AI-DRIVEN WILDLIFE DETERRENCE AND MOVEMENT MONITORING SYSTEM

A PROJECT REPORT

Submitted by

AARIFF M

BERBIN JOE J

MOHAMED AQEEL M

THARUN ATITHYA B

in partial fulfillment of the requirement
for the award of the degree

of

BACHELOR OF TECHNOLOGY

In

ARTIFICIAL INTELLIGENCE AND DATA SCIENCE

EXCEL ENGINEERING COLLEGE
(An Autonomous Institution, affiliated to Anna University Chennai and Approved by AICTE, New Delhi)

KOMARAPALAYAM – 637303

APRIL 2025

EXCEL ENGINEERING COLLEGE (AUTONOMOUS)

KOMARAPALAYAM – 637303

BONAFIDE CERTIFICATE

Certified that this Project report titled "**AI DRIVEN WILDLIFE DETERRENCE AND MOVEMENT MONITORING SYSTEM"** is the bonafide work of **AARIFF M (730921201001)**, **BERBIN JOE J (730921201010)**, **MOHAMED AQEEL M (730921201033)** and **THARUN ATITHYA B (730921201058)** who carried out the Project under my supervision. Certified further, that to the best of my knowledge the work reported herein does not form part of any other Project report or dissertation on the basis of which a degree or award was conferred on an earlier occasion on this or any other candidate.

SIGNATURE	**SIGNATURE**
Prof. P. JAYAPRABHA, M.E.,	**Mrs. A.YASMIN, M.E.,**
HEAD OF THE DEPARTMENT	**SUPERVISOR**
ASSISTANT PROFESSOR	ASSISTANT PROFESSOR
Department of Artificial	Department of Artificial
Intelligence and Data science	Intelligence and Data science
Excel Engineering College (Autonomous)	Excel Engineering College (Autonomous)
Komarapalayam - 637303.	Komarapalayam - 637303.

Submitted for the Viva-Voce examinations held on ……………

INTERNAL EXAMINER **EXTERNAL EXAMINER**

DECLARATION

We jointly declare that the Project report on **AI DRIVEN WILDLIFE DETERRENCE AND MOVEMENT MONITORING SYSTEM** is the result of original work done by us and best of our knowledge, similar work has not been submitted to **“ANNA UNIVERSITY CHENNAI”** for the requirement of Degree of Bachelor of Technology in Artificial Intelligence and Data Science. This Project report is submitted on the partial fulfilment of the requirement of the award of Degree Bachelor of Technology.

Signature

AARIFF M

BERBIN JOE J

MOHAMED AQEEL M

THARUN ATITHYA B

Place: Komarapalayam

Date:

ACKNOWLEDGEMENT

We wish to express our sincere gratitude to our honourable chairman **Dr. A.K.NATESAN, M.Com., MBA(NIT)., M.Phil., Ph.D., FTA,** for providing immense facilities at our institution.

We express our heartiest gratitude and thanks to our respected Vice Chairman **Dr.N.MATHAN KARTHICK, M.B.B.S., MISTE., PHIE.,** for providing all the facilities for successfully completing the Project

We are very proudly rendering our thanks to our Principal **Dr.K.BOMMANNA RAJA., M.E., Ph.D.,** for the facilities and the encouragement given by him to the progress and completion of our Project.

We sincerely thank our respected Dean, **Dr. G.M. TAMILSELVAN, M.E., Ph.D.,** for his guidance, encouragement, and support throughout this project.

We proudly render our immense gratitude to the Head of the Department **Prof. P. JAYAPRABHA M.E.,** for her effective leadership, encouragement and guidance in the Project.

We are highly indebted to provide our heartfull thanks to our supervisor **Mrs.A.YASMIN, M.E.,** for her valuable ideas, encouragement and supportive guidance throughout the Project.

We wish to extend our sincere thanks to all faculty members of our Artificial Intelligence and Data Science Department for their valuable suggestions, kind co-operation and constant encouragement for successful completion of this Project.

We take the privilege to record our everlasting and loving thanks to our parents for their kind help and support which render is bringing our project fruitful manner

ABSTRACT

This project presents an AI-Driven Wildlife Deterrence and Movement Monitoring System aimed at protecting agricultural fields from wild animal intrusions while promoting sustainable and non-invasive wildlife management. The system employs a custom-trained YOLOv11 deep learning model to detect wild animals in real-time. Upon detection, the center of the animal is calculated, and a pan-tilt mechanism controlled by two servo motors aligns to point an ultrasonic emitter toward the animal. This ultrasound deters the animal without causing harm, encouraging it to retreat from the field. The system also establishes a virtual boundary, and if the animal crosses this threshold, an alert is immediately sent to local officials via the MQTT protocol. Movement data is logged for further analysis of migration patterns from forest reserves to human settlements, supporting long-term wildlife monitoring. The system is designed using low-cost, energy-efficient components powered by renewable energy sources, making it suitable for rural deployment. This intelligent solution ensures effective crop protection while supporting ethical wildlife conservation practices.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF ABBREVIATIONS

CSS	-	Casading Style Sheet
GPS	-	Global Positioning System
HTML	-	Hyper Text Markup Language
MQTT	-	Message Queuing Telemetry Transport
UI	-	User Interface
YOLO	-	You Looking Only Once

CHAPTER 1
INTRODUCTION

1.1 Background and Motivation

In recent years, human-wildlife conflict has intensified, particularly in regions where agricultural fields border forest reserves. Wild animals such as elephants, wild boars, and deer often stray into these fields in search of food, leading to significant crop damage, financial losses for farmers, and sometimes even endangering human lives. Traditional methods like electric fencing, scarecrows, firecrackers, or manual patrolling are either harmful to animals, ineffective over time, or labor-intensive and expensive.

With the rapid advancement of artificial intelligence, the integration of smart surveillance and autonomous systems provides a new opportunity to address this issue in an efficient, ethical, and scalable manner. This project is motivated by the need to develop a low-cost, eco-friendly, and intelligent system that not only detects and deters wildlife intrusions in real-time but also helps monitor their movement patterns for long-term wildlife conservation and human safety. By leveraging AI, IoT, and ultrasonic deterrence, the system aims to create a sustainable solution for protecting both crops and wildlife.

1.2 Problem Statement

Farmers living near forest areas frequently face severe crop damage due to the intrusion of wild animals such as boars, elephants, and deer. These animals often venture into farmlands in search of food, leading to significant agricultural loss, damage to property, and potential harm to human lives. Existing deterrent methods such as fencing, noise-based systems, and manual intervention are either costly, harmful to wildlife, or prove ineffective over time as animals adapt to them.

Moreover, there is a lack of intelligent monitoring systems that can provide real-time alerts to local authorities and collect data for studying animal movement patterns. This absence of proactive and eco-friendly solutions results in poor wildlife management and ineffective protection of agricultural zones.

Therefore, there is a critical need for a smart, affordable, and sustainable system that can detect, deter, and monitor wild animals while ensuring minimal environmental impact and promoting human-wildlife coexistence.

1.3 Objective of the Project

The primary objective of this project is to design and implement an AI-driven, cost-effective, and eco-friendly wildlife deterrence and monitoring system to protect agricultural fields from wild animal intrusions, while promoting ethical and sustainable wildlife management.

The specific objectives include:

- Real-time detection of wild animals using a custom-trained YOLOv11 deep learning model.
- Tracking and alignment of a pan-tilt mechanism using two servo motors to point toward the detected animal.
- Safe deterrence through the use of ultrasonic sound waves that repel animals without causing harm.

- Monitoring of animal proximity using ultrasonic distance sensors to assess approach behavior.
- Alert generation via the MQTT protocol when an animal crosses a defined virtual boundary.
- Data logging to study animal movement patterns and support long-term wildlife behavior analysis.

Through these objectives, the project aims to ensure agricultural protection, reduce human-wildlife conflict, and support intelligent conservation practices.

1.4 Significance of the Study

This study holds significant value in addressing one of the most pressing challenges in rural and agricultural communities—human-wildlife conflict. By integrating artificial intelligence, automation, and IoT technologies, the project provides an innovative and ethical approach to protecting crops from wild animal intrusions. Unlike traditional methods, this system offers a non-lethal, sustainable, and real-time solution that benefits both humans and wildlife.

The use of ultrasonic deterrence ensures animals are repelled without physical harm, promoting humane wildlife management. Simultaneously, the system collects movement data, offering insights into animal behavior and migration patterns, which can support forest officials and researchers in planning better conservation strategies.

Additionally, the low-cost hardware components such as the ESP8266, servo motors, and ultrasonic sensors make the system affordable and scalable, making it ideal for deployment in under-resourced farming communities. Overall, this study contributes to agricultural sustainability, rural safety, and environmental conservation through intelligent technology.

1.5 Scope of the Project

The scope of this project encompasses the design, development, and testing of an AI-driven system that detects, tracks, and deters wild animals from entering agricultural fields. The system integrates multiple components including a custom-trained YOLOv11 model for real-time animal detection, a pan-tilt mechanism for physical tracking, ultrasonic deterrence for safe repulsion, and an ESP8266 microcontroller for wireless communication.

Key features covered in the project scope include:

- Detecting animals using a camera and YOLOv11 deep learning algorithm.
- Calculating the animal's position and aligning the servo motors accordingly.
- Using ultrasonic waves to deter animals upon detection.
- Monitoring the distance between the animal and the field boundary using an ultrasonic sensor.
- Sending real-time alerts via MQTT protocol to notify local authorities when a boundary breach occurs.
- Powering the system using a DC-DC buck converter for regulated, efficient operation.

However, the project does not involve physical fencing or long-range animal tracking using GPS. It is focused on localized, automated monitoring and deterrence using low-cost, energy-efficient components.

1.6 Overview of Methodology

The methodology of this project involves a multi-stage integration of artificial intelligence, embedded systems, and IoT communication to create a responsive wildlife deterrence and monitoring system.

1. **Data Collection and Model Training:** A dataset of commonly intruding animals is collected and annotated. YOLOv11, a deep learning object detection model, is trained to accurately identify animals in real-time.
2. **Real-Time Detection and Tracking:** A live video feed is processed using the trained YOLOv11 model. Once an animal is detected, the system calculates its center coordinates and aligns a pan-tilt mechanism using two 180-degree servo motors.
3. **Ultrasonic Deterrence:** When an animal enters a predefined detection zone, an ultrasonic emitter is activated to safely repel it using high-frequency sound waves.
4. **Distance Monitoring:** An ultrasonic sensor continuously measures how close the animal is to the field. If it crosses the virtual boundary, it's considered a threat.
5. **Alert System via MQTT**: The ESP8266 microcontroller sends alerts to local authorities or farmers using the MQTT protocol, enabling timely intervention.
6. **Power Supply Management:** A DC-DC buck converter ensures safe and stable voltage supply to all components, making the system efficient and field-ready.

CHAPTER 2
LITERATURE SURVEY

2.1 Existing Solutions

Several conventional and modern techniques have been used to prevent wild animals from entering agricultural fields. However, many of these methods have limitations in terms of effectiveness, cost, sustainability, and ethical considerations.

1. **Manual Surveillance and Patrolling**
   Farmers or forest guards often patrol the fields to drive animals away. While somewhat effective, this approach is labor-intensive, inconsistent, and not feasible for large areas or during nighttime.
2. **Electric Fencing**
   Electric fences are widely used to deter animals, but they are expensive to install and maintain. They also pose risks to wildlife, sometimes causing injuries or fatalities, raising ethical and legal concerns.
3. **Scare Devices (Sound/Light-Based)**
   Devices such as firecrackers, loudspeakers, and flashing lights are used to scare animals. However, animals can adapt to these methods over time, rendering them ineffective.
4. **GPS Tracking and Camera Traps**
   These are useful for monitoring animal movement but are passive tools that do not actively prevent intrusion. They also require high-cost infrastructure and expertise.

These limitations highlight the need for a smarter, automated, and humane solution—motivating the development of the proposed AI-based system.

2.2 The Evolution of Attendance Technology

Let’s look at the evolution of technology in this space:

- Paper-based roll calls: Time-consuming and prone to errors.
- Excel-based management: Some automation but still requires manual input.
- RFID & Biometric systems: More secure but not contactless. High setup costs.
- Mobile apps with QR/Location tracking: Better but often unreliable indoors.
- Facial recognition systems: Emerging as the most promising due to contactless, real-time, and reliable verification.

Educational institutions across countries like Japan, the US, and India have started experimenting with facial recognition. However, widespread adoption is limited due to privacy concerns, accuracy issues, and cost of commercial software.

2.3 Comparison with Our System

Our system differs in significant ways:

- Built using open-source Python libraries, minimizing cost.
- No need for expensive hardware – only a webcam is needed.
- Contactless and hygienic, addressing COVID-era concerns.
- Incorporates a chatbot for enhanced accessibility.

- Customizable UI built with HTML/CSS, easy to adapt for different institutions.

Compared to RFID systems:

- We eliminate card-swapping misuse.
- No risk of forgetting the ID card.

Compared to fingerprint systems:

- No physical touch needed.
- No additional biometric hardware required.

Compared to GPS-based attendance:

- Not location-reliant, hence works indoors with poor GPS signals.
- Doesn't consume battery life excessively like continuous tracking apps.

2.3 Proposed Work

The proposed work aims to develop an innovative, AI-driven wildlife deterrence and movement monitoring system that offers an effective, humane, and sustainable solution to the problem of wild animal intrusion in agricultural fields. The system integrates several cutting-edge technologies, including object detection, automated tracking, ultrasonic deterrence, and real-time alerting.

Key Components of the Proposed System:

1. **AI-Based Animal Detection**

   The system uses a custom-trained YOLOv11 deep learning model to detect animals in real-time through a camera feed. YOLOv11's high accuracy and efficiency make it ideal for processing images in real-time, enabling quick detection and tracking.

2. **Pan-Tilt Mechanism**

   Upon detecting an animal, the system utilizes a pan-tilt mechanism with two 180-degree servo motors to track the animal's movement. This dynamic tracking ensures that the animal is continuously monitored without human intervention.

3. **Ultrasonic Deterrence**

   Once the animal is detected within a predefined proximity, an ultrasonic emitter is activated to emit high-frequency sound waves that deter the animal. These ultrasonic waves are designed to be uncomfortable for animals but harmless, ensuring humane deterrence.

4. **Distance Monitoring**

   An ultrasonic distance sensor is used to measure the proximity of the animal to the field boundary. The system triggers an alert when the animal crosses the virtual boundary, enabling real-time responses.

5. **Real-Time Alerts via MQTT**

   The system utilizes the MQTT protocol to send alerts to local authorities or farmers via the ESP8266 microcontroller when an animal breaches the boundary. This allows for immediate action and intervention.

6. **Data Logging and Analysis**

   The system logs animal movement data to help understand migration patterns, providing valuable insights for long-term wildlife management and conservation efforts.

The proposed system combines AI, IoT, and automation to create a scalable, low-cost, and eco-friendly solution that ensures the protection of crops while fostering wildlife conservation.

2.4 Use Cases from Reserve Forests

The integration of AI-driven wildlife deterrence systems in reserve forests and their surrounding agricultural zones can significantly enhance both crop protection and wildlife management. The following use cases highlight the practical applications of the proposed system in real-world scenarios:

1. Buffer Zone Monitoring

Reserve forests often have buffer zones where wildlife activity is high. Installing this system along the forest boundary enables real-time detection and tracking of animals such as elephants, wild boars, or deer that may attempt to enter farmland, ensuring quick deterrence and early warning

2. Migration Path Observation

During seasonal migrations, animals often follow natural paths that intersect with human settlements. The system can track migration patterns using logged movement data, helping forest officials take preventive actions or reroute migration safely without human-wildlife conflict.

3. Early Warning to Local Communities

When an animal breaches a virtual boundary, the system can instantly send MQTT-based alerts to local farmers and authorities, allowing communities to prepare and prevent damage without panic.

4. Ethical Wildlife Control

Unlike electric fences or traps, this system provides non-lethal deterrence using ultrasonic sound, aligning with forest department ethics and conservation policies.

2.5 Research Insights

The development of AI-driven wildlife deterrence systems draws from interdisciplinary research in the fields of artificial intelligence, embedded systems, wildlife ecology, and sustainable agriculture. Several key insights support the relevance and feasibility of the proposed system:

1. Effectiveness of AI in Real-Time Detection

Studies have shown that deep learning models such as YOLO (You Only Look Once) can accurately detect and classify animals in real-time, even under varying lighting and environmental conditions. This makes them highly suitable for outdoor wildlife monitoring applications.

2. Behavioral Response to Ultrasonic Waves

Research in animal behavior indicates that ultrasonic frequencies (typically above 20 kHz) are effective in disturbing and repelling certain species like wild boars and deer without causing physical harm. This insight supports the use of ultrasonic deterrents as a humane method of protection.

3. Importance of Movement Data in Conservation

Tracking the movement of wildlife can help identify frequent intrusion zones, migration paths, and seasonal patterns. This data is vital for forest officials and ecologists to make informed decisions on wildlife conservation and land use planning.

4. Role of IoT and MQTT in Remote Monitoring

MQTT (Message Queuing Telemetry Transport) is a lightweight and efficient protocol for real-time communication in IoT systems, especially in rural and low-bandwidth areas. This ensures timely alerts with minimal data usage.

These research insights validate the technical and ethical foundation of the proposed project and its potential for real-world deployment.

2.6 Role of YOLOv11

Key Contributions of YOLOv11:

1. **Real-Time Animal Detection**
   YOLOv11 processes camera input and detects wild animals with minimal delay, allowing the system to respond instantly when an intrusion occurs.
2. **High Accuracy in Diverse Conditions**
   The model is trained on a custom dataset of target species, enabling it to recognize animals under various lighting, weather, and

background conditions commonly found in forest and field environments.

3. **Lightweight and Edge-Compatible**

   YOLOv11 is optimized for performance on low-power devices, making it compatible with systems running on microcontrollers or compact processing units without the need for cloud computing.

4. **Detection with Localization**

   YOLOv11 provides bounding box coordinates that help calculate the center point.

CHAPTER 3
TECHNICAL INSIGHTS

3.1 Tools and Technologies Used

The development of the Attendance Tracking System using Facial Recognition involved a blend of technologies carefully chosen for performance, simplicity, and compatibility. The following tools were used:

- Python – Core language for backend development and facial recognition logic.
- Face recognition (Python Library) – Used for detecting and recognizing faces in real-time using machine learning models.
- OpenCV – Assists in camera integration and real-time image processing.
- HTML & CSS – For building a lightweight, responsive front-end.
- Flask – Python web framework used to connect the front end with the backend functionalities.
- SQLite – Chosen for lightweight and quick data storage.
- Chatbot Framework (Rasa or similar) – Integrated to provide an interactive chat experience for users to retrieve attendance data.

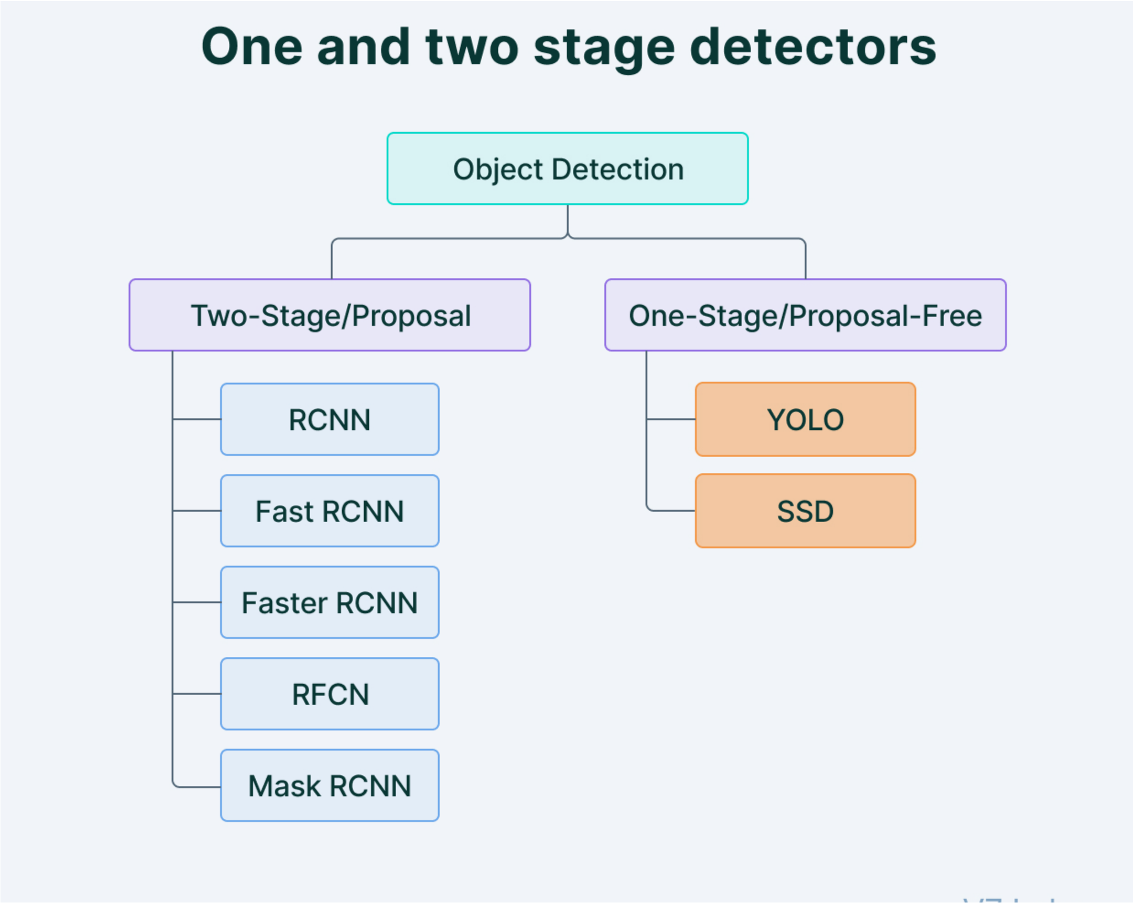
These tools were selected to create a low-cost, high-efficiency solution that doesn’t require heavy hardware or deep technical expertise to operate. 3.1 Custom Trained YOLOv11 Animal Detection

YOLOv11 (You Only Look Once, Version 11) is a powerful deep learning-based object detection algorithm known for its speed and accuracy, making it ideal for real-time applications. In this project, YOLOv11 is custom-trained to detect specific wild animals—such as elephants, wild boars, and deer—that commonly intrude into agricultural fields near forest reserves.

The training process began with the collection of a diverse dataset of wildlife images sourced from open databases and real-world captures. These images were annotated using tools like LabelImg or Roboflow, where each animal was labeled with bounding boxes and class identifiers. The dataset was then used to train YOLOv11 using transfer learning techniques to improve performance even with limited data.

The model was trained to recognize animals under different lighting conditions, poses, angles, and backgrounds to improve robustness. Post-training, the YOLOv11 model is capable of performing real-time object detection with high precision and low computational load, making it suitable for deployment on edge devices. The detected object’s bounding box coordinates are then used for tracking and deterrence mechanisms, forming the foundation of the system's intelligent response.

3.2 Real-Time Object Detection



Real-time object detection is a critical feature of the proposed AI-driven wildlife deterrence system, enabling instant identification and response to potential animal intrusions. This functionality is achieved by deploying the custom-trained YOLOv11 model on a processing unit capable of handling video input from a live camera feed.

As the system continuously captures video frames, each frame is passed through the YOLOv11 algorithm, which analyzes the image and identifies the presence of animals by drawing bounding boxes around them along with class labels and confidence scores. The high-speed processing capability of YOLOv11 allows it to detect multiple animals simultaneously with minimal latency, making it well-suited for real-world field environments.

3.3 Pan-Tilt Servo Motor Animal Tracking

To ensure continuous focus on a detected animal, the system integrates a pan-tilt mechanism controlled by two 180-degree servo motors. These servos enable horizontal (pan) and vertical (tilt) rotation of the camera, allowing it to dynamically follow the movement of the animal within the camera's field of view.

Once an animal is detected using YOLOv11, the system calculates the center point of the bounding box generated by the model. This center point serves as the reference for tracking. The difference between the bounding box center and the frame center is translated into servo angle adjustments, aligning the camera towards the moving animal in real-time.

The tracking logic is implemented in a lightweight embedded script that runs on a microcontroller (such as ESP8266), ensuring low power consumption and fast actuation. This real-time tracking enables accurate targeting of the ultrasonic deterrent and also improves detection persistence, ensuring the system doesn't lose sight of the animal as it moves.

3.4 Ultrasound-Based Animal Deterrence

The system incorporates ultrasonic sound technology as a non-lethal and eco-friendly method to deter wild animals from entering agricultural fields. Ultrasonic waves—typically above 20 kHz—are inaudible to humans but can be highly discomforting to many animal species such as wild boars, deer, and elephants.

Once an animal is detected and tracked by the YOLOv11 model and pan-tilt mechanism, the system checks its distance using an ultrasonic distance sensor. If the animal crosses a pre-set virtual boundary or enters a critical proximity zone, the system triggers the ultrasonic emitter. This device emits high-frequency sound pulses that create an aversive stimulus for the animal, encouraging it to retreat without causing harm.

Unlike electric fencing or physical barriers, ultrasonic deterrents are safe, ethical, and sustainable, making them ideal for use near reserve forests. The ultrasonic component is only activated when needed, preserving energy and reducing the risk of animals becoming desensitized to the sound over time.

This mechanism adds a crucial protection layer, effectively balancing crop security with wildlife conservation.

3.5 Virtual Boundary Monitoring

Virtual boundary monitoring plays a crucial role in this project by defining the area of protection without the need for physical fences. This system uses a combination of camera-based detection, ultrasonic distance sensors, and logical boundary conditions to identify when a wild animal crosses a predefined "safe zone."

The system defines a virtual perimeter—a set of coordinate limits in the camera’s field of view or a critical distance threshold from the ultrasonic sensor. When the YOLOv11 model detects an animal, the system calculates its position based on bounding box coordinates and/or distance measurements. If the animal’s position violates the set threshold, it is considered a boundary breach.

3.6 MQTT Protocol for Alert Messaging

This message is then subscribed to by local authority dashboards, farmer alert systems, or mobile applications, which notify stakeholders immediately through SMS, app alerts, or automated speaker announcements.

The MQTT protocol's key benefits in this project include:

- Low Power Consumption: Perfect for ESP8266-based microcontrollers running on battery or solar power.
- Fast and Reliable Transmission: Ensures alerts are sent instantly, even over unstable networks.
- Scalability: Multiple devices can publish and subscribe to different topics for managing various zones.
- Integration with IoT Platforms: Compatible with platforms like Node-RED, ThingSpeak, or custom dashboards.

3.7 Power Management and Eco-Friendly Design

In designing this AI-driven wildlife deterrence and monitoring system, one of the primary considerations was sustainability and low energy consumption, especially in remote areas where access to continuous power may be limited. Therefore, power management and an eco-friendly design are integral components of the system.

Low Power Consumption with Efficient Components

The system relies on low-power microcontrollers, such as the ESP8266, which is well-suited for Internet of Things (IoT) applications due to its minimal energy requirements. The use of servo motors with low power ratings ensures that the pan-tilt mechanism does not consume excessive power, even during continuous movement or tracking of animals. The camera and sensor modules are selected for their energy efficiency, allowing them to operate for extended periods without excessive drain on the power supply.

CHAPTER 4

SYSTEM ARCHITECTURE

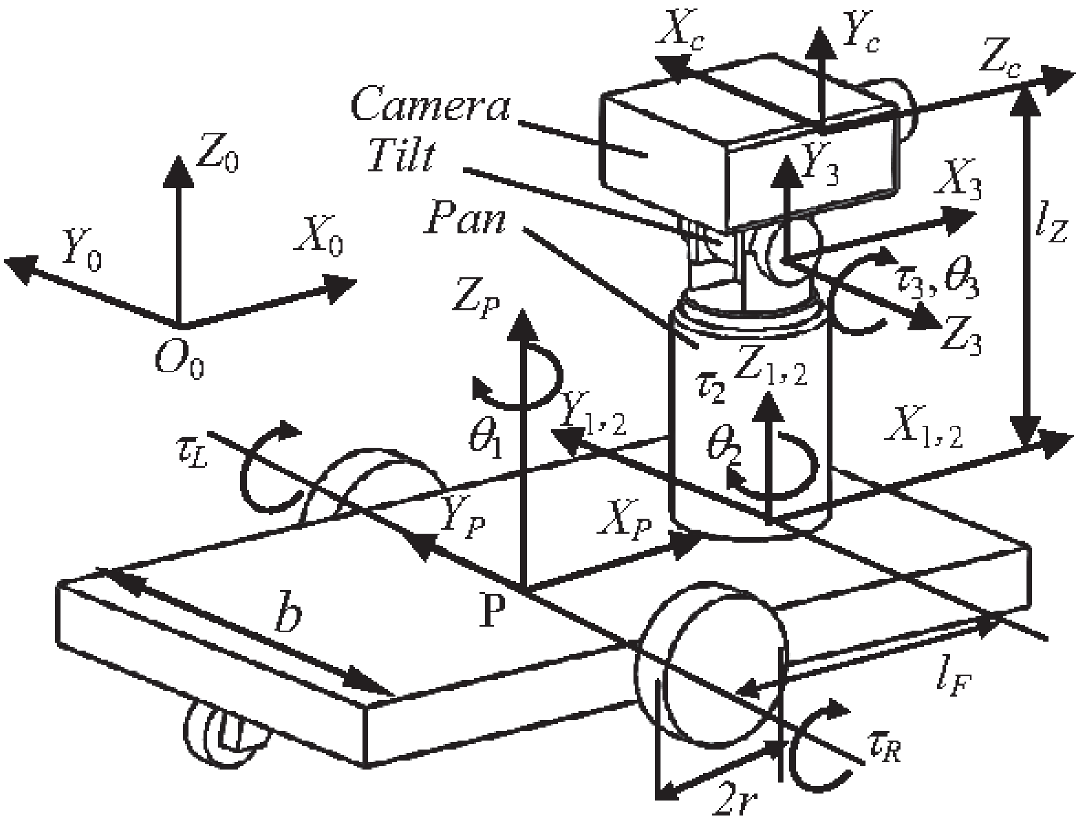
4.1 How It All Connects

The Animal Detection Module is the heart of the system, responsible for detecting and classifying animals in real-time. This module consists of a camera, such as an IP camera or an external camera module, which captures video feed in the environment. The feed is continuously processed using the custom-trained YOLOv11 model. YOLOv11 performs object detection by identifying animals, classifying them into species (e.g., wild boars, elephants, or deer), and generating bounding boxes around detected animals. This real-time object detection ensures that even brief or fleeting intrusions are detected immediately.

The camera module feeds data to an embedded processing unit (like ESP8266) where the YOLOv11 model runs. The processing unit evaluates whether the detected object meets the criteria for being an animal and calculates its position. This information is then passed to the Pan-Tilt Tracking Mechanism for further action.

4.2 Pan-Tilt Tracking Mechanism

The step-by-step workflow of the system:



The Pan-Tilt Tracking Mechanism enables the system to autonomously track animals once they are detected by the animal detection module. This is achieved by using a pan-tilt setup equipped with two servo motors. The servos control horizontal (pan) and vertical (tilt) movements of the camera

4.3 Ultrasonic Distance Measurement Unit

The Ultrasonic Distance Measurement Unit is responsible for determining the proximity of the detected animal to the camera setup. It uses an ultrasonic sensor to send out sound waves that bounce off objects (in this case, animals) and return to the sensor. By calculating the time it takes for the sound wave to return, the system can calculate the distance between the animal and the sensor with high precision.4.4 Deployment and Testing Environment

4.4 Animal Deterrence System

The Animal Deterrence System consists of the ultrasonic emitter, which emits high-frequency sound waves above the human hearing range. Once the system detects an animal within the critical range using the animal detection and ultrasonic sensor modules, the deterrence system activates the ultrasonic emitter.

The sound produced by the ultrasonic emitter causes discomfort to the animal, making it reluctant to stay in the area, thereby encouraging it to move away from agricultural zones. This is a non-lethal, humane deterrent that avoids harming wildlife while effectively protecting crops. The frequency of the ultrasonic waves is carefully selected to ensure it is disruptive to wildlife but not harmful, aligning with ethical wildlife management practices.

4.5 Communication and Alert System

The Communication and Alert System ensures that all relevant stakeholders, such as farmers, wildlife authorities, or conservationists, are promptly informed when an animal is detected and deterrence measures are activated. This is achieved using the MQTT protocol, a lightweight communication protocol ideal for real-time, low-bandwidth environments.

4.6 Power Supply and Regulation Unit

The Power Supply and Regulation Unit ensures that all components of the system operate efficiently and sustainably. The system relies on a solar power setup with solar panels that charge a battery pack, allowing the system to function independently of the main power grid. This is especially important in rural areas where electricity access may be unreliable.

The power regulation unit is designed to ensure that all components, including the camera, servo motors, ultrasonic sensor, ultrasonic emitter, and MQTT communication module, receive consistent and regulated voltage. DC-DC buck converters are used to step down the voltage from the battery to the required levels for each component, optimizing energy consumption. Additionally, the system is designed to automatically enter low-power modes during periods of inactivity, ensuring energy conservation and extending the overall battery life.

CHAPTER 5
FUNCTIONALITIES

5.1 Real-Time Animal Detection and Tracking

1. **Live Video Feed:** The system continuously captures live video using a camera module.
2. **YOLOv11 Object Detection:** The captured video is processed using the custom-trained YOLOv11 model, which detects and classifies wild animals.
3. **Bounding Box Creation:** Once an animal is detected, the model generates a bounding box around the animal with species and confidence scores.
4. **Tracking Mechanism Activation**: The system calculates the animal's center coordinates and activates the pan-tilt servo mechanism to keep the animal centered in the camera's field of view.
5. **Continuous Tracking:** The camera adjusts automatically in real-time to track the animal's movement and ensure it remains in focus.

5.2 Ultrasonic Deterrence

1. **Animal Detection and Tracking:** Once the animal is detected and tracked, the system checks its proximity to the defined virtual boundary.
2. **Ultrasonic Emitter Activation:** If the animal crosses the virtual boundary or approaches too close, the system triggers the ultrasonic emitter.
3. **Sound Emission:** The ultrasonic emitter produces high-frequency sound waves that are inaudible to humans but irritating to animals.

4. **Animal Movement Deterrence:** The sound deters the animal from staying within the area, encouraging it to retreat without causing harm.
5. **Eco-Friendly:** The ultrasonic deterrent is a humane, eco-friendly method that does not harm the animal or the environment.

5.3 Boundary Breach Detection and Alert

1. **Virtual Boundary Setup:** A virtual boundary is defined around the agricultural field using the camera's field of view or distance sensors.
2. **Breach Detection**: Once the animal is detected, its position relative to the boundary is monitored. If it crosses the threshold, it is considered a boundary breach.
3. **Real-Time Detection:** The system continuously monitors animal movement and triggers breach detection immediately upon crossing the boundary.
4. **Alert Messaging:** Upon detection of a breach, the system sends an alert to local authorities or farmers using the MQTT protocol, providing information on the type of animal, location, and breach time.
5. **Proactive Action:** The real-time alert enables immediate action to prevent crop damage and mitigate potential wildlife-human conflicts.

5.4 Data Logging and Wildlife Movement

1. **Data Collection:** The system logs animal movement data, including the time, location, species, and behavior observed.
2. **Continuous Monitoring:** The system records continuous data throughout the day to track wildlife activity patterns and identify frequent boundary breaches.

3. **Real-Time Database Updates**: The logged data is updated in real-time and stored for further analysis or review.
4. **Pattern Analysis:** Historical data helps in analyzing animal movement trends, identifying high-risk periods, and providing insights for improved future management.
5. **Report Generation:** The collected data can be used to generate reports for wildlife conservation efforts and local authorities to make informed decisions on deterrent strategies.

CHAPTER 6
ADVANCEMENT AND IMPACTS

6.1 Advancements in Non-Invasive Wildlife Management

1. **Humane Wildlife Deterrence:** The system employs ultrasonic sound waves to deter animals without causing them harm or distress, offering a non-lethal alternative to traditional deterrent methods like traps or electric fences.
2. **Reduced Wildlife-Human Conflict:** By preventing animals from entering agricultural zones, the system helps to reduce conflicts between wildlife and humans, ensuring that both can coexist peacefully.
3. **Sustainable Management Practices:** The use of eco-friendly methods, like solar power and ultrasonic sound, reduces the environmental footprint of wildlife management solutions and promotes sustainability.
4. **Real-Time Detection and Response:** The ability to detect, track, and deter animals in real-time allows for immediate action, minimizing damage and improving the effectiveness of wildlife management.
5. **Low Maintenance and Cost-Effective:** Unlike physical barriers or chemical deterrents, the system requires minimal maintenance, offering a cost-effective solution for wildlife management in remote areas.
6. **Scalable Solution:** The modular nature of the system allows for easy scalability, enabling its use in diverse wildlife environments, from small farms to large reserves.

6.2 Agricultural Sustainability and Rural Economy

1. **Crop Protection:** The system helps safeguard agricultural fields from wildlife damage, reducing losses due to animal intrusions and ensuring a stable food supply for local communities.
2. **Increased Crop Yield:** By reducing wildlife interference, farmers can experience higher crop yields, contributing to food security and improved livelihoods.
3. **Eco-Friendly Approach to Agriculture:** The integration of solar power and ultrasonic deterrents supports an eco-friendly farming model that aligns with sustainable agriculture practices, reducing dependence on chemical pesticides or harmful physical barriers.
4. **Economic Benefits for Farmers:** With effective wildlife deterrence, farmers can focus on improving productivity and reduce crop loss, leading to improved income stability in rural areas.
5. **Reduced Conflict Costs:** The system helps minimize the costs associated with human-wildlife conflicts, such as damage to crops, property, and the potential risks to human safety.
6. **Job Creation and Technological Adoption:** The deployment of AI-driven solutions like this can stimulate job creation in rural areas by promoting the adoption of modern, technology-driven solutions for farming, boosting the local economy.
7. **Long-Term Sustainability:** By promoting wildlife conservation while supporting agricultural production, the system contributes to the long-term sustainability of both farming and wildlife preservation in rural economies.

CHAPTER 7
CONCLUSION AND FUTURE ENHANCEMENT

7.1 Conclusion

1. **AI-Driven Animal Detection and Tracking:** The system successfully integrates a custom-trained YOLOv11 model for real-time animal detection and tracking, capable of identifying various species in agricultural and forested areas.
2. **Pan-Tilt Mechanism for Tracking:** The pan-tilt servo mechanism enables precise animal tracking by continuously adjusting the camera's orientation, ensuring that the detected animal remains within the field of view at all times.
3. **Ultrasonic Deterrence System:** The project introduced an eco-friendly ultrasonic deterrent to safely and humanely repel animals without causing harm, offering a non-invasive method to protect crops and preserve wildlife habitats.
4. **Real-Time Boundary Breach Alerts:** The system utilizes MQTT protocol to send real-time alerts to local authorities and farmers when an animal crosses the predefined virtual boundary, ensuring immediate intervention to prevent crop damage.
5. **Sustainable Power Management:** The system operates on solar power, making it suitable for rural areas with limited access to electricity. It includes efficient power regulation and energy-saving features, ensuring long-term sustainability.
6. **Data Logging for Wildlife Movement:** The system continuously logs data on animal movement patterns, which can be analyzed to gain

insights into wildlife behavior and optimize future deterrence strategies.

7. **Impact on Agricultural Sustainability**: By reducing crop damage, the system supports sustainable agricultural practices, enhances the economic stability of rural communities, and minimizes human-wildlife conflict.

7.2 Future Enhancements

Advanced Species Recognition and Behavior Prediction:

- Upgrade the YOLOv11 model to recognize not just species but also specific behaviors (e.g., grazing, approaching, retreating), allowing more intelligent responses based on the animal's intent.
- Implement adaptive ultrasonic deterrents that adjust frequency, pattern, or activation based on the specific animal detected, improving deterrence efficiency without habituation.
- Introduce drone-based cameras that can autonomously launch when a boundary breach is detected, providing aerial tracking and extended monitoring capabilities.
- Combine thermal imaging, infrared sensors, and acoustic monitoring with visual detection to improve tracking performance, especially during night-time or foggy conditions.
- Use machine learning algorithms on the logged wildlife movement data to predict future animal paths and optimize boundary placements or deterrent activations preemptively.

APPENDICES

APPENDIX I: SOURCE CODE

```
import streamlit as st
import cv2, time
import pandas as pd
import matplotlib.pyplot as plt
plt.matplotlib.use('Agg')

from entity.ProcessAllAnimal import ProcessAllAnimal
from dao.ChartsImpl import ChartsImpl
from entity.StorePoints import StorePoints

from forStreamLit import getFrameForStreamlit

st.title("AI-Driven Fencing")

charts = ChartsImpl()
points = StorePoints()

imgHolder = st.empty()
cameraHolder = st.sidebar.title("Camera: 1")
locationHolder = st.sidebar.title("South Zone TamilNadu")

escapeCountHolder = st.sidebar.empty()
escapedAnimalHolder = st.sidebar.empty()
```

```
xMovements, yMovements = st.sidebar.columns(2)
with xMovements:
xMetricHolder = st.empty()
with yMovements:
yMetricHolder = st.empty()

fullAnimalDetails = st.sidebar.empty()
warningMessagePlaceHolder = st.empty()

lineChartHolder = st.sidebar.empty()
subPlotHolder = st.sidebar.empty()

old_x, old_y = 90, 90

while True:
img, escapeCount, escapedAnimal, newEscapes, allAnimal, new_x, new_y,
allXpoints, allYpoints = getFrameForStreamlit()

img_rgb = cv2.cvtColor(img, cv2.COLOR_BGR2RGB)

escapeCountHolder.text(f"Escaped: {escapeCount}")

if escapedAnimal is not None:
escapedAnimalHolder.text(f"Escaped Animal Category: {', '.join(str(i) for i
in escapedAnimal)}")

if allAnimal:
```

```
newAllAnimals =
ProcessAllAnimal.processallAnimal(allAnimal=allAnimal)
dfAllAnimal = pd.DataFrame(newAllAnimals)
fullAnimalDetails.table(dfAllAnimal)

if newEscapes:
for i in newEscapes:
with warningMessagePlaceHolder:
st.warning(f"{i[1]} is likely to escaped the marked boundary. {i[2]} +5:30
UTC")

newEscapes = None

if new_x and new_y:
diffX = new_x - old_x
diffY = new_y - old_y

xMetricHolder.metric("X-Movement", f"{new_x}", f"{diffX}",
border=True)

yMetricHolder.metric("Y-Movement", f"{new_y}", f"{diffY}",
border=True)

old_x = new_x
old_y = new_y

lineChartHolder.line_chart(charts.lineChart(), x_label="Frames",
y_label="Distance")
```

```
fig = charts.subPlots()
subPlotHolder.pyplot(fig=fig)
plt.close(fig)

imgHolder.image(img_rgb, channels="RGB")
import time

from dao.EspActivity import EspActivity
from entity.Normalize import Normalize
from entity.ProcessESPData import ProcessESPData

from util.ESP8266Connection import ESP8266Connection

class EspActivityImpl(EspActivity):
def _init_(self):
self.defaultPositionSent = False
self.esp = ESP8266Connection.getConnection()

def espManipulation(self, shortestObjId, closestCoords):
try:
espData = ProcessESPData(esp=self.esp)
distance = espData.process_esp_data()

currentDistance = None

if distance is not None:
currentDistance = distance
```

```
if shortestObjId is not None and closestCoords is not None:
cx, cy = closestCoords
new_x, new_y = Normalize.map_coordinates(cx, cy)
if self.esp:
data = f"{new_x},{new_y}\n"
self.esp.write(data.encode())
self.defaultPositionSent = False

return (new_x, new_y, currentDistance)

except Exception as e:
print(f"Error: {e}")

def skipTime(self, lastDetectionTime):
timeDiff = time.time() - lastDetectionTime

if timeDiff > 3 and self.defaultPositionSent == False and self.esp is not None:
data = "90,90\n"
self.esp.write(data.encode())
self.defaultPositionSent = True
with open("log.txt", 'w') as f:
f.write(f"{self.defaultPositionSent}, {timeDiff}")
return True
import streamlit as st
import matplotlib.pyplot as plt
```

```
plt.matplotlib.use('Agg')
import time
import pandas as pd

from dao.RecieveViaMQTTImpl import RecieveViaMQTTImpl

st.title("MQTT Chart Viewer")

debug = st.empty()
lineChart = st.empty()
subPlot = st.empty()

reciever = RecieveViaMQTTImpl()
reciever.connect()

while True:
lineChartData = reciever.recieveForLineChart()
subPlotData = reciever.recieveForSubPlot()

# Debug information
debug.text(f"Line data type: {type(lineChartData)}, Subplot data type: {type(subPlotData)}")

if lineChartData is not None:
try:
# If it's a DataFrame, use line_chart
if isinstance(lineChartData, pd.DataFrame):
lineChart.line_chart(lineChartData)
```

```
# If it's raw data, convert to DataFrame first
else:
debug.text(f"Converting line data of type {type(lineChartData)} to DataFrame")
except Exception as e:
debug.text(f"Error displaying line chart: {e}")

if subPlotData is not None:
try:
# Make sure we're passing a figure object to pyplot
if isinstance(subPlotData, plt.Figure):
subPlot.pyplot(subPlotData)
plt.close(subPlotData)
else:
debug.text(f"Subplot data is not a matplotlib Figure: {type(subPlotData)}")
except Exception as e:
debug.text(f"Error displaying subplot: {e}")

time.sleep(1)
import cv2, cvzone, time, sys
import numpy as np

sys.path.append(r"C:\Stack overflow\Main-Project_2k25\Development Phase\Review-4\Project")

from dao.DetectObjectsImpl import DetectObjectsImpl
from dao.TrackObjectsImpl import TrackObjectsImpl
from dao.EspActivityImpl import EspActivityImpl
```

```
from dao.SendViaMQTTImpl import SendViaMQTTImpl

from entity.Camera import Camera
from entity.StorePoints import StorePoints

from util.ModelLoading import ModelLoading
from util.TrackerLoading import TrackerLoading

sender = SendViaMQTTImpl()
modelLoad = ModelLoading()
detectObjects = DetectObjectsImpl()
trackObjects = TrackObjectsImpl()
points = trackObjects.all_points
espActivity = EspActivityImpl()

model = modelLoad.loadModel()
tracker = TrackerLoading.loadTracker()
cam = Camera(points.getoriginalWidth(), points.getoriginalHeight())
lastDetectionTime = time.time()

escapeCount = 0
escapedAnimal = None
newEscapes = None
allAnimal = None
new_x, new_y = 0, 0
status = False
pt = StorePoints()
```

```
def getXYpoints():
return allXpoints, allYpoints

def getFrameForStreamlit():
global lastDetectionTime
global escapeCount
global escapedAnimal
global newEscapes
global allAnimal
global new_x
global new_y
global status
global allXpoints
global allYpoints
allXpoints = []
allYpoints = []

img = cam.getFrame()

results = model(img, stream=True)
dets = np.empty((0, 5))
curCls = None

conf, curCls, x1, y1, x2, y2, w, h, cx, cy, dets, lastDetectionTime = detectObjects.forLoopResults(results=results, dets=dets, curCls=curCls, lastDetectionTime=lastDetectionTime)

resultTracker = tracker.update(dets=dets)
```

```
if len(resultTracker) > 0:

x1, y1, x2, y2, w, h, cx, cy, new_x_, new_y_, escapedAnimal, shortestObjId, closestCoords, minDist, id, dist, trackedObjects, newEscapes, allAnimal = trackObjects.forLoopResults(resultTracker=resultTracker, curCls=curCls)

new_x, new_y, currentDistance = espActivity.espManipulation(shortestObjId=shortestObjId, closestCoords=closestCoords)
points.updatedistanceHisory(currentDistance)
points.updatepixelDistanceHistory(minDist)

for i in trackedObjects:
x1, y1, x2, y2, w, h, cx, cy, id, dist = i
cvzone.cornerRect(img, (x1, y1, w, h), )
cvzone.putTextRect(img, f"conf:{conf}, cls:{curCls}", (max(0, x1), max(30, y2 + 40)), offset=2)
cvzone.putTextRect(img, f"id:{id}, dist:{dist}", (max(0, x1), max(30, y1 - 10)), offset=2)
cv2.circle(img, (cx, cy), 5, (255, 0, 255), cv2.FILLED)
allXpoints.append(pt.getoriginalWidth() - cx)
allYpoints.append(pt.getoriginalHeight() - cy)
sender.sendForLineChart()
sender.sendForSubplot()

escapeCount = points.getEscapeCount()
cvzone.putTextRect(img, f"Cam:1 Object Detected - Escaped: {escapeCount}", (max(0, 10), max(30, 10)), offset=2)
```

```
else:
status = espActivity.skipTime(lastDetectionTime=lastDetectionTime)
cvzone.putTextRect(img, f"Cam:1 No Objects Detected", (max(0, 10),
max(30, 10)), offset=2)

return img, escapeCount, escapedAnimal, newEscapes, allAnimal, new_x,
new_y, allXpoints, allYpoints
from dao.TrackObjects import TrackObjects
from entity.Normalize import Normalize
from entity.StorePoints import StorePoints
from entity.Animal import Animal
from datetime import datetime
import time

class TrackObjectsImpl(TrackObjects):
def _init_(self):
self.all_points = StorePoints()

def forLoopResults(self, resultTracker, curCls):
all_x_points = []
all_y_points = []
escaped_animal = []
trackedObjects = []
shortestObjId = None
minDist = float('inf')
closestCoords = None
id, dist = None, None
```

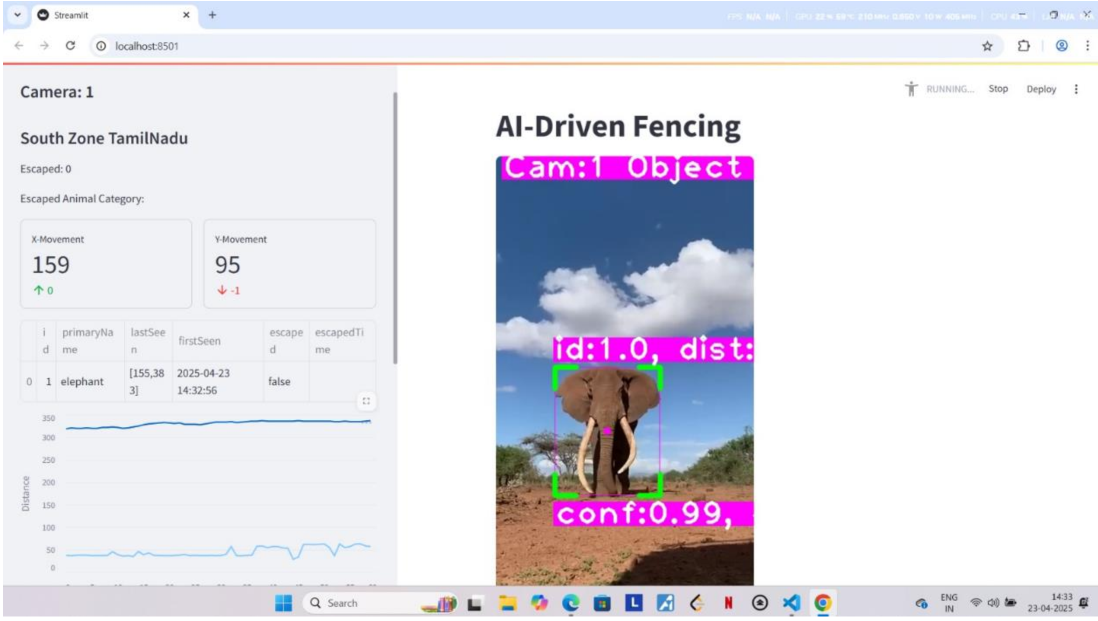
```
x1, y1, x2, y2, w, h, cx, cy, new_x_, new_y_ = None, None, None, None, None, None, None, None, None, None
newEscapes = []

for i in resultTracker:
x1, y1, x2, y2, id = i
x1, y1, x2, y2 = int(x1), int(y1), int(x2), int(y2)
w, h = x2 - x1, y2 - y1
cx, cy = x1 + w // 2, y1 + h // 2
new_x_, new_y_ = Normalize.map_coordinates(x=cx, y=cy, cameraOption=self.all_points.getcameraOption(), originalWidth=self.all_points.getoriginalWidth(), originalHeight=self.all_points.getoriginalHeight())
all_x_points.append(self.all_points.getoriginalWidth() - cx)
all_y_points.append(self.all_points.getoriginalHeight() - cy)

if self.all_points.processAnimal(id, curCls, cx, cy):
now = datetime.now().strftime("%Y-%m-%d %H:%M:%S")
newEscapes.append([id, curCls, now])

dist = self.all_points.getoriginalHeight() - cy

if dist < minDist:
minDist = dist
shortestObjId = id
closestCoords = (cx, cy)

objData = (x1, y1, x2, y2, w, h, cx, cy, id, dist)
```

trackedObjects.append(objData)

escapedAnimal = set([i.primaryName for i in self.all_points.getEscapedAnimals()])
allAnimal = self.all_points.allAnimals()

return x1, y1, x2, y2, w, h, cx, cy, new_x_, new_y_, escapedAnimal, shortestObjId, closestCoords, minDist, id, dist, trackedObjects, newEscapes, allAnimal

APPENDIX II: OUTPUT SCREENSHOTS



REFERENCES

1. Arduino Project Hub. “How to Control a Pan-Tilt System with Arduino.” Accessed April 23, 2025. https://create.arduino.cc/projecthub.
2. Components101. “HC-SR04 Ultrasonic Sensor – Working and Application.”Accessed April 23,https://components101.com/sensors/hc-sr04-ultrasonic-sensor.
3. Espressif Systems. *ESP8266EX Datasheet*. Accessed April 23, 2025. https://www.espressif.com/sites/default/files/documentation/0a-esp8266ex_datasheet_en.pdf.
4. Gonzalez, Luis F., et al. “Unmanned Aerial Vehicles (UAVs) and Artificial Intelligence Revolutionizing Wildlife Monitoring and Conservation.” *Sensors* 16, no. 1 (2016): 97. https://www.mdpi.com/1424-8220/16/1/97.
5. Hart, Josh K., and Kirk Martinez. “Environmental Sensor Networks: A Revolution in the Earth System Science?” *Earth-Science Reviews* 78, no. 3–4 (2006): 177–91. https://doi.org/10.1016/j.earscirev.2006.05.001.
6. MQTT.org. *MQTT Version 3.1.1 Specification*. Last modified 2014. https://mqtt.org/specification/.
7. Redmon, Joseph, and Ali Farhadi. *YOLOv3:An Improvement*. April 8, 2018. arXiv. https://arxiv.org/abs/1804.02767.
8. Saleh, M., and H. Kim. “Solar Energy Harvesting for Low-Power Wireless Sensor Networks.” *IEEE Sensors Journal* 16, no. 10 (2016): 3605–3613. https://ieeexplore.ieee.org/document/7439531.