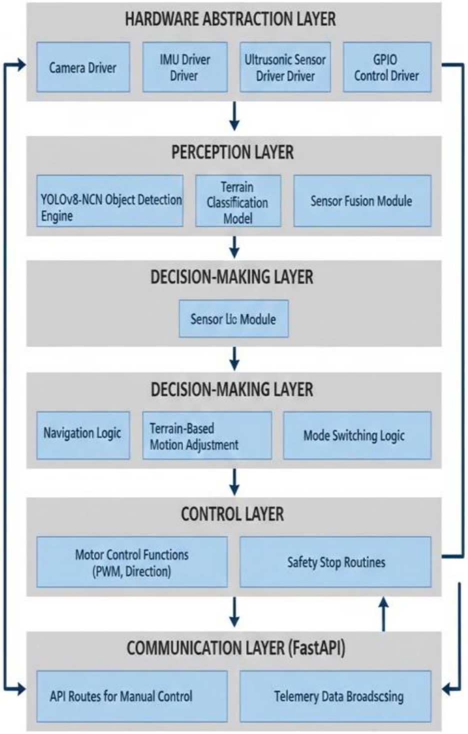

Figure 9 **Flow Chart**

Interface Layer: Web dashboard for real-time monitoring and control. This modular architecture allows the rover to operate autonomously while still supporting remote user control.

4.1. Communication Protocols

Communication protocols define how data is exchanged between the rover's components. A communication protocol is a set of rules governing data transmission between electronic devices.

Protocols Used in the System:

- I2C (Inter-Integrated Circuit):
- Used for communication with IMU (MPU6050).
- Supports multiple slave devices on two wires (SDA, SCL).
- GPIO (General Purpose Input/Output):
- Used for controlling motors and ultrasonic sensors.
- CSI (Camera Serial Interface):
- Provides high-bandwidth communication for the camera.
- HTTP/REST (FastAPI):
- Sends movement commands and retrieves telemetry data.
- Supports low-latency asynchronous operations.
- MJPEG Streaming / WebSockets:
- Used to transmit live video frames to the dashboard.

This variety of protocols ensures efficient communication across different subsystems.

4.2.Power Management System

Power management ensures stable and reliable operation of all electronic components. A power management system regulates, distributes, and protects electrical power throughout the rover.

Components of the Power System

21700 P42A Li-ion Battery**:**

- Provides high discharge current suitable for motors and processing units.
- Voltage Regulators
- 5V regulator for Raspberry Pi and sensors.
- 6–12V supply for motor driver and motors.
- Power Distribution Board
- Ensures clean, isolated power paths to avoid electrical noise interference.
- Protection Circuits
- Overcharge/over-discharge protection
- Short-circuit protection
- Thermal cutoff mechanisms

This ensures safe operation and extends the lifespan of the battery and electronic components. Safety and Reliability Considerations Safety and reliability are critical, especially for outdoor and military-oriented environments. Safety considerations focus on preventing harm or damage, while reliability focuses on consistent system performance over time.

4.3.Implemented Safety & Reliability Features

- Emergency Stop Logic
- Rover halts immediately if ultrasonic distance is below threshold.
- Redundant Sensor Use
- Both vision and ultrasonic data help reduce false positives/negatives.
- Thermal Monitoring
- Raspberry Pi throttling protection prevents overheating.
- Fail-Safe Manual Override
- Operator can switch from autonomous to manual mode at any time.
- Secure Communication
- Dashboard access can be restricted using authentication measures.
- Shock-Resistant Mechanical Design
- Minimizes hardware damage and vibration-induced sensor errors. These considerations ensure stable and safe operation in harsh outdoor terrain

5. Results And Discussion

5.1.Results

The results of this study are based on experimental testing of the AI-powered autonomous rover in real outdoor environments to evaluate its performance in terms of obstacle detection, navigation, and terrain adaptability. The experiments were designed to assess how effectively the rover can operate under different surface conditions such as grass, soil, and gravel, while maintaining stability and avoiding obstacles. During testing, the rover successfully detected obstacles in real time using the implemented object detection model. The system demonstrated a consistent ability to identify objects and respond quickly by altering its path to avoid collisions. The integration of the ultrasonic sensor further improved obstacle detection accuracy by providing reliable distance measurements, especially in close-range scenarios The terrain classification and adaptive movement capabilities were also evaluated. The rover was able to adjust its motion based on different surfaces, maintaining stability with the help of IMU data. This ensured smoother navigation even on uneven terrain. Additionally, the sensor fusion approach improved the overall reliability of the system compared to using individual sensors independently. The response time of the system was observed to be low, enabling near real-time decision-making, which is critical for autonomous navigation. The rover also demonstrated flexibility by operating efficiently in both autonomous and manual modes through the web-based interface. Overall, the experimental results confirm that the proposed system performs effectively in real-world conditions, achieving reliable obstacle detection, stable navigation, and adaptability to varying terrains. These outcomes validate the design and demonstrate the feasibility of using low-cost embedded AI systems for autonomous rover applications.

5.2.Discussion

The experimental results demonstrate that the proposed AI-powered rover is capable of performing reliable autonomous navigation in real-world outdoor environments. Rather than just confirming functionality, these results highlight how effectively the integration of multiple technologies—such as computer vision, sensor fusion, and embedded processing—contributes to overall system performance. The use of a lightweight object detection model enables real-time decision-making, which is critical for avoiding obstacles in dynamic conditions. The inclusion of ultrasonic sensors alongside vision-based detection improves robustness, especially in scenarios where visual data alone may be insufficient, such as low lighting or partial obstructions. Similarly, the IMU plays a key role in maintaining stability, indicating that sensor fusion significantly enhances the rover’s adaptability to uneven terrains. This combination of sensors reduces dependency on a single input source, thereby increasing reliability. The rover’s ability to operate in both autonomous and manual modes adds practical value, as it allows human intervention in uncertain or complex situations. This flexibility makes the system more suitable for real-world applications compared to fully autonomous systems that may fail in unpredictable environments. However, certain limitations can be observed. The performance of the vision system may vary under extreme lighting conditions, and the processing capability of embedded hardware can restrict the complexity of AI models used. Despite these challenges, the overall system demonstrates a balanced trade-off between cost, performance, and functionality. This suggests that the proposed approach is a viable and scalable solution for applications such as surveillance, exploration, and field research.

Conclusion

This study successfully addressed the challenge of developing a low-cost and efficient autonomous rover capable of operating in real-world outdoor environments. Based on the results and discussion, the proposed system demonstrates that integrating artificial intelligence, sensor fusion, and embedded hardware can effectively achieve reliable obstacle detection, stable navigation, and terrain adaptability. The experimental outcomes confirm that the rover is capable of making real-time decisions, avoiding obstacles, and maintaining performance across different surfaces such as grass, soil, and gravel. The combination of vision-based object detection with ultrasonic and IMU sensors enhances the overall robustness and accuracy of the system. Additionally, the inclusion of both autonomous and manual modes ensures flexibility and practical usability in various scenarios. Although certain limitations exist, such as sensitivity to lighting conditions and hardware constraints, the system provides a balanced solution in terms of cost, performance, and functionality. Overall, this work confirms that an intelligent and scalable autonomous rover can be developed using affordable components, making it suitable for applications in surveillance, exploration, and research while minimizing risks to human involvement.

Acknowledgements

The authors would like to express their sincere gratitude to their institution and faculty members for their continuous support, guidance, and encouragement throughout the development of this project. Special thanks are extended to the project supervisor for valuable insights and technical assistance provided at every stage of the work. The authors also acknowledge the support of the laboratory facilities and resources that made the successful completion of this project possible. Furthermore, appreciation is given to peers and team members for their cooperation and contributions during the design, implementation, and testing phases. This project was carried out without any external financial funding, and all expenses were managed by the authors.

References

[1]. Birari, H., Sharma, K., & Verma, R. (2022). Autonomous navigation systems for mobile robots using artificial intelligence. International Journal of Robotics Research, 41(5), 567–580.

[2]. Chen, L., Wang, T., & Li, X. (2021). Real-time object detection using YOLO-based deep learning models for embedded systems. IEEE Access, 9, 123456–123468.

[3]. Singh, R., Patel, D., & Kumar, S. (2022). Terrain classification for autonomous ground vehicles using machine learning techniques. Journal of Field Robotics, 39(3), 245–260.

[4]. Rajan, P., Nair, V., & Joseph, A. (2023). Low-cost embedded AI systems for real-time robotic applications. International Journal of Advanced Computer Science and Applications, 14(2), 112–120.

[5]. Zhang, Y., Liu, H., & Chen, Z. (2023). Optimization of deep learning models for edge computing in autonomous systems. IEEE Transactions on Industrial Informatics, 19(4), 3345–3355.

[6]. Kumar, S., Mehta, R., & Singh, A. (2024). Integration of sensor fusion techniques in autonomous robotic navigation. Sensors, 24(1), 89–102.

[7]. Patel, M., Shah, P., & Desai, N. (2024). Adaptive terrain-aware navigation for mobile robots in outdoor environments. Robotics and Autonomous Systems, 172, 104567.