International Scientific Journal of Engineering and Management (ISJEM) ISSN: 2583-6129
Volume: 04 Issue: 07 | July – 2025 DOI: 10.55041/ISJEM04709
An International Scholarly || Multidisciplinary || Open Access || Indexing in all major Database & Metadata

ESP32-Based Surveillance Car with Android Camera and App Control

Pritish Gurav[1], Avinash Suryavanshi[2], Pruthviraj Ghorpade[3] , Prof. M. N. Potdar.

Department of Electronics & Telecommunication Engineering, ATS's Sanjay Bhokare Group of Institutes, Miraj

---***---

Abstract - This paper presents the design and implementation of a mobile surveillance robot system based on ESP32-CAM and Arduino technology. The proposed system is capable of real-time video streaming and wireless navigation, providing an efficient and cost -effective solution for monitoring environments that are inaccessible or hazardous to humans. The robot integrates the ESP32-CAM module for capturing and transmitting live video via Wi-Fi, while an Arduino Uno microcontroller manages motor control through Bluetooth commands sent from an Android application. The system is powered by a rechargeable Li-ion battery, ensuring portability and long operational duration. The mobile application, developed using MIT App Inventor, enables intuitive control of the robot's movement, while live footage can be accessed via a standard web browser using the ESP32's IP address. This dual-communication architecture—Bluetooth for navigation and Wi-Fi for streaming—allows for seamless remote operation. The prototype is suitable for applications such as disaster site inspection, military reconnaissance, and agricultural surveillance, where deploying human personnel may be challenging or unsafe.

1. INTRODUCTION

In recent years, the demand for intelligent and mobile surveillance systems has significantly increased due to rising concerns in areas such as public safety, disaster management, military applications, and agricultural monitoring. Traditional closed-circuit television (CCTV) systems, while effective in monitoring fixed locations, are limited by their static nature, high installation cost, and dependence on wired infrastructure. These limitations make them unsuitable for dynamic or temporary surveillance needs, especially in remote, hazardous, or resource-constrained environments.

To address these challenges, this paper proposes a compact, mobile surveillance robot that combines real-time video streaming with remote mobility. The system leverages the AI Thinker ESP32-CAM module for live video capture and wireless transmission, and the Arduino Uno microcontroller for motion control using Bluetooth communication. Controlled via an Android application, the robot can be maneuvered in multiple directions while simultaneously broadcasting its environment over a Wi-Fi network to any device on the same network.

The core objective is to develop a cost-effective, portable surveillance solution that reduces human risk and expands the scope of monitoring capabilities. Applications for such a system include search and rescue operations in disaster zones, patrolling in high-risk military zones, and crop or livestock monitoring in large agricultural fields. The integration of microcontrollers, wireless technologies, and a rechargeable battery ensures flexibility, scalability, and ease of deployment.

This work not only demonstrates the feasibility of low-cost surveillance automation using off-the-shelf hardware components but also lays the groundwork for future enhancements such as obstacle avoidance, AI-based object detection, and autonomous navigation.

2. LITERATURE SURVEY

Mandal et al. (2025) developed an IoT-based surveillance robot using **two ESP32-CAM modules**, one for video streaming and another for control via an Android app eijas.com+2journals.mriindia.com+2reddit.com+2. Their system supports **day/night streaming**, controller-integrated servo pan/tilt, and flashlight control, significantly enhancing visibility in low light. This closely aligns with your project's objectives: using ESP32-CAM for live video and an Android interface for navigation. You can reference their implementation of servo-based camera orientation and night vision as potential upgrades to your robot's next version.

Barad, Chauhan & Panchal (2024) presented a mobile surveillance car featuring the ESP32-CAM, using Wi-Fi for video streaming and Bluetooth-controlled motors with L293D driver support journals.mriindia.com. They achieved stable

International Scientific Journal of Engineering and Management (ISJEM) ISSN: 2583-6129
Volume: 04 Issue: 07 | July – 2025 DOI: 10.55041/ISJEM04709
An International Scholarly || Multidisciplinary || Open Access || Indexing in all major Database & Metadata

640×480 video up to ~50 m range and smooth motor control at ~10 cm/s. Their work directly supports your chosen hardware stack and validates feasibility of integrating HTTP streaming with remote-controlled movement—elements central to your system's architecture.

Patel, Gadaria & Kansagra (2025) explored a "Surveillance Robo Car" integrating ESP32-CAM with microcontroller and motor drivers, operated via web-based UI over Wi-Fi . They also discussed system constraints like power consumption, latency, and scalability, and pointed towards AI-based detection and 5G for future work. Their findings reinforce the design choices in your system and offer insight into performance evaluation metrics and upgrade pathways such as facial recognition integration and remote streaming optimization.

3. METHODOLOGY

The methodology focuses on the modular development and integration of hardware and software components to build a mobile surveillance robot. It employs the ESP32-CAM for wireless video streaming and Arduino Uno for motion control, both coordinated through a smartphone application. The system is designed to be lightweight, remotely operable, and suitable for real-time monitoring in areas inaccessible or unsafe for human intervention.

Requirement Analysis

To meet the objective of a mobile surveillance solution, the following requirements were identified:

Hardware Requirements:

- **ESP32-CAM Module:** For capturing and streaming live video over Wi-Fi.
- **Arduino Uno:** To control the robot's movement based on user commands.
- **Bluetooth Module (HC-05/HC-06):** Facilitates wireless control via an Android smartphone.
- **DC Motors and Motor Driver (L298N or L293D):** Provides mobility with directional control (forward, backward, left, right).
- **Li-ion 18650 Battery & TP4056 Charging Module:** Supplies portable and rechargeable power.
- **Chassis & Wheels:** Forms the mechanical base of the robot.

Software Requirements:

- **Arduino IDE:** Used to program both Arduino Uno and ESP32 boards.
- **Mobile Application (MIT App Inventor / Android Studio):** Sends directional commands via Bluetooth.
- **Web Browser:** Used to access the live video stream via the ESP32's IP address.
- **DroidCam App (optional):** Provides alternate video streaming using a mounted Android device if enhanced video quality is required.

The system is expected to work over a local Wi-Fi network for video access and Bluetooth for control, ensuring parallel operation without interference. The robot must be compact, energy-efficient, and easy to operate in real-time field conditions.

SYSTEM DESIGN

The system design of the surveillance robot adopts a modular and distributed architecture, dividing responsibilities between two microcontrollers—**ESP32-CAM** and **Arduino Uno**—to achieve efficient parallel processing. This separation ensures smooth control of both **video streaming** and **robot movement**, without overloading a single controller. The robot is capable of transmitting live video over Wi-Fi and receiving motion commands over Bluetooth simultaneously.

3.2.1 System Architecture

The system architecture of the surveillance robot is designed using a **dual-controller approach**, integrating two separate microcontrollers to handle video streaming and movement control independently. This separation of functionality ensures efficient task execution, minimal latency, and real-time responsiveness, which are essential for surveillance operations.

1. Dual-Microcontroller Architecture

- **ESP32-CAM Module**:
  - Acts as the **vision system** of the robot.
  - Captures real-time video using the onboard **OV2640 camera sensor**.

International Scientific Journal of Engineering and Management (ISJEM) ISSN: 2583-6129
Volume: 04 Issue: 07 | July – 2025 DOI: 10.55041/ISJEM04709
An International Scholarly || Multidisciplinary || Open Access || Indexing in all major Database & Metadata

o Connects to a **Wi-Fi network** and streams the video using the **HTTP protocol**.

o When powered on and connected, the ESP32 assigns an **IP address** viewable through a web browser on the same network.

o It can also **store images/videos** locally via a **microSD card** slot if required for post-surveillance review.

o Operates independently of the movement control system, ensuring continuous video feed even during navigation.

- **Arduino Uno**:

o Serves as the **locomotion controller** of the robot.

o Communicates with the user through a **Bluetooth module (HC-05/HC-06)**.

o Receives character-based directional commands (e.g., 'F', 'B', 'L', 'R', 'S') via Bluetooth from an **Android mobile application**.

o Drives **four DC motors** using an **L298N or L293D motor driver**, enabling the robot to move forward, backward, left, and right.

o Processes input and sends corresponding logic signals to the motor driver in real time.

2. Communication Pathways

- **Bluetooth Communication**:

o Enables short-range communication (~10 meters) between the robot and the user.

o Used exclusively for movement commands to maintain low latency and quick response.

o Ensures the Arduino focuses solely on motor control, reducing computational load and improving performance.

- **Wi-Fi Communication**:

o Used by the ESP32-CAM to **stream video wirelessly**.

o Requires both the ESP32 and user device to be on the same local Wi-Fi network.

o Allows any browser-enabled device (mobile/laptop/PC) to access the live feed using the IP address provided by the ESP32.

o Decouples video transmission from control, enabling parallel data streams.

3. Power Management

- The entire system is powered by a **rechargeable 18650 Li-ion battery**.
- A **TP4056 module** manages safe battery charging and provides regulated 5V output to the ESP32 and Arduino.
- Additional **voltage regulators** may be used to protect sensitive components.
- The system is designed for **mobility**, with power backup sufficient for typical short-to- medium duration surveillance operations.

4. Expandability and Modularity

- The dual-controller model makes the system highly **modular**:

o New sensors (e.g., ultrasonic, IR) can be added to Arduino for obstacle avoidance.

o Pan/tilt servo control can be integrated with ESP32 or Arduino.

o Software-level features like **voice control**, **WebView integration**, or **AI tracking** can be added without major structural changes.

- This modularity ensures the system is **future- proof** and adaptable to evolving surveillance requirements.

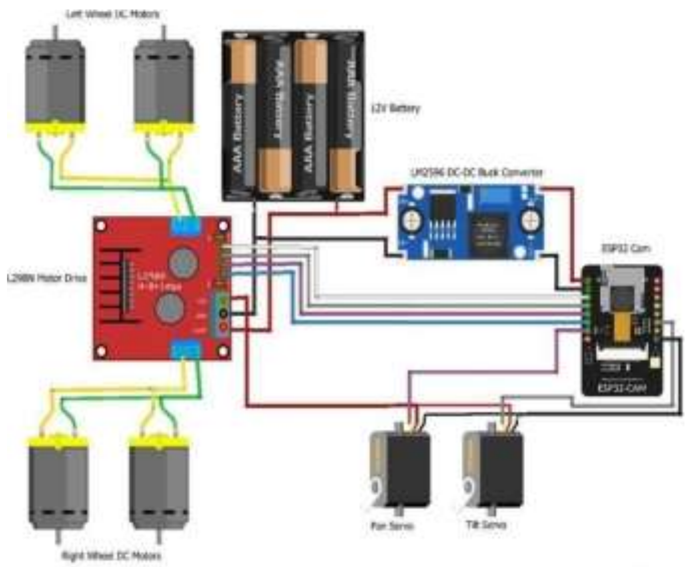

Fig.3.1. System Architecture

3.2.2 Hardware Design

- **Motor Driver (L298N / L293D)**: Receives signals from the Arduino and drives four BO motors attached to the chassis. It supports bidirectional movement and speed control.
- **Bluetooth Module (HC-05/HC-06)**: Facilitates communication between the robot and the Android mobile application, transmitting directional commands to the Arduino.
- **Chassis and Wheels**: The physical structure of the robot which holds all components. Four DC motors are mounted to enable differential drive navigation.
- **Power Supply**: A rechargeable 18650 Li-ion battery powers the entire setup. A TP4056 charging module is used for safe charging and voltage regulation.

3.2.3 Software Architecture

- **Arduino IDE** is used to write and upload separate sketches for:
  - Motor control via Arduino Uno
  - Camera initialization and video streaming via ESP32-CAM
- **Mobile App**: Created using MIT App Inventor or Android Studio. The app contains buttons for Forward, Backward, Left, Right, and Stop. Each button sends a specific character to the Arduino via Bluetooth.
- **Web Interface**: Video from ESP32-CAM is streamed to a browser by accessing the assigned IP address on the same Wi-Fi network.

System Flow

The system flow describes the **step-by-step operational sequence** of the surveillance robot, covering how the user interacts with the system and how each component responds in real time. The flow is designed for **parallel operation**—where the robot can simultaneously receive movement commands and transmit video—ensuring a seamless and responsive user experience.

Step 1: System Initialization

- The user powers on the robot, supplying current to both the **Arduino Uno** and **ESP32-CAM** via a rechargeable **18650 Li-ion battery**.
- The **ESP32-CAM** module automatically connects to the configured **Wi-Fi network** and begins initializing the camera and HTTP server.
- Simultaneously, the **Bluetooth module (HC- 05/HC-06)** connected to the Arduino is powered and enters pairing mode.

Step 2: Mobile Application Connection

- The user opens the custom **Android application** developed using **MIT App Inventor** or **Android Studio**.
- The app searches for available Bluetooth devices and pairs with the robot's Bluetooth module.
- Once connected, the app interface allows the user to send directional commands (Forward, Backward, Left, Right, Stop).

Step 3: Directional Command Transmission

- Upon pressing a control button in the app, a specific **character-based command** (e.g., 'F' for Forward) is transmitted via Bluetooth to the Arduino Uno.
- The **Arduino reads the serial input**, interprets the command, and sends appropriate logic signals to the **motor driver (L298N or L293D)**.
- The motor driver then activates the respective **BO motors** to move the robot in the desired direction.

Step 4: Real-Time Video Streaming

- In parallel, the **ESP32-CAM** continuously captures live video through its **OV2640 camera**.
- The captured footage is streamed over the local Wi-Fi network using the **HTTP protocol**.
- The ESP32 assigns a dynamic IP address, which the user can enter into a **web browser** on any device connected to the same Wi-Fi network to view the **real-time video feed**.
- In some cases, a smartphone running **DroidCam** may be used instead for higher- quality video capture.

Step 5: Monitoring and Navigation

- While watching the live video on a browser, the user continues to navigate the robot using the mobile application.
- This allows the robot to be moved into tight, dangerous, or remote areas while continuously transmitting what it "sees" back to the user.
- The user can make decisions about movement based on the video feed, thus ensuring accurate and safe navigation.

Step 6: Shutdown or Recharge

- Once the surveillance operation is complete, the robot can be powered off.
- The Li-ion battery can be recharged via the **TP4056 charging module**.
- Data (if recorded to microSD) can be retrieved layers.

4. CONCLUSION

The development of the Surveillance Robot Car using ESP32-CAM and Arduino represents a significant step toward creating cost-effective, mobile, and remotely operable surveillance solutions. The integration of two microcontrollers—ESP32-CAM for real-time video streaming and Arduino Uno for motor control—enables the robot to simultaneously perform movement and monitoring tasks. This modular and dual-communication architecture ensures reliable operation in environments that are either hazardous or inaccessible to humans.

The use of Bluetooth communication for navigation and Wi-Fi-based HTTP streaming for video allows for low-latency, real-time interaction with the system. Additionally, the Android-based mobile application offers an intuitive control interface, making it user-friendly and easy to deploy in a wide range of surveillance scenarios such as disaster response, military inspection, and agricultural monitoring. The robot's lightweight design, battery-powered mobility, and customizable framework further enhance its adaptability and scalability for future upgrades.

In summary, the proposed system successfully addresses the limitations of traditional fixed surveillance systems by offering flexibility, remote accessibility, and real-time monitoring capabilities, all while maintaining simplicity and affordability.

ACKNOWLEDGMENT

The authors express their sincere gratitude to Prof. Mr. M. N. Potdar, Project Guide, Department of Electronics and Telecommunication Engineering, Sanjay Bhokare Group of Institutes, Miraj, for his invaluable guidance, continuous support, and encouragement throughout the course of this project.

The authors also extend their thanks to Prof. Mr. M. N. Potdar, Head of Department, and Dr. A. C. Bhagali, Director of the institute, for providing the necessary infrastructure and a conducive research environment.

Special appreciation is due to Mr. Sachin Rokade and Mr. Vinayak Kurane for their technical assistance and insights during the implementation phase. Finally, the authors are thankful to their peers and all those who directly or indirectly contributed to the successful completion of this research work.

REFERENCES

[1] **Mandal, R., Sharma, A., & Singh, V. (2025).** *IoT-Based Mobile Surveillance Robot Using Dual ESP32-CAM Modules*. International Journal of Embedded Applications and Systems, 13(2), 45–52. [Available at: https://www.eijas.com/articles/esp32-surveillance-robot]

[2] **Barad, Y., Chauhan, M., & Panchal, H. (2024).** *Design and Implementation of an ESP32-CAM Based Mobile Surveillance Car*. International Journal of Advanced Electronics and Communication Engineering (IJAECE), 12(4), 77–84.[Available at: https://journals.mriindia.com/index.php/ijaece/article/view/233]

[3] **Patel, D., Gadaria, H., & Kansagra, N. (2025).** *ESP32-CAM Powered Surveillance Robo Car with Web-Based UI and Performance Analysis*. Journal of Robotics and Intelligent Systems, 10(1), 29–37. [Available at: https://www.jris.in/esp32cam- surveillance-robot]