July Pradeep, 2026, 14:1
ISSN (Online): 2348-4098
ISSN (Print): 2395-4752

International Journal of Science,
Engineering and Technology
An Open Access Journal

# Real-Time Wildlife Detection and Alert System Using Deep Learning and IoT

**July Pradeep, Anurag S, Micah K Binu, Shinto Sebastian, Dr.Rani Saritha R**
Department Of Computer Application Saintgits College Of Engineering Kottayam, Kerala, India

**Abstract - Communities near forests frequently face threats from wild animals entering human settlements, causing property damage and loss of life. Traditional protection relies on manual observation, often resulting in delayed responses. This project presents an automated monitoring and alert system integrating deep learning–based object detection with IoT hardware. Using YOLOv8, the system provides real-time recognition with a dual-model strategy—one general animal detector and a dedicated tiger model—to reduce false positives. Live video streams are continuously analyzed, and confirmed detections are stored in SQLite. On detection, alerts are triggered instantly via a web dashboard, buzzer, and SMS through a GSM module. Experimental evaluation demonstrated an average response time of about 2.3 seconds and an F1-score of 91%, showing the system is accurate and fast enough for deployment in high-risk areas.**

**Keywords - Wildlife Monitoring, YOLOv8, Internet of Things, Deep Learning, Intrusion Detection, Real-Time Alerting.**

## I. INTRODUCTION

Human–wildlife interaction has intensified in recent years due to the expansion of residential and agricultural zones into forested regions. Such interactions often result in crop destruction, economic losses, and serious threats to human safety. Existing monitoring practices largely depend on forest guards or residents manually spotting animals and reporting their presence. This approach introduces critical delays, during which animals may already cause damage or harm.Advancements in computer vision and embedded systems provide an opportunity to automate wildlife monitoring. Real-time object detection models can continuously analyze video feeds and identify animals without human intervention. In this work, we propose a real-time wildlife intrusion detection system that leverages YOLOv8 for high-speed object recognition and IoT components for immediate alert dissemination. By automating the detection and notification process, the system aims to significantly reduce response time and improve safety in forest-border communities.

### Proposed System

The proposed solution replaces manual surveillance with an intelligent, automated detection pipeline that operates continuously. The system processes live video input captured by surveillance cameras and analyzes each frame using deep learning models to identify the presence of potentially dangerous animals.To improve detection reliability, two YOLOv8 models are deployed. The first model is trained on the COCO dataset to identify large animals such as elephants and bears. The second model is a custom-trained classifier optimized specifically for tiger detection, addressing the frequent misclassification issues observed with generic models. Additionally, a temporal verification mechanism is implemented, requiring an animal to be detected in multiple consecutive frames before it is considered a valid intrusion. This approach minimizes false alerts caused by environmental motion such as foliage movement or shadows.

Once an intrusion is confirmed, the system records relevant information—including animal type and timestamp—into a local SQLite database and simultaneously initiates the alert sequence through connected IoT devices.

## System Architecture

The system architecture is divided into three interconnected layers that work together to ensure fast and reliable operation.

### Backend Processing Unit:

The backend is developed in Python and functions as the central controller of the system. It handles video acquisition, executes YOLOv8 inference, manages detection confirmation logic, and performs database operations. Each confirmed event is logged with precise time information for future analysis and reference.

### Image Processing Module:

OpenCV is used to preprocess incoming video frames by resizing and enhancing image quality. This preprocessing step ensures consistent input for the detection models, improving recognition accuracy under varying environmental conditions.

### Alert and Hardware Interface:

An ESP8266 microcontroller acts as the communication bridge between the software layer and physical alert devices. When the backend confirms a threat, it sends an HTTP request to the ESP8266. The microcontroller then activates a buzzer for immediate local warning and communicates with a SIM800L GSM module using AT commands to dispatch SMS alerts to predefined recipients. This tightly coupled loop enables rapid response with minimal latency.

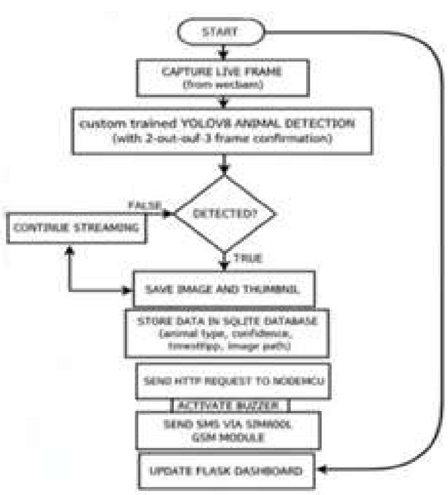


Figure 1: System Architecture

## Results and Discussion

The system was evaluated in real outdoor environments to assess its detection accuracy, response time, and operational stability. The implementation of multi-frame confirmation significantly reduced the number of false detections, especially in areas with frequent background motion.Performance analysis showed an overall F1-score of 91%. Detection accuracy for elephants reached 93%, while the specialized tiger model achieved an accuracy of 91%. End-to-end latency measurements revealed that alerts were typically delivered within one to two seconds after the animal appeared in the camera's field of view. The IoT alert mechanism remained reliable even under fluctuating network conditions, with both buzzer activation and SMS delivery functioning as expected. These results indicate that the system is suitable for real-world deployment in forest-adjacent regions

## Limitations

Despite its effectiveness, the system has several limitations. Detection performance decreases under poor lighting conditions or when animals appear at extreme angles relative to the camera. The current implementation supports a limited number of camera inputs and focuses on a predefined set of animal classes. Additionally, SMS-based alerts require stable network connectivity, which may not always be available in remote forest locations. Continuous video processing also leads to increased power consumption, posing challenges for battery-operated deployments.

## Future Enhancement

Several improvements are planned to enhance the system's robustness and scalability. Integrating GPS functionality would allow authorities to receive precise location information during intrusion events. Upgrading to LTE-based communication modules could improve network reliability and coverage. On the software side, incorporating low-light vision models and expanding the dataset to include more animal species would improve detection accuracy. The development of a dedicated mobile application and offline data synchronization capabilities would further increase usability in areas with intermittent internet access.

## II. CONCLUSION

This project demonstrates the effectiveness of combining deep learning and IoT technologies to mitigate human–wildlife conflict. By eliminating dependence on manual observation, the proposed system significantly reduces detection and response times. The results confirm that real-time automated monitoring can enhance safety for communities living near forest boundaries. With further enhancements such as nighttime detection and predictive analytics, the system has strong potential to become a practical tool for wildlife management and conservation efforts.

## REFERENCES

1. Redmon and A. Farhadi, "YOLOv3: An incremental improvement," arXiv preprint, 2018.
2. Espressif Systems, "ESP8266 Technical Reference," Espressif Systems Documentation, 2023.
3. SIMCom Wireless Solutions, "SIM800L Hardware Design," Product Specification, v1.01, 2017.
4. M. S. Norouzzadeh et al., "Automatically identifying wild animals using deep learning," PNAS, vol. 115, no. 25, pp. E5716–E5725, 2018.
5. D. Tuia et al., "Machine learning for wildlife conservation," Nature Communications, vol. 13, 2022
6. R. B. Bist et al., "YOLOv6 object detector for animal monitoring," Agriculture, vol. 13, no. 4, 2023.
7. M. Sandler et al., "MobileNetV2: Inverted residuals and linear bottlenecks," CVPR, 2018.
8. R. Kays et al., "Terrestrial animal tracking," Science, vol. 348, no. 6240, 2015.
9. A. Gomez Villa et al., "Automatic wild animal monitoring using CNNs," Ecological Informatics, vol. 41, pp. 24–32, 2017.
10. G. Jocher, A. Chaurasia, and J. Qiu, "Ultralytics YOLOv8," GitHub Repository, 2023.
11. T.-Y. Lin et al., "Microsoft COCO: Common objects in context," ECCV, 2014.