Animal Detection and Monitoring System Using IoT

Aadhavan G[1], Dushan S[1], Giriraj N[1], Mani P[2]

[1]Student, [2]Assistant Professor

Department of Electronics and Communication Engineering

K. Ramakrishnan College of Technology, Trichy, Tamil Nadu, India.

Abstract: Animal Detection and Monitoring System Using IoT is an economical system that monitors the presence of any animals and alerts the people before hand through IoT technology. Damage to crops, accidents caused due to vehicles and conflicts between animals and humans near the farms or at the border of forests are major concerns, and manually keeping watch is a laborious process which also comes with cost constraints and is not reliable at night time. In this project, we have used ESP32-CAM, which is a module which consists of both a microcontroller and a camera. Images of the area are captured continuously by the system and then those captured frames are analyzed for the presence of any animal. Once any animal is detected, then the system gives an alarm through a buzzer and an LED light, the status is shown on OLED and it sends the information via Wi-Fi network.This design uses cheap components that are easily obtainable, and it can be used on farms, gardens, village borders, and edges of forests. The ways through which this design could be enhanced include cloud storage, mobile notifications, solar energy, PIR sensors, and advanced machine learning algorithms.

Key Words: Animal Detection, IoT, ESP32-CAM, Monitoring System, Buzzer Alert, Wildlife Intrusion, Low-Cost Embedded System.

INTRODUCTION

A perennial issue that faces farming and village activities in areas adjacent to forests is the incursion of wild animals into human habitation areas. This includes damage to farms by animals like wild boar, elephants, deer, and monkeys, as well as road and railway accidents due to these animals crossing these routes. The conventional means of dealing with such problems include watchmen, fences, scarecrows, and patrolling. This requires constant manpower and is hard at night and during bad weather as well. It also does not necessarily offer a timely response to the situation.

Embedded systems technology, cheap cameras, and IoT have allowed development of automatic monitoring systems. The IoT-based system utilizes sensors, microcontroller, and wireless communication to gather data from the environment, analyze it, and issue alerts without human intervention all the time. The Animal Detection and Monitoring System Using IoT is a solution that provides simplicity and economic feasibility through the use of ESP32-CAM, where there is a combination of a Wi-Fi microcontroller and OV2640 camera on one board.

The proposed system is built using an ESP32-CAM (AI-Thinker) board, a regulated 5V power source, an active buzzer board, LEDs and a 0.96 inch OLED display, all connected with the help of jumper wires and breadboards. The camera will be used to capture images from the surveillance area. The ESP32-CAM will analyze each frame taken by the camera to determine whether there is an animal. In case of an animal's detection, the buzzer will be turned on along with the red LED, a message will appear on the OLED display, and the information will be transmitted via the Wi-Fi network. In case there is no animal, the green LED will be turned on and the monitoring will continue.

This can be used on the edges of farms, gardens, plantations, villages, and roadside forests. For the future, this could be improved through cloud storage of images, mobile alerts, solar energy, motion sensing, and improvement in the detection model for classification of the animal.

HISTORIOGRAPHY AND RESEARCH GAPS

Animal protection has evolved from primitive techniques like watch towers, fences, noise production, and patrolling to more advanced electronic systems. The early electronic systems employed sensors like PIR and ultrasonic sensors which detect motion around an area that needs to be safeguarded. The sensors mentioned above are inexpensive, but

they do not differentiate between an animal, a human being, a car, or moving plants and hence generate many false alarms. With the emergence of low-cost microcontrollers like Arduino and ESP32, camera detection systems became available, and machine learning techniques such as YOLO and MobileNet were able to recognize objects from images.

However, despite these improvements, some challenges are still evident. Some of the devices detect the object's movement but fail to confirm whether it is an animal or not. Devices that require powerful processors and cloud server are costly and require stable internet connection, which might not always be feasible in rural locations. Some of the devices just trigger local alarms without transmitting the information to other places. Furthermore, most of the current devices are not feasible for small farmers and learners due to high costs and complexity. Thus, there is a need for simple and inexpensive device capable of combining visual recognition, triggering of alarms and remote monitoring. The Animal Detection and Monitoring System Using IoT is the solution as it is based on the ESP32-CAM along with buzzer, LEDs, and OLED display.

The performance of the system in terms of detection rate, false alarm generation, response time and power consumption can be assessed through field testing.

PRIMARY SOURCES AND METHODOLOGY

The information sources for the current work involve the images and status output generated by the ESP32-CAM, the specifications of the hardware used, and the data recorded when the system was operational. Datasheets provide the working principle, electrical specifications, pin out details, and limitations of ESP32-CAM, buzzer module, LEDs, and OLED screen. The aim of the work is to develop an inexpensive and easy-to-use system for animal detection.

The first step in the method involves identifying the shortcomings of existing protection systems which include need for constant human intervention, delayed action, false alarms and high cost. This is followed by setting of system requirements and selection of system components. The ESP32-CAM was selected because it incorporates the processor, camera, WiFi and PSRAM in one board. The USB programmer base is employed to upload the program while a regulated 5V power source is used to power the board. Buzzer module, green and red LEDs and OLED are connected to appropriate GPIO pins of ESP32-CAM. The program is developed using the Arduino IDE and uploaded into the board.

The working of the system is as follows. Once the system is initialized, the camera is turned on along with the Wi-Fi connection. The camera takes a frame of the area and detects any animal. In case the animal is detected, the buzzer produces a sound, the red LED is turned on, "Animal Detected" message is displayed on the OLED screen and the detection event is communicated to the monitoring interface. In case the animal is not detected, the green LED is on, "Area Clear" message is displayed on the OLED screen

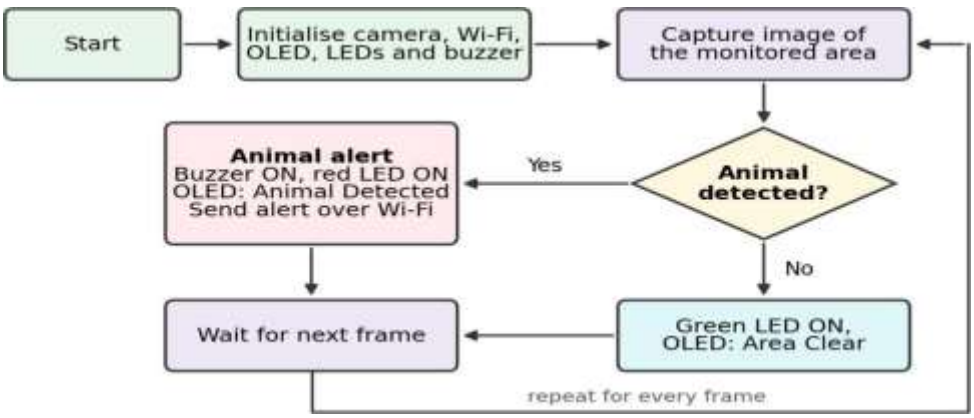


Fig. 1: Work flow diagram

Table 1: Inputs and System Responses

S. No.	Input Condition	ESP32-CAM Decision	System Output
1	System powered on	Initialise camera, Wi-Fi, OLED, LEDs and buzzer	System ready; monitoring starts
2	New image frame captured	Analyse the frame for an animal	Detection result is obtained
3	Animal detected	Activate buzzer and red LED	Alarm sounds; "Animal Detected" shown on OLED
4	Animal detected	Send detection event over Wi-Fi	Remote monitoring is updated
5	No animal detected	Keep buzzer OFF, green LED ON	"Area Clear" shown on OLED
6	Animal leaves the area	Evaluate the next frames	Alarm is switched off; monitoring continues

After programming the system, it is then tested in various conditions like presence of an animal, no object in the frame, presence of a person/vehicle, and varying light conditions. The outcome of detection, reaction of the buzzer and LED, OLED message together with reaction time in every condition is recorded. Comparison of the expected and actual outcomes is done to determine any connection error, programming issues or detection error, and then rectification is done to the wiring, threshold and programming. Evaluation will be done on the detection efficiency, alarm, response time, Wi-Fi connection stability and energy consumption of the system. Measurement should be done before any quantitative performance outcome of the system is declared.



Fig. 2: Output status (illustrative sample output

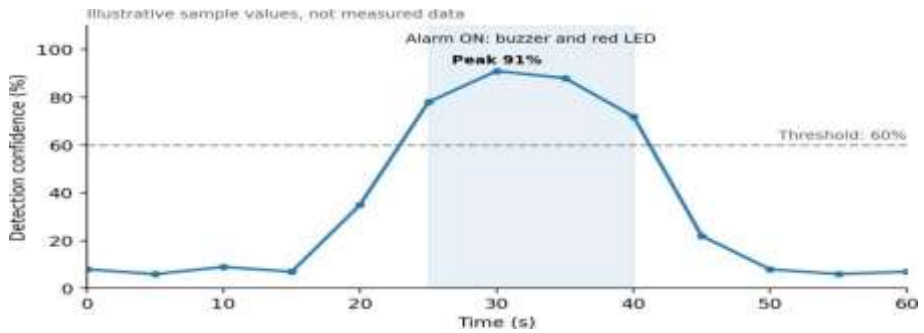


Fig. 3: Response graph (illustrative sample response)

RESULTS AND DISCUSSION

The objective of the Animal Detection and Monitoring System Using IoT is to provide advance alert of animal intrusion through the use of the ESP32-CAM together with some output devices. The image is captured using the camera, and the ESP32-CAM processes the image, and then the buzzer, LEDs, and OLED provide alert to people around.

If the animal is detected, then the buzzer and red light are turned on, if there is no animal, then there is no buzz and the green light means that everything is fine. In comparison with the system where only the motion sensor works, the presence of a camera provides more data about the object which is entering the area and helps to decrease the number of false alarms. The use of only one ESP32-CAM module, few components and low cost make it a good choice for small farms, gardens, and villages. ESP32-CAM has relatively low computing power and capacity, so the accuracy and speed of detection depends on the algorithm and conditions of illumination, distance and camera angle. At night the system might need additional infrared illumination or infrared camera.

Its real efficiency can be tested only in practice. As experimental data have not been obtained yet, it is too early to talk about accuracy of detection, false alarm rate and response time. However, the system is a feasible and economical way to detect animals automatically

CONCLUSION

The Animal Detection and Monitoring System Using IoT is a budget-friendly project that aims at minimizing the loss of assets from the intrusion of animals. The ESP32-CAM is used in capturing the images of the surroundings and identifying animals and sending out an alert through the buzzer, LED and OLED screen with data being transferred via Wi-Fi for remote monitoring. Unlike manual surveillance, the project is automated and eliminates the requirement of continuous human intervention. The project is constructed using easily obtainable materials that can be used around farms, gardens, villages borders and forests. The project is also a good educational project on embedded systems, interfacing cameras, IoT connectivity and automatic alert systems.

Further improvement can help in improving the performance of the project. In the future, storing images in cloud, notification through mobile application, utilizing solar power with battery back up, PIR motion sensor for saving energy, infrared camera for night vision and machine learning algorithms for identification of species of the animal can be introduced. Tests have to be conducted in various environmental conditions including lighting and distance to find the performance of the project in terms of accuracy and energy consumption. Since no results have been obtained

REFERENCES

[1] Arduino, Arduino IDE Documentation. Arduino Documentation.

[2] Espressif Systems, ESP32 Series Datasheet. Espressif Systems.

[3] Espressif Systems, ESP-WHO: Face Detection and Recognition Framework. Espressif Systems.

[4] Howard, A. G., et al., “MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications,” arXiv:1704.04861, 2017.

[5] Lin, T.-Y., et al., “Microsoft COCO: Common Objects in Context,” in European Conference on Computer Vision (ECCV), pp. 740–755, 2014.

[6] Liu, W., et al., “SSD: Single Shot MultiBox Detector,” in European Conference on Computer Vision (ECCV), pp. 21–37, 2016.

[7] NXP Semiconductors, I²C-Bus Specification and User Manual, UM10204, Rev. 7.0, 2021.

[8] Redmon, J., Divvala, S., Girshick, R., and Farhadi, A., "You Only Look Once: Unified, Real-Time Object Detection," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 779–788, 2016.

[9] Solomon Systech Limited, SSD1306: 128 × 64 Dot Matrix OLED/PLED Segment/Common Driver with Controller, Technical Datasheet.

[10] Wang, J., et al., "Deep High-Resolution Representation Learning for Visual Recognition," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 43, no. 10, pp. 3349–3364, 2021.

[11] Espressif Systems, ESP32 Camera Driver Documentation. Espressif Systems.

[12] OmniVision Technologies, OV2640 CMOS Image Sensor Technical Documentation. OmniVision Technologies.