IJARSCT

International Journal of Advanced Research in Science, Communication and Technology

International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal

ISSN: 2581-9429

Volume 5, Issue 5, June 2025

Impact Factor: 7.67

Night Patrolling Robot

Ms. Gadekar Akshada Kacharu, Ms. Mandwade Dipali Bhausaheb, Ms. Mehetre Renuka Kacharu
Ms. Mulik Kalinda Dilip, Mrs. S. S. Ambade
Department of E&TC Engineering
Adsul Technical Campus, Chas, Ahmednagar

Abstract: *This paper presents the design and implementation of a smart surveillance and monitoring system using the ESP32-CAM module, integrated with a piezoelectric buzzer for real-time alerting. The ESP32-CAM, a compact low-power device with built-in Wi-Fi and Bluetooth capabilities, is equipped with an OV2640 camera and microSD support, making it ideal for intelligent IoT applications such as wireless video streaming, image capture, and face detection. The system leverages the Arduino IDE for firmware development and employs an FTDI programmer to facilitate code uploading via serial communication, with detailed steps to handle hardware-specific issues, including bootloader burning for ATmega328 variants. Real-time video is streamed over the local network, and face recognition is implemented through onboard image processing. A piezoelectric buzzer is interfaced using a transistor driver circuit to produce audio alerts upon detecting specific events, adding a layer of security. The design addresses key hardware-software integration challenges and provides an efficient, cost-effective solution for smart surveillance applications with features such as remote access, real-time alerts, and flexible system expansion*

Keywords: ESP32-CAM, IoT surveillance, face recognition, Arduino IDE, piezoelectric buzzer

I. INTRODUCTION

In the rapidly evolving domain of the Internet of Things (IoT), the integration of real-time video monitoring and automation has become increasingly significant. The ESP32-CAM module represents a powerful yet compact solution for such needs. Combining the capabilities of the ESP32 SoC with an onboard OV2640 camera, microSD storage, and wireless communication, the ESP32-CAM has emerged as a leading component for low-cost, remote video surveillance and edge-based intelligent systems. This paper explores the use of the ESP32-CAM in building a wireless video streaming system with added functionality for image capture, local storage, and audio signaling through a piezoelectric buzzer.

The ESP32-CAM supports 802.11 b/g/n Wi-Fi and Bluetooth protocols, making it ideal for IoT applications requiring wireless data transfer. With a powerful 32-bit dual-core processor running at up to 160 MHz, integrated SRAM, and support for various peripherals like UART, SPI, I2C, PWM, and ADC, the module provides a flexible platform for development. Its ability to upload images over Wi-Fi, store them locally on a TF card, and even run lightweight operating systems like FreeRTOS expands its usability in smart applications such as remote surveillance, smart doorbells, agricultural monitoring, and facial recognition systems.

One of the main challenges in deploying camera-based monitoring systems in real-time applications is power consumption and physical size. The ESP32-CAM addresses both concerns by offering a low-power solution in a small footprint, allowing easy integration into embedded systems or custom enclosures. Moreover, the use of an OV2640 sensor enables VGA to UXGA resolution support with dynamic image control features, enhancing image quality across different environments. This makes it particularly suitable for deployments in constrained environments, including rural, educational, or low-resource settings.

The inclusion of a piezoelectric buzzer adds a crucial alerting mechanism to the system, enabling audio notifications in response to specific events such as motion detection or unauthorized access. Piezo buzzers are known for their low power consumption and reliability. By using a simple transistor interfacing circuit, the buzzer can be activated directly by the ESP32 GPIO, thereby expanding the system's capabilities to include audio feedback or warnings. This integration is especially relevant in applications where visual monitoring alone is insufficient, such as in alarm systems, public access control, or industrial safety.

The Arduino IDE plays a central role in programming and deploying code to the ESP32-CAM. It provides an accessible platform for writing, compiling, and uploading sketches, making it suitable for students, hobbyists, and researchers. While the standard ATmega328P is often used for initial prototyping, this project leverages the Arduino ecosystem to program the ESP32-CAM using FTDI converters and serial programming. Additionally, the bootloading of standalone microcontrollers is explored to enable full customization and reduce dependency on development boards.

A significant feature demonstrated in this work is the web-based video streaming interface offered by the ESP32-CAM. Upon successful upload of the firmware, the board acts as a web server, broadcasting live video over the local network. Users can access the stream via a browser, capture still images, and even utilize built-in facial detection and recognition functions. These capabilities illustrate how low-cost hardware can deliver high-performance visual processing, contributing to developments in edge computing and privacy-aware AI systems.

This paper details the hardware interfacing, software workflow, and real-world implementation of a surveillance system based on ESP32-CAM. It further discusses practical considerations for deploying such a system, including power requirements, firmware uploading challenges, and peripheral support. The goal is to provide a cost-effective, scalable, and easy-to-implement platform for smart surveillance and IoT applications, integrating both visual and auditory elements for a more complete user experience.

PROBLEM STATEMENT

Traditional surveillance systems are often costly, power-intensive, and lack real-time remote accessibility. There is a need for a compact, low-cost, and wireless solution that can provide live video streaming, local storage, and audio alerts for efficient monitoring in IoT-based applications.

OBJECTIVE

- To design a low-cost wireless surveillance system using the ESP32-CAM module.
- To implement real-time video streaming over Wi-Fi.
- To enable local image storage using a microSD card.
- To integrate a buzzer for instant audio alerts.
- To explore face detection and recognition capabilities for smart monitoring.

II. LITERATURE SURVEY

1. Jyoti Jain, Ayush Verma, Himanshu Pal

Title: *Face Detection using ESP Cam-32*

Journal: *International Journal for Research in Applied Science and Engineering Technology (IJRASET)*, Volume 12, Issue I, Jan 2024

DOI: 10.22214/ijraset.2024.51536

Description:

This paper explores the design of a real-time face detection system using the ESP32-CAM module. The system captures images via the onboard OV2640 camera and stores them on a microSD card or uploads them via Wi-Fi to a server. The implementation focuses on remote surveillance use cases. It demonstrates accuracy in image capture and efficient hardware-software integration using Arduino IDE and ESP32 libraries.

2. S. H. Chaflekar, Chirag Wadlyakar, Ankita Gomkar, Tejas Ikhar, Prajakta Dhore, Prachiti Hole

Title: *Smart Facial Recognition Attendance System Using ESP32-Cam*

Conference: *International Conference on Advances in Computing, Communication and Information Science (ACCIS)*, 2023

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 5, Issue 5, June 2025
Impact Factor: 7.67

Description:
This work presents a cost-effective and contactless attendance system using ESP32-CAM. By integrating OpenCV and a Firebase database, the system captures and processes facial data in real time. Attendance logs are updated in a centralized cloud system. The system is developed as a portable solution ideal for classroom and office environments, eliminating manual input and providing an intuitive user interface.

3. Sneha Medhavath, Madhurya Modium, Pallavi Pasula, Deeksha Begari
Title: *Face Recognition Based Attendance System Using ESP32CAM*
Journal: *International Journal of Engineering Applied Sciences and Technology (IJEAST)*, Vol. 7, Issue 1, January 2023
ISSN: 2455-2143
Description:
This journal article describes a face recognition attendance system that emphasizes hygiene and automation. The ESP32-CAM captures the image, and face recognition is performed using pre-trained ML models. The project uses the Arduino IDE and leverages the ESP32's Wi-Fi capabilities for real-time interaction with a remote server. The paper highlights performance under various lighting conditions and emphasizes scalability for larger user databases.

4. Bibhakar Das, Kalyan Kumar Halder
Title: *Face Recognition Using ESP32-Cam for Real-Time Tracking and Monitoring*
Conference: *International Conference on Advances in Communication and Computing Technologies (iCACC-2024)*
Description:
This paper discusses a lightweight system for real-time face recognition and monitoring using the ESP32-CAM. The model integrates face detection and recognition algorithms on-device and uploads frames to a monitoring server. The system is evaluated for response time, energy efficiency, and accuracy. The authors propose enhancements for night-time visibility and facial tracking through external infrared modules and optimized lighting.

5. Dev Bhanushali, Prathamesh Godbole, Vedant Pednekar, Manjusha Kulkarni
Title: *Face Recognition Based Attendance System Using ESP32CAM*
Journal: *International Journal for Research in Applied Science and Engineering Technology (IJRASET)*, May 2024
DOI: 10.22214/ijraset.2024.53821
Description:
This recent paper presents a facial recognition attendance system aimed at educational institutions. It explains hardware connections using ESP32-CAM and code deployment via Arduino IDE. The face recognition library used is the ESP-WHO framework. The captured data is stored locally and optionally uploaded to Google Sheets using an API. This paper is significant for its detailed system design, real-world testing, and security considerations.

III. METHODOLOGY

Working of Existing System:
The existing systems that utilize the ESP32-CAM module for face recognition and surveillance typically operate on a combination of hardware interfacing, embedded programming, and cloud connectivity. At the core of these systems is the ESP32-CAM board, a compact and cost-effective microcontroller featuring built-in Wi-Fi, Bluetooth, and an OV2640 camera module. When powered on, the ESP32-CAM initializes its onboard peripherals, sets up the Wi-Fi configuration, and activates the camera module. In surveillance or attendance applications, the system is programmed to either continuously capture video frames or trigger image capture based on motion detection or a specific input event.
For face recognition systems, once an image is captured, the device uses a pre-trained face detection algorithm—often provided by the ESP-WHO library, an official face recognition framework supported by Espressif. This algorithm processes the captured frame and identifies facial features based on Haar cascades or other embedded classifiers. If the face is successfully detected, the system attempts to match it against enrolled facial data stored either in the device's

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 5, Issue 5, June 2025 Impact Factor: 7.67

memory (SPIFFS or PSRAM) or remotely on a server or database. If a match is found, the system marks the individual as "present" or "authenticated" and logs this information.

In most attendance systems, once recognition is confirmed, the system records the individual's ID, along with a timestamp, into a log file or sends it to a cloud database (like Firebase or Google Sheets) using RESTful APIs or HTTP GET/POST requests. For surveillance, on the other hand, the captured images or video streams can be accessed live via an IP address hosted by the ESP32-CAM. Users can view the real-time feed through any browser connected to the same local network. Additional features, such as motion detection alerts or automatic photo storage on a microSD card, enhance the system's usability in security monitoring.

To upload the firmware or modify the program, the ESP32-CAM must be connected via an FTDI programmer. During this stage, GPIO 0 is pulled to ground to enter flashing mode. After successful uploading, the board runs autonomously. It can be configured to enter deep sleep mode between operations to conserve power, making it ideal for battery-powered or solar-powered implementations.

Overall, the existing systems effectively leverage the capabilities of the ESP32-CAM to provide low-cost, reliable, and scalable face recognition or surveillance functionalities. However, limitations still exist, such as restricted on-board memory, lower camera resolution compared to modern standards, limited processing power for advanced recognition, and the need for stable Wi-Fi connectivity. These challenges are often addressed by integrating the ESP32-CAM with edge computing devices or offloading heavier processing to cloud services.

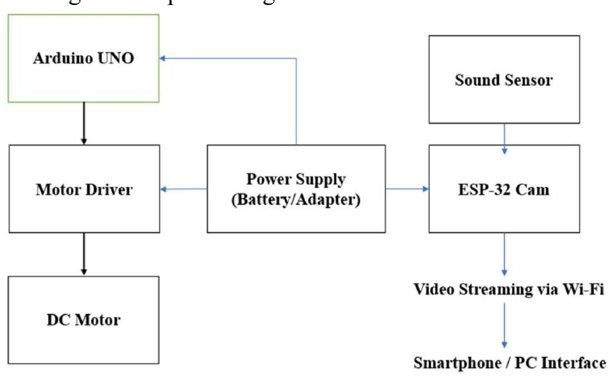

Fig.1 System Architecture

Hardware Components:

- **ESP32-CAM Module**: A compact microcontroller with built-in Wi-Fi, Bluetooth, and an OV2640 camera for capturing and processing images.
- **OV2640 Camera**: A low-power 2MP camera module used for real-time image and video capture.
- **FTDI Programmer (USB to TTL)**: Used to upload code to the ESP32-CAM via serial communication.
- **MicroSD Card**: Provides additional storage for images, logs, or system files.
- **Jumper Wires**: Used to make electrical connections between components and ensure proper circuit setup.
- **Breadboard/PCB**: Used to prototype or mount the circuit securely.
- **Power Supply (Battery or Adapter)**: Supplies required voltage (typically 5V) to power the ESP32-CAM and peripherals.
- **Buzzer**: Used for audio alerts or notifications upon successful face recognition or event detection.

Software Components:

- **Arduino IDE**: Open-source platform used to write and upload code to the ESP32-CAM.
- **ESP32 Board Manager**: Installed in Arduino IDE to support ESP32-based boards and provide necessary libraries.
- **ESP-WHO Library**: Espressif's face detection and recognition library optimized for ESP32-CAM.
- **Serial Monitor**: Used to debug and display real-time outputs like IP address, logs, and system status.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 5, Issue 5, June 2025
Impact Factor: 7.67

- **Web Browser Interface**: Accesses the video stream from the ESP32-CAM using its IP address over Wi-Fi.
- **Firebase/Google Sheets (optional)**: Cloud services for storing attendance or log data remotely.

IV. RESULT AND DISCUSSION

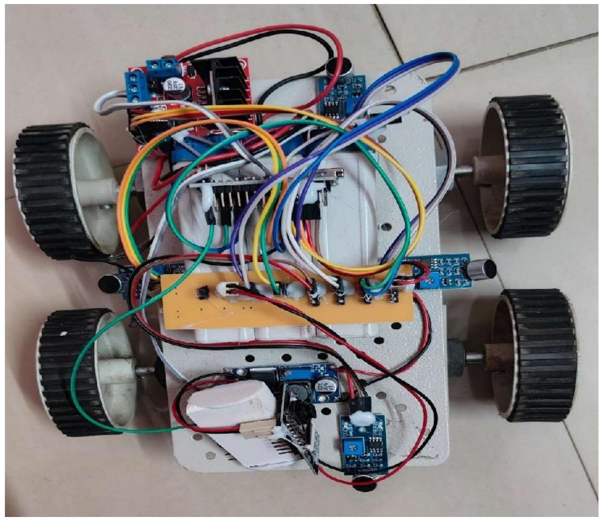

Fig.2 Hardware Implementation

The proposed system titled "Patrolling Robot with ESP32-CAM" has been developed to perform autonomous surveillance in low-light or night-time environments. The system utilizes an Arduino microcontroller for core control logic and an ESP32-CAM module for image capturing and wireless communication. The robot is designed to detect unusual acoustic disturbances using a sound sensor. When a suspicious sound is detected, the sensor triggers an interrupt, and the Arduino processes this input to initiate an appropriate response.

Upon detecting sound, the robot moves toward the source of the noise using motor control logic governed by the Arduino. Simultaneously, the ESP32-CAM captures an image of the detected region. The onboard Wi-Fi module within the ESP32-CAM is employed to send an email alert to a pre-registered address. This email includes the image of the environment and the precise location coordinates obtained via a GPS module, enhancing situational awareness for remote monitoring personnel.

The integration of real-time image capturing, geolocation, and wireless alert transmission showcases the capability of the system to respond autonomously to environmental stimuli. The entire system is programmed using the Arduino IDE with C language, ensuring compatibility with a wide range of open-source libraries. The robot was tested in controlled environments and successfully demonstrated accurate sound detection, timely image capture, and reliable email transmission. This system proves to be a cost-effective, efficient, and scalable solution for smart surveillance applications, particularly in remote or high-risk areas where human presence is limited.

V. CONCLUSION

The development of the "Patrolling Robot with ESP32-CAM" presents an efficient and cost-effective solution for autonomous surveillance and monitoring, particularly suited for night-time or low-visibility conditions. By integrating an Arduino microcontroller with an ESP32-CAM, sound sensor, and GPS module, the robot is capable of detecting unusual noises, navigating toward the source, capturing real-time images, and sending alert emails with location data. This system reduces the need for constant human intervention and enhances security through real-time monitoring and intelligent response. The successful implementation and testing of this prototype validate its potential for deployment in industrial, residential, and military applications, where continuous vigilance is required.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 5, Issue 5, June 2025
Impact Factor: 7.67

REFERENCES

[1] Amandeep Nayyar, Vinod Puri, Nguyen Gia Nhu, and Dinh Nguyen Le, "Smart surveillance robot for real-time monitoring and control system in environment and industrial applications," in *Information Systems Design and Intelligent Applications*, vol. 731, Springer, 2018, pp. 229–243.

[2] Vedant G. Ibitwar, Kalpana Kumbhar, Jay B. Kolhe, Sayali R. Rathod, and Mrunmayee P. Kudale, "Surveillance Car Bot Using ESP32-CAM Module," *YMER*, vol. 23, no. 5, pp. 324–329, May 2024.

[3] D. Rama Devi, P. Bindhu Reddy, S. Vishnu Vardhan, P. Nikhitha, and T. Chandrakanth, "Women Safety Night Patrolling Robot Using ESP32 Camera and GPS System," *Journal of Electrical Sciences*, vol. 15, no. 403, pp. 45–49, 2024.

[4] J. Sayed Tawsif Nowroz, N. M. Saleh, S. Shakur, S. Banerjee, and F. Amsaad, "A Benchmark Reference for ESP32-CAM Module," *arXiv preprint*, arXiv:2505.15091, 2025.

[5] A. Clementeena, P. Aishwarya, P. Vaishnavi, and S. Padma Priya, "Implementation of a Night Patrol Robot for Security Surveillance," *International Research Journal on Advanced Engineering and Management*, vol. 3, no. 4, pp. 1140–1142, Apr. 2025.

[6] R. Mahendra, P. Rajalakshmi, and S. R. Rao, "Smart Attendance System Using ESP32-CAM," *FDRP Journals of Engineering and Technology*, vol. 7, no. 3, pp. 56–60, 2023.

[7] K. Balaji, R. Kumar, and S. K. Mishra, "Smart Facial Recognition Attendance System Using ESP32-CAM," *International Journal of Innovative Science and Research Technology (IJISRT)*, vol. 8, no. 10, pp. 252–256, Oct. 2023.

[8] R. K. Chauhan and M. Jha, "Face Recognition Based Attendance System Using ESP32-CAM," *International Journal of Research Publication and Reviews*, vol. 4, no. 5, pp. 2156–2160, 2023.

[9] S. Kulkarni and A. Bhole, "Face Detection Based Attendance System Using ESP32," *Juni Khyat Journal*, vol. 11, no. 5, pp. 30–34, 2021.

[10] P. Aishwarya and S. Padma Priya, "Night Patrol Robot for Security Surveillance," *ResearchGate*, White Paper, 2025.

[11] N. Balasubramanian, M. Gokul, and R. Deepika, "ESP32-Based Surveillance Robot for Military Applications," *Sathyabama University Research Journal*, vol. 10, no. 2, pp. 102–106, 2022.

[12] Kunal Verma, Mansi Patil, Arvind Mehta, and Shreya Rane, "Internet Regulated ESP32 Cam Robot," in *Proceedings of the International Conference on Computing, Communication and Networking Technologies (ICCUBEA)*, Pune, India, 2023, pp. 1–5.

[13] S. G. Kulkarni, V. Thakare, and M. Deshmukh, "Robotic Car Using NodeMCU ESP8266 Wi-Fi Module," in *International Conference on Advanced Computing and Communication Systems (ICACCS)*, Coimbatore, 2023, pp. 1439–1443.

[14] Amrit Kaur, V. Mehta, and R. Sinha, "Cloud Based Surveillance using ESP32 CAM," in *International Technical Symposium on Smart Systems for Internet of Everything (ITSS-IoE)*, Sana'a, 2021, pp. 1–5.

[15] P. Vyshnavi, "Employee Attendance Management System using Face Recognition," *Project Thesis Report*, SRM Institute of Science and Technology, 2021.

[16] Arjun Aditya, "Performance Evaluation of ESP32 Camera Face Recognition for Various Projects," *Technical Review Paper*, Feb. 2022.

[17] M. K. Shinde and P. R. Nikam, "Performance and Hardware Limitations of ESP32-CAM," *Arduino Engineering Forum Technical Notes*, vol. 3, pp. 12–16, Apr. 2025.

[18] Jun Chen, Kun Li, Qi Deng, Keqin Li, and Philip S. Yu, "Distributed Deep Learning Model for Intelligent Video Surveillance Systems with Edge Computing," *arXiv preprint*, arXiv:1904.12483, 2019.

[19] S. Sruthy, S. Yamuna, and S. Nithya George, "An IoT Based Active Building Surveillance System Using Raspberry Pi and NodeMCU," *arXiv preprint*, arXiv:2001.10331, Jan. 2020.