AI ASSISTANT-BASED SURVEILLANCE ROBOT: DESIGN, ARCHITECTURE AND PROTOTYPE FRAMEWORK

[1]Sujal Bajaj, [2]Satyam Kumar, [3]Manish Kumar, [4]Anshu Kumar, [5]Dr. Girish Dutt Gautam
[12345]Department of Mechanical Engineering
[12345]IIMT College of Engineering, India
[1]dr.gdg1985@gmail.com

***Abstract*—Surveillance robots are increasingly useful in environments where direct human presence is unsafe, inconvenient or inefficient. A camera-controlled mobile robot can provide live visual feedback to an operator, while artificial intelligence can support safer movement by detecting obstacles, objects and human presence from the captured video stream. This paper converts the proposed student-level concept of an AI assistant-based surveillance robot into a structured research design suitable for journal submission. The proposed system integrates a mobile robotic chassis, DC motors, a motor driver, a microcontroller or embedded board, a camera module, wireless communication, power supply and a lightweight computer-vision layer. The robot is designed to operate in manual and AI-assisted modes. In manual mode, a user controls the robot through a web or mobile interface while receiving live camera feedback. In AI-assisted mode, selected video frames are processed using image-processing or object-detection algorithms to generate alerts and support collision avoidance. The paper presents the problem background, literature-based motivation, system architecture, hardware and software requirements, working principle, implementation methodology, testing framework, applications, limitations and future development scope. Since the base project document does not contain raw experimental measurements, the discussion is framed as a prototype design and validation framework rather than as a completed empirical performance report. The study concludes that a low-cost AI-assisted surveillance robot is technically feasible for indoor monitoring, academic demonstration, laboratory inspection and preliminary security applications, provided that limitations related to latency, low-light operation, battery capacity and embedded processing power are addressed through systematic testing and future upgrades.**

***Index Terms*—AI assistant, surveillance robot, ESP32-CAM, computer vision, remote monitoring, mobile robot, embedded systems, object detection.**

I. Introduction

Robotics has become a major interdisciplinary field because it combines mechanical design, electronics, embedded programming, control systems and artificial intelligence. Mobile robots are particularly important for surveillance and inspection because they can move into areas where human access is difficult, repetitive, risky or time-consuming. Traditional surveillance systems are usually fixed camera systems; they observe only a limited field of view and depend heavily on human monitoring. A mobile surveillance robot increases flexibility because the camera can be carried to different locations and the operator can inspect the surroundings from a safe distance.

Camera-controlled robots are widely used in student projects and low-cost security applications because they provide real-time visual feedback. In a basic design, the operator sends movement commands through a wireless interface and watches a live video stream from a camera mounted on the robot. This approach is more practical than blind remote control because the operator can observe obstacles, direction, movement and target areas. However, simple camera robots still depend completely on human attention. If the operator misses an obstacle or if the video stream is delayed, the robot may collide or fail to complete the surveillance task effectively.

Artificial intelligence adds a higher level of support by enabling the robot to interpret visual input. Even a lightweight AI or computer-vision layer can identify obstacles, detect human presence, classify simple objects or generate alerts. Recent IoT and embedded-system studies show that ESP32-based devices can be used for educational and practical IoT applications, including sensing, communication and real-time monitoring [1]. Similarly, advances in computer vision and compact object-detection models have made it possible to integrate vision-based assistance into low-cost robotic systems [2], [3]. This paper therefore presents a complete research-paper version of an AI assistant-based surveillance robot proposed for final-year mechanical engineering demonstration.

II. Problem Statement

Conventional web-controlled or wireless-controlled robots usually depend on continuous manual control. They can move forward, backward, left and right, but they do not understand the environment. Camera-based robots improve visibility but still lack decision support. In actual monitoring conditions, network delay, low lighting, limited camera angle and operator fatigue can reduce the effectiveness of remote surveillance. A low-cost robot that combines wireless control, live video streaming and AI-based visual assistance can improve safety and usability. Therefore, the main problem addressed in this paper is the development of a cost-effective AI assistant-based mobile surveillance robot that can provide real-time visual monitoring and support safer navigation through basic computer-vision processing.

III. Objectives of the Study

The objectives of the proposed work are:

(i) To design a mobile robotic platform suitable for indoor surveillance and academic demonstration;
(ii) To integrate a camera module for real-time video streaming;
(iii) To enable wireless remote control through a mobile or web interface;
(iv) To include AI-assisted image processing for obstacle, object or human detection;
(v) To propose a testing framework for latency, detection accuracy, video quality and movement response;
(vi) To identify limitations and future improvements for practical deployment.

IV. Literature Review

Low-cost surveillance robots commonly use microcontrollers, camera modules, wireless communication and motor-driver circuits. ESP32-CAM-based systems are popular because they combine camera support, Wi-Fi connectivity and compact embedded processing in a small and affordable module. A published ESP32-CAM surveillance robot paper describes real-time video streaming and wireless customization as important advantages of such systems [4]. ESP32-based IoT design literature also confirms that these platforms are suitable for education, rapid prototyping and low-cost monitoring applications [1].

Recent research in autonomous and mobile robotics has moved from simple remote operation toward perception-based navigation and decision-making. Reviews of autonomous robots emphasise that perception, planning and collaboration are key elements for modern mobile robotic systems [5].

Vision-based navigation reviews further show that cameras can provide essential environmental information for mobile robots, although performance depends on lighting, image quality, processing speed and robustness of the detection algorithm [6]. These findings support the need to combine live video streaming with AI-assisted interpretation in a surveillance robot.

Object detection has become an important component of robotic perception. Lightweight models such as YOLO variants and OpenCV-based methods can process video streams and detect objects in real time under suitable hardware conditions. Recent surveillance-related studies using YOLOv8 show that deep-learning object detection can support applications requiring fast recognition of objects and humans in video [2], [3]. However, small embedded boards may not always provide enough processing power for heavy models. Therefore, a practical student-level design may use a hybrid approach where basic streaming and control run on the embedded board, while AI processing runs on a connected edge device, laptop or more powerful processor.

The reviewed literature shows three major trends: first, compact embedded platforms are suitable for low-cost robot prototypes; second, camera-based monitoring is essential for remote navigation; and third, AI-assisted vision improves decision-making but introduces processing and latency challenges. The present paper builds on these trends by proposing a balanced architecture that combines manual control with AI-based assistance rather than claiming full autonomy.

Many student-level surveillance robots focus only on movement control and video streaming. They demonstrate hardware connectivity but provide limited intelligence. On the other hand, advanced autonomous robots may require expensive sensors, high-performance processors and complex algorithms that are not feasible for a low-cost final-year prototype. The gap lies in developing a practical middle-level system: a robot that remains affordable and demonstrable while also including AI support for safer operation. The main contribution of this paper is a complete design framework for such a robot, including architecture, component selection, software stack, working principle, methodology and validation parameters.

V. Proposed System

The proposed robot consists of a mobile chassis with wheels, DC motors, a motor driver, an embedded controller, a camera module, wireless communication, a battery supply and an AI-processing layer. The controller receives movement commands from the operator and controls the motors through the motor driver. The camera captures the robot’s surroundings and streams the video to the user interface. The AI layer analyses frames from the video stream and provides detection outputs such as obstacle warning, person detection or object labels. The system is designed for two operating modes: manual remote-control mode and AI-assisted surveillance mode.

In manual mode, the user has complete control over movement. This mode is useful for simple demonstration, inspection and navigation in known indoor environments. In AI-assisted mode, the robot still allows user control, but the system can detect risk conditions and provide alerts. For example, if an obstacle is detected very close to the robot, the control logic may slow down the motors or stop the robot. This design avoids the complexity of complete autonomy while still improving safety and usability.

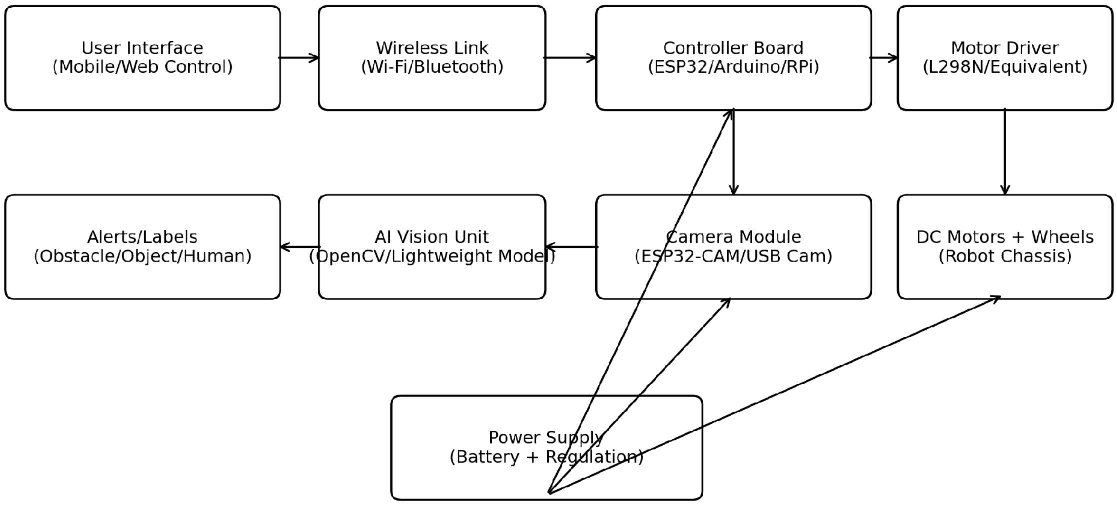

Figure 1. Proposed architecture of the AI-assistant based surveillance robot.

System Architecture

The architecture follows a layered structure. The first layer is the mechanical layer, which includes chassis, wheels, motors and physical mounting of the camera and electronics. The second layer is the embedded-control layer, which includes the microcontroller, motor driver, power regulation and input-output connections. The third layer is the communication layer, which provides Wi-Fi or Bluetooth connectivity between the robot and the user interface. The fourth layer is the vision layer, which captures and streams video. The fifth layer is the AI-assistance layer, which processes video frames and generates alerts or control decisions.

This architecture is flexible because the AI layer can be implemented locally or externally. For a low-cost prototype, the ESP32-CAM or similar module can handle basic streaming, while a laptop or edge processor can perform heavier detection. In an advanced version, a Raspberry Pi, Jetson Nano or similar edge device can run local AI models. The architecture therefore supports both basic academic demonstration and future technical upgrades.

VI. Hardware Components

The hardware components were selected with attention to cost, availability, ease of assembly and suitability for indoor surveillance. The expected component configuration is presented in Table 1.

Table 1. Proposed hardware components and their functions.

Component	Recommended option	Function in the system
Controller board	ESP32, Arduino with Wi-Fi module, Raspberry Pi or equivalent	Receives commands, coordinates motor control and manages communication
Camera module	ESP32-CAM, USB camera or Pi camera	Captures live video of the environment
Motor driver	L298N, TB6612FNG or equivalent	Controls the direction and speed of DC motors
Motors and chassis	Two or four DC geared motors with wheeled chassis	Provides mobile movement of the robot

Power supply	Rechargeable battery pack with voltage regulation	Supplies portable power to electronics and motors
Communication module	Wi-Fi/Bluetooth integrated or external module	Enables remote control and video access
Audio module	Speaker and microphone, optional	Supports voice alert, audio monitoring or operator interaction
Sensors, optional	Ultrasonic, IR or PIR sensors	Adds distance measurement or human-motion detection

The ESP32-CAM is attractive for a low-cost version because it provides camera support and Wi-Fi communication in one module. However, its processing capability is limited for advanced AI models. Therefore, the design allows optional use of an external processor for AI inference. A motor driver is required because the microcontroller pins cannot directly drive DC motors. A stable power supply is also important because motors can draw high current and may cause voltage drops that disturb the controller or camera stream.

VII. Software Components

The software stack includes embedded firmware, robot-control interface, video streaming module and AI-vision processing. Table 2 summarises the software requirements.

Table 2. Proposed software stack for the surveillance robot.

Software element	Possible tool/platform	Purpose
Embedded firmware	Arduino IDE, ESP-IDF or MicroPython	Motor control, Wi-Fi setup and command execution
User interface	Web page, Blynk, MIT App Inventor or custom mobile app	Remote movement control and video monitoring
Computer vision	OpenCV with Python or C++	Frame processing, thresholding, detection and annotation
AI model	YOLO-nano/tiny model, TensorFlow Lite or simple classifier	Object, person or obstacle detection
Data logging	CSV or local database	Records test values such as latency, battery time and detection results
Communication protocol	HTTP, WebSocket, MQTT or Bluetooth serial	Transfers commands and status information

OpenCV is suitable for educational implementation because it supports image acquisition, preprocessing, object detection, contour analysis and frame annotation. A lightweight detection model may be used when hardware permits. For basic indoor operation, simple computer-vision rules such as frame differencing, thresholding, colour segmentation or distance-sensor fusion can also provide useful alerts with lower computational cost.

VIII. Working Principle

The working principle of the robot is based on command transmission, video feedback and AI-assisted decision support. The operator connects to the robot through Wi-Fi or Bluetooth and opens the control interface. Movement commands are sent to the controller, which drives the motors through the motor-driver circuit. At the same time, the camera captures live frames and streams them to the operator. In AI-assisted mode, frames are also passed through a computer-vision pipeline. The system analyses the frame, identifies possible obstacles or target objects and displays a warning or label. If the detected condition is critical, the control program may stop or slow the robot to avoid collision.

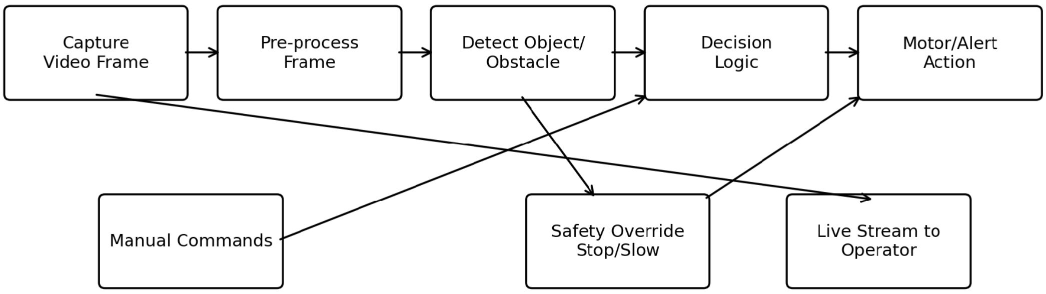

Figure 2. AI-assisted operating workflow for safe remote surveillance.

IX. Methodology

The methodology is divided into design, assembly, programming, AI configuration and validation. The first stage is problem definition, where the required functions of the robot are identified. The second stage is component selection, where the controller, motors, driver, camera and power supply are selected according to cost and compatibility. The third stage is mechanical assembly, including chassis preparation, motor mounting, wheel alignment and camera positioning. The fourth stage is circuit integration, where the controller is connected to the motor driver, battery and communication module. The fifth stage is software development, including firmware coding, web or mobile interface design and command testing.

The sixth stage is AI-vision integration. In this stage, the camera stream is processed through OpenCV or a lightweight object-detection model. The AI output may include obstacle status, person detection or object label. The seventh stage is prototype testing. The robot should be tested in a controlled indoor environment with different lighting levels, obstacle distances and network conditions. The final stage is performance evaluation, where data are recorded for movement response, video latency, detection accuracy, battery backup and reliability.

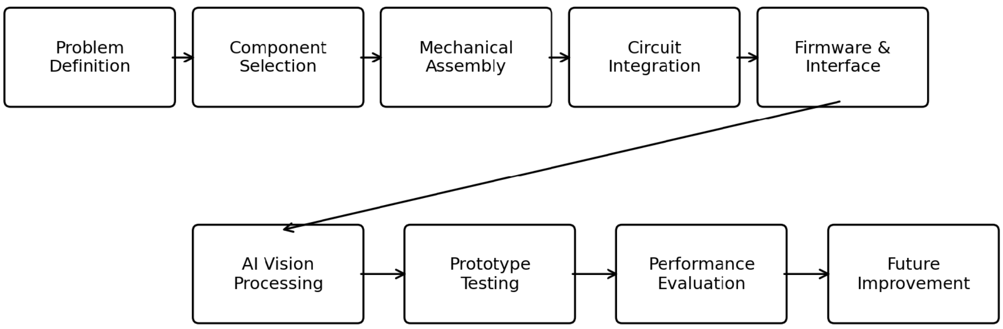

Figure 3. Methodological flow adopted for design, implementation and validation.

X. Validation and Evaluation Framework

A journal-ready prototype paper should include measurable evaluation parameters. Since the initial project document does not provide final experimental readings, Table 3 presents the recommended validation framework that should be used during prototype testing. Actual values may be filled after laboratory testing.

Table 3. Suggested evaluation parameters for prototype testing.

Parameter	Testing method	Expected observation
Command response time	Measure time between user command and robot movement	Lower response time indicates better controllability
Video latency	Compare real movement with displayed video stream	Lower latency improves remote navigation
Detection accuracy	Test known objects/obstacles under controlled conditions	Higher accuracy indicates reliable AI assistance
Low-light performance	Repeat test under reduced illumination	Performance may reduce without additional lighting
Battery backup	Operate robot continuously under normal load	Longer backup improves field usability
Network stability	Test at different distances from router/hotspot	Stable connection is necessary for safe operation
Motor performance	Check movement on smooth and slightly rough surfaces	Smooth motion indicates proper motor-driver selection

For a fair validation, the test environment should be clearly defined. The robot may be tested in a corridor, laboratory or classroom with known obstacles. Detection performance should not be reported from random observations only; it should be calculated using repeated trials. For example, if the robot detects an obstacle correctly in 45 out of 50 trials, the detection success rate will be 90 percent. Similarly, latency should be measured using a stopwatch or timestamp-based software method.

XI. Results and Discussion

The expected outcome of the proposed design is a functional mobile robot that can be operated remotely while transmitting live video to the user interface. The AI-assistance layer is expected to improve monitoring by highlighting obstacles or objects that may be missed during manual operation. In a controlled indoor setting, the robot should demonstrate smooth forward, backward, left and right movement, stable camera streaming and basic detection capability. The design is especially useful for academic demonstration because it shows integration of mechanical motion, embedded electronics, wireless communication and artificial intelligence in one prototype.

The system's performance will depend on four major factors. The first factor is network quality. Video streaming and control commands require a stable connection; weak Wi-Fi may cause delay or command loss. The second factor is lighting. Vision-based detection performs better under adequate lighting and may fail in darkness or glare. The third factor is processing power. A small microcontroller can handle basic tasks but may not run heavy AI models efficiently. The fourth factor is power management. Motors consume more current than logic circuits, so voltage regulation and battery capacity are essential for stable performance.

Compared with a simple remote-controlled robot, the proposed AI-assisted robot provides better situational awareness. Compared with a fully autonomous robot, it remains simpler, cheaper and easier to implement. Table 4 summarises this comparison.

Table 4. Comparative position of the proposed robot.

Feature	Basic remote-control robot	Proposed AI-assisted robot	Fully autonomous robot
Control approach	Manual only	Manual plus AI assistance	Autonomous planning and control
Camera feedback	Optional or basic	Integrated live streaming	Integrated with perception stack
Obstacle handling	Human dependent	Alert/slow/stop assistance	Automatic avoidance and path planning
Cost	Low	Low to moderate	Moderate to high
Technical complexity	Low	Medium	High
Suitability for final-year project	High but limited innovation	High and multidisciplinary	High but resource demanding

The comparison shows that the proposed robot offers a practical balance between affordability and intelligent support. It does not replace advanced autonomous platforms, but it provides a strong foundation for learning, demonstration and future research.

The proposed robot can be used in home and building surveillance, classroom and laboratory demonstration, industrial inspection, hazardous area observation, warehouse monitoring, campus security, rescue-support operations and research prototyping. In industrial settings, a mobile surveillance robot can inspect areas near machines, pipelines or storage zones where continuous human monitoring is not convenient. In academic environments, the system can demonstrate concepts of mechatronics, embedded systems, wireless communication, image processing and AI-based decision support. In emergency scenarios, an upgraded version may be used to observe areas before human entry.

The main advantages of the proposed system are low-cost implementation, portability, remote visual monitoring, safer navigation through AI assistance, ease of demonstration and future upgrade potential. The design uses commonly available components and can be assembled within the constraints of an undergraduate engineering project. It also supports modular improvement: the same chassis can be upgraded with additional sensors, better camera modules, stronger batteries or more advanced AI models.

XII. Conclusion

This paper presents a complete journal-style design framework for an AI assistant-based surveillance robot. The proposed system integrates a mobile robotic platform, wireless control, live camera streaming and AI-assisted visual processing. The robot can operate manually while receiving intelligent support for obstacle and object detection. The design is suitable for indoor surveillance, academic demonstration and preliminary inspection tasks. The study also identifies the main technical challenges, including latency, lighting variation, limited embedded processing power, battery life and network dependence. Overall, the proposed robot provides a practical and affordable foundation for smart surveillance and future autonomous robotic development.

References

[1] D. Hercog, T. Lerher, M. Truntič and O. Težak, “Design and implementation of ESP32-based IoT devices,” Sensors, vol. 23, no. 15, p. 6739, 2023, doi: 10.3390/s23156739.

[2] D. Nimma, “Object detection in real-time video surveillance using YOLOv8-based approaches,” Alexandria Engineering Journal, 2025.

[3] M. Talib, “Improved YOLOv8 for real-time object detection,” Karbala International Journal of Modern Science, 2024.

[4] “Research paper on surveillance robot using ESP32-CAM,” International Journal of Creative Research Thoughts, 2024.

[5] W. Chen et al., “A survey of autonomous robots and multi-robot navigation,” Machine Learning with Applications, 2025.

[6] E. A. Rodríguez-Martínez et al., “Vision-based navigation and perception for autonomous mobile robots: A review,” Automation, 2025.

[7] S. Nahavandi et al., “A comprehensive review on autonomous navigation of mobile robots,” ACM Computing Surveys, 2025.

[8] G. Bradski and A. Kaehler, Learning OpenCV: Computer Vision with the OpenCV Library. Sebastopol, CA: O’Reilly Media, 2008.

[9] R. Szeliski, Computer Vision: Algorithms and Applications, 2nd ed. Cham: Springer, 2022.

[10] J. Redmon, S. Divvala, R. Girshick and A. Farhadi, “You only look once: Unified, real-time object detection,” Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 779–788.

[11] A. Howard et al., “MobileNets: Efficient convolutional neural networks for mobile vision applications,” arXiv preprint arXiv:1704.04861, 2017.

[12] M. Quigley et al., “ROS: An open-source Robot Operating System,” ICRA Workshop on Open Source Software, 2009.

[13] I. Goodfellow, Y. Bengio and A. Courville, Deep Learning. Cambridge, MA: MIT Press, 2016.

[14] “Smart surveillance robotic rover using ESP32-CAM and IoT-based control,” International Journal of Scientific Research in Engineering and Technology, 2024.

[15] “Surveillance robo car using ESP32-CAM module,” International Journal of Scientific Research in Science, Engineering and Technology, 2025.