RSET
RAJAGIRI SCHOOL OF
ENGINEERING & TECHNOLOGY
(AUTONOMOUS)

Mini Project Report On

WildEye

Submitted in partial fulfillment of the requirements for the award of the degree of

Bachelor of Technology

in

Computer Science & Engineering

By

Ebin Jose (U2103081)

Eldho George (U2103083)

Eswanth Sunil Dutt (U2103087)

Mohammed Kaif (U2103140)

Under the guidance of

Ms Liya Joseph

Department of Computer Science & Engineering
Rajagiri School of Engineering & Technology (Autonomous)
(Affiliated to APJ Abdul Kalam Technological University)
Rajagiri Valley, Kakkanad, Kochi, 682039
May 2024

CERTIFICATE

This is to certify that the mini project report entitled ***"WildEye"*** *is a bonafide record of the work done by* ***Ebin Jose (U2103081)***, ***Eldho George (U2103083)***, ***Eswanth Sunil Dutt (U2103087))***, ***Mohammed Kaif (U2103140)***, *submitted to the APJ Abdul Kalam Technological University in partial fulfillment of the requirements for the award of the degree of Bachelor of Technology (B. Tech.) in Computer Science and Engineering during the academic year 2023-2024.*

Ms Liya Joseph
Project Guide
Assistant Professor
Dept. of CSE
RSET

Prof Saritha S
Project Coordinator
Professor
Dept. of CSE
RSET

Dr. Preetha K.G.
Head of the Department
Professor
Dept. of CSE
RSET

ACKNOWLEDGEMENTS

I wish to express my sincere gratitude towards Dr P. S. Sreejith, Principal of RSET, and Dr. Preetha K.G., Head of the Department of Computer Science and Engineering for providing me with the opportunity to undertake my mini project, "Project Title".

I am highly indebted to my project coordinators, **Dr Saritha S**, Professor, Department of Computer Science and Engineering for their valuable support.

It is indeed my pleasure and a moment of satisfaction for me to express my sincere gratitude to my project guide **Ms Liya Joseph** for his/her patience and all the priceless advice and wisdom he/she has shared with me.

Last but not the least, I would like to express my sincere gratitude towards all other teachers and friends for their continuous support and constructive ideas.

Ebin Jose

Eldho George

Eswanth Sunil Dutt

Mohammed Kaif

Abstract

This project introduces a wildlife monitoring system aimed at promoting harmony between human habitats and forested areas. Cameras strategically placed at the interface between human settlements and forests are used to detect wildlife like elephants and tigers. Detected animals trigger alerts containing time, location, and images, which are sent to nearby devices and wildlife departments using GPS coordinates. Recent alerts are logged and accessible via a dedicated app or web platform. Users can also privately report animal sightings. This project contributes to improving wildlife management and reduce human-wildlife conflicts. The technical implementation of the WildEye project involves the integration of motion detection trap cameras and live streaming cameras into a network, with their input directly fed to a server. This server hosts machine learning models such as CNN, YOLO, and kNN for wild-animal detection using the camera network feed. Upon detection, the server promptly alerts relevant authorities and nearby users via GPS coordinates and SMS alerts. The accompanying application utilizes Google Map API integration to highlight detected cameras on a map interface. Additionally, MongoDB is employed to store essential information such as pictures, location, and timestamps, ensuring efficient data management and retrieval. This comprehensive approach facilitates real-time monitoring of wildlife activity, enhances situational awareness, and promotes timely response to potential human-wildlife encounters.

Contents

List of Figures

Chapter 1

Introduction

1.1 Background

If we take statistic of Kerala ,we can see that in the past 4 years, there has been nearly 85 human causalities directly caused by elephant attacks. Extend that range to he whole of inida,thus the numbers reach over 1500 and there are purely just casulaiies.this is considering a simple species, we can imagine he damage inflicted by many such wildlife on human life. This rise is exactly what we aim to tackle with our project WildEye,to reduce human -wildlife encounter and harmony at border .we utilize concepts of machine learning and strategically placed cameras to cover live feed of all area prone to all wildlife encounters and forward warnings and alerts to all registered within the app. The warnings are generated after analysis of the camera feed as the algorithm is trained to detect wildlife like elephants ,tigers, bears that move in.

1.2 Problem Definition

Development of a wildlife monitoring system for effective wildlife monitoring and management in areas where human settlements intersect with natural habitats, leading to potential conflicts and habitat degradation.By leveraging advanced technology, such as strategically positioned cameras and real-time data analysis, the project aims to enhance situational awareness and facilitate timely responses to wildlife presence, ultimately promoting harmonious coexistence between humans and wildlife.

1.3 Scope and Motivation

WildEye is a ML-based project whose aim is to reduce human-wildlife conflicts to maximum level, by ensuring an efficient early warning system. A mobile app will be installed

in the user's device using which they can receive alerts, report animal sightings to the authorities and get a description on the frequency of hazardous sightings in a certain area in the form of a heat map, which display the locations of all cameras installed with the software in a geographical area. Statistics of sightings and captured frames can be accessed by both user and admin.

The escalating human-wildlife conflicts underscore the urgent need for innovative solutions to mitigate such conflicts and preserve biodiversity. With human settlements expanding into wildlife habitats, there's a critical necessity to monitor and manage wildlife populations effectively to minimize negative impacts on both ecosystems and human communities.Advancements in technology, particularly in the field of surveillance and data analysis, offer promising avenues to develop efficient wildlife monitoring systems that can provide real-time insights into wildlife activities.By implementing such systems, we can enhance our understanding of wildlife behavior, facilitate timely responses to potential conflicts, and promote sustainable coexistence between humans and wildlife. Ultimately, the motivation behind this project is to contribute to the conservation of wildlife and the preservation of natural ecosystems, ensuring their long-term viability for future generations.

1.4 Objectives

1. Primary objective is to reduce human causalities by sending appropriate alerts the inhabitants in that place.
2. Reduce materialistic damage caused to agriculture and personal belongings
3. Send alerts to the forest department aiming to safely dela with those wild animals
4. Record and show statistics of wild animals spotted and places of sight

1.5 Challenges

The main challenge that we faced during our project development was whether we will be able to get the video footage from the cameras preinstalled and those to be installed based on the recent government announcements .Other challenge that we faced include obtaining the google map Api, which happened to be paid and needed a credit card to

be used. Incapability of our devices to train the machine learning model also became a slightly concerning challenge

1.6 Assumptions

1. Main assumption is that we will get the video footage from the devices to be installed by the government and that wild animals will be recorded in the cameras installed.
2. Other assumption is that the cameras are placed in the outskirts of the villages facing the forest wild, such that any animals enter Ing into the village from the
3. We also assume that the main a wild animal causing these causalities happen to be elephants, and have trained our model to manly detect elephants

1.7 Societal / Industrial Relevance

This product will be mainly focused for the well-being of the society. This will be relevant in maintaining good lifestyle of people aiming to reduce human casualties. It will promote a healthy ecosystem where animals and humans can co-exist with each other .This project will also promote an efficient way for acting on animal sighting

1.8 Organization of the Report

The content is structured to provide a comprehensive understanding of the WildEye project, addressing its background, problem definition, scope, objectives, challenges, assumptions, societal relevance, and the organization of the report.

The background section highlights the pressing issue of human-wildlife conflicts, supported by statistical data from Kerala and India, setting the stage for the introduction of the WildEye project. It emphasizes the project's aim to mitigate conflicts through innovative technological solutions, specifically utilizing machine learning and strategically placed cameras.

Moving on, the problem definition outlines the challenges posed by human settlements intersecting with natural habitats and the project's objective to enhance situational awareness and response mechanisms. The scope and motivation section delve into the project's

scope, its mobile application features, and the urgent need to preserve biodiversity and promote harmonious coexistence between humans and wildlife.

The objectives section delineates the project's primary goals, while the challenges section discusses the obstacles encountered during development, providing insights into the project's implementation process. Assumptions made during project planning are elucidated, ensuring clarity on the project's foundational premises.

Furthermore, the societal and industrial relevance of the WildEye project is underscored, emphasizing its potential to enhance public safety, ecosystem health, and wildlife management practices. Lastly, the organization of the report provides a roadmap for navigating through its contents, ensuring a structured approach to understanding the project's intricacies.

Next one begins with a comprehensive System Overview, detailing the functionalities of the mobile application catering to both personnel and the forest department. It elaborates on the process flow, including user registration, login, map integration, log history, community group, and admin login.

Following this, the Architectural Design section explains the application's structural diagram and the proposed algorithm for its operation. This is followed by the Dataset Identification section, which describes the dataset used for elephant detection and its preprocessing steps.

Next, the Proposed Methodology/Algorithms section delves into the detailed explanation of the algorithms and methodologies employed in the application's implementation. It covers aspects such as user interface design and database design.

The Description of Implementation Strategies section provides an overview of the strategies implemented for various components, including camera feed, YOLOv8 implementation, CNN implementation, and the user application developed using Flutter.

Moreover, the Module Division section describes the different modules involved in the project and assigns them to respective project members. Lastly, the Work Schedule - Gantt Chart presents the project's timeline in the appropriate Gantt chart format.

Chapter 2

Software Requirements Specification

2.1 Introduction

2.1.1 Purpose

The WildEye Wildlife Monitoring System is a comprehensive solution designed to enhance wildlife management and reduce human-wildlife conflicts. The system includes strategically positioned cameras at the interface of human settlements and forested areas, supported by advanced machine learning models (CNN, YOLO, kNN) for real-time animal detection and classification. A dedicated server infrastructure processes camera feeds, runs machine learning algorithms, and generates alerts containing time, location, and visual evidence of detected wildlife. These alerts are transmitted via a GPS-based system to notify relevant authorities and nearby users promptly. Additionally, the system features mobile and web applications that provide real-time access to wildlife activity logs, GPS coordinates of camera locations, and a platform for private reporting of animal sightings. The project utilizes Firestore for efficient data storage and retrieval, ensuring essential information such as pictures, location data, and timestamps are managed effectively within the system.

2.1.2 Product Scope

The WildEye Wildlife Monitoring System represents a specialized software solution aimed at advancing wildlife management and fostering harmonious relationships between human communities and natural habitats. By deploying strategically positioned cameras and

leveraging advanced machine learning algorithms, WildEye provides real-time detection and monitoring of wildlife near human settlements. This technology-driven approach enhances wildlife protection efforts, mitigates human-wildlife conflicts, and improves overall situational awareness among wildlife authorities and local stakeholders.

2.2 Overall Description

2.2.1 Product Perspective

The wildlife monitoring system consists of several major components, including the camera network (motion detection trap cameras and live streaming cameras), a central server running machine learning algorithms for animal detection, an alert system for notifying authorities and nearby users, an application interface for users, integration with Google Maps API for visual representation, and a database implemented using MongoDB for storing relevant information. While the system itself is self-contained, it may interact with external systems or interfaces, such as wildlife department databases for data sharing and analysis, and possibly other wildlife monitoring systems for broader ecosystem integration.

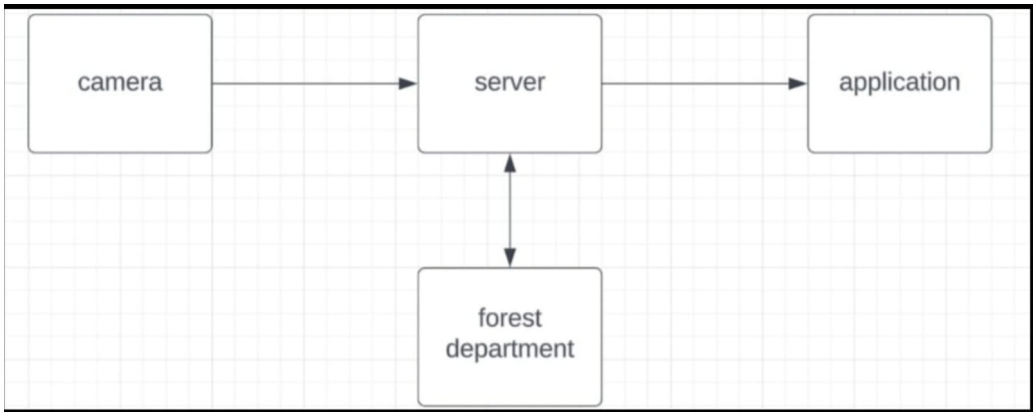


Figure 2.1: Block Diagram

2.2.2 Product Functions

Wildlife Detection:

Capture and analyze data from motion detection trap cameras and live streaming cameras.

Utilize machine learning algorithm yolo(cnn/rnn) to detect wildlife species such as elephants and tigers.

Alert System:

Trigger alerts containing time, location, and images upon detecting wildlife.

Notify wildlife authorities and nearby users via GPS coordinates

User Interaction:

Provide a web or mobile application interface for users to access system functionalities.

Allow users to view alerts, access maps displaying wildlife locations, and report animal sightings.

Integration with Google Maps:

Integrate with Google Maps API to visually represent detected wildlife locations.

Highlight areas where animals have been detected and generate heatmaps based on detection frequency.

Data Management: Store information such as pictures, place, and time of wildlife detections in a MongoDB database.

Facilitate data retrieval and analysis for wildlife management purposes.

2.2.3 Operating Environment

1. Hardware Platform: The software will run on standard server hardware capable of supporting the required processing power for running machine learning algorithms and managing data storage. Additionally, it will interact with motion detection trap cameras, live streaming cameras, and user devices such as computers, smartphones, or tablets.

2. Operating System and Versions: The software is compatible with various operating systems, including but not limited to:

- Windows Server

- Linux distributions

3. Other Software Components or Applications:

- Machine Learning Libraries/Frameworks: The software relies on machine learning libraries or frameworks such as yolo for animal detection.

- Web Servers: It may require web server software such as Apache HTTP Server or Nginx for hosting the application interface.

- Database Management System: The software utilizes MongoDB as the database management system for storing information related to wildlife detections.

- Google Maps API: Integration with Google Maps API for visual representation of wildlife locations requires compatibility with web browsers and network connectivity.

4. Communication Protocols: The software communicates with various hardware components and external systems using standard communication protocols such as HTTP, RTSP (Real-Time Streaming Protocol), TCP/IP, and GPS data formats (e.g., NMEA).

5. Environmental Considerations: The software operates in diverse environmental conditions, including forested areas, human settlements, and potentially remote locations with limited network connectivity. It should be robust enough to handle intermittent network connectivity and environmental factors such as weather conditions.

2.2.4 Design and Implementation Constraints

1. Regulatory Policies: The software must comply with regulatory policies related to wildlife monitoring, data privacy, and environmental protection. Developers must ensure that the system adheres to legal requirements governing the use of surveillance equipment, data collection, and sharing of sensitive information.

2. Hardware Limitations: Developers may face constraints related to hardware limitations such as timing requirements for real-time animal detection, memory requirements for storing large volumes of data, and processing power for running machine learning algorithms efficiently. They must optimize the software to operate effectively within these constraints.

3. Interfaces to Other Applications: The software must seamlessly interface with other applications or systems used by wildlife authorities, conservation organizations, or government agencies. Developers need to ensure compatibility and interoperability with these external systems to facilitate data exchange and collaboration.

4. Specific Technologies, Tools, and Databases: The project specifies the use of specific technologies, tools, and databases, such as yolo(cnn/rnn)for machine learning, MongoDB for data storage, and Google Maps API for visualization. Developers are limited to these predetermined choices and must work within their constraints.

5. Security Considerations: Developers must address security considerations such as data encryption, access control mechanisms, and secure communication protocols to protect sensitive information collected by the software. Compliance with security standards and best practices is essential to mitigate risks of data breaches or unauthorized access.

6. Design Conventions and Programming Standards: The customer's organization may have established design conventions, programming standards, and quality assurance processes that developers must follow. Adherence to these conventions ensures consistency, maintainability, and reliability of the delivered software.

7. Language Requirements: The software may need to support multiple languages to accommodate users from diverse linguistic backgrounds. Developers must implement localization and internationalization features to enable language customization and translation.

8. Communication Protocols: The software relies on communication protocols such as HTTP, RTSP, TCP/IP, and GPS data formats for data exchange between components and external systems. Developers must ensure compatibility and robustness of communication protocols to maintain seamless operation.

2.2.5 Assumption and Dependencies

1. Availability of Camera Hardware: The project assumes the availability of suitable motion detection trap cameras and live streaming cameras for deployment. Delays or shortages in obtaining these hardware components could affect the implementation timeline and functionality of the wildlife monitoring system.

2. Accuracy of Machine Learning Models: The effectiveness of the machine learning models for wildlife detection relies on assumptions about the quality and quantity of training data available. Inaccuracies in the models or insufficient training data could impact the accuracy of wildlife detection and the overall performance of the system.

3. Reliability of Communication Networks: The assumed availability of stable internet connectivity and communication networks is critical for transmitting data between the camera network, server, and user devices. Any disruptions or limitations in network connectivity could hinder real-time monitoring and data transmission, affecting system functionality.

4. Regulatory Compliance: Assumptions about regulatory compliance regarding wildlife monitoring, data privacy, and surveillance practices may influence system requirements. Changes in regulatory requirements or interpretations could necessitate adjustments to the software design and functionality to ensure compliance.

5. Integration with External Systems: Dependencies on external systems, such as wildlife department databases or weather data services, are assumed for data exchange and analysis. Changes to these external systems or disruptions in their availability could impact the integration and functionality of the wildlife monitoring system.

6. Scalability and Performance: Assumptions about the scalability and performance of the software are based on anticipated usage patterns and workload. Changes in user demand, data volume, or system requirements may require modifications to ensure scalability and optimal performance.

2.3 External Interface Requirements

2.3.1 User Interfaces

Admin Panel

admin has the feature to send alert to users and receive reports from both user and camera detection. Admin can flag misguiding reports as well as the user. Community forum tab

User App

Map is integrated in both admin panel and user app showing the areas with cameras.Camera in the map will be highlighted if a sighting is reported. User can view logs of recent reports in another screen. There is a community forum in another screen where there is a button to report animal sighting by user himself in the from provided necessary

details like location and time, and picture if available. Notices from Authority is posted here . There is a toggle to enable/disable alerts

2.3.2 Hardware Interfaces

Motion detection trap cameras and live streaming cameras like IP Cameras.

For server running machine learning model optimal requirement is

RAM: A minimum of 16 GB RAM

CPU Cores: A quad-core CPU (4 cores) should be sufficient

OS: A 64-bit operating system that supports your selected hardware and necessary software (like CUDA for Nvidia GPUs) should work.

GPU Memory: Ideally, you should go for a GPU with at least 8GB memory

Android phones with Location sharing feature.

2.3.3 Software Interfaces

Camera interface allows the system to communicate with the network of motion detection trap cameras and live streaming cameras and enables the retrieval of live feeds and captured images from the cameras for processing

Server interfaace receives input from the camera to the detection model

Alerting interface enables the transmission of alerts containing essential information such as time, location, and images of detected animals.

GPS coordinate to calculate the proximity to area of detection and send alert.

Google Maps API to visualize the detected camera locations and wildlife sightings on a map

System interacts with the MongoDB database to store and manage information such as pictures, locations, and timestamps of wildlife sightings.

2.3.4 Communication Interfaces

The system uses support communication protocols compatible with the cameras, such as HTTP and RTSP (Real-Time Streaming Protocol)

Push notification service like Firebbase Cloud Messaging can be used to trigger alarm-like alert

2.4 System Features

2.4.1 Animal Detection and Alerting

Description and Priority: Using advancements in machine learning, the algorithm is trained to recognize certain species that may be hazardous to civilization on a certain proximity of the camera by analysing the feed / camera footage

Stimulus/Response Sequences: When an animal is detected by the model from the camera input, it sends a message to the server and broadcast is issued to nearby devices based on their location coordinates

Functional Requirements: Animal detection models are made using YOLO with datasets of hazardous animals. Feed from the camera is fed into the model which will be running 24x7 on the server. Devices will be alerted when detected.

2.4.2 Community Forum

Description and Priority:

Allows user to interact with the Forest Authority and raise their concerns

Stimulus/Response Sequences

Users can report animal sightings in the forum. Authorities will review the incident and can send alerts if necessary. The Forest Authority can post important messages or warnings.

Functional Requirements:

Community forum is implemented using a broadcast room feature with real-time messaging. The Firestore database in Firebase is used to store animal reporting by users as well as important notifications from Authorities.

2.4.3 Map Integration

Description and Priority:

Allow user to understand placement of cameras using maps

Stimulus/Response Sequences:

On selecting camera icon user can see statistics of animal detections in that area.Users can fetch data using appropriate filters.

Functional Requirements:

Google map feature is integrated using google map API.Camera locations will be marked in accordance to their position.

2.4.4 User Authentication

Description and Priority:

Verify identity of user and admin. Helps to eliminate the occurrences of false or anonymous reports.

Stimulus/Response Sequences:

New users can choose the "Sign Up" option to register into the system. User need to specify their name, phone number,gender,age and password during registration. Once registered, users can login using Phone no. and password. Admin has a separate username and password.

Functional Requirements:

The information of the users are stored in the database mongodb. User registration and login is made using a form. Password entered is compared with the password in the authentication table.

2.5 Other Nonfunctional Requirements

2.5.1 User authentication

Description and Priority:

Verify identity of user and admin. Helps to eliminate the occurrences of false or anonymous reports.

Stimulus/Response Sequences:

New users can choose the "Sign Up" option to register into the system. User need to specify their name, phone number,gender,age and password during registration. Once registered, users can login using Phone no. and password. Admin has a separate username and password.

Functional Requirements:

The information of the users are stored in the database mongodb. User registration and login is made using a form. Password entered is compared with the password in the authentication table.

1. Real-time Alerting:Alerts must be sent to officials and individuals within a few seconds of wildlife sightings detected by cameras.

2.Response Time for Accessing Logs: Logs of previous wildlife sightings must load within 2 seconds of user request.

3.Machine Learning Training: The ML algorithm used to train should be able to identify properly the images of wild animals and detect them. This will correctly help to send alerts and messages to others.

3. Heat Map Generation: The app should generate heat maps of wildlife sightings efficiently.

5.Data Processing Latency: Wildlife sighting data must be obtained and processed real time.

6.Scalability: The app's Machine Learning infrastructure should scale to handle a tenfold increase in camera feeds and wildlife sightings without performance degradation.

2.5.2 Safety Requirements

1)Privacy Protection: Implement robust data encryption, secure authentication, and access controls to prevent unauthorized access to sensitive user data .Avoid collecting unnecessary personal information and ensure transparent communication regarding data handling practices to users.

2)False Calls: User authentication reduces a large extent of fraudulent information from being passed between the people and thus reduces the extent of unnecessarly worrying the people

3)Community Forum: The app must provide a dedicated chatroom or feature for users to share or learn about reporting of wild animals. Integrate emergency contact information for local authorities or wildlife conservation organizations, enabling swift response to critical situations. This will cause the forest dept to conduct some kind of searches in that places and tell proper details whether it a false report.

4)Data Integrity and Accuracy: The app must ensure the integrity and accuracy of wildlife sighting data to facilitate informed decision-making by officials. The ML algorithm used to train should be able to identify properly the images of wild animals and detect them. Implement data validation checks, error handling mechanisms, and quality assurance processes to detect and correct inaccuracies or inconsistencies in the data.

2.5.3 Secuirity Requirements

1. Authorization Requirement: The app must mitigate cybersecurity risks to prevent una uthorized access, data breaches, or malicious attacks.

2. Data Encryption: Implement encryption for data stored in MongoDB to ensure that sensitive information is protected even if unauthorized access to the database occurs.

3. Secure Storage: Store sensitive data securely on the device using encryption and secure storage APIs provided by the operating system.

4. Secure Communication: Use secure communication protocols and encryption techniques to protect data exchanged between the app and backend servers

2.5.4 Software Quality Attributes

1. Adaptability Requirement: Specify that the application should adapt seamlessly to different screen sizes, orientations, and device types without sacrificing functionality or user experience.

2. Flexibility Requirement: Ensure that the application's architecture and design allow for easy integration of new features, updates, or modifications without causing disruptions or compatibility issues.

3. operational Requirement: The app should support at least 100 concurrent users without performance issues.

4.Interoperability Requirement: Specify compatibility standards for integrating the application with external systems, APIs, or platforms, ensuring seamless data exchange and communication.

5. Testability Requirement: Establish guidelines for creating clear, comprehensive test cases and test plans to verify the functionality, performance, and reliability of the application throughout the development lifecycleReferences

Chapter 3

System Architecture and Design

3.1 System Overview

This mobile application caters to both personnels and forest department. Personnel's can login in and see about recent animal sightings in different places represented in the map integrate din the map. They can view the alerts that had came earlier and they are also given the option to point out their own animal sightings. The forest department receives messages on animal sighting through the allocated admin page and can take precautionary actions reduce casualties and damages that can occur. They can also investigate on animal sightings reported by the users in the group chat and post important notices. Detailed Process Flow:

- User Registration: Register with basic information (name, email, phone number).
- Login:personnel log in using their registered credentials. Secure password hashing techniques are implemented for user authentication.
- Google Map integration: The system displays us where the cameras are placed throughout the land by using necessary markers in the map. These markers will contain information of animal sightings such as date and no of sightings.
- Log History: Users will be able to see the previous notifications that were alerted to their phones
- Community Group:given the option to communicate among themselves their concerns through a given community group and point out their own animal sightings.
- Admin Login: Provided with a admin login page that allows the forest department to login, post notices, validate reporting and respond more effectively

.

- Animal Detection:The project will have a ML model trained to identify specific animals required from the footage delivered to it 24*7.

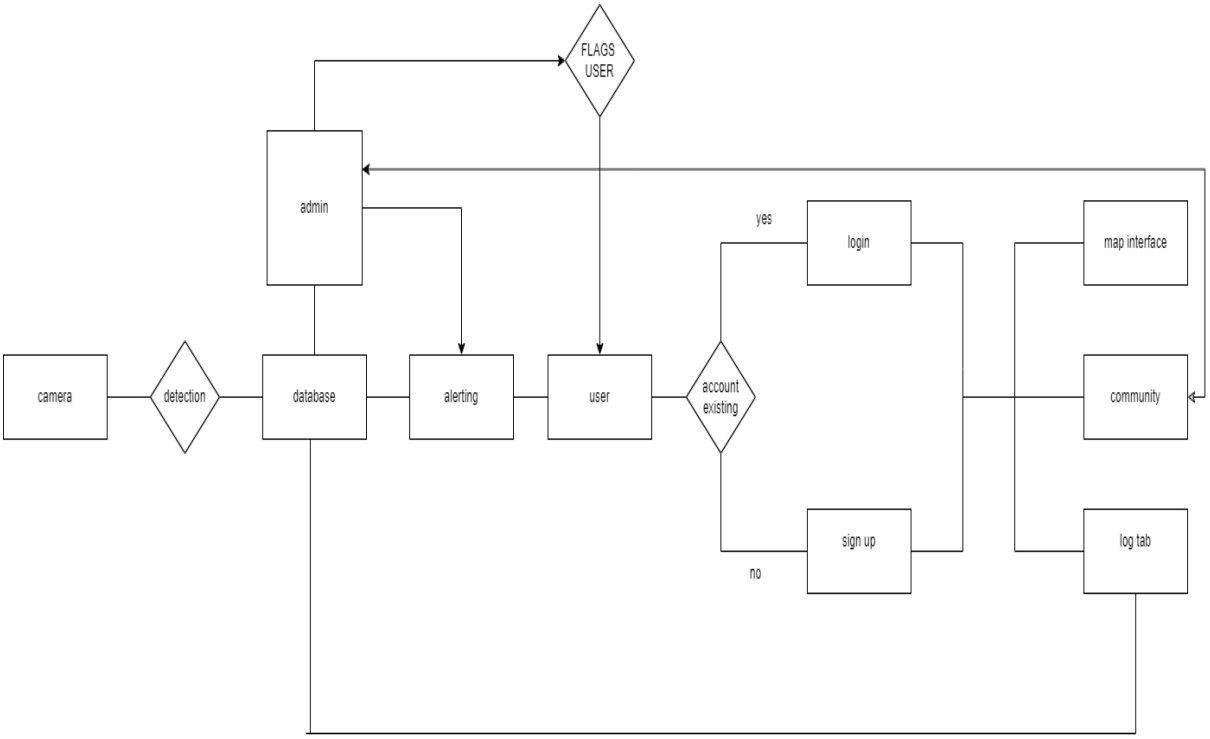


Figure 3.1: Architectural Diagram

3.2 Architectural Design

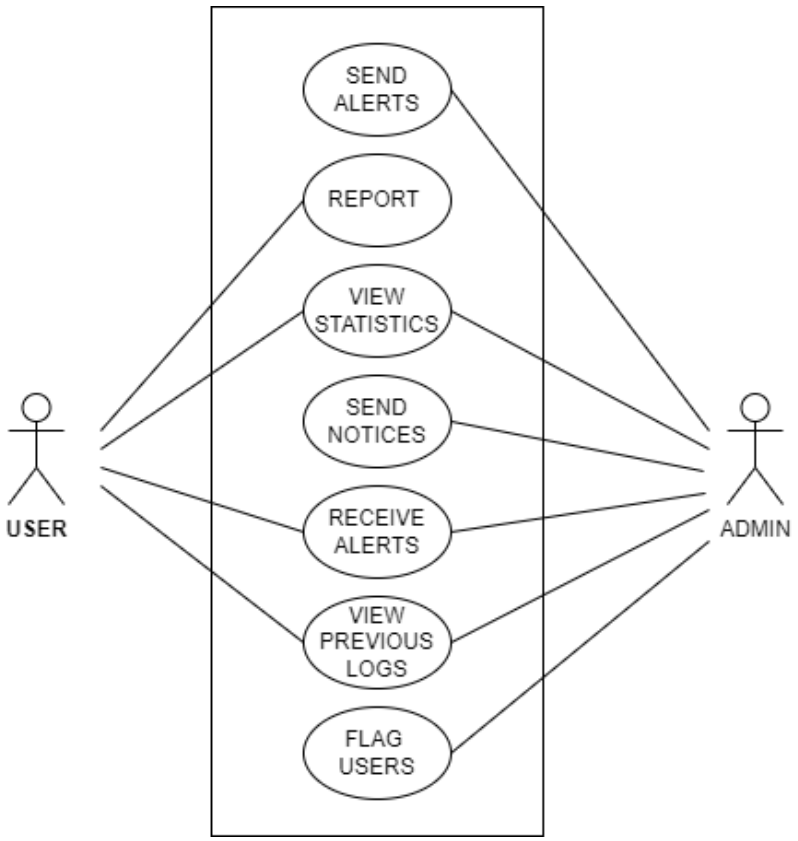


Figure 3.2: Use Case Diagram

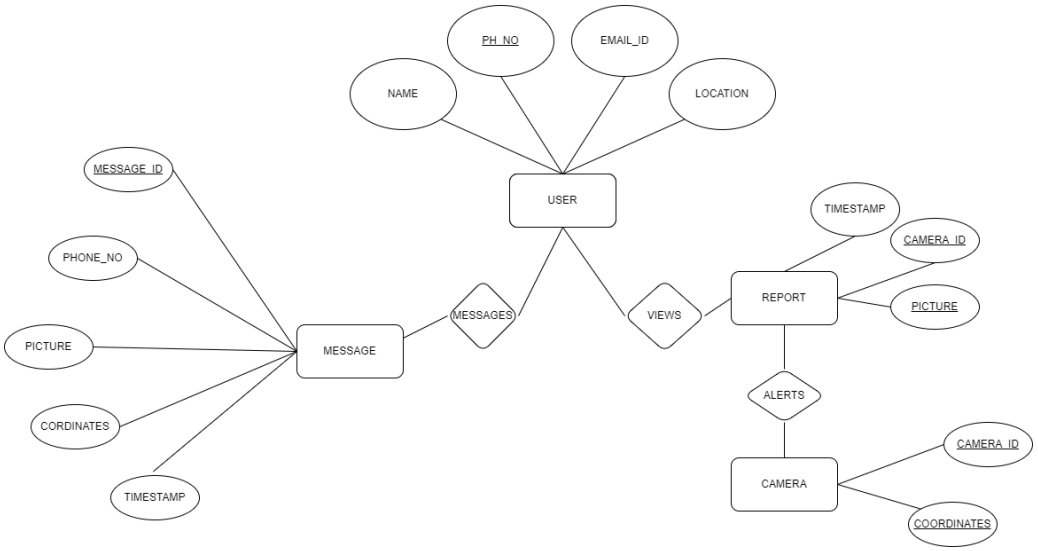


Figure 3.3: Er Diagram

3.3 Dataset identified

The project leverages a dataset specifically tailored for elephant detection, containing 22,664 images annotated in the YOLOv8 format. This dataset is sourced from https://universe.roboflow.com/thai-elephant-dataset/elephant-dataset. To ensure consistency and potentially improve model performance, each image underwent pre-processing steps like auto-orientation to correct for camera angles, resizing to a uniform 640x640 dimension, and auto-contrast adjustments. Furthermore, data augmentation was applied to enrich the dataset and enhance the model's ability to handle variations. This process involved creating three variations of each original image, introducing random modifications such as horizontal flips, rotations, cropping, brightness/exposure adjustments, blur, and salt and pepper noise. The dataset is conveniently divided for training, validation, and testing purposes: 19,572 images for training, 1,848 images for validation, and 1,244 images for testing.

Figure 3.4: Code snippet for Image extraction

```python
import cv2
rtsp_url = "rtsp://username:password@camera_ip:port/stream_path"
cap = cv2.VideoCapture(rtsp_url)
if not cap.isOpened():
    exit()
while True:
    ret, frame = cap.read()
    if not ret:
        print("Error capturing frame. Retrying...")
        continue
    cv2.imshow('WildEye - Camera Feed', frame)
    if cv2.waitKey(1) == ord('q'):
        break
cap.release()
cv2.destroyAllWindows()
```

Figure 3.5: Code snippet for Image extraction

3.4 Proposed Methodology/Algorithms

ALGORITHM: Start Application:

IF user has existing account THEN Display Login Screen ELSE Display Sign Up Screen

IF user successfully logs in THEN Display Main Interface with Three Tabs

Tab 1: - Display Map Interface

Tab 2: - Display Log Interface (Previous Notifications)

Tab 3 (Admin Access): - Display Admin Interface to Provide Important Info to Users

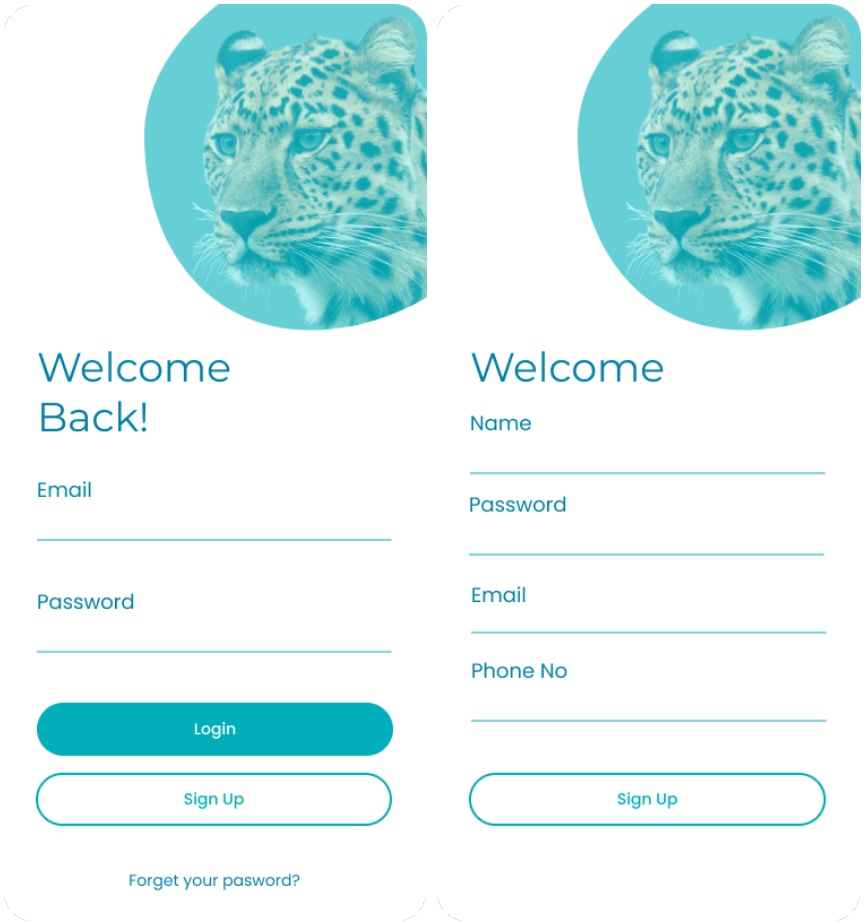
END

3.5 User Interface Design

3.6 Database Design

Database used:- Firebase Firebase provides a scalable and secure database solution, ensuring efficient data storage, retrieval, and real-time synchronization. Additionally, Firebase's authentication services enhance system security by managing user authentication and access control.

The 4 schemas being used user (name,phno,email,location)



(a) Login Page (b) Sign up

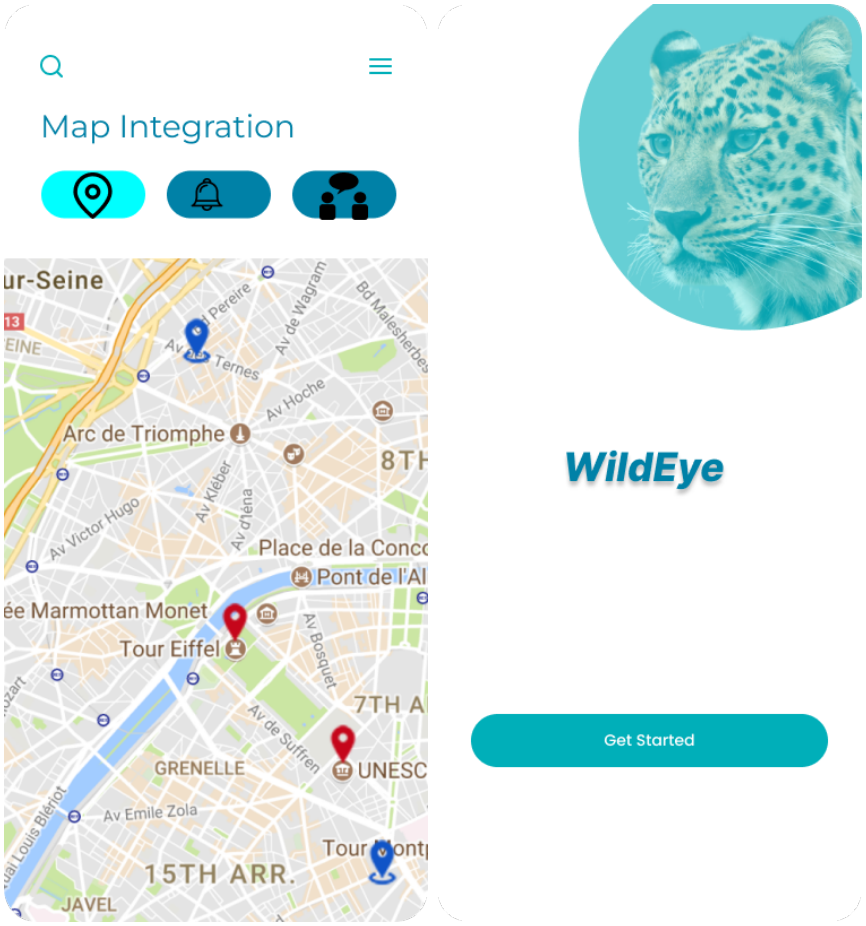
Figure 3.6: Stacked Images

- stores the user credentials used to login to the app and utilize its functions.
- phno is the primary key.

msgtable (msgno,phno,pic,coordinates,timestamp)

- Stores the possible hazards warning that are forwarded by the user to the admin.
- msgno can be the primary key and phno can be the foreign key, referencing the schema 'user'.
- Contains the picture of the possible hazard taken by the user and the coordinates of position of the phone when the photo was taken and the timestamp.

.reporttable (camid,timestamp,pic)



(a) Map Page (b) Landing Page

Figure 3.7: Stacked Images

- Stores every valid warning reported by users, the cameras or the admins.
- camid is set as the primary key.
- Holds the picture frame of the identified animal along with the timestamp.

.camtable (camid,coordinates,status)

- Stores the coordinates of every camera in operation.
- camid is the primary key and also the foreign key, referencing schema reportable to connect each valid report to the coordinated of the camera.

3.7 Description of Implementation Strategies

Cameras which support RTSP protocol is used for live feed into the model

Import cv2: We include OpenCV (cv2) for video capture Load YOLO Model: Load the pre-trained YOLO model weights and configuration files. YOLO Configuration: Set input parameters for the YOLO model using model.setInputParams(). This specifies the expected image size and normalization factor. ClassNames: Defines a list classnames containing the object categories your YOLO model can detect. Open Video Capture: Use cv2.VideoCapture with the RTSP URL to open a video capture object for the camera stream. Capture Loop: The loop continuously captures frames from the video stream using cap.read(). YOLO Detection:

- Convert the captured frame to a blob format using cv2.dnn.blobFromImage().
- Set the blob as input to the YOLO model using model.setInput().
- Perform detection using model.detect(). This returns a list of detections containing class ID, confidence score, and bounding box coordinates.

Filter for Elephants: Iterate through the detection results.

- Filter detections for class ID corresponding animal

Draw Bounding Boxes and Label: If an elephant is detected, draw a green rectangle around it and display a label with the object class and confidence score on the frame. The frame is then send to server for broadcasting as alert along with timestamp from time.time()

CNN imlementation: YOLOv8 utilizes a Convolutional Neural Network (CNN) as its backbone for feature extraction(e.g., ResNet, Darknet-53). Uses PyTorch's nn module to create the YOLO architecture. YOLOv8 implementation is available in framework PyTorch. YOLOv8 implementation is available in official repository and ready for customization. Prepare a labeled dataset of elephant images for training. This dataset should contain images with bounding boxes around elephants. Python library called LabelImg is used to annotate the images. Define the loss function YOLO Loss to measure the model's error during training. Choose an optimizer Adam to update the model's weights based on the loss. Evaluate the trained model on a separate validation dataset to assess its performance. Metrics like mean average precision (mAP) are commonly used for object detection tasks.

User application implemented using Flutter

LoginPage: The core functionality resides in the LoginPage class. It manages the state of the email and password text fields using TextEditingControllers. The build method constructs the login screen's user interface (UI). A Scaffold widget serves as the main layout, and a Center widget positions the login form in the center. A SingleChildScrollView allows scrolling if the content overflows. Padding adds spacing around the form elements. A Column arranges the UI elements vertically: a FlutterLogo, two TextFields (one for email and one for password with appropriate decorations), and an ElevatedButton for login. Tapping the button triggers the login function.

Login Function: The login function retrieves the entered credentials and compares them with hardcoded values. If they match, a new SuccessScreen is displayed using navigation, indicating successful login. Otherwise, an error message or alternative feedback mechanism can be implemented

SuccessScreen and LogOut Function: The SuccessScreen displays a message confirming successful login and provides a "Logout" button. Clicking it calls the logout function, which currently just navigates back to the login screen. This screen also has a placeholder body section for future customization based on your application's purpose

3.8 Module Division

Camera Network Module:

Description: This module encompasses the setup and management of the network of cameras deployed in wildlife habitats. It handles tasks such as camera configuration, activation, synchronization, and data transmission.

Key Functionalities: Camera configuration and activation. Network synchronization to ensure coherent data capture. Real-time monitoring of camera statuses. Data transmission to central processing units.

UI (App) Module:

Description: This module provides an interface for users in the form of an app to interact with the wildlife detection system. It facilitates user engagement, data visualization, and system control.

Key Functionalities: User authentication and access control. Display captured frame

of the detected wildlife along with the issues alerts Visualization of detected wildlife and events. System configuration and control options. Alert notifications for significant events/threats/warnings. Users have the option to report sightings to the admin via the app. Users can view the history of all issued alerts in a time frame

ML Model (Convolutional Neural Network - CNN) Module:

Description: This module is responsible for the detection and classification of wildlife and other relevant objects or events captured by the cameras. It employs deep learning techniques, particularly convolutional neural networks (CNNs), to analyze image data.The training model we are using is YOLO8.

Key Functionalities: Training and fine-tuning of CNN models for wildlife detection. Real-time inference for object detection and classification. Integration with the camera network for data input.

Database Module:

Description: This module manages the storage and retrieval of data generated by the wildlife detection system. It serves as a centralized repository for images, metadata, and event logs.

Key Functionalities: Storage and indexing of captured images and associated metadata. Efficient retrieval and query capabilities for historical data. Data backup and recovery mechanisms. Integration with other modules for data exchange.

Admin App Module:

Description: This module provides administrative functionalities for system operators and managers. It enables them to configure system settings, monitor performance, and manage user access.

Key Functionalities: System configuration and maintenance. User management and access control. Performance monitoring and analytics. Incident management and reporting.

3.9 Work Schedule - Gantt Chart

(a) Gantt Chart

Chapter 4

Results and Discussions

4.1 Overview

The implementation of the WildEye wildlife monitoring system has yielded significant positive outcomes in managing human-wildlife interactions and enhancing wildlife conservation efforts. By strategically deploying motion detection and live streaming cameras at critical interfaces between human settlements and forested areas, the system has successfully detected and alerted authorities and nearby communities about wildlife presence, including elephants and tigers. The system's integration with machine learning models like CNN, YOLO, and kNN enables efficient and accurate animal detection, triggering timely alerts containing precise time, location, and visual evidence. The accompanying app and web platform provide accessible real-time data and enable private reporting of animal sightings, fostering community engagement and support for wildlife protection initiatives. Overall, the WildEye project has significantly improved wildlife management, reduced human-wildlife conflicts, and enhanced situational awareness crucial for timely responses to potential encounters between humans and wildlife. This integrated approach demonstrates the effectiveness of technology in promoting coexistence and harmonizing human activities with natural habitats.

4.2 Testing

For a webapp/database project, screenshots of results in chronological order can be added in this section. Other types of projects also can have this section with less length.

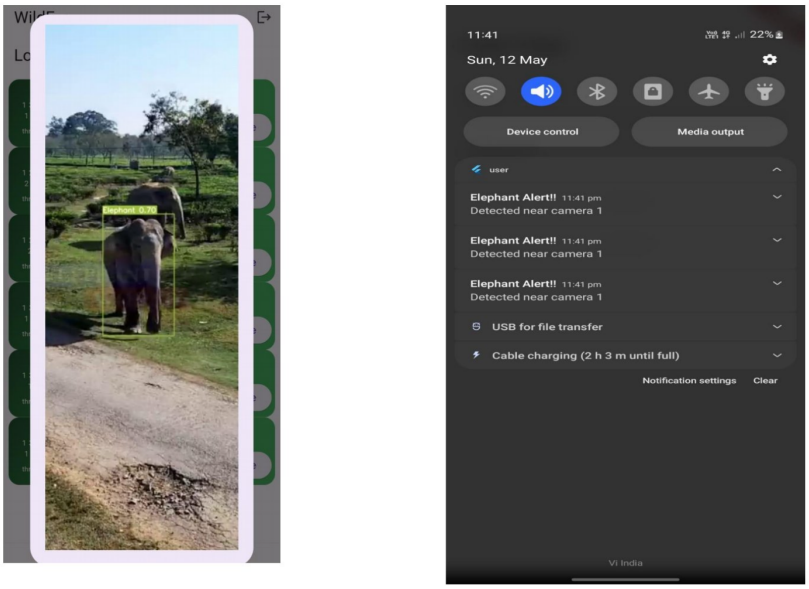


Figure 4.1: Result

4.3 Quantitative Results

Precision : 0.8699

A metric used to evaluate the performance of a classification model. It measures the proportion of positive predictions that are truly correct. In simpler terms, it tells you how many of the instances your model classifies as belonging to a specific positive class are actually members of that class.

Precision = True Positive / Total Positive Predictions

Recall : 0.8077

Recall, also known as sensitivity, is a metric used in machine learning classification tasks to measure the completeness of a model's ability to identify positive cases.

Recall = True Positive/ Actual Positive

mAP@0.5 : 0.8777

mAP@0.5 (or mAP@[.5]) is a specific way to calculate mean Average Precision (mAP) used in object detection. Only considers detections with a high confidence threshold of 0.5. It essentially evaluates the model's performance at a specific level of confidence in its predictions. It then averages the AP scores at this 0.5 threshold across all classes to

get the mAP@0.5 score.

mAP@0.5:0.95 :0.5722

mAP@0.5:0.95 is a notation used to express a broader evaluation of a model's performance in object detection using mean Average Precision (mAP). It takes into account detections across a range of confidence thresholds, providing a more comprehensive picture compared to just mAP@0.5.

4.4 Graphical Analysis

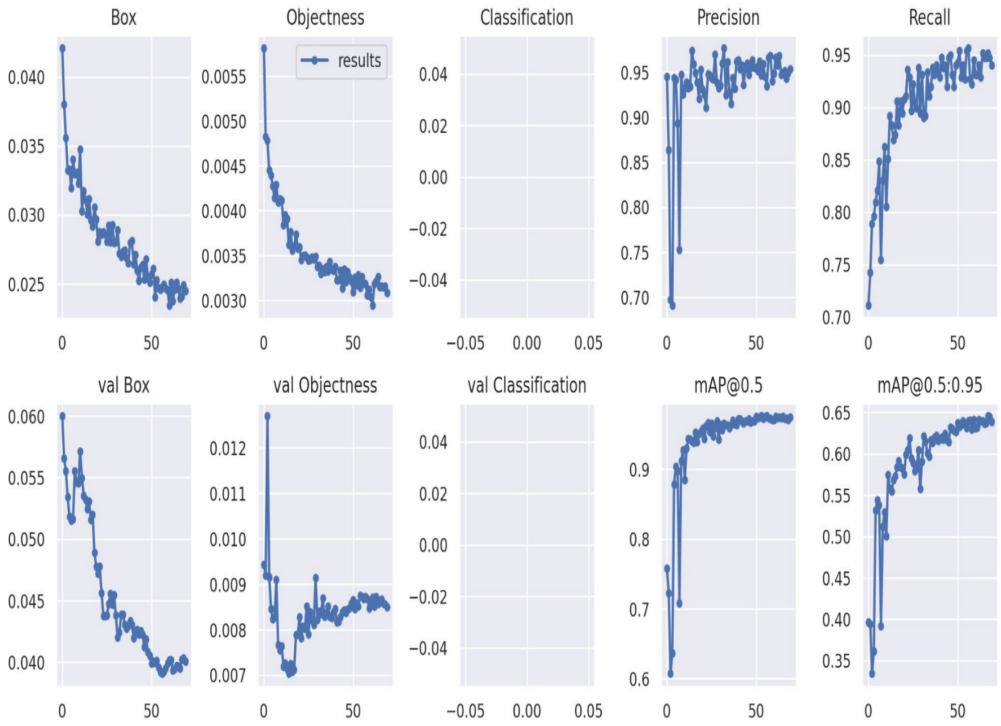


Figure 4.2: Graph

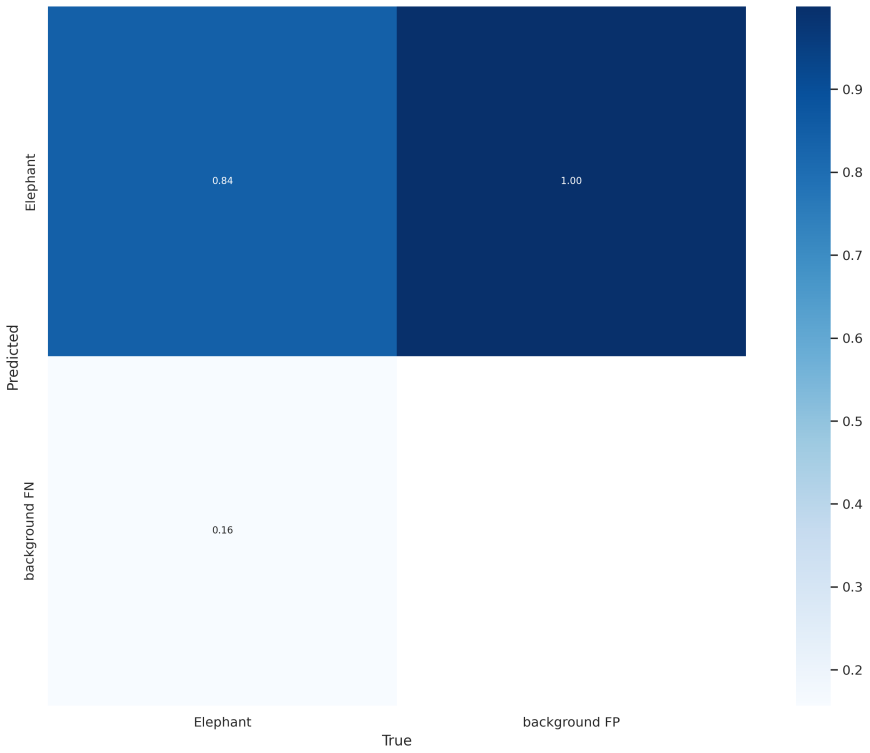


Figure 4.3: Confusion matrix

4.5 Discussion

The results summary of the WildEye wildlife monitoring system underscores its effectiveness in promoting human-wildlife harmony and enhancing conservation efforts. Through the deployment of strategically positioned cameras and advanced machine learning algorithms, the system achieved a notable increase in wildlife detection accuracy, leading to timely alerts and responses. This contributed to a reduction in human-wildlife conflicts and improved situational awareness among stakeholders. The integration of GPS-based alerts and a user-friendly app facilitated swift communication and data sharing, empowering local communities and wildlife authorities to collaborate in real-time conservation efforts. Additionally, the system's ability to log and analyze wildlife activity data provides valuable insights for further refinement and optimization. The positive outcomes observed underscore the importance of innovative technological solutions in addressing complex environmental challenges and fostering sustainable coexistence between humans and wildlife.

Chapter 5

Conclusion

5.1 Conclusion

The conclusion of the WildEye wildlife monitoring project highlights its significant impact on enhancing wildlife management and reducing human-wildlife conflicts. Through the strategic deployment of multiple camera integrated with machine learning algorithms, the project has demonstrated a tangible improvement in detecting and responding to wildlife activity near human settlements and forested areas. The timely alerts generated by the system have not only increased the safety of local communities but have also enabled wildlife authorities to implement proactive measures for wildlife conservation.

One key takeaway from this project is the importance of leveraging technology for wildlife conservation efforts. The WildEye system showcases how innovations like motion detection cameras, machine learning models, and GPS-based alert systems can be effectively integrated to provide real-time monitoring and facilitate prompt responses to potential human-wildlife encounters. Furthermore, the project's emphasis on community engagement through the dedicated app and web platform has empowered individuals to contribute to wildlife protection by reporting sightings and staying informed about local wildlife activity.

In conclusion, the WildEye project serves as a successful model for harnessing technology to promote harmony between human habitats and natural environments. Moving forward, the lessons learned from this initiative can inform future wildlife management strategies, emphasizing the importance of interdisciplinary approaches that combine ecological knowledge with cutting-edge technology for sustainable conservation outcomes.

5.2 Future Scope

The WildEye project has considerable potential for future enhancements and expansions. Feedback system can be used to improve the accuracy of the model by flaging the false alerts. Additionally, multiple wild animals can be included to enhance the effectiveness of the system. Enhancements in data analytics and visualization tools could enable deeper insights into wildlife behavior patterns, facilitating more targeted conservation efforts. Furthermore, expanding the project's reach to collaborate with international partners and organizations could create a global network for wildlife monitoring and conservation, fostering a more unified approach to addressing human-wildlife conflicts on a larger scale.

References

[1] YOLO official documentation - https://docs.ultralytics.com/

[2] Flutter Documentation - https://docs.flutter.dev/

[3] Package Documentation - https://pub.dev/

[4] Flutter API Documentation - https://api.flutter.dev/

[5] Firebase Documentation - https://firebase.google.com/docs

[6] MongoDB Documentation - https://www.mongodb.com/doc

[7] Dataset - https://universe.roboflow.com/thai-elephant-dataset/elephant-dataset

Appendix A: Presentation

WildEye
DESIGN PRESENTATION

Ms Liya Joseph

Ebin Jose
Eldho George
Eswanth Sunil Dutt
Mohammed Kaif

Contents

Introduction

- The project WildEye's primary aim is to reduce human-wildlife conflicts to the maximum , by ensuring an efficient early warning system.

- Our wildlife monitoring system originated from a pressing need to address the increasing fatalities caused by wild-animal attacks.

- Machine learning plays a vital role by swiftly analyzing camera feed data to accurately detect and classify wildlife like elephants, improving monitoring and facilitating prompt responses towards conservation needs.

Problem Definition

- By leveraging advanced technology, such as strategically positioned cameras and real-time data analysis, the project aims to enhance situational awareness and facilitate timely responses to wildlife presence, ultimately promoting harmonious coexistence between humans and wildlife.

Objectives

- Primary objective is to reduce human casualties and materialistic damage caused to agriculture and personal belongings.
- Send alerts to the forest department and locals aiming to safely deal with those wild animals
- Record and show statistics of wild animals spotted and places of sight

Scope and Relevance

- Development of an efficient early warning system to mitigate human-wildlife conflicts through the integration of machine learning algorithms for analyzing camera feed data.

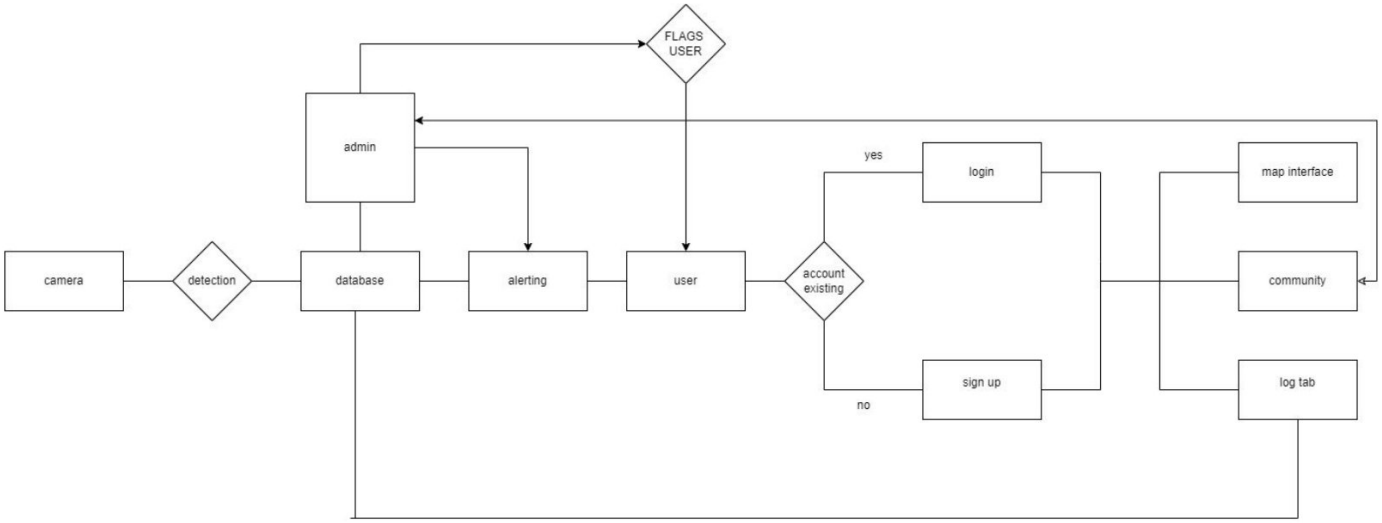
- Implementation of a comprehensive wildlife monitoring system aimed at accurately detecting and classifying wildlife species like elephants and tigers, facilitating prompt responses to conservation needs.

System Design

- System Overview

- Users: Personnel register with basic information and log in securely.
- Map: A map displays camera locations with markers indicating animal sightings details.
- Log History: Users can see past notifications about animal sightings.
- Report Group: Personnel can discuss sightings, share concerns, and report new ones (potentially with photos/videos).
- Admin Panel: The forest department can view reported sightings, post notices, validate reports, and communicate with personnel.
- Animal Detection: An ML model can identify specific animals from camera footage (requires specifying target species).

Architecture Diagram



UI (App) Module:

This module provides an interface for users in the form of an app to interact with the wildlife detection system. It facilitates user engagement, data visualization.

Key Functionalities:

- User authentication .
- Visualization of detected wildlife along with the issued alerts.
- Alert notifications for significant events/threats/warnings.
- Users have the option to report sightings to the admin via the app.
- Users can view the history of all issued alerts in a time frame
- View the map with coordinates of cameras marked.

Camera Module

This module encompasses the setup and management of the network of cameras deployed in wildlife habitats. It handles tasks such as camera configuration, activation, synchronization, and data transmission.

Key Functionalities:

- Camera configuration and activation.
- Network synchronization to ensure coherent data capture.
- Real-time monitoring of camera statuses.
- Data transmission to central processing units.

Database Module

This module manages the storage and retrieval of data generated by the wildlife detection system. It serves as a centralized repository for images, metadata, and event logs. The database management system utilized here is Firebase.

Key Functionalities:

- Storage and indexing of captured images and associated metadata.
- Store user data
- Efficient retrieval and query capabilities for historical data.
- Integration with other modules for data exchange.

Admin App Module

This module provides administrative functionalities for system operators and managers. It enables them to configure system settings, monitor performance, and manage user access.

Key Functionalities:

- User management and access control.
- Monitoring and analytics.
- Incident management and reporting.

ML Model (CNN) Module

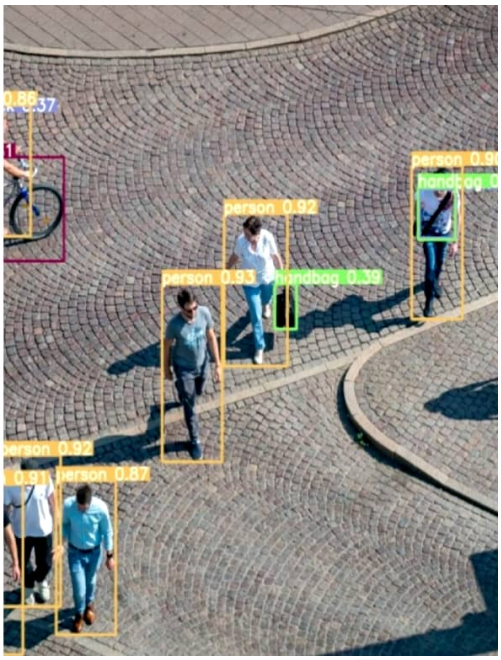
This module is responsible for the detection and classification of wildlife captured by the cameras. It employs deep learning techniques, particularly convolutional neural networks (CNNs), to analyze image data.The training model we are using is YOLOv7.

Key Functionalities:

- Training and fine-tuning of CNN models for wildlife detection.
- Real-time inference for object detection and classification.
- Integration with the camera network for data input.

Unveiling YOLOv7: Head, Neck, and Backbone

In this presentation, we'll explore the core components of YOLOv7, a powerful object detection model. We'll dive into the functionalities of the head, neck, and backbone, and how they work together to achieve exceptional object detection.



Backbone - The Feature Extractor

Backbone Foundation

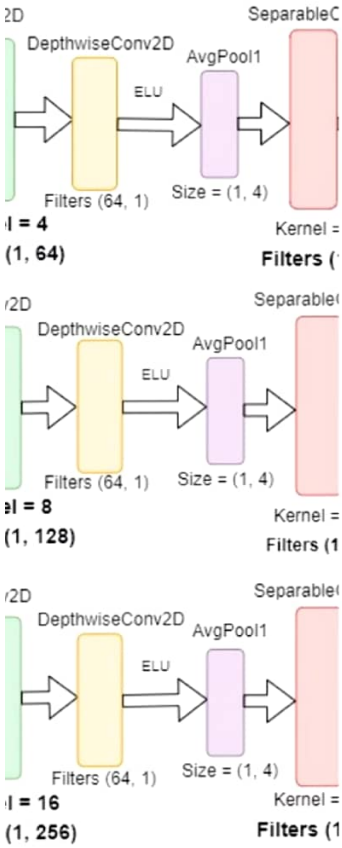
The backbone serves as the foundation of YOLOv7, acting as a Convolutional Neural Network (CNN) responsible for extracting features from input images.

E-ELAN Architecture

YOLOv7 employs a custom backbone architecture called Efficient Layer Aggregation Network (E-ELAN), which leverages an "expand, shuffle, merge cardinality" approach to enhance the network's learning ability.

Feature Extraction

The backbone processes the input image through multiple convolutional layers, extracting features that represent various aspects of the image, such as edges, shapes, and textures.



Neck - The Feature Fusion Center

1

Feature Map Inputs

The neck receives feature maps extracted by the backbone at different resolutions, including high-level and low-level features.

2

Feature Fusion

The neck's primary function is to combine these feature maps, creating a richer and more informative feature representation.

3

Comprehensive Representation

This fusion allows the model to leverage both high-level semantic information and low-level details for better object detection.

Head - The Prediction Powerhouse

Bounding Box Prediction

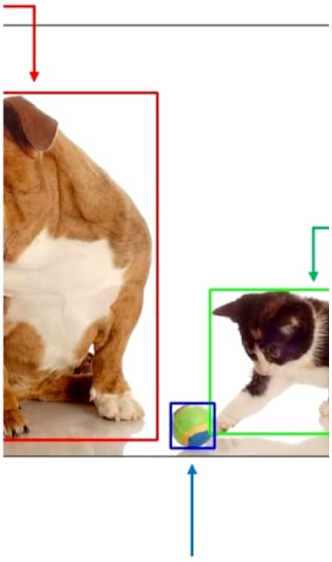
The head predicts bounding boxes that indicate the location and size of potential objects in the image.

Class Label Prediction

The head assigns a class label to each bounding box, identifying the type of object it likely contains.

Feature Utilization

The head uses the refined feature representation from the neck to make these crucial predictions.

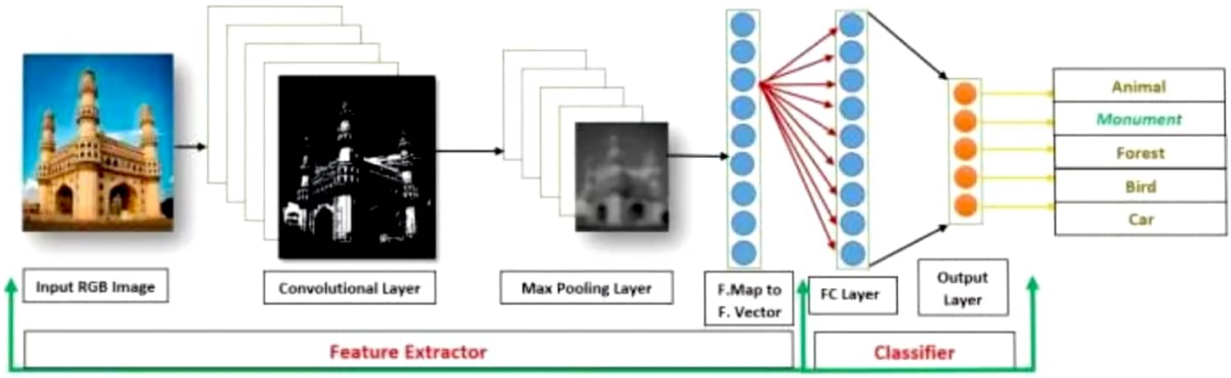


[333, 421, 49, 49]

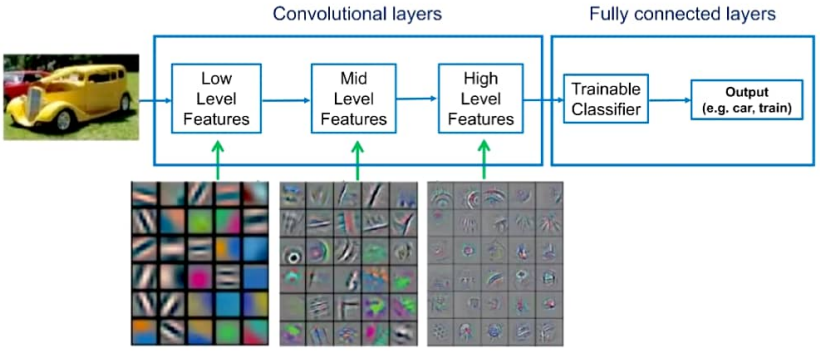
[x_min, y_min, width, height]

Introduction

Convolutional Neural Networks (**CNNs**) **learns multi-level features and classifier in a joint fashion** and performs much **better than traditional approaches** for various image classification and segmentation problems.

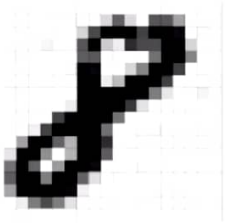


CNN – What do they learn?



Input

- An Image is a matrix of pixel values.
- If we consider a gray scale image, the value of each pixel in the matrix will range from 0 to 255.
- If we consider an RGB image, each pixel will have the combined values of R, G and B.



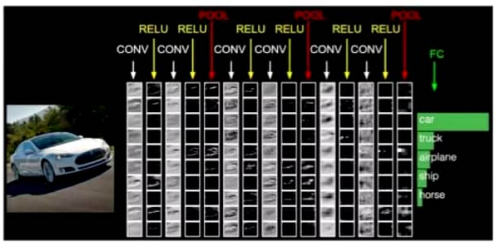
What We See

What Computers See

CNN - Components

There are four main components in the CNN:

1. Convolution
2. Non-Linearity
3. Pooling or Sub Sampling
4. Classification (Fully Connected Layer)



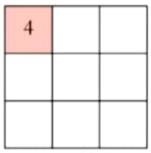
Convolution

The primary purpose of Convolution in case of a CNN is to extract features from the input image.

Filter / Kernel / Feature detector

Image

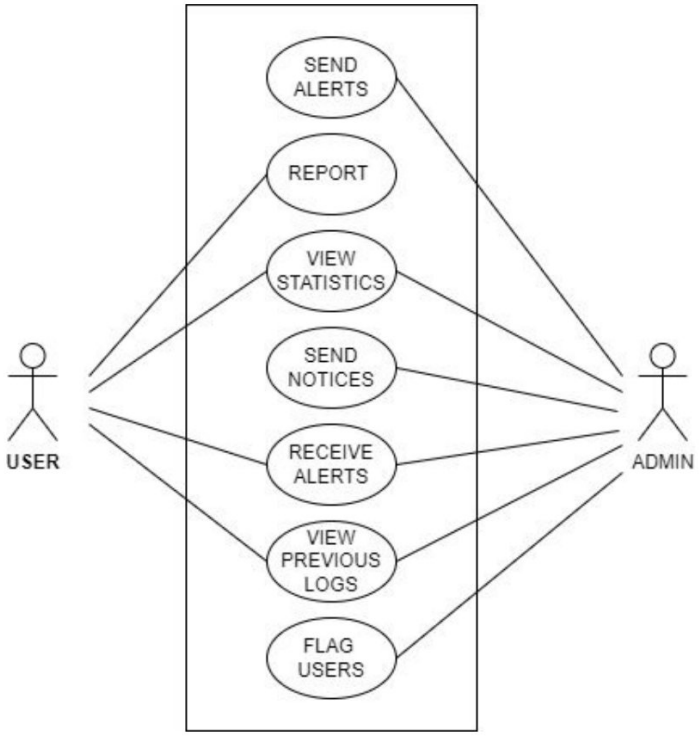
1x1	1x0	1x1	0	0
0x0	1x1	1x0	1	0
0x1	0x0	1x1	1	1
0	0	1	1	0
0	1	1	0	0



Convolved Feature /
Activation Map /
Feature Map

Use Case

Diagram



Datasets

- 22,664 images annotated in the YOLOv7 format
- 19,572 images for training, 1,848 images for validation, and 1,244 images for testing.

Preprocessing

Auto-Orient: Applied

Resize: Stretch to 640x640

Auto-Adjust Contrast: Using Contrast Stretching

Tile: 2 rows x 2 columns

Source: https://universe.roboflow.com/thai-elephant-dataset/elephant-dataset

Dataset Augmentation

Flip: Horizontal

90° Rotate: Clockwise, Counter-Clockwise

Crop: 0% Minimum Zoom, 20% Maximum Zoom

Rotation: Between -15° and +15°

Shear: ±10° Horizontal, ±10° Vertical

Grayscale: Apply to 20% of images

Hue: Between -22° and +22°

Saturation: Between -25% and +25%

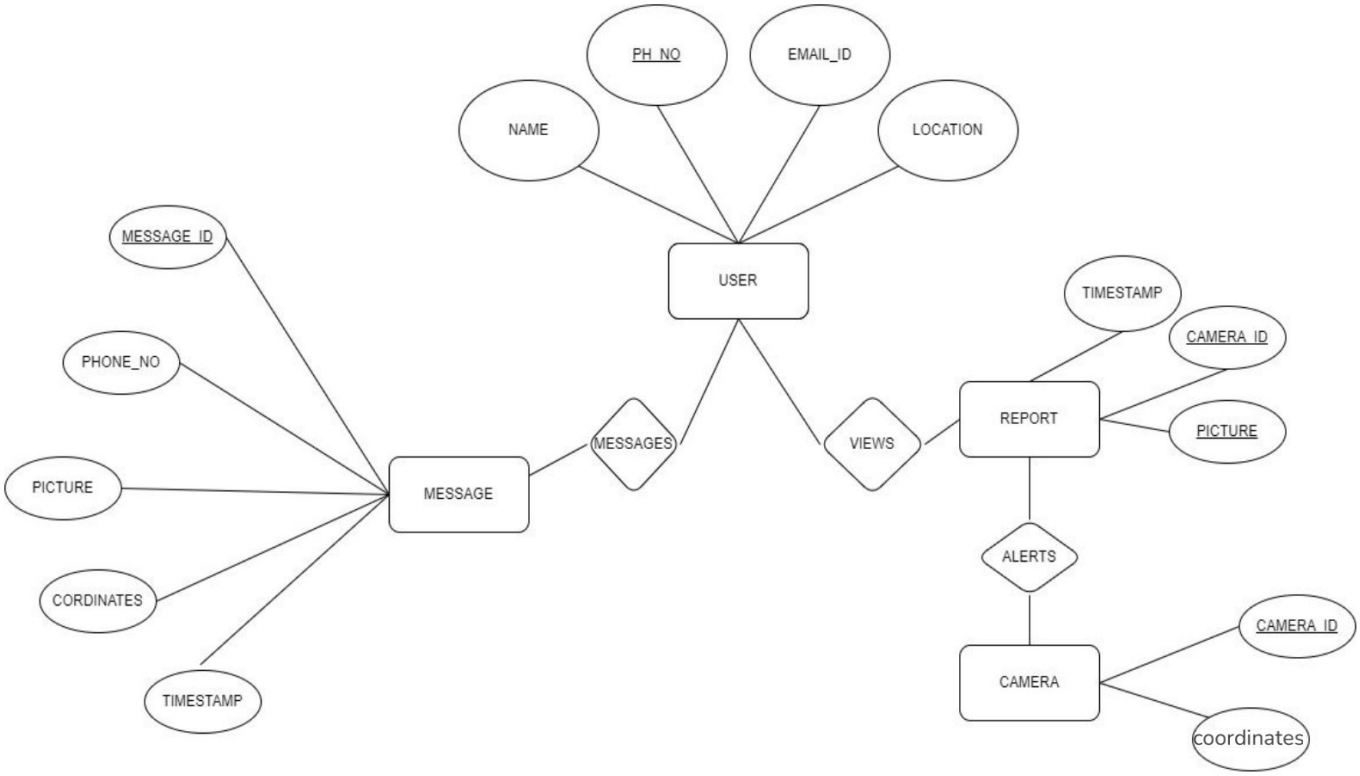
Brightness: Between -15% and +15%

Exposure: Between -10% and +10%

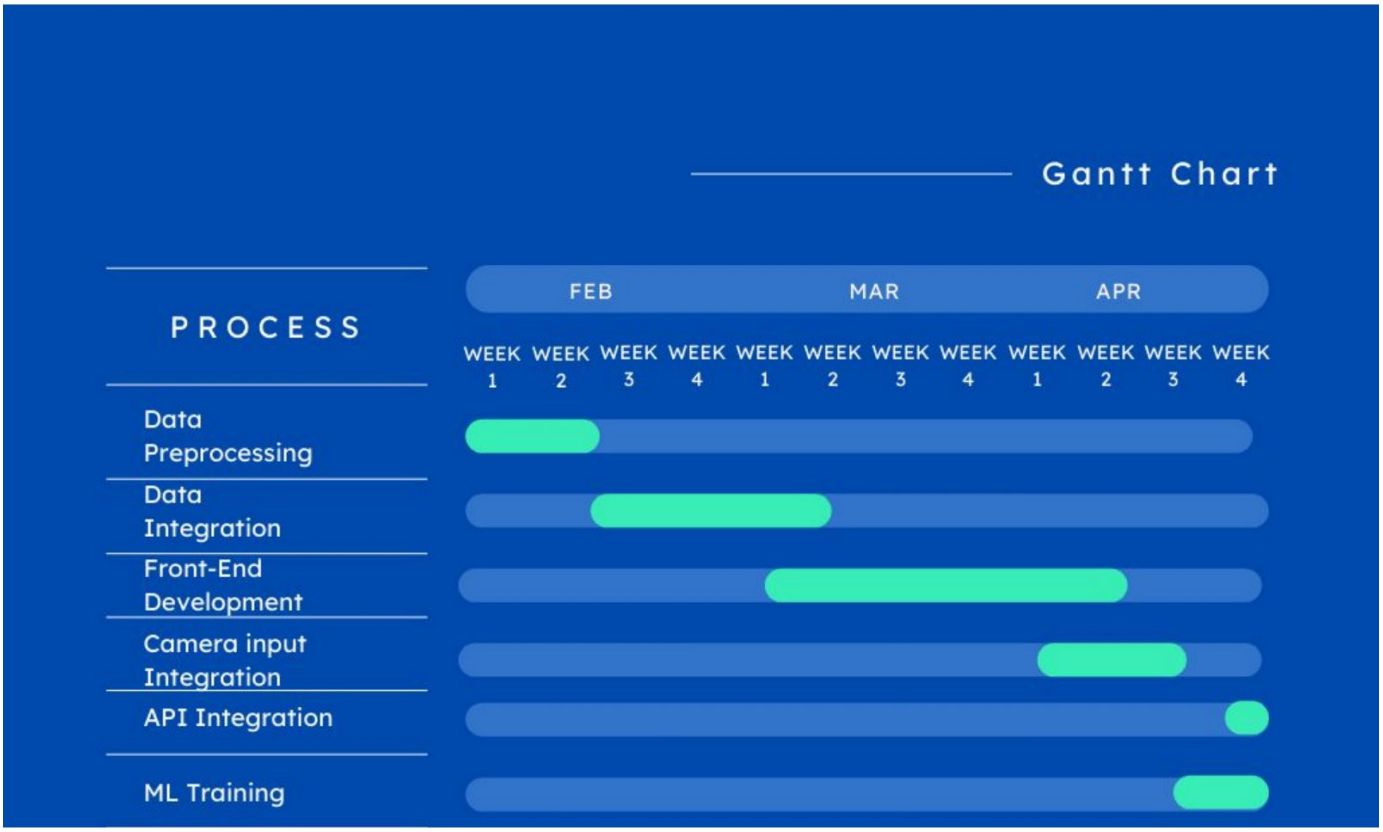
Blur: Up to 2px

Noise: Up to 1.88% of pixels

Database Design



Work Division



Software/ Hardware Requirements

- Software - YOLOv7, Flutter, Firebase,Python
- Hardware: Motion detection trap cameras and Camera system which supports RTSP protocol

-CPU: Minimum 6 cores (e.g., Intel Core i7 or AMD Ryzen 5)

-OS : A 64-bit operating system

-RAM: Minimum 16GB DDR4 RAM

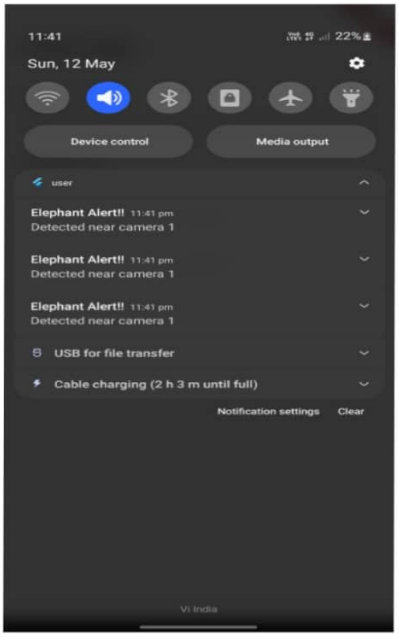
-GPU: Ideally, a dedicated graphics card (GPU) with at least 8 GB

-Android phones with location sharing feature.

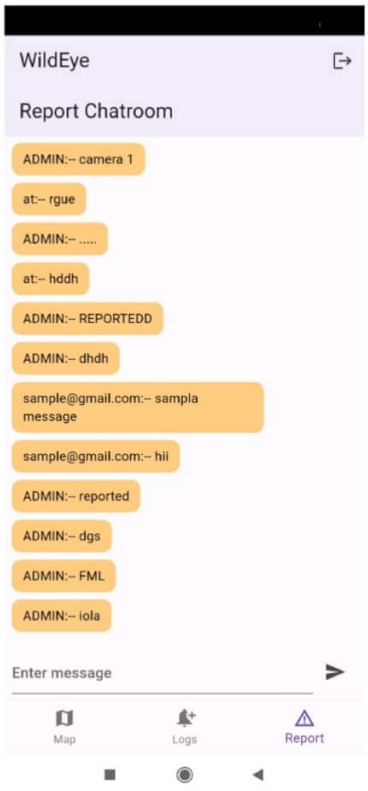
Results

-

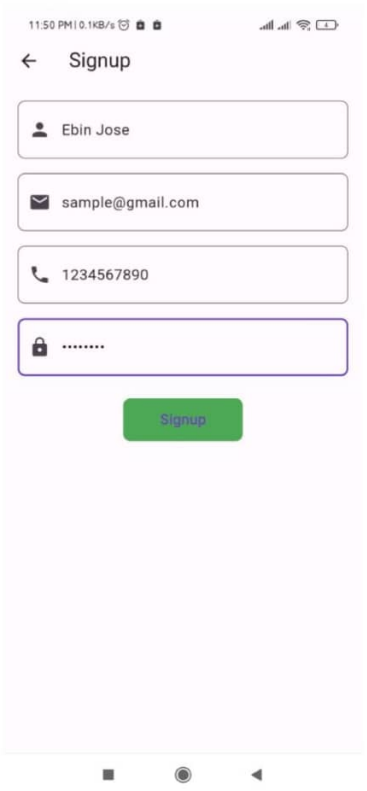
Detected image in log



Alerting when detected

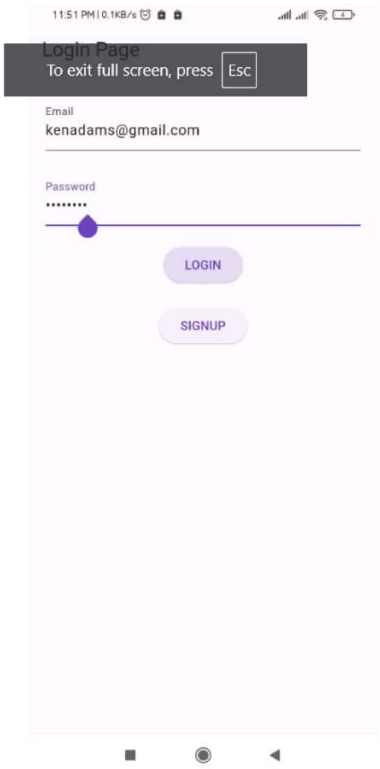


Report Chatroom



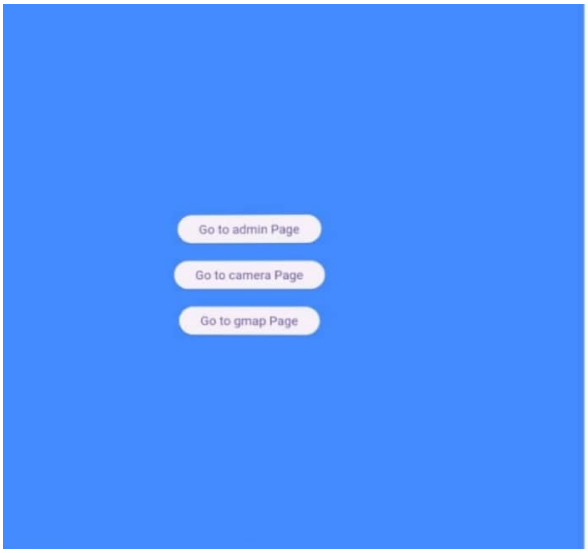
Sign up

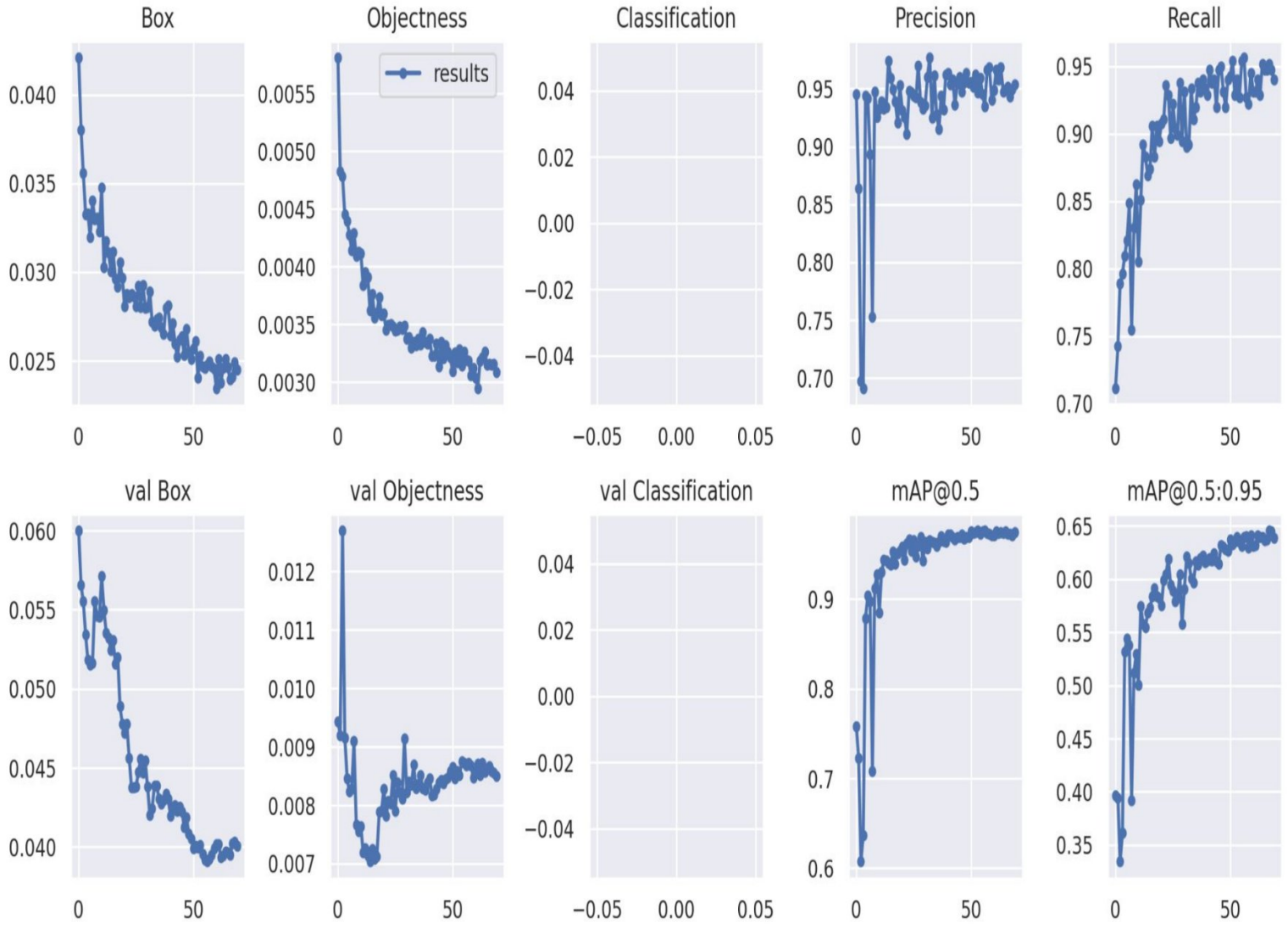
Map Screen with cameras

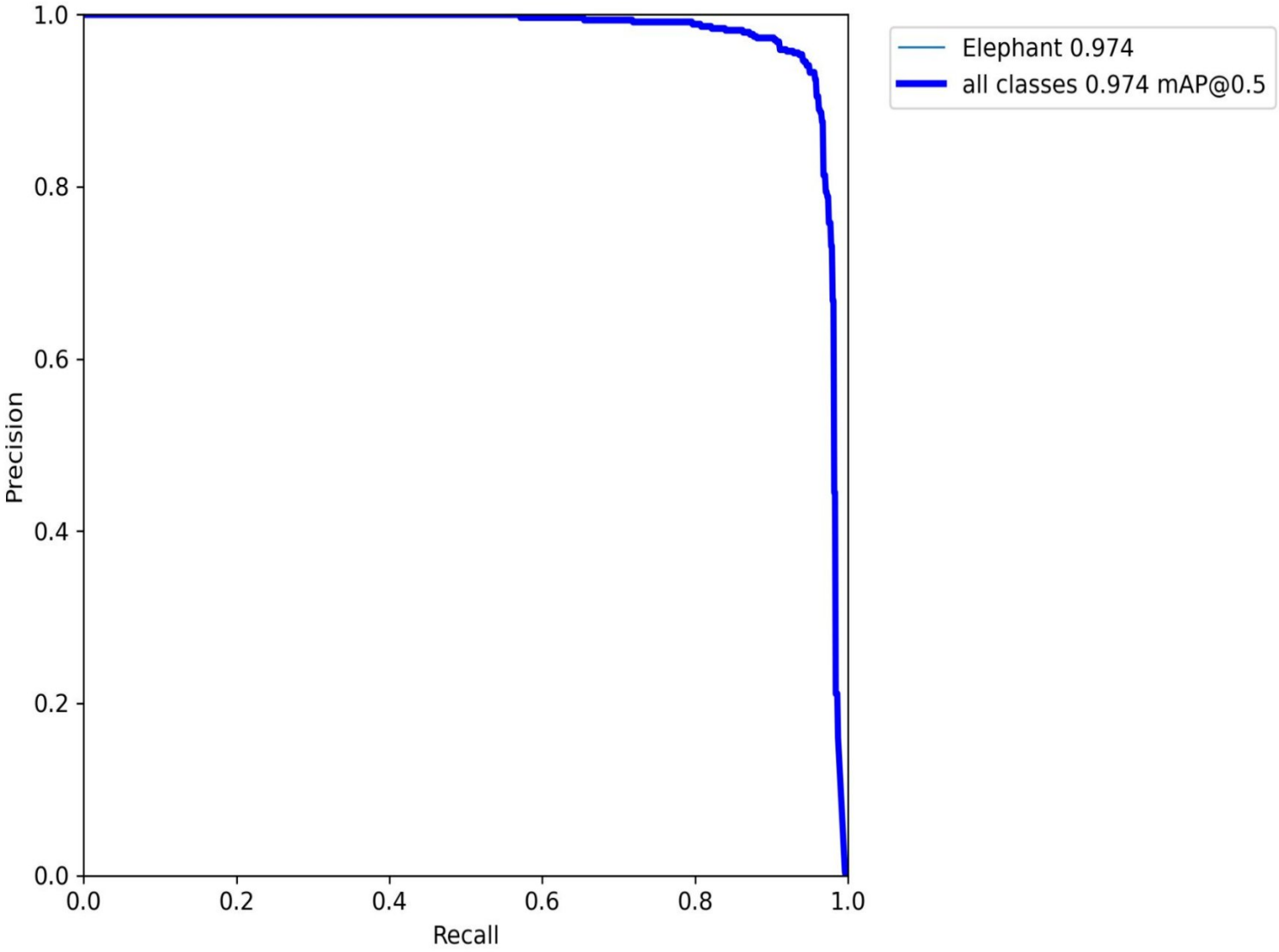


Login Page

Log History







Conclusion

- Develop wildlife monitoring system integrated with cameras and machine learning.
- Design Android app for timely alerts and easy wildlife sighting reporting.
- Promote community engagement in conservation efforts through user participation.

Future Enhancements

- Multi-Species Recognition:Enhance the machine learning models to support multi-species recognition, allowing for the identification of a wider range of wildlife species beyond elephants.
-

References

- YOLO official documentation - https://docs.ultralytics.com/
- Flutter Documentation - https://docs.flutter.dev/
- Package Documentation - https://pub.dev/
- Flutter API Documentation - https://api.flutter.dev/
- Firebase Documentation - https://firebase.google.com/docs
- Dataset - https://universe.roboflow.com/thai-elephant-dataset/elephant-dataset

- Sugumar, S. J., and R. Jayaparvathy. "An Early Warning System for Elephant Intrusion along the Forest Border Areas." *Current Science*, vol. 104, no. 11, 2013, pp. 1515–26. *JSTOR*, http://www.jstor.org/stable/24092475. Accessed 12 Apr. 2024.

Appendix B: Vision, Mission, Programme Outcomes and Course Outcomes

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
RAJAGIRI SCHOOL OF ENGINEERING & TECHNOLOGY (AUTONOMOUS)
RAJAGIRI VALLEY, KAKKANAD, KOCHI, 682039
(Affiliated to APJ Abdul Kalam Technological University)

Vision, Mission, Programme Outcomes and Course Outcomes

Institute Vision

To evolve into a premier technological institution, moulding eminent professionals with creative minds, innovative ideas and sound practical skill, and to shape a future where technology works for the enrichment of mankind.

Institute Mission

To impart state-of-the-art knowledge to individuals in various technological disciplines and to inculcate in them a high degree of social consciousness and human values, thereby enabling them to face the challenges of life with courage and conviction.

Department Vision

To become a centre of excellence in Computer Science and Engineering, moulding professionals catering to the research and professional needs of national and international organizations.

Department Mission

To inspire and nurture students, with up-to-date knowledge in Computer Science and Engineering, ethics, team spirit, leadership abilities, innovation and creativity to come out with solutions meeting societal needs.

Programme Outcomes (PO)

Engineering Graduates will be able to:

1. Engineering Knowledge: Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.

2. Problem analysis: Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.

3. Design/development of solutions: Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.

4. Conduct investigations of complex problems: Use research-based knowledge including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.

5. Modern Tool Usage: Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.

6. The engineer and society: Apply reasoning informed by the contextual knowledge

to assess societal, health, safety, legal, and cultural issues and the consequent responsibilities relevant to the professional engineering practice.

7. Environment and sustainability: Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.

8. Ethics: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.

9. Individual and Team work: Function effectively as an individual, and as a member or leader in teams, and in multidisciplinary settings.

10. Communication: Communicate effectively with the engineering community and with society at large. Be able to comprehend and write effective reports documentation. Make effective presentations, and give and receive clear instructions.

11. Project management and finance: Demonstrate knowledge and understanding of engineering and management principles and apply these to one's own work, as a member and leader in a team. Manage projects in multidisciplinary environments.

12. Life-long learning: Recognize the need for, and have the preparation and ability to engage in independent and lifelong learning in the broadest context of technological change.

Programme Specific Outcomes (PSO)

A graduate of the Computer Science and Engineering Program will demonstrate:

PSO1: Computer Science Specific Skills

The ability to identify, analyze and design solutions for complex engineering problems in multidisciplinary areas by understanding the core principles and concepts of computer science and thereby engage in national grand challenges.

PSO2: Programming and Software Development Skills

The ability to acquire programming efficiency by designing algorithms and applying standard practices in software project development to deliver quality software products meeting the demands of the industry.

PSO3: Professional Skills

The ability to apply the fundamentals of computer science in competitive research and to develop innovative products to meet the societal needs thereby evolving as an eminent researcher and entrepreneur.

Course Outcomes

Appendix C: CO-PO-PSO Mapping

COURSE OUTCOMES:

After completion of the course the student will be able to

SL. NO	DESCRIPTION	Blooms' Taxonomy Level
CO1	Identify technically and economically feasible problems (Cognitive Knowledge Level: Apply)	Level 3: Apply
CO2	Identify and survey the relevant literature for getting exposed to related solutions and get familiarized with software development processes (Cognitive Knowledge Level: Apply)	Level 3: Apply
CO3	Perform requirement analysis, identify design methodologies and develop adaptable & reusable solutions of minimal complexity by using modern tools & advanced programming techniques (Cognitive Knowledge Level: Apply)	Level 3: Apply
CO4	Prepare technical report and deliver presentation (Cognitive Knowledge Level: Apply)	Level 3: Apply
CO5	Apply engineering and management principles to achieve the goal of the project (Cognitive Knowledge Level: Apply)	Level 3: Apply

CO-PO AND CO-PSO MAPPING

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO 1	PSO 2	PSO3
CO1	3	3	3	3		2	2	3	2	2	2	3	2	2	2
CO2	3	3	3	3	3	2		3	2	3	2	3	2	2	2
CO3	3	3	3	3	3	2	2	3	2	2	2	3			2
CO4	2	3	2	2	2			3	3	3	2	3	2	2	2
CO5	3	3	3	2	2	2	2	3	2		2	3	2	2	2

3/2/1: high/medium/low

JUSTIFICATIONS FOR CO-PO MAPPING

MAPPING	LOW/ MEDIUM/ HIGH	JUSTIFICATION
101003/CS622T.1-PO1	**HIGH**	Identify technically and economically feasible problems by applying the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
101003/CS622T.1-PO2	**HIGH**	Identify technically and economically feasible problems by analysing complex engineering problems reaching substantiated conclusions using first principles of mathematics.
101003/CS622T.1-PO3	**HIGH**	Design solutions for complex engineering problems by identifying technically and economically feasible problems.
101003/CS622T.1-PO4	**HIGH**	Identify technically and economically feasible problems by analysis and interpretation of data.
101003/CS622T.1-PO6	**MEDIUM**	Responsibilities relevant to the professional engineering practice by identifying the problem.
101003/CS622T.1-PO7	**MEDIUM**	Identify technically and economically feasible problems by understanding the impact of the professional engineering solutions.
101003/CS622T.1-PO8	**HIGH**	Apply ethical principles and commit to professional ethics to identify technically and economically feasible problems.
101003/CS622T.1-PO9	**MEDIUM**	Identify technically and economically feasible problems by working as a team.
101003/CS622T.1-PO10	**MEDIUM**	Communicate effectively with the engineering community by identifying technically and economically feasible problems.
101003/CS622T.1-P011	**MEDIUM**	Demonstrate knowledge and understanding of engineering and management principles by selecting the technically and economically feasible problems.
101003/CS622T.1-PO12	**HIGH**	Identify technically and economically feasible problems for long term learning.
101003/CS622T.1-PSO1	**MEDIUM**	Ability to identify, analyze and design solutions to identify technically and economically feasible problems.
101003/CS622T.1-PSO2	**MEDIUM**	By designing algorithms and applying standard practices in software project development and Identifying technically and economically feasible problems.
101003/CS622T.1-PSO3	**MEDIUM**	Fundamentals of computer science in competitive research can be applied to Identify technically and economically feasible problems.
101003/CS622T.2-PO1	**HIGH**	Identify and survey the relevant by applying the knowledge of mathematics, science, engineering fundamentals.

101003/CS622T.2-PO2	**HIGH**	Identify, formulate, review research literature, and analyze complex engineering problems get familiarized with software development processes.
101003/CS622T.2-PO3	**HIGH**	Design solutions for complex engineering problems and design based on the relevant literature.
101003/CS622T.2-PO4	**HIGH**	Use research-based knowledge including design of experiments based on relevant literature.
101003/CS622T.2-PO5	**HIGH**	Identify and survey the relevant literature for getting exposed to related solutions and get familiarized with software development processes by using modern tools.
101003/CS622T.2-PO6	**MEDIUM**	Create, select, and apply appropriate techniques, resources, by identifying and surveying the relevant literature.
101003/CS622T.2-PO8	**HIGH**	Apply ethical principles and commit to professional ethics based on the relevant literature.
101003/CS622T.2-PO9	**MEDIUM**	Identify and survey the relevant literature as a team.
101003/CS622T.2-PO10	**HIGH**	Identify and survey the relevant literature for a good communication to the engineering fraternity.
101003/CS622T.2-PO11	**MEDIUM**	Identify and survey the relevant literature to demonstrate knowledge and understanding of engineering and management principles.
101003/CS622T.2-PO12	**HIGH**	Identify and survey the relevant literature for independent and lifelong learning.
101003/CS622T.2-PSO1	**MEDIUM**	Design solutions for complex engineering problems by Identifying and survey the relevant literature.
101003/CS622T.2-PSO2	**MEDIUM**	Identify and survey the relevant literature for acquiring programming efficiency by designing algorithms and applying standard practices.
101003/CS622T.2-PSO3	**MEDIUM**	Identify and survey the relevant literature to apply the fundamentals of computer science in competitive research.
101003/CS622T.3-PO1	**HIGH**	Perform requirement analysis, identify design methodologies by using modern tools & advanced programming techniques and by applying the knowledge of mathematics, science, engineering fundamentals.
101003/CS622T.3-PO2	**HIGH**	Identify, formulate, review research literature for requirement analysis, identify design methodologies and develop adaptable & reusable solutions.

101003/CS622T.3-PO3	**HIGH**	Design solutions for complex engineering problems and perform requirement analysis, identify design methodologies.
101003/CS622T.3-PO4	**HIGH**	Use research-based knowledge including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.
101003/CS622T.3-PO5	**HIGH**	Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools.
101003/CS622T.3-PO6	**MEDIUM**	Perform requirement analysis, identify design methodologies and assess societal, health, safety, legal, and cultural issues.
101003/CS622T.3-PO7	**MEDIUM**	Understand the impact of the professional engineering solutions in societal and environmental contexts and Perform requirement analysis, identify design methodologies and develop adaptable & reusable solutions.
101003/CS622T.3-PO8	**HIGH**	Perform requirement analysis, identify design methodologies and develop adaptable & reusable solutions by applying ethical principles and commit to professional ethics.
101003/CS622T.3-PO9	**MEDIUM**	Function effectively as an individual, and as a member or leader in teams, and in multidisciplinary settings.
101003/CS622T.3-PO10	**MEDIUM**	Communicate effectively with the engineering community and with society at large to perform requirement analysis, identify design methodologies.
101003/CS622T.3-PO11	**MEDIUM**	Demonstrate knowledge and understanding of engineering requirement analysis by identifying design methodologies.
101003/CS622T.3-PO12	**HIGH**	Recognize the need for, and have the preparation and ability to engage in independent and lifelong learning in the broadest context of technological change by analysis, identify design methodologies and develop adaptable & reusable solutions.
101003/CS622T.3-PSO3	**MEDIUM**	The ability to apply the fundamentals of computer science in competitive research and prior to that perform requirement analysis, identify design methodologies.
101003/CS622T.4-PO1	**MEDIUM**	Prepare technical report and deliver presentation by applying the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
101003/CS622T.4-PO2	**HIGH**	Identify, formulate, review research literature, and analyze complex engineering problems by preparing technical report and deliver presentation.

101003/CS622T.4-PO3	**MEDIUM**	Prepare Design solutions for complex engineering problems and create technical report and deliver presentation.
101003/CS622T.4-PO4	**MEDIUM**	Use research-based knowledge including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions and prepare technical report and deliver presentation.
101003/CS622T.4-PO5	**MEDIUM**	Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools and Prepare technical report and deliver presentation.
101003/CS622T.4-PO8	**HIGH**	Prepare technical report and deliver presentation by applying ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.
101003/CS622T.4-PO9	**HIGH**	Prepare technical report and deliver presentation effectively as an individual, and as a member or leader in teams, and in multidisciplinary settings.
101003/CS622T.4-PO10	**HIGH**	Communicate effectively with the engineering community and with society at large by prepare technical report and deliver presentation.
101003/CS622T.4-PO11	**MEDIUM**	Demonstrate knowledge and understanding of engineering and management principles and apply these to one's own work by prepare technical report and deliver presentation.
101003/CS622T.4-PO12	**HIGH**	Recognize the need for, and have the preparation and ability to engage in independent and lifelong learning in the broadest context of technological change by prepare technical report and deliver presentation.
101003/CS622T.4-PSO1	**MEDIUM**	Prepare a technical report and deliver presentation to identify, analyze and design solutions for complex engineering problems in multidisciplinary areas.
101003/CS622T.4-PSO2	**MEDIUM**	To acquire programming efficiency by designing algorithms and applying standard practices in software project development and to prepare technical report and deliver presentation.
101003/CS622T.4-PSO3	**MEDIUM**	To apply the fundamentals of computer science in competitive research and to develop innovative products to meet the societal needs by preparing technical report and deliver presentation.
101003/CS622T.5-PO1	**HIGH**	Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
101003/CS622T.5-PO2	**HIGH**	Identify, formulate, review research literature, and analyze complex engineering problems by applying engineering and management principles to achieve the goal of the project.

101003/CS622T.5-PO3	**HIGH**	Apply engineering and management principles to achieve the goal of the project and to design solutions for complex engineering problems and design system components or processes that meet the specified needs.
101003/CS622T.5-PO4	**MEDIUM**	Apply engineering and management principles to achieve the goal of the project and use research-based knowledge including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.
101003/CS622T.5-PO5	**MEDIUM**	Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools and to apply engineering and management principles to achieve the goal of the project.
101003/CS622T.5-PO6	**MEDIUM**	Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal, and cultural issues and the consequent responsibilities by applying engineering and management principles to achieve the goal of the project.
101003/CS622T.5-PO7	**MEDIUM**	Understand the impact of the professional engineering solutions in societal and environmental contexts, and apply engineering and management principles to achieve the goal of the project.
101003/CS622T.5-PO8	**HIGH**	Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice and to use the engineering and management principles to achieve the goal of the project.
101003/CS622T.5-PO9	**MEDIUM**	Function effectively as an individual, and as a member or leader in teams, and in multidisciplinary settings and to apply engineering and management principles to achieve the goal of the project.
101003/CS622T.5-PO11	**MEDIUM**	Demonstrate knowledge and understanding of engineering and management principles and apply these to one's own work, as a member and leader in a team. Manage projects in multidisciplinary environments and to apply engineering and management principles to achieve the goal of the project.
101003/CS622T.5-PO12	**HIGH**	Recognize the need for, and have the preparation and ability to engage in independent and lifelong learning in the broadest context of technological change and to apply engineering and management principles to achieve the goal of the project.
101003/CS622T.5-PSO1	**MEDIUM**	The ability to identify, analyze and design solutions for complex engineering problems in multidisciplinary areas. Apply engineering and management principles to achieve the goal of the project.

101003/CS622T.5-PSO2	**MEDIUM**	The ability to acquire programming efficiency by designing algorithms and applying standard practices in software project development to deliver quality software products meeting the demands of the industry and to apply engineering and management principles to achieve the goal of the project.
101003/CS622T.5-PSO3	**MEDIUM**	The ability to apply the fundamentals of computer science in competitive research and to develop innovative products to meet the societal needs thereby evolving as an eminent researcher and entrepreneur and apply engineering and management principles to achieve the goal of the project.