# EcoTrap: Protecting Wildlife – Smart Traps for a Greener Tomorrow

BINDUSHREE T[1], M R MANOHAR[2], AMRUTHAVARSHINI P[3], DR. CHANDRIKA J[4], M K SANDEEP[5], SHRUTHI A S[6]

[1, 2, 3, 5]*Dept of Computer Science and Engg., Malnad College of Engineering Hassan, India*

[4]*Guide, Head of Dept Computer Science and Engg., Malnad College of Engineering Hassan, India*

[6]*Co-Guide,Assistant professor Computer Science and Engg. Malnad College of Engineering Hassan, India*

***Abstract— Human-wildlife conflict is a significant global chal- lenge, with increased interactions between wildlife and human activities raising ecological, economic, and ethical concerns. This requires innovative and humane solutions for wildlife manage- ment. The project, EcoTrap, is an intelligent trapping system that integrates advanced image processing techniques and IoT technology to enhance the precision and efficiency of wildlife management practices. The system employs computer vision algorithms to detect and identify specific species of wildlife, meaning that only targeted animals are captured with a minimal probability of capturing nontargeted species. EcoTrap has also used motion detection and real-time image analysis to minimize false triggers, optimizing the mechanism of trapping. Additionally, it uses IoT sensors that provide essential information such as location, timestamp, and photographic evi- dence of the animal trapped. This information is automatically transmitted to wildlife guards and forest officers so that rescue operations can be executed in time and with adherence to ethics. By integrating technology with humane practices, EcoTrap aims at improving the efficacy of wildlife management efforts, conservation of biodiversity, and reduction in human-wildlife conflicts.***

## I. INTRODUCTION

Human-wildlife conflict, driven by habitat loss and urbanization, poses ecological and safety challenges, demanding hu- mane and innovative solutions. The EcoTrap project addresses these issues by developing an intelligent trapping system that ensures precise, ethical wildlife management.

Traditional traps often unintentionally capture non-target species and require frequent human intervention, compromising animal welfare and reducing efficiency. EcoTrap leverages image processing and IoT technology to overcome these challenges. Using computer vision algorithms, it identifies target species in real time, while motion detection minimizes false triggers from environmental factors or non-target movements. IoT sensors enhance functionality by providing the location of the trap, a timestamp, and photographic evidence of captures, instant notification to wildlife officials to facilitate timely rescue operations. Automation and ethical compliance make EcoTrap a significant advancement in humane and efficient wildlife management. EcoTrap contributes to biodiversity conservation by reducing conflict and promoting coexistence and sets a new standard in wildlife management systems.

## II. LITERATURE SURVEY OVERVIEW

The growing problems of human-wildlife conflict have made it necessary to evolve sophisticated systems that ensure humane and effective animal management. This survey syn- thesizes critical insights into the evolution of animal detection and trapping technologies, focusing on their contributions and limitations in addressing destructive medium-sized animals.

### *A. Evolution of Animal Detection Systems*

Early animal detection systems were primitive in approach, with merely static detection techniques and manual mon- itoring. Many approaches lacked precision, efficiency, and adaptability toward changing environmental conditions. For instance, camera-based systems based on handcrafted features work poorly for small species or target versus non-target animals in true detection scenarios.

### *B. Advancements in Technology*

Recent advancements in animal detection systems have greatly improved their performance, making them more ef- ficient and adaptable for wildlife management. Deep learning models, such as YOLOv5 and Cascade R-CNN, have con- tributed to high-accuracy animal detection. With YOLOv8, real-time detection capabilities and overall model efficiency have seen substantial improvements, enabling faster and more precise animal identification.

Moreover, IoT sensors contribute to remote monitoring by offering real-time, lightweight detection solutions. These sensors collect environmental data, which can assist in tracking movements of animals in order to provide more responsive systems.. These AI-based systems offer rapid processing of information on embedded devices and ensure efficient moni- toring in dynamic environments.

*C. Persistent Challenges*

Despite the advancement, there are still several issues in an- imal detection and trapping systems. High-complexity systems tend to be poor at detecting small objects and consume a lot of computational resources. Moreover, models may become less effective if they are trained with limited or imbalanced datasets. Environmental factors, such as power consumption, false positives, interference from lighting, or dense vegetation, also decrease the reliability of the system and make it diffi- cult to achieve consistent performance in various conditions. Manual intervention is still required in parts of many systems. This restricts scalability and full automation. This reliance on human operators cannot allow large-scale or distributed deployment.

*C. Trapping Mechanism*

EcoTrap uses electromechanical actuators to trigger the trap in response to detection inputs. The mechanism is designed to eliminate delay as much as possible and to catch the target species with speed and humanness. In addition, it is optimized so that the accidental triggering is minimized.

*D. Software Development*

The Arduino IDE is used to write microcontroller software. Its characteristics ensure that such microcontrollers operate smoothly and cause no complications. Control logic and algorithms are developed using Python and the OpenCV and TensorFlow libraries for rapid processing and decision-making capabilities.

*E. Power Supply*

Rechargeable lithium-ion batteries provide uninterrupted power, ensuring continuous operation in remote and off-grid environments. The power supply is complemented by energy- efficient components and low-power modes to extend battery life.

*D. Research Gaps*

Key research gaps are in the case of animal detection and trapping systems, such as robust small object detection, where current systems are mainly challenged in difficult environments. Issues with high false positive rates remain, as does the lack of night-time detection. The second gap is concerning complete end-to-end automation as there is partial integration between pipelines of detection, notification, and management, leaving much to human intervention for possible reduction. Additionally, systems often lack adaptability to diverse environmental conditions, such as dense vegetation or extreme weather, affecting their reliability and performance.

## III. METHODOLOGY

*A. Image Processing and Species Identification*

The EcoTrap system uses a high resolution camera in con- junction with machine learning for the accurate detection of a species. Processing real-time imagery, the actual processing unit of this system would be a Raspberry Pi, thus not requiring very much power at all. To efficiently identify the particular species of animals being targeted, various computer vision methods are used, including CNNs.

*B. IoT Integration*

The trap activity and location are monitored by motion sensors and GPS modules. The captured data are transmitted to the system in the form of trap status, animal identification, and geolocation by using Wi-Fi modules. IoT connectivity thus ensures real-time communication with forest officers and wildlife guards.

## IV. TOOLS AND LIBRARIES USED HARDWARE COMPONENTS

*A. Camera Module (Raspberry Pi)*

Captures images or videos of the area to monitor and detect animals, aiding in animal recognition and tracking.

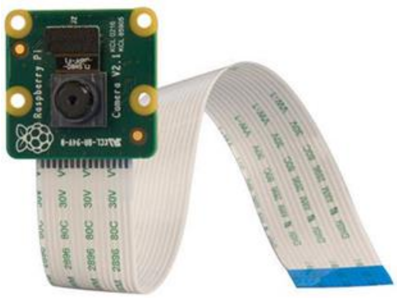

Fig. 1. Raspberry Pi Camera V2

*B. Motion Sensor (PIR)*

Detects movement near the trap, triggering the system to activate other components like the camera or weight sensor. They trigger the IoT components (such as cameras, microcon- trollers, and cloud systems) to get the information whether the detected entity is a target species.

*C. IR Sensor*

Detects proximity of animals to the trap, providing addi- tional confirmation before activating the cage mechanism.

*D. Connectivity Modules ( ESP8266 )*

Facilitates communication by sending notifications through the internet (via Wi-Fi) or SMS (via GSM) in areas with limited connectivity.

Fig. 2. PIR Motion Sensor

Fig. 3. IR Sensor

*E. Microcontroller or Microprocessor (Raspberry Pi, Arduino, or ESP32)*

Acts as the central processing unit for managing sensor inputs, controlling actuators like the servo motor, and handling communication modules.

## V. SOFTWARE COMPONENTS

*A. Raspbian Operating System*

Serves as the lightweight and optimized operating system for the Raspberry Pi, managing hardware and software inter- actions.

*B. Python Programming Language*

Handles image recognition, sensor controls, communication protocols, and integration of machine learning models.

*C. TensorFlow or PyTorch Frameworks*

Enables deployment of models for detecting and recogniz- ing animals, utilizing advanced algorithms for object detection.

*D. Pre-trained Models (YOLO)*

Provides ready-to-use object detection models, ensuring efficient and accurate identification of animals.

*E. Message Sending APIs (Twilio, Firebase Cloud Messaging)*

Sends alerts or notifications about trapped animals through SMS, push notifications, or emails.

Fig. 4. Esp8266 Wifi Module

Fig. 5. ESP32-C3 Microcontroller

*F. Database (Firebase Realtime Database)*
Stores animal data logs and syncs information across de- vices, enabling real-time monitoring and analysis.

## VI. IOT AND DEVELOPMENT TOOLS

*A. Cloud Integration (AWS, Google Cloud)*
Enables remote monitoring and scalability by connecting the system to the cloud for data storage and analysis.

*B. Arduino IDE*
Serves as the programming environment for developing and uploading code to microcontrollers like Arduino and ESP32.

*C. PyCharm or Visual Studio Code*
Acts as the primary development environment for writing and debugging Python code and other application logic.

*D. Dashboards (ThingsBoard, Grafana)*
Provides graphical interfaces for real-time monitoring of animal activity and system performance.

*E. SimulationTools(TinkercadorProteus)*
Simulates electronic circuits and components, allowing test- ing and validation of designs before actual implementation.

## VII. SYSTEM WORKFLOW OVERVIEW

*A. System Initialization*
Power on the system using rechargeable lithium- ion batter- ies. Activate sensors (PIR and ultrasonic) and the Raspberry Pi camera module. Establish a Wi-Fi connection for real-time data exchange with the cloud.

Fig. 6. Arduino Nano Microcontroller

Fig. 7. Raspberry Pi

*B. Motion Detection*
PIR motion sensors detect any movement within the desig- nated area. Ultrasonic sensors confirm the size of the moving object to ensure it matches the target criteria, reducing false activations.

*C. Image Capture*
The Raspberry Pi camera captures high-resolution images or video of the detected object for further analysis.

*D. Species Identification*
In the species identification phase, the system processes the captured media—such as images or videos—using a YOLO deep learning model. This model is specifically trained to rec- ognize and classify various animal species with high accuracy in real time. Once the species is identified, the system evaluates whether it matches the predefined target. If it does, the system transitions to the trap activation phase. Otherwise, the system disregards the detection and continues monitoring for relevant activity, ensuring non-target animals are not unnecessarily affected.

*E. Trap Activation*
The actuator triggers the smart cage to safely trap the identified target species without causing harm.

### *F. Notification Dispatch*

Alerts are sent to wildlife officials in real time via SMS (Twilio) and push notifications (Firebase Cloud Messag- ing).Emails (SMTP).

## VIII. SYSTEM ARCHITECTURE AND IMPLEMENTATION

### *A. System Architecture*

The system architecture is designed with multiple modules that work together to provide real-time sign language transla- tion. The flow starts with the camera capturing video input, which is processed by the hand detection and tracking module. After recognizing and classifying the gestures, the system converts the output into text and speech. The architecture ensures smooth integration between these components with minimal latency.

### *B. Implementation Details*

1) Camera Setup: A camera captures live video feed, and OpenCV is used to process each frame.
2) Hand Detection and Tracking: MediaPipe detects hand landmarks and tracks the hand's motion in real-time.
3) Gesture Classification: A pre-trained CNN- LSTM model classifies the hand gestures based on the captured data.
4) Text Generation: The recognized gestures are translated into text using a Transformer- based model.
5) Speech Synthesis: The generated text is converted into speech using the Pyttsx3 library, ensuring audible output for the user.

The system is designed to be highly efficient, with minimal processing time between the input gesture and the output response, ensuring near real-time performance.

## CONCLUSION

This research presents the EcoTrap system, an intelligent wildlife detection and management system that uses real-time photography and deep learning models, such as in the YOLO IoT sensor, to track the identification and tracking of target species.

The system demonstrates a high level of species identifica- tion accuracy with real-time alerts to wildlife watchers. To make wildlife management and conservation a reality. The system works well in a controlled environment. But they are generally not sensitive to discriminating animal behavior. Brightness classification and real-time performance on high- definition video streams Habitat diversity and habitat diversity Using advanced models that can improve detection accuracy And the use of adaptive features across sites to improve analysis and performance of survivability behavior across categories will enhance the future. These will help ensure that EcoTrap remains a reliable and adaptable solution. Flexible for wildlife conservation.

## REFERENCES


[1] Aniket Gat, Hrishikesh Gaikwad, Rahul Giri. *Animal Detector System for Forest Monitoring Using OpenCV and Raspberry-Pi*, 2022.

[2] Mengyu Tan, Wentao Chao, Jo-Ku Cheng. *Animal Detection and Clas- sification from Camera Trap Images Using Different Mainstream Object Detection Architectures*

[3] Tushar Atkare, Snehal Khalkar, Pallavi Mandekar, Ganesh Kolhe, Anita Gawali. *Wild Animal Detection in Farm using Image Processing and Machine Learning*,2021.

[4] Dr. V Nagagopiraju, Suvarna Pinninti, Anjamma Tamma. *Advanced Wild Animal Detection and Alert System Using YOLOv*,2024.

[5] Brahm Davea, Meet Moria, Anurag Bathania, Parth Goela. *Wild Animal Detection Using YOLOv8*,2023..

[6] Nathan van Dijk. *Advanced Wildlife Camera Trapping Using Embedded AI Machine Vision*,2023.

[7] Balakrishnan, Ramalingam, Rajesh Elara Mohan, Sathian Pookkuttath. *Remote Insects Trap Monitoring System Using Deep Learning Frame- work and IoT*,2020.

[8] Matthew J. Darr. *Advanced Embedded Systems and Sensor Networks for Animal Environment Monitoring*,2007.