MAIN PROJECT REPORT

On

FAUNA TRACK AI

Submitted in partial fulfilment for the award of degree

Of

Master of Computer Applications

By

REVATHI S KUMAR (MLM23MCA-2039)

Under the Guidance of

MS. GREESHMAMOL P KURIAN

(Assistant Professor, Dept. of Computer Applications)

DEPARTMENT OF COMPUTER APPLICATIONS

MANGALAM COLLEGE OF ENGINEERING, ETTUMANOOR

(Affiliated to APJ Abdul Kalam Technological University)

APRIL 2025

दिर्श शास्त्रे प्रतिष्ठितम्।

NAAC A

NBA NATIONAL BOARD OF ACCREDITATION

CE CSE ME ECE

MANGALAM COLLEGE OF ENGINEERING

Accredited by NAAC& ISO 9001:2000 Certified Institution

DEPARTMENT OF COMPUTER APPLICATIONS

VISION

To become a centre of excellence in computer applications,competent in the global ecosystem with technical knowledge,innovation with a sense of social commitment.

MISSION

- To serve with state of the art education,foster advanced research and cultivate innovation in the field of computer applications.
- To prepare learners with knowledge skills and critical thinking to excel in the technological landscape and contribute positively to society.

Program Educational Objectives

- PEO I :Graduates will possess a solid foundation and in-depth understanding of computer applications and will be equipped to analyze real-world problems, design and create innovative solutions, and effectively manage and maintain these solutions in their professional careers.
- PEO II: Graduates will acquire technological advancements through continued education, lifelong learning and research, thereby making meaningful contributions to the field of computing.
- PEO III: Graduates will cultivate team spirit, leadership, communication skills, ethics, and social values, enabling them to apply their understanding of the societal impacts of computer applications effectively.

Program Specific Outcomes

- **PSO I**: Apply advanced technologies through innovations to enhance the efficiency of design development.
- **PSO II**: Apply the principles of computing to analyze, design and implement sustainable solutions for real world challenges.

MANGALAM COLLEGE OF ENGINEERING, ETTUMANOOR

DEPARTMENT OF COMPUTER APPLICATIONS

APRIL 2025

विश्वं शास्त्रे प्रतिष्ठितम्।

<u>CERTIFICATE</u>

*This is to certify that the Project titled **"FAUNA TRACK AI"** is the bonafide record of the work done by **REVATHI S KUMAR (MLM23MCA-2039)** of Masters of Computer Applications towards the partial fulfilment of the requirement for the award of the **DEGREE OF MASTERS OF COMPUTER APPLICATIONS by APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY**, during the academic year 2024-25.*

PROJECT GUIDE	***HEAD OF THE DEPARTMENT***
Ms. Greeshmamol P Kurian	**Ms. Divya S.B.**
Assistant Professor	**Associate Professor**
Department of Computer Applications	**Department of Computer Applications**

PROJECT COORDINATOR	***EXTERNAL EXAMINER***
MS. Banu Sumayya S	
Assistant Professor	
Department of Computer Applications	

ACKNOWLEDGEMENT

I am greatly indebted to the authorities of Mangalam College of Engineering for providing the necessary facilities to successfully complete my Project on the topic “Smart Electric Meter”.

I express my sincere thanks to **Dr. Vinodh P Vijayan**, Principal, Mangalam College of Engineering for providing the facilities to complete my Project successfully.

I thank and express my solicit gratitude to **Ms. Divya S.B**. , HOD, Department of Computer Applications, Mangalam College of Engineering, for her invaluable help and support which helped me a lot in successfully completing this Project work.

Furthermore, I would like to acknowledge with much appreciation the crucial role of the faculty, especially the project coordinator, **Ms. Banu Sumayya S**, Assistant Professor, Department of Computer Applications, who granted permission to use all the required equipment and necessary resources to complete the presentation and report.

I express my gratitude to my Internal Guide, **Ms. Greeshmamol P Kurian** Assistant professor, Department of Computer Applications, for the suggestions and encouragement which helped in the successful completion of our Project.

Finally, I would like to express my heartfelt thanks to my parents who were very supportive both financially and mentally and for their encouragement to achieve my goal.

REVATHI S KUMAR

(MLM23MCA-2039)

ABSTRACT

Fauna Track AI is an intelligent system designed to monitor and analyze animal tracks using artificial intelligence. By leveraging computer vision, deep learning, and geospatial mapping, it identifies wildlife species, tracks movement patterns, and assesses population distribution. The system processes data from trail cameras, drones, and mobile devices, enabling researchers and conservationists to study wildlife with greater accuracy and efficiency.

Fauna Track AI is an advanced artificial intelligence system designed to monitor, track, and analyze wildlife populations using machine learning and computer vision. This technology leverages satellite imagery, drone surveillance, and sensor data to provide real-time insights into animal movements, habitat changes, and conservation threats. By automating data collection and analysis, Fauna Track AI enhances biodiversity research, aids in anti-poaching efforts, and supports environmental sustainability. The system enables researchers, conservationists, and policymakers to make informed decisions for wildlife protection, ensuring a balance between human activities and ecological preservation.

This AI-driven tool supports biodiversity conservation by detecting endangered species, monitoring habitat changes, and identifying potential threats such as poaching. It also integrates environmental data to understand how climate change influences animal behavior. Beyond conservation, Fauna Track AI has applications in ecotourism, agriculture, and forensic investigations, making it a valuable resource for sustainable wildlife management.

Fauna Track AI is an innovative artificial intelligence system designed for wildlife monitoring and conservation. It integrates machine learning, computer vision, and IoT-based sensors to track animal movements, analyze behavioral patterns, and detect potential threats such as poaching and habitat destruction. Utilizing satellite imagery, drone footage, and real-time data processing, the system provides accurate insights for researchers, conservationists, and policymakers. Fauna Track AI enhances biodiversity protection by enabling proactive decision-making, improving species conservation strategies, and supporting sustainable ecosystem management. This technology plays a crucial role in bridging the gap between technological advancements and wildlife preservation.

TABLE OF CONTENTS

List of Figures

List of Abbreviations

- GPS: Global Positioning System
- AI: Artificial intelligence
- YOLO: You Only Look Once
- V5: Version 5
- CCTV: Closed-Circuit Television
- UAV: Unmanned Aerial Vehicles
- GIS: Geographic Information System
- IoT: Internet of Things
- UAT: User Acceptance Testing
- CNN: Convolutional Neural Networks
- OOD: Object-Oriented Design
- RNN: Recurrent Neural Networks
- RFID: Radio-Frequency IDentification
- DFD: Data Flow Diagram
- UML: Unified Modeling Language

CHAPTER 1

INTRODUCTION

FaunaTrack AI is an intelligent system designed to monitor and analyze wildlife using advanced technologies like artificial intelligence, GPS tracking, and remote sensing. It helps researchers, conservationists, and wildlife organizations track animal movements, study behavior, and protect endangered species. FaunaTrack AI is an innovative technology designed to monitor and analyze wildlife using artificial intelligence and tracking systems. This tool helps conservationists, researchers, and environmental organizations collect real-time data on animal movements, behaviors, and habitat changes. It is advanced technology that leverages artificial intelligence and tracking systems to monitor wildlife, analyze animal behavior, and support conservation efforts through real-time data collection and predictive insights.

FaunaTrack AI is an innovative wildlife monitoring and tracking system that leverages artificial intelligence, GPS technology, and remote sensing to enhance conservation efforts and ecological research. Designed to provide real-time insights into animal behavior, movement patterns, and habitat changes, this intelligent system enables researchers, conservationists, and environmental organizations to make data-driven decisions for species protection and biodiversity management. By utilizing advanced image recognition, machine learning, and data analytics, FaunaTrack AI can identify various species, track their locations, and analyze migration trends with high accuracy. The integration of drone and satellite imagery further enhances its capabilities, allowing for large-scale monitoring of wildlife populations without direct human intervention. Additionally, the system plays a crucial role in detecting threats such as poaching, habitat destruction, and human-wildlife conflicts by providing early warnings and predictive analysis. Its user-friendly dashboard and interactive data visualization tools make it accessible for researchers and conservation agencies, ensuring that critical information is available for proactive decision-making. By bridging the gap between technology and wildlife conservation, FaunaTrack AI represents a significant step toward sustainable environmental management, helping protect endangered species and preserve natural ecosystems for future generations. It provides valuable insights into biodiversity and ecosystem health. It also plays a crucial role in detecting potential threats such as poaching, habitat destruction, and human-wildlife conflicts, enabling proactive conservation efforts.

1.1 Background

Fauna Track AI is an advanced AI-powered system that leverages computer vision and sound detection to monitor and classify wildlife in real time. By utilizing CCTV cameras and acoustic sensors, the system can automatically identify, track, and analyze animal movement and behavior in various environments. Advanced Wild Animal Detection and Alert System Using you only look once version5 (YOLO V5) Model is a proposed analyses the images and detect the presence of wild animals. If the system detects the presence of any wild animal, it sends an alert to the authorities through an alarm or message. The proposed system is expected to provide an effective solution to prevent any potential harm caused by wild animals and help preserve wildlife by reducing human-animal conflict. Fauna Track AI has applications in wildlife conservation, biodiversity research, and environmental monitoring, providing valuable insights for researchers and conservationists. Fauna Track AI is an innovative artificial intelligence (AI) system that integrates computer vision and sound detection classification to monitor wildlife using CCTV cameras and acoustic sensors. This technology enables automated detection, identification, and analysis of animal species in diverse environments, aiding conservation efforts and biodiversity research. Fauna Track AI is an innovative system that applies computer vision and AI-powered sound detection to monitor and classify animals using CCTV and audio sensors. This technology is particularly useful for wildlife conservation, biodiversity monitoring, and human-wildlife conflict mitigation.

1.2 Introduction

FaunaTrack AI is an advanced technology designed to monitor wildlife and track animal movements using artificial intelligence and real-time data analytics. This system helps researchers, conservationists, and environmental organizations study animal behavior, migration patterns, and habitat changes with greater accuracy and efficiency. By integrating AI-powered image recognition, GPS tracking, and remote sensing, FaunaTrack AI provides valuable insights into biodiversity and ecosystem health. It also plays a crucial role in detecting potential threats such as poaching, habitat destruction, and human-wildlife conflicts, enabling proactive conservation efforts. With its ability to process large volumes of data and generate predictive insights, FaunaTrack AI is transforming the way wildlife is studied and protected, contributing to more sustainable environmental management. It enables researchers and conservationists to gather real-time data on species movement, habitat usage, and population trends. By combining AI-driven image recognition, GPS tracking, and data analytics, FaunaTrack AI helps in identifying potential threats such as poaching, habitat loss, and human-wildlife conflicts. This technology enhances conservation efforts by providing accurate insights that support decision-making for protecting biodiversity and maintaining ecological balance.

1.3 Problem Statement

Wildlife conservation faces significant challenges due to habitat destruction, poaching, climate change, and human-wildlife conflicts. Traditional methods of monitoring animal populations, such as manual tracking and camera traps, are often time-consuming, labor-intensive, and limited in coverage. These conventional approaches struggle to provide real-time insights, making it difficult for conservationists and researchers to respond quickly to emerging threats. Additionally, the lack of accurate and large-scale data on animal movements and behaviors hinders effective policy-making and conservation strategies. The increasing human encroachment into wildlife habitats further exacerbates the problem, leading to more frequent conflicts between animals and local communities. Without efficient monitoring, authorities are unable to detect poaching activities in time to prevent illegal hunting and protect endangered species. Climate change adds another layer of complexity by altering ecosystems and affecting species migration patterns, requiring advanced tracking solutions to understand and mitigate its impact.

FaunaTrack AI addresses these challenges by integrating artificial intelligence, real-time tracking, and data analytics to provide accurate, automated, and large-scale wildlife monitoring. By leveraging AI-driven image recognition, GPS-based movement tracking, and predictive analytics, this system enables conservationists to gather reliable data, detect threats early, and implement timely interventions. FaunaTrack AI bridges the gap between technology and conservation, ensuring that efforts to protect wildlife are more efficient, data-driven, and sustainable.

1.4 Motivation

The need for effective wildlife conservation has never been greater, as human activities, climate change, and habitat destruction continue to threaten biodiversity. Traditional wildlife monitoring methods, such as manual tracking and camera traps, often fail to provide timely and comprehensive data, making it difficult to protect vulnerable species. The lack of real-time insights into animal movements, behavior, and population trends limits conservation efforts and hinders proactive decision-making. Additionally, poaching and illegal wildlife trade remain persistent threats, further endangering species that are already at risk of extinction.

FaunaTrack AI was developed to address these challenges by leveraging advanced technology to enhance wildlife monitoring and conservation efforts. By integrating artificial intelligence, GPS tracking, and data analytics, this system enables researchers and conservationists to track animal movements in real time, analyze behavioral patterns, and detect potential threats with greater accuracy. The ability to automate data collection and analysis reduces the need for human intervention, making conservation efforts more efficient and scalable.

Another key motivation behind FaunaTrack AI is the prevention of human-wildlife conflicts. As urbanization expands into natural habitats, interactions between humans and animals increase, often leading to conflicts that result in harm to both parties. FaunaTrack AI helps mitigate these issues by predicting animal movement patterns and providing early warnings to authorities and communities.

By harnessing the power of technology, FaunaTrack AI aims to revolutionize wildlife conservation, ensuring that future generations inherit a world rich in biodiversity. The goal is to create a more sustainable and data-driven approach to protecting wildlife while supporting ecological balance and coexistence between humans and nature.

1.5 Scope

FaunaTrack AI has a broad scope in wildlife conservation, ecological research, and biodiversity management. By integrating artificial intelligence, GPS tracking, and data analytics, it provides a comprehensive solution for monitoring and studying animal behavior, movement patterns, and habitat changes. This technology can be used across various environments, including forests, savannas, wetlands, and marine ecosystems, allowing conservationists to track species in diverse habitats.One of the key areas where FaunaTrack AI can make a significant impact is endangered species protection. By continuously monitoring at-risk populations, researchers can identify trends, detect threats, and take proactive measures to prevent extinction. Additionally, the system can help enforce anti-poaching laws by detecting illegal activities in protected areas, providing real-time alerts to authorities.

CHAPTER 2

LITERATURE REVIEW

Artificial intelligence (AI) is increasingly being integrated into wildlife monitoring, offering innovative solutions for tracking and conserving animal populations. One notable example is Nature Track AI, developed by Robotto, which utilizes unmanned aerial vehicles (UAVs) equipped with AI-driven tracking capabilities. This system enables real-time detection and monitoring of wildlife, providing conservationists and livestock farmers with crucial data to prevent human-wildlife conflicts and support conservation efforts. Nature Track AI's collaboration with organizations like WWF Denmark and WWF Thailand has demonstrated its effectiveness in mitigating human-elephant conflicts in national parks such as Kaeng Krung and Kui Buri in Thailand.

1.Introduction to fauna Track ai

Fauna Track AI is an advanced artificial intelligence system designed to monitor, track, and analyze wildlife movements using cutting-edge technologies such as computer vision, machine learning, and remote sensing. This innovative approach enhances conservation efforts by providing real-time data on animal populations, habitat usage, and migration patterns. By integrating AI-powered surveillance with tools like drones, acoustic sensors, and satellite imagery, Fauna Track AI helps researchers and conservationists make informed decisions to protect endangered species and mitigate human-wildlife conflicts. The system's automation reduces the need for manual tracking, increasing efficiency while minimizing disturbance to natural ecosystems.

2. Role of Fauna Track Ai

Fauna Track AI plays a crucial role in wildlife monitoring and conservation by leveraging artificial intelligence to track animal movements, analyze behavioral patterns, and assess habitat conditions. This technology aids researchers, conservationists, and policymakers by providing real-time data that enhances species protection efforts. By using AI-driven tools such as drones, thermal imaging, and acoustic sensors, Fauna Track AI helps in identifying endangered species, preventing poaching, and reducing human-wildlife conflicts. Additionally, it supports ecological research by offering insights into migration trends, population health, and environmental changes, ultimately contributing to sustainable biodiversity management.

3. Technological Developments

The technological advancements in Fauna Track AI have significantly improved wildlife monitoring and conservation. This system integrates artificial intelligence with modern tracking technologies such as drone surveillance, satellite imagery, and acoustic monitoring. AI-powered image recognition and machine learning algorithms enable the identification of animal species and their behaviors with high accuracy. Additionally, real-time data processing allows for quick decision-making in conservation efforts. Innovations like infrared sensors and GPS tagging enhance the tracking of nocturnal and elusive species. These developments have made wildlife monitoring more efficient, reducing human intervention while improving data collection for ecological research and species protection.

4. Applications

Fauna Track AI has a wide range of applications in wildlife conservation and ecological research. It is used to monitor animal populations, track migration patterns, and assess habitat changes, providing valuable insights for conservation efforts. This technology helps prevent human-wildlife conflicts by detecting and predicting animal movements near human settlements, reducing risks to both people and animals. Additionally, Fauna Track AI supports anti-poaching initiatives by using AI-powered surveillance systems to identify illegal activities in protected areas. It also aids researchers in studying endangered species by analyzing behavioral patterns and environmental interactions, contributing to biodiversity preservation and sustainable wildlife management.

5. Advantages of Fauna Track Ai

Fauna Track AI offers several advantages in wildlife conservation and ecological research. It enhances accuracy in tracking animal movements and behaviors, reducing reliance on manual observation. The technology enables real-time monitoring, allowing for quick responses to threats such as poaching and habitat destruction. By integrating AI with tools like drones and satellite imaging, it provides extensive coverage of remote and inaccessible areas. Additionally, Fauna Track AI minimizes human disturbance to wildlife, promoting ethical and non-intrusive research methods. Its data-driven insights improve conservation planning, helping protect endangered species and maintain ecological balance efficiently.

6. Challenges and Limitations

Despite its numerous benefits, Fauna Track AI faces several challenges and limitations. One major issue is the high cost of implementing and maintaining AI-driven tracking systems, which may not be accessible to all conservation organizations. Additionally, the technology relies heavily on data accuracy, and any errors in AI models can lead to misinterpretations of animal behavior or population estimates. Environmental factors, such as dense vegetation or extreme weather conditions, can also interfere with tracking accuracy. Furthermore, ethical concerns regarding data privacy and potential disruptions to wildlife habitats must be carefully managed. Lastly, the need for skilled personnel to operate and interpret AI-driven systems remains a challenge for widespread adoption.

7. Case Studies and Real-World Applications

Case Studies and Real-World Applications of Fauna Track AI

Fauna Track AI has been successfully implemented in various real-world conservation projects, demonstrating its effectiveness in wildlife monitoring and protection.

1. **Elephant Conservation in Thailand**

   In collaboration with conservation groups, AI-powered drone surveillance has been used to monitor elephant movements in Thailand’s national parks. By tracking their migration patterns, researchers can mitigate human-elephant conflicts by implementing early warning systems that prevent crop damage and ensure both human and animal safety.

2. **Acoustic Monitoring of Endangered Species in Costa Rica**

   AI-driven acoustic sensors have been deployed in Costa Rica’s Osa Peninsula to study the calls of Geoffrey’s spider monkeys. The technology helps researchers analyze large volumes of audio data, identifying patterns in vocalizations to track population dynamics and habitat preferences without direct human interference.

3. **Anti-Poaching Efforts in Africa**

   AI-powered surveillance systems, including drones and motion-sensing cameras, have been used in African wildlife reserves to combat poaching. By detecting unusual human activities in protected areas, authorities can respond quickly, reducing illegal hunting of endangered species like rhinos and elephants.

4. **Tracking Snow Leopards in the Himalayas**

   Camera traps combined with AI image recognition have improved the monitoring of snow leopards in the remote Himalayan region. The technology automatically identifies individual leopards based on unique fur patterns, providing valuable data for population studies and conservation planning.

These real-world applications highlight the potential of Fauna Track AI in supporting biodiversity conservation, reducing human-wildlife conflicts, and enhancing ecological research through advanced technological solutions.

8. Future Directions

The future of Fauna Track AI is expected to bring further advancements in wildlife monitoring and conservation through emerging technologies. One key development is the integration of more sophisticated artificial intelligence models, which will enhance the accuracy of species identification and behavioral analysis. The use of advanced machine learning algorithms will enable automated data processing, reducing the need for human intervention and increasing efficiency.

Another promising direction is the expansion of real-time monitoring using satellite technology and high-resolution imaging. These innovations will provide more comprehensive coverage of remote and inaccessible areas, improving conservation strategies. Additionally, the incorporation of Internet of Things (IoT) devices and edge computing will allow for faster data collection and analysis, enabling immediate responses to environmental threats such as poaching and habitat destruction.

Collaboration between conservation organizations, governments, and technology companies will also play a crucial role in scaling Fauna Track AI applications globally. Efforts to make AI-driven tracking systems more cost-effective and accessible will help smaller conservation groups benefit from this technology. Furthermore, ethical considerations, including minimizing disruption to wildlife and ensuring data privacy, will shape the responsible development of AI in ecological research.

Overall, the continued evolution of Fauna Track AI holds great potential for improving wildlife conservation, advancing scientific research, and promoting sustainable coexistence between humans and nature.

9. Conclusion

Fauna Track AI represents a transformative approach to wildlife monitoring and conservation, utilizing advanced artificial intelligence to track, analyze, and protect animal populations. By integrating technologies such as drones, satellite imaging, and acoustic sensors, it enhances data accuracy and enables real-time decision-making in conservation efforts. Despite challenges such as high costs, environmental limitations, and ethical considerations, continuous technological advancements and collaborative initiatives are driving its growth and accessibility. With further innovation, Fauna Track AI has the potential to play a crucial role in biodiversity preservation, mitigating human-wildlife conflicts, and supporting sustainable environmental management for future generations. In conclusion, Fauna Track AI is a groundbreaking tool that offers immense potential for revolutionizing wildlife conservation and research. By harnessing the power of artificial intelligence, it enables more efficient, accurate, and non-invasive monitoring of animal populations and ecosystems. Through technologies like drones, AI-based image recognition, and acoustic sensors, Fauna Track AI enhances our ability to track species, protect endangered animals, and prevent human-wildlife conflicts.

Artificial Intelligence (AI) has become an integral tool in wildlife monitoring, offering innovative solutions for tracking and conserving fauna. This literature review explores the application of AI in fauna tracking, emphasizing methodologies, technological advancements, and practical implementations. The integration of AI in fauna tracking has revolutionized wildlife monitoring, offering precise, efficient, and scalable solutions for conservation challenges. While significant progress has been made, ongoing research is essential to address existing gaps and further enhance the effectiveness of AI applications in this field.

CHAPTER 3

PROPOSED SYSTEM

The proposed system for Fauna Track AI aims to revolutionize wildlife monitoring and conservation by integrating advanced technologies such as artificial intelligence, machine learning, and remote sensing tools. This system will provide a comprehensive, real-time solution to track and analyze animal behaviors, populations, and ecosystems across diverse landscapes.

1. **Data Collection and Integration**

   The system will leverage various data collection methods, including drones, camera traps, acoustic sensors, and satellite imagery. These tools will gather information on animal movements, environmental changes, and habitat conditions. Data from remote sensors will be transmitted to a centralized cloud platform for processing.

2. **AI-Powered Data Analysis**

   Artificial intelligence algorithms will be used to analyze the collected data, identifying patterns in animal behavior, migration routes, and population density. Machine learning models will be trained to recognize specific species based on visual or acoustic data, enabling automatic species identification without the need for manual intervention. This system will also be capable of distinguishing individual animals through features such as fur patterns or vocalizations.

3. **Real-Time Monitoring and Alerts**

   The proposed system will offer real-time monitoring of wildlife, with AI algorithms continuously analyzing incoming data. The system will generate instant alerts in the event of unusual activity, such as a poaching attempt or a human-wildlife conflict. Early warning systems will allow authorities and conservationists to respond quickly, reducing risks to both humans and animals.

4. **Geospatial Mapping and Visualization**

   The system will integrate geospatial mapping to visualize animal movement patterns, habitat utilization, and population distribution across various ecosystems. This mapping tool will help researchers and conservationists make informed decisions regarding protected areas, migration corridors, and biodiversity conservation strategies.

5. **Sustainability and Scalability**

   The proposed system will be designed to scale across different geographic regions, from remote forests to urban interfaces, providing flexible solutions for various conservation needs. The system will also be designed to be cost-effective, ensuring that even smaller conservation projects and local communities can benefit from its capabilities.

6. **Ethical Considerations and Data Security**

   The system will be developed with strict ethical guidelines in mind, ensuring minimal disruption to wildlife and their habitats. Data security protocols will also be implemented to protect sensitive information, ensuring that wildlife data is handled responsibly and securely.

In conclusion, the proposed Fauna Track AI system offers a powerful, integrated solution for enhancing wildlife conservation efforts. By combining AI with real-time monitoring and geospatial mapping, it will provide valuable insights into animal behavior and habitat management, contributing to more effective and sustainable conservation strategies.

1. Advantages of the Proposed System

The **Fauna Track AI** system offers numerous benefits in wildlife monitoring and conservation efforts. Below are some of its key advantages:

1. **Enhanced Wildlife Monitoring**
   The system provides real-time tracking of animal movements, behaviors, and populations, allowing conservationists to monitor species more effectively.
2. **Improved Conservation Efforts**
   By identifying endangered species and their habitats, Fauna Track AI helps in implementing proactive conservation strategies, reducing threats such as poaching and habitat destruction.
3. **Automation and Efficiency**
   The use of AI-driven technology minimizes the need for manual tracking, reducing labor costs and enabling faster data collection and analysis.
4. **Accurate Data Collection**
   With advanced AI algorithms, the system ensures precise identification of species, movements, and population trends, improving the reliability of ecological studies.
5. **Early Threat Detection**
   The system can detect anomalies in animal behavior, indicating potential dangers such as disease outbreaks, poaching activities, or environmental changes, allowing authorities to take timely action.
6. **Sustainable and Non-Intrusive Tracking**
   Unlike traditional tracking methods that may involve tagging or physical intervention, Fauna Track AI operates through remote sensing, camera traps, and drones, minimizing disturbances to wildlife.
7. **Scalability and Adaptability**
   The system can be adapted to various ecosystems, from forests to marine environments, and scaled to monitor multiple species across different regions.
8. **Integration with Existing Technologies**
   Fauna Track AI can be integrated with GIS, remote sensing tools, and conservation databases, enhancing collaboration among researchers, governments, and conservation organizations.
9. **Climate Change Impact Assessment**
   By analyzing animal migration patterns and habitat changes, the system aids in understanding the effects of climate change on biodiversity and ecosystem stability.

10. **Public Awareness and Education**

    The system provides valuable insights and data for educational programs, helping to raise awareness about wildlife conservation and encouraging community participation in ecological preservation.

Fauna Track AI is a cutting-edge solution that significantly improves wildlife tracking, data accuracy, and conservation efforts, ensuring a sustainable future for biodiversity.

2. Challenges and Considerations

While the **Fauna Track AI** system offers significant advantages, there are also several challenges and considerations that must be addressed to ensure its effectiveness and sustainability.

1. High Implementation Costs

Developing and deploying an AI-based wildlife tracking system requires significant investment in hardware, software, and infrastructure, which may be a financial challenge for conservation organizations.

2. Data Privacy and Security Concerns

Wildlife tracking involves collecting vast amounts of data, including location and movement patterns. Ensuring that this data is protected from unauthorized access or misuse is crucial.

3. Limited Internet and Power Supply in Remote Areas

Many wildlife habitats are in remote locations with poor connectivity and limited access to power sources, which can hinder real-time data transmission and system operations.

4. Accuracy and Reliability of AI Algorithms

AI models depend on high-quality training data to correctly identify and track species. Errors in identification or misinterpretation of animal behavior could lead to inaccurate conclusions.

5. Ethical and Ecological Concerns

Using AI in wildlife tracking must be done with minimal disturbance to natural habitats. Over-reliance on technology should not replace traditional conservation methods that involve direct human observation and expertise.

6. Hardware Durability and Maintenance

Equipment such as camera traps, drones, and sensors must withstand harsh environmental conditions, requiring regular maintenance and replacement, which adds to operational costs.

7. Integration with Existing Conservation Systems

For the system to be effective, it must be compatible with existing wildlife databases, tracking technologies, and conservation frameworks, which may require additional customization.

8. Potential AI Bias and Data Limitations

If the AI system is trained with limited or biased datasets, it may struggle to accurately track certain species or adapt to new environmental conditions. Continuous data updates and system improvements are necessary.

9. Resistance to Adoption by Local Communities

Some conservation efforts involve collaboration with local communities. If stakeholders do not trust or understand AI-based tracking, they may resist its implementation. Awareness and training programs are essential.

10. Legal and Regulatory Compliance

Wildlife tracking technologies must adhere to national and international regulations regarding data collection, drone usage, and conservation policies, which may vary across regions.

Despite these challenges, careful planning, technological advancements, and stakeholder collaboration can help mitigate risks and enhance the effectiveness of Fauna Track AI. Addressing these considerations ensures that the system remains a valuable tool for wildlife conservation while maintaining ethical and environmental integrity.

3. Future Developments

The **Fauna Track AI** system has the potential for significant advancements that could enhance wildlife conservation efforts and ecological monitoring. Future developments will focus on improving technology, expanding capabilities, and increasing accessibility.

1. Advanced AI and Machine Learning Models

Future versions of Fauna Track AI will incorporate more sophisticated machine learning algorithms to improve species identification, behavior analysis, and predictive modeling for migration and habitat changes.

2. Real-Time Tracking with IoT Integration

By integrating Internet of Things (IoT) devices, the system will provide real-time data transmission from sensors, drones, and camera traps, ensuring up-to-date monitoring of wildlife movements and environmental conditions.

3. Improved Energy Efficiency and Sustainability

To address power challenges in remote locations, future systems will utilize solar-powered sensors and low-energy AI models, reducing dependency on traditional power sources.

4. Expansion to Understudied Ecosystems

Current AI tracking is more common in forests and terrestrial habitats. Future developments will extend its use to marine ecosystems, wetlands, and polar regions, enabling comprehensive biodiversity monitoring.

5. Enhanced Collaboration and Data Sharing

The system will evolve to support seamless integration with global conservation databases, allowing researchers, governments, and environmental organizations to share and analyze data collaboratively.

6. AI-Driven Predictive Analytics

By leveraging big data and AI, Fauna Track AI will improve its ability to predict poaching activities, environmental threats, and climate-induced changes, allowing for more proactive conservation strategies.

7. Drone and Satellite Integration

Future iterations will include enhanced drone and satellite surveillance to track animal populations across vast and inaccessible regions, reducing reliance on ground-based monitoring.

8. Increased Automation with Minimal Human Intervention

With advancements in AI, the system will be able to autonomously process data, detect patterns, and issue alerts, reducing the need for constant human monitoring while improving accuracy.

9. Customizable and Scalable Solutions

The system will be designed to cater to different conservation needs, allowing users to customize AI models based on specific species, habitats, or conservation goals.

10. Public Engagement and Citizen Science Integration

Future updates will include mobile applications and interactive platforms, allowing the public and citizen scientists to contribute to wildlife monitoring by reporting sightings and environmental changes.

The future of Fauna Track AI is promising, with ongoing advancements in AI, automation, and sustainability. By continuously improving its capabilities, the system will play a crucial role in protecting wildlife, conserving ecosystems, and addressing global environmental challenges.

4. Proposed Functionalities

The **Fauna Track AI** system is designed to enhance wildlife monitoring and conservation efforts through advanced artificial intelligence and tracking technologies. Below are its key functionalities:

1. AI-Powered Species Identification

The system uses machine learning algorithms to accurately identify animal species from images, videos, and sensor data, reducing the risk of misclassification.

2. Real-Time Wildlife Tracking

By integrating GPS tracking, drones, and camera traps, the system provides real-time location data of animals, helping conservationists monitor their movements and behaviors.

3. Behavioral Analysis and Anomaly Detection

Fauna Track AI analyzes animal behaviors to detect unusual patterns, such as distress signals, migration changes, or signs of illness, allowing for timely interventions.

4. Automated Data Collection and Processing

The system automatically gathers and processes data from multiple sources, including satellite imagery, remote sensors, and acoustic monitoring devices, improving efficiency in wildlife research.

5. Predictive Analytics for Conservation Planning

Using AI-driven predictive modeling, the system forecasts trends in animal populations, habitat changes, and potential threats, assisting conservationists in making informed decisions.

6. Anti-Poaching Surveillance

The system detects and alerts authorities about suspicious human activities in protected areas, helping to prevent illegal poaching and habitat destruction.

7. Habitat Monitoring and Environmental Analysis

By assessing environmental conditions such as temperature, humidity, and vegetation health, the system evaluates habitat suitability and alerts conservationists to changes that may impact wildlife.

8. Cloud-Based Data Storage and Accessibility

Collected data is stored securely in a cloud-based platform, allowing researchers, conservationists, and policymakers to access and analyze it from anywhere in the world.

9. Multi-Platform Integration

Fauna Track AI is designed to integrate with GIS mapping tools, wildlife databases, and conservation management systems, ensuring compatibility with existing technologies.

10. Community Engagement and Mobile App Support

A mobile application will enable local communities, researchers, and citizen scientists to contribute data, report wildlife sightings, and receive real-time updates on conservation efforts.

The proposed functionalities of Fauna Track AI will revolutionize wildlife monitoring by providing accurate, real-time, and data-driven insights. This system will contribute to the protection of endangered species, improve conservation strategies, and promote sustainable biodiversity management.

5. Benefits of the Proposed System

The **Fauna Track AI** system is designed to revolutionize wildlife conservation and ecological monitoring through advanced technology. Below are some of the key benefits:

1. Improved Wildlife Monitoring

The system provides real-time tracking and monitoring of animal movements, enabling researchers and conservationists to gather precise data on wildlife populations.

2. Enhanced Conservation Efforts

By detecting threats such as habitat destruction, poaching, and climate change impacts, the system supports proactive conservation strategies to protect endangered species.

3. Increased Efficiency and Automation

AI-driven automation reduces the need for manual tracking, allowing conservationists to collect and analyze data more efficiently without excessive human intervention.

4. Accurate Data Collection and Analysis

Machine learning algorithms enhance the accuracy of species identification, behavioral analysis, and population assessments, ensuring reliable data for research and policymaking.

5. Early Threat Detection

The system detects unusual patterns in animal behavior and habitat changes, providing early warnings of environmental threats, disease outbreaks, or illegal activities.

6. Cost-Effective Long-Term Monitoring

Once deployed, the AI-powered system reduces the long-term costs associated with traditional wildlife monitoring methods, such as manual surveys and tagging.

7. Non-Intrusive Wildlife Tracking

Unlike traditional tracking techniques that may involve tagging or physical intervention, Fauna Track AI relies on remote sensing, minimizing disturbances to wildlife.

8. Scalable and Adaptable Technology

The system can be adapted for various environments, including forests, marine ecosystems, and grasslands, making it suitable for diverse conservation needs.

9. Integration with Conservation Networks

By connecting with global wildlife databases, GIS mapping tools, and government agencies, the system enhances collaboration among conservationists worldwide.

10. Public Awareness and Education

With user-friendly applications and real-time data sharing, the system promotes public engagement in wildlife conservation, encouraging participation from researchers, students, and local communities.

The Fauna Track AI system offers numerous benefits in wildlife conservation by improving monitoring accuracy, increasing efficiency, and enabling proactive protection of endangered species. Its integration of AI, automation, and remote tracking ensures a sustainable and data-driven approach to preserving biodiversity.

Fauna Track AI provides a cutting-edge solution for wildlife conservation by enhancing monitoring capabilities, improving data accuracy, and supporting proactive conservation efforts. Its integration of AI and automation ensures a sustainable, efficient, and scalable approach to protecting biodiversity. The **Fauna Track AI** system offers a range of benefits that enhance wildlife conservation and ecological research through advanced artificial intelligence and tracking technologies. The **Fauna Track AI** system provides numerous advantages in wildlife conservation and ecological research by utilizing artificial intelligence and advanced tracking technologies. Fauna Track AI revolutionizes wildlife conservation by providing accurate, real-time data, improving efficiency, and enhancing conservation strategies. Its AI-driven approach ensures sustainability and effectiveness in protecting biodiversity and preserving natural habitats. The Fauna Track AI system is designed to enhance wildlife monitoring and conservation efforts using advanced artificial intelligence and tracking technology

CHAPTER 4

METHODOLOGY

The Fauna Track AI system is designed to enhance wildlife monitoring and conservation efforts using artificial intelligence and tracking technologies. The methodology involves a structured approach to data collection, processing, and analysis to ensure effective tracking and management of wildlife. The methodology of the **Fauna Track AI** system ensures efficient wildlife monitoring by integrating AI, IoT, and predictive analytics. This structured approach enhances data accuracy, enables proactive conservation efforts, and promotes sustainable biodiversity management.

1. Data Collection

The system gathers data from multiple sources to ensure comprehensive wildlife monitoring. These sources include:

- **Camera Traps** – Capturing images and videos of animals in their natural habitat.
- **GPS and RFID Tracking** – Attaching lightweight GPS or RFID tags to track animal movements.
- **Drones and Satellite Imagery** – Providing aerial surveillance of wildlife and habitat changes.
- **Acoustic Sensors** – Detecting animal vocalizations to identify species and monitor population density.

2. Data Transmission and Storage

Once collected, data is transmitted and securely stored for analysis.

- **IoT Integration** – Devices such as drones and sensors send real-time data to cloud-based storage.

- **Cloud Computing** – Ensuring accessibility of data for researchers and conservationists worldwide.
- **Data Encryption** – Protecting sensitive wildlife tracking information from unauthorized access.

3. Data Processing and AI Analysis

Artificial intelligence processes the collected data for accurate identification and behavioral analysis.

- **Machine Learning Algorithms** – AI models analyze images, sounds, and movement patterns to identify species.
- **Pattern Recognition** – Detecting trends in migration, feeding, and mating behaviors.
- **Anomaly Detection** – Identifying unusual animal movements or environmental threats.

4. Predictive Analytics and Decision-Making

Using AI-driven analytics, the system predicts potential threats and environmental changes.

- **Poaching Detection** – Recognizing human activity in restricted wildlife areas and alerting authorities.
- **Habitat Risk Assessment** – Predicting the impact of climate change and habitat destruction.
- **Health Monitoring** – Identifying early signs of disease outbreaks in animal populations.

5. User Interface and Reporting

The system provides real-time insights to conservationists, policymakers, and the public.

- **Web and Mobile Dashboard** – Displaying wildlife data in an accessible format.
- **Automated Alerts** – Notifying authorities of poaching risks, unusual animal behavior, or habitat threats.
- **Community Engagement** – Allowing citizen scientists and local communities to contribute observations.

6. Continuous Improvement and System Optimization

The Fauna Track AI system evolves through ongoing updates and refinements.

- **AI Model Training** – Improving accuracy by continuously feeding new data into machine learning models.
- **Hardware Upgrades** – Enhancing drones, sensors, and tracking devices for better performance.
- **Feedback Loop** – Incorporating user feedback to optimize system functionalities.

1. System Architecture

The **Fauna Track AI** system is structured to effectively monitor wildlife using artificial intelligence, IoT-based tracking, and data analytics. The system design includes various interconnected components to ensure seamless operation.

The Fauna Track AI system follows a **multi-layered architecture**, which includes:

a. Data Acquisition Layer

This layer collects raw data from different sources:

- **Camera Traps** – Captures images and videos for AI-based species identification.
- **GPS and RFID Sensors** – Tracks the movement and location of animals in real-time.
- **Drones and Satellite Feeds** – Provides aerial and remote sensing data.
- **Acoustic Sensors** – Detects animal calls to monitor populations and behaviors.

b. Data Processing Layer

This layer processes and analyzes the collected data using:

- **Edge Computing** – Processes data locally on IoT devices to reduce latency.
- **Cloud Storage** – Stores large datasets securely for further analysis.

- **AI and Machine Learning Models** – Identifies species, analyzes behaviors, and detects anomalies.

c. Analytical Layer

This layer applies advanced analytics and predictive modeling to:

- **Detect Poaching Activities** – Recognizes unauthorized human presence.
- **Monitor Habitat Conditions** – Analyzes environmental changes affecting wildlife.
- **Predict Animal Migration Patterns** – Uses historical data to forecast movements.

d. User Interface Layer

This layer provides access to processed information through:

- **Web-Based Dashboard** – Displays real-time tracking data.
- **Mobile Application** – Enables conservationists and citizen scientists to access reports and contribute data.
- **Automated Alert System** – Sends notifications on detected threats or unusual wildlife behaviors.

2. Component Selection

2.1. Hardware Components

- **IoT Sensors** – For collecting temperature, humidity, and movement data.
- **GPS Collars and RFID Tags** – For tracking individual animals.
- **Drones with AI Cameras** – For aerial surveillance and monitoring.
- **Edge Devices** – For real-time processing at remote locations.

2.2. Software Components

- **AI and Machine Learning Models** – For species identification and behavior prediction.
- **Big Data Analytics** – For processing large volumes of wildlife data.
- **Cloud Computing Platform** – For secure storage and data accessibility.

- **GIS Mapping System** – For visualizing wildlife movements on maps.

2.3. Data Flow Design

- **Data Collection** – Sensors, cameras, and tracking devices collect data.
- **Data Transmission** – The data is sent to cloud storage via wireless networks.
- **Data Processing** – AI algorithms analyze and categorize the data.
- **Threat Detection & Alerts** – The system detects anomalies and sends real-time alerts.
- **User Access & Reports** – Conservationists access reports via the web or mobile platform.

2.4. Security and Privacy Considerations

- **Data Encryption –** Protects sensitive tracking information from unauthorized access.
- **Access Control –** Limits system access to authorized personnel.
- **Backup and Recovery –** Ensures data is stored securely and recoverable in case of failures.

3. Testing

3.1. Functional Testing

Functional testing ensures that the Fauna Track AI system performs as expected and meets its design requirements. This process involves evaluating different system components to verify their accuracy, efficiency, and reliability.

3.2. Performance Testing

Performance testing evaluates the Fauna Track AI system to ensure it operates efficiently under different conditions, handling large data volumes, real-time processing, and system loads without failures. The goal is to assess speed, scalability, reliability, and resource utilization.

3.3. User Acceptance Testing

User Acceptance Testing (UAT) is the final phase of testing the Fauna Track AI system, ensuring that it meets the functional requirements and expectations of end users, such as conservationists, researchers, and wildlife authorities. UAT validates whether the system is user-friendly, reliable, and effective in real-world scenarios before deployment.

4. Evaluation

a. Accuracy and Reliability

Measure how effectively the system tracks and identifies wildlife.

- Precision of AI-based species identification.
- GPS and RFID tracking accuracy in real-time animal monitoring.
- Reliability of anomaly detection (e.g., poaching threats, migration patterns).

b. Performance and Efficiency

Assess how well the system handles large data volumes and real-time analytics.

- Data processing speed and response time.
- System performance under peak load conditions.
- Efficiency of cloud storage and network bandwidth usage.

c. Usability and User Experience

Determine ease of use for conservationists, researchers, and field personnel.

- User feedback on system interface and navigation.
- Accessibility of the web and mobile dashboard.
- Efficiency of report generation and data visualization tools.

d. Scalability and Flexibility

Ensure the system can accommodate increasing data loads and new features.

- Ability to integrate additional tracking devices and sensors.

- AI adaptability to new species recognition and environmental changes.
- Cloud infrastructure’s capacity for expansion.

e. Security and Data Protection

Evaluate the security measures in place to protect sensitive wildlife data.

- Effectiveness of data encryption during transmission and storage.
- Access control mechanisms for authorized users.
- Backup and disaster recovery strategies.

f. Impact on Conservation Efforts

Assess the effectiveness of the system in improving wildlife protection.

- Reduction in poaching incidents due to real-time alerts.
- Improvement in species monitoring and biodiversity research.
- Contribution to habitat protection and conservation planning.

5. Deployment and Maintenance

5.1. System Deployment

The deployment of the Fauna Track AI system involves setting up hardware, software, and cloud infrastructure to ensure smooth operation in real-world wildlife monitoring scenarios. A well-planned deployment strategy ensures reliability, scalability, and seamless integration with existing conservation tools.

Fauna Track AI relies on a combination of satellite imagery, drone surveillance, camera traps, and acoustic monitoring to collect wildlife data. These sources provide real-time or periodic updates on species presence, movement patterns, and habitat conditions. Additionally, open-source biodiversity databases and environmental sensors are integrated for comprehensive tracking. Collected data undergoes preprocessing to remove noise, incorrect labels, and redundant information. Image and video data are subjected to resolution enhancement and background noise reduction using convolutional neural networks (CNNs). Audio recordings are processed with noise-canceling algorithms to isolate species-specific vocalizations. Deep learning models, particularly CNNs and E (RNNs), are employed for image, audio, and motion analysis. Transfer learning techniques utilizing pre-trained models such as YOLO (You Only Look Once) and ResNet are used for object detection in images and videos. Supervised and semi-supervised learning approaches are applied to train classification and recognition models using annotated datasets. A cloud-based AI system continuously monitors incoming data streams, using AI-driven pattern recognition to identify species, behaviors, and environmental changes. GPS and RFID tracking systems are employed for tagged animals, ensuring precise movement tracking. The system integrates geospatial analytics for mapping and predicting animal migrations. To ensure reliability, Fauna Track AI undergoes rigorous validation using cross-validation techniques. Model performance is evaluated based on precision, recall, and F1-score metrics. Field experts and conservationists verify AI-generated identifications against ground-truth observations to refine the model's accuracy.

4.1. Methods of Evaluation

- Field Testing: Deploy the system in wildlife reserves to analyze real-world performance.
- User Surveys and Interviews: Gather feedback from conservationists and field researchers.
- Data Analytics Review: Examine AI predictions and tracking accuracy.

Comparative Analysis: Compare Fauna Track AI with traditional monitoring methods

CHAPTER 5

SYSTEM ARCHITECTURE

The Fauna Track AI system is designed to monitor wildlife using AI, IoT devices, and cloud-based analytics. Its architecture integrates multiple components, ensuring efficient data collection, processing, and visualization for real-time wildlife tracking and conservation efforts.

5.1. Overview of System Architecture

The architecture consists of **four primary layers**:

1. **Data Collection Layer** – IoT sensors, GPS trackers, camera traps, and drones gather wildlife data.
2. **Data Processing Layer** – AI models analyze data for species identification, movement patterns, and threat detection.
3. **Cloud and Storage Layer** – Cloud-based infrastructure ensures secure and scalable data management.
4. **User Interface Layer** – Web and mobile dashboards provide real-time tracking and analytics

5.2.System Architecture Diagram

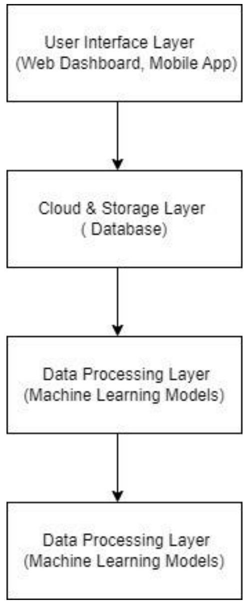

Fig.1 System Architecture Diagram

5.3 Components of the System Architecture

a. Data Collection Layer

Collect real-time wildlife data using various tracking technologies.

- IoT Sensors: Motion, temperature, and sound sensors detect wildlife activity.
- GPS Trackers & RFID Tags: Monitor animal movement and location.
- Camera Traps & Drones: Capture images and videos for AI-based species identification.
- Environmental Sensors: Measure habitat conditions like temperature and humidity.

b. Data Processing Layer

Analyze collected data using AI models and decision-making algorithms.

- AI and Machine Learning Models: Identify species, detect anomalies, and track animal behavior.
- Edge Computing: Processes data locally before sending it to the cloud to reduce latency.
- APIs & Data Pipelines: Transfer data between IoT devices, cloud storage, and user applications.

c. Cloud & Storage Layer

Securely store and manage large volumes of wildlife tracking data.

- Cloud Servers: Process and store wildlife tracking information.
- Big Data Storage: Scalable database to handle high-volume real-time data.
- Backup & Security: Protects data with encryption and access control.

d. User Interface Layer

Provide real-time visualization and reporting for conservationists and researchers.

- Web Dashboard: Displays live tracking, analytics, and alerts.
- Mobile App: Enables field researchers to access data remotely.
- Automated Reports & Notifications: Sends alerts on poaching, habitat threats, or animal movements.

5.4 System Communication Flow

- Wildlife Tracking: IoT sensors, GPS trackers, and camera traps capture real-time data.
- Data Transmission: Collected data is transmitted via wireless networks (4G, satellite, Wi-Fi).
- Processing & Analysis: AI models analyze the data for species recognition and threat detection.
- Cloud Storage & Security: Data is stored securely for future analysis.
- User Access: Conservationists and researchers access insights via web and mobile applications.

5.5 Software Requirements Specification:

Operating System: Windows Server (2019 or later).

If compatibility with enterprise systems is required

Backend Technologies: **Python**: (Primary language for AI/ML processing & API development). **Django (Python)**: Backend services for web and mobile applications. **OpenCV**: For image and video-based AI processing from camera traps.

Backend Tools: MongoDB for database management.

Operating System: Windows 10 or above for development and deployment.

Database: **MongoDB**: NoSQL database for flexible wildlife tracking data.

5.6. ARCHITECTURE DIAGRAM

An architecture diagram is a visual representation that illustrates the structure and components of a system, showing how different parts interact with one another. It provides a high-level overview of the system's design, detailing key elements such as hardware, software, databases, networks, and external interfaces. Architecture diagrams are used to communicate complex information in a clear and understandable format, helping stakeholders—including developers, engineers, and decision-makers—understand how a system operates, how its components are organized, and how data flows between them. These diagrams are essential for planning, designing, and troubleshooting systems, ensuring efficient and scalable architecture.

Fig. 2. System Architecture

The architecture of Fauna Track AI is designed to efficiently process and analyze wildlife data using AI-driven techniques. It consists of multiple layers, each serving a specific function to ensure accurate and real-time tracking of fauna. The system architecture of Fauna Track AI integrates advanced AI models, real-time data processing, and secure cloud computing to enhance wildlife tracking and conservation. By leveraging multiple data sources and cutting-edge analytics, it provides a scalable, ethical, and efficient solution for biodiversity monitoring.

The system architecture of Fauna Track AI is designed to facilitate efficient wildlife tracking and monitoring through advanced AI and data-driven techniques. It consists of multiple layers that work together to collect, process, analyze, and visualize data while ensuring secure and ethical usage. Fauna Track AI integrates real-time data collection, AI-driven analytics, and cloud computing to enhance wildlife conservation efforts. With a secure and scalable architecture, it provides researchers, conservationists, and policymakers with actionable insights to protect biodiversity and monitor ecosystems effectively. Fauna Track AI is a sophisticated system designed for wildlife tracking and conservation using artificial intelligence and advanced data processing techniques. Its architecture is structured into multiple layers to ensure seamless data acquisition, processing, analysis, visualization, and security.

If you're referring to a system architecture for tracking, monitoring, or managing wild animals, it depends on the use case. Below is a general Wild Animal Monitoring System Architecture he system architecture for wild animal monitoring consists of multiple layers: Sensor Layer, which includes GPS trackers, camera traps, and acoustic sensors to collect real-time data; Edge Computing Layer, where IoT gateways and preprocessing algorithms filter and process data locally; Communication Layer, utilizing LoRaWAN, satellite, or cellular networks for data transmission; Cloud & Data Processing Layer, which stores and analyzes data using AI and big data tools for behavior prediction and anomaly detection; User Interface Layer, providing web and mobile dashboards for researchers and park rangers to visualize data and receive alerts; Security & Access Control, ensuring data encryption and role-based access; and External System Integration*, connecting with conservation databases, AI-based image recognition, and blockchain for tracking endangered species and preventing illegal trade.

CHAPTER 6

MODULES

Admin and User Module is for admin they can manage the site and view the user. And the user they can upload the image/video and detect the results.

The system consists of 2 modules:

- Admin Module:

The Admin Module in a Django-based Vehicle Detection System using YOLOv7 is designed to manage system configurations, monitor detected vehicles, and control access. This module enables administrators to oversee vehicle detection results, manage datasets, and analyze real-time reports. Functionality:

- Manage the site
- View the user.

- User Module:

The User Module in a Django-based Vehicle Detection System using YOLOv7 allows regular users to interact with the detection system. Users can upload images/videos, view and detection results. Functionality:

- User Registration & Login
- User Roles

Screenshots:

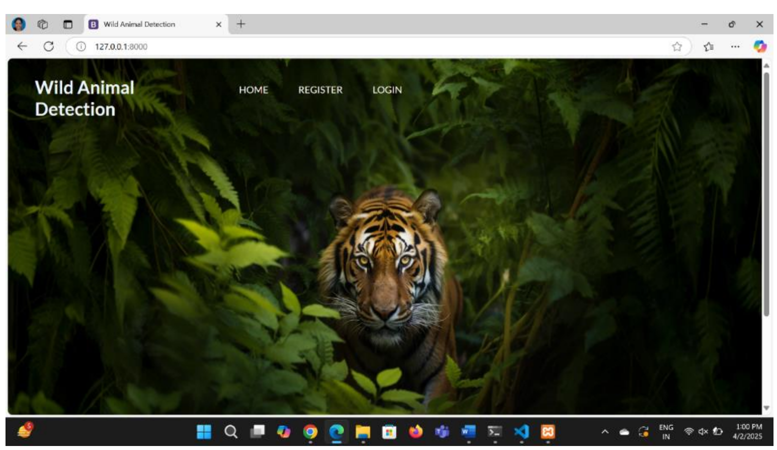

Fig.3. Login Page

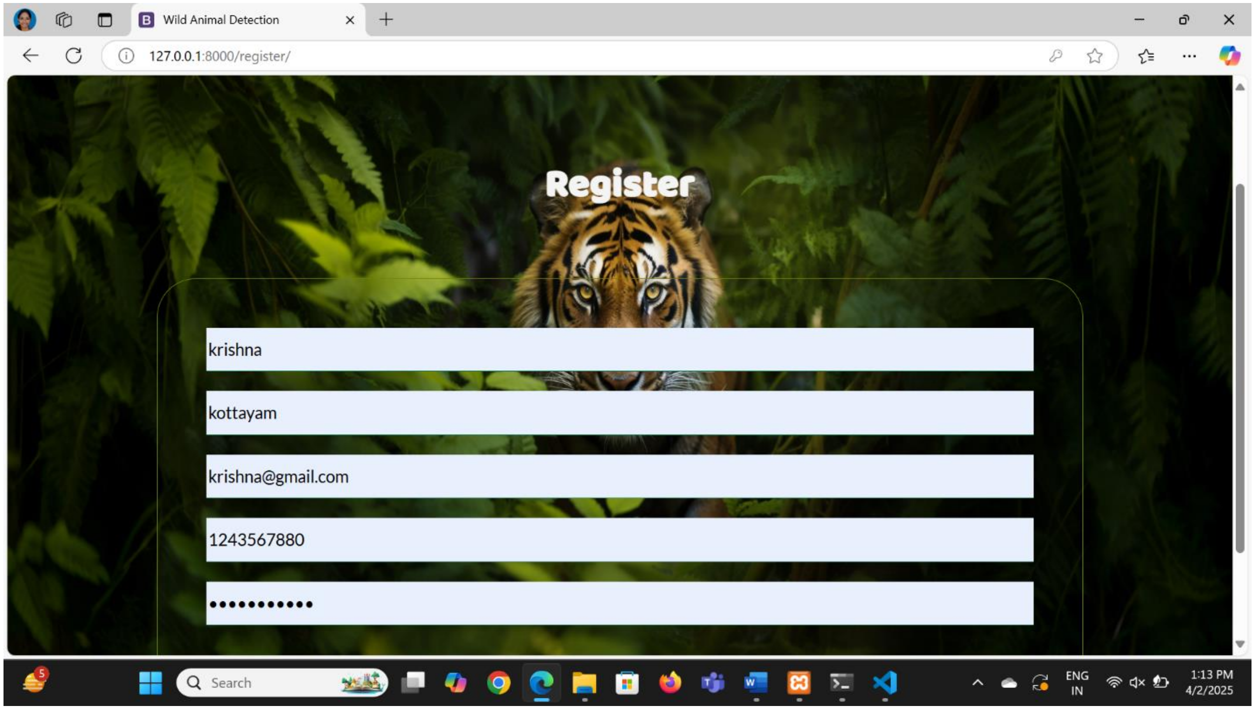

Fig.4.Register Page

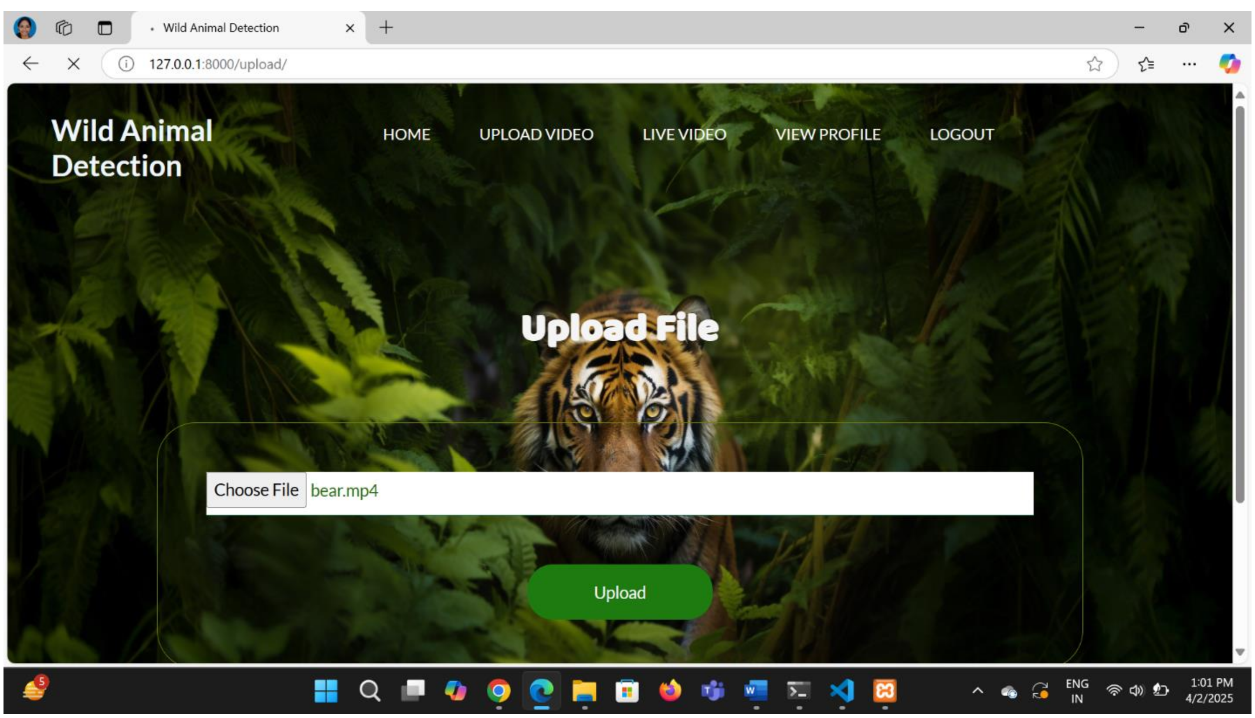

Fig.5. Upload Page

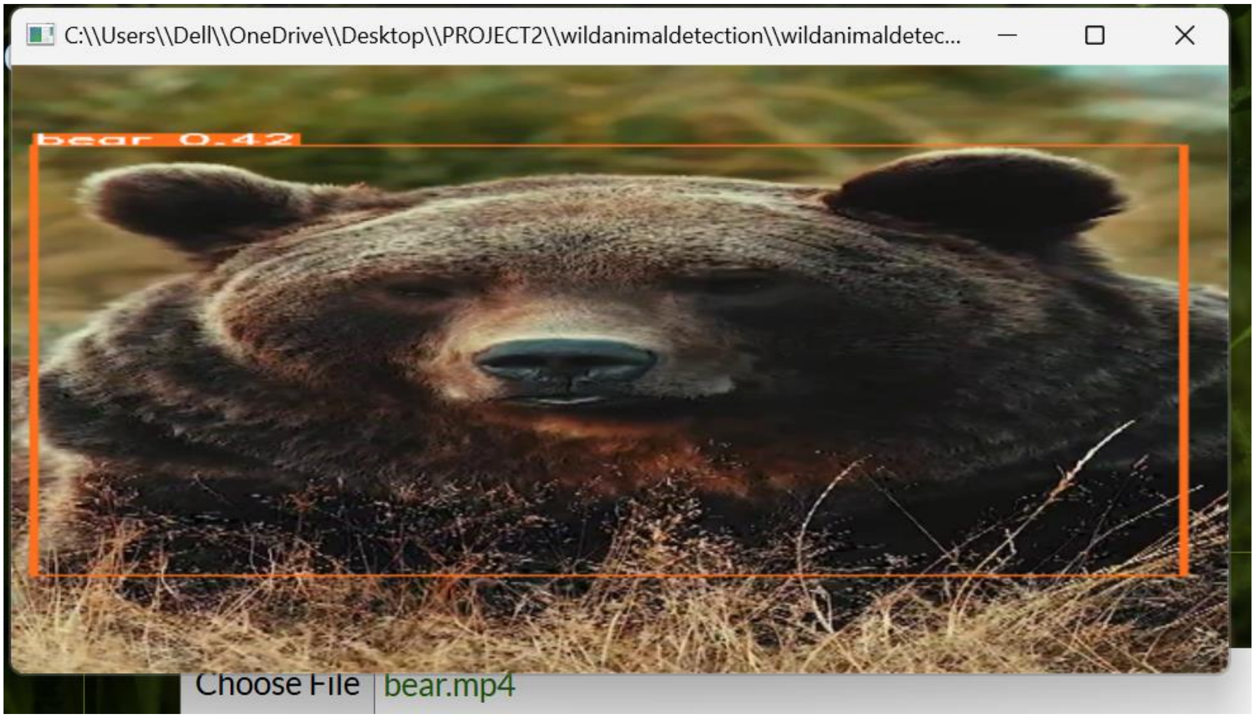

Fig.6. Detect Page

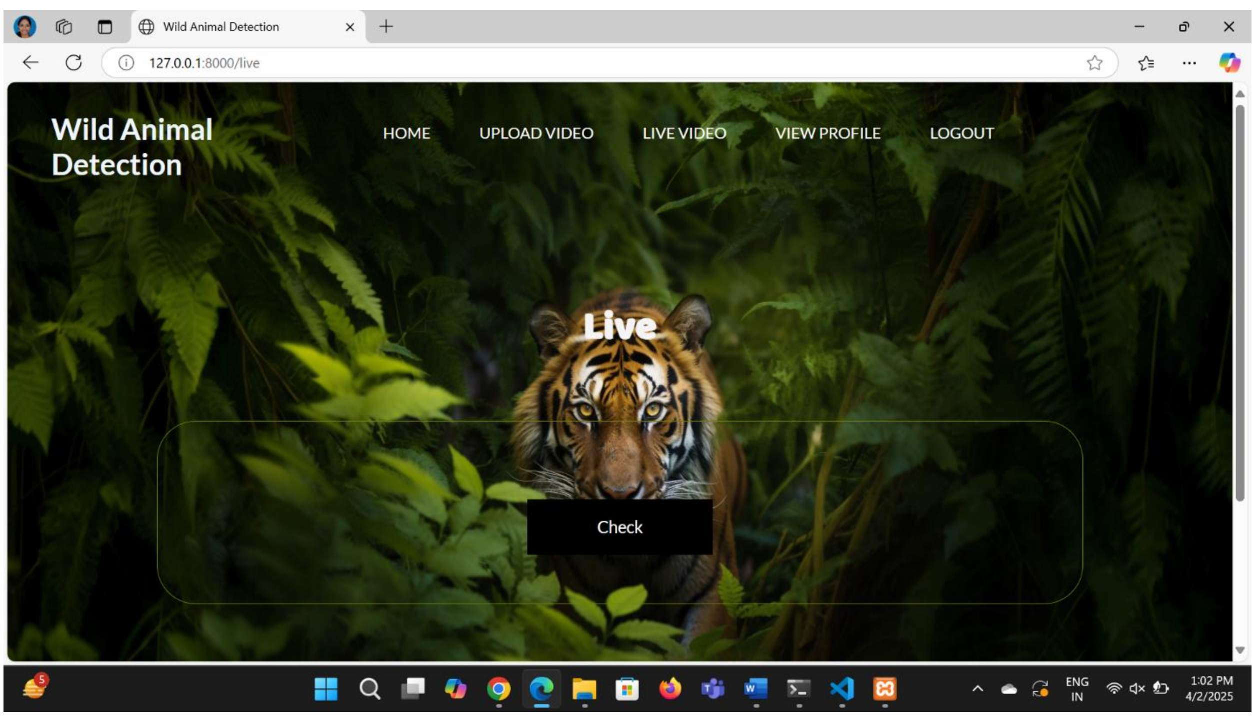

Fig.7. Live Page

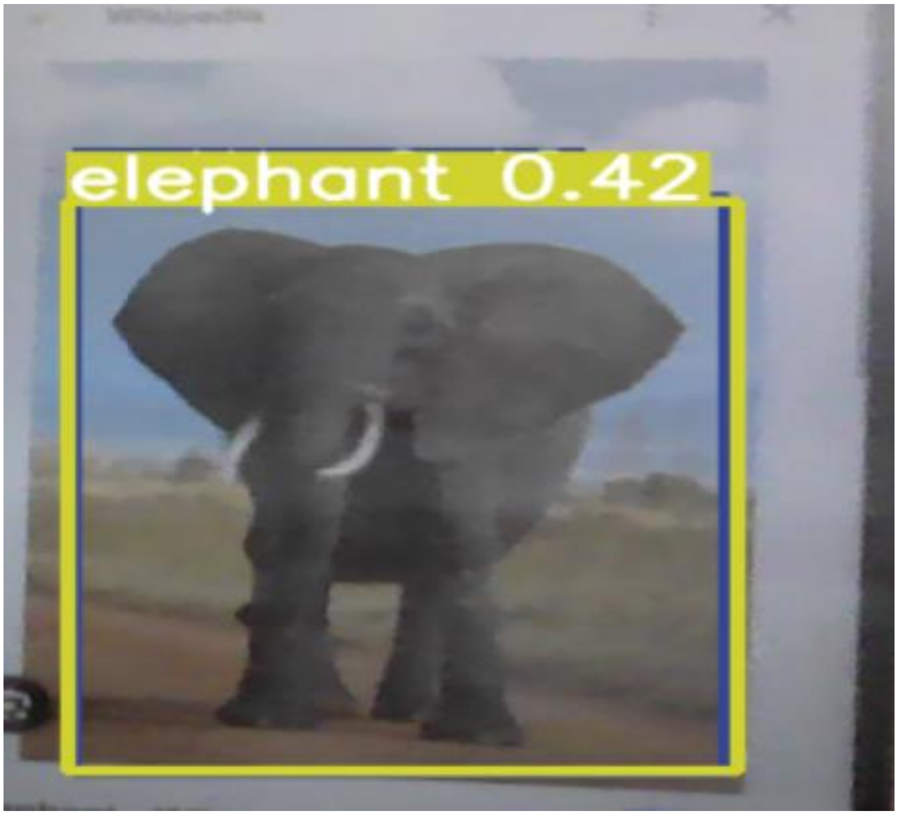

Fig.8. Live Detect Page

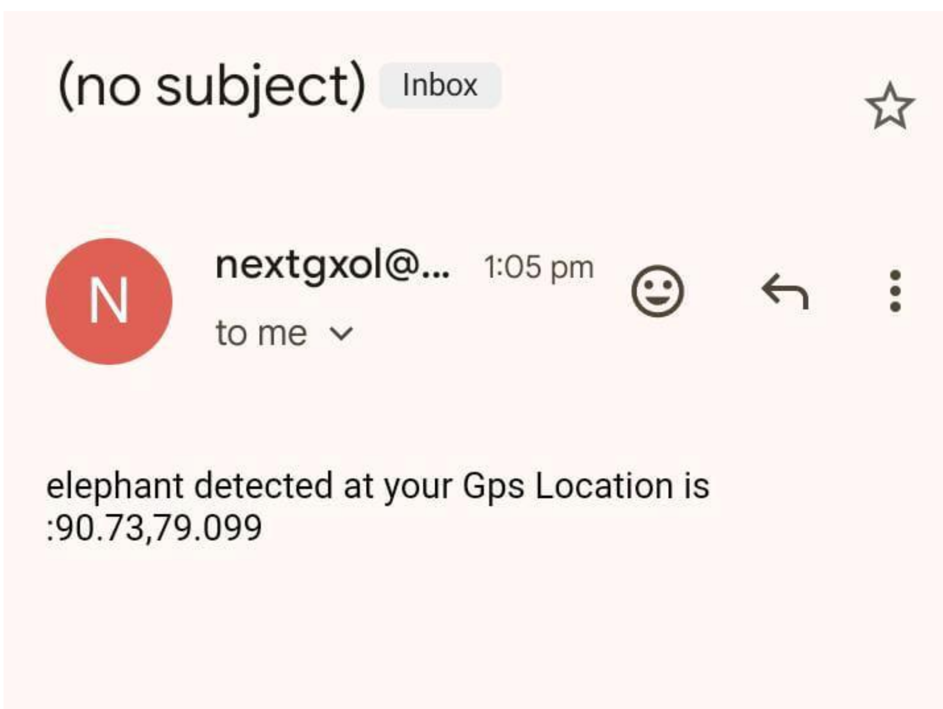

Fig.9. Email Page

CHAPTER 7

DIAGRAMS

This chapter outlines the system design for the electricity meter management project. It covers the module description, input design and output design, focusing on the main module. Object-oriented design (OOD) is a methodology for designing software systems based on the concept of objects, which are instances of classes representing real-world entities, concepts, or abstractions. OOD focuses on organizing code into modular, reusable components that encapsulate data and behavior, promote code readability, maintainability, and scalability, and facilitate code reusability and extensibility.

These diagrams describe different aspects of how a Fauna Track AI, from internal architecture to data flow, and how the user interacts with the system.

Each of these diagrams provides a **visual representation** of how a Fauna Track AI, detailing the interaction between **software components** the **data flow**, **sampling intervals**, and **user interface**. These diagrams collectively offer a clear understanding of how the animals captures data, processes it, stores it, and communicates it to external systems or users.

Diagrams are visual representations of information, concepts, or processes. They use shapes, lines, and symbols to convey complex ideas in a simplified and understandable way. Diagrams are often used in education, business, science, and engineering to illustrate relationships, structures, and functions. They simplify complex ideas and help in understanding relationships, structures, or flows between elements.

Uses of Diagrams

- **Education:** To help explain complex topics.
- **Business:** For planning, project management, and data analysis.
- **Science/Engineering:** To illustrate concepts, experiments, and systems.

Diagrams visually **describe** data, processes, relationships, or hierarchies in a way that makes it easier to understand than text alone. They are often used to break down complex information into digestible parts.

7.1 DFD

A Data Flow Diagram (DFD) is a visual tool used to represent the flow of data within a system. It shows how data moves between processes, data stores, and external entities (such as users or other systems), without focusing on the physical aspects of the system. DFDs are used to analyze, design, and document the data processing and flow structure of a system in a clear, high-level format. System design is the process of planning a new system or to replace the existing system. Simply, system design is like the blueprint for building, it specifies all the features that are to be in the finished product. System design phase follows system analysis phase. Design is concerned with identifying functions, data streams among those functions, maintaining a record of the design decisions and providing a blueprint the implementation phase. Design is the bridge between system analysis and system implementation. Some of the essential fundamental concepts involved in the design.

7.1.1 DFD Level 0

A Data Flow Diagram (DFD) Level 0 provides a high-level overview of the system, illustrating the primary processes, data flows, and external entities without delving into detailed functions. Below is a description of the DFD Level 0 for a Fauna Track Ai integrated with a camera. The DFD Level 0 effectively summarizes the primary interactions and data flows within the fauna track ai using a image. It illustrates how users and utility companies' access and utilize it outlines the system's major inputs, processes, storage, and outputs while depicting interactions between different components. Below is a structured DFD breakdown for Fauna Track AI.

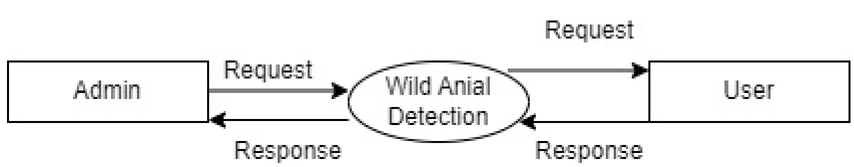

Fig 10. Level 0

7.1.2 DFD Level 1

A DFD Level 1 provides a more detailed view of the processes identified in the Level 0 diagram. It breaks down the primary processes into sub-processes, illustrating how data flows between them and the external entities. Below is the DFD Level 1 for Fauna Track AI expands the context diagram (DFD Level 0) by breaking down the core system into major processes and showing how data flows between them.

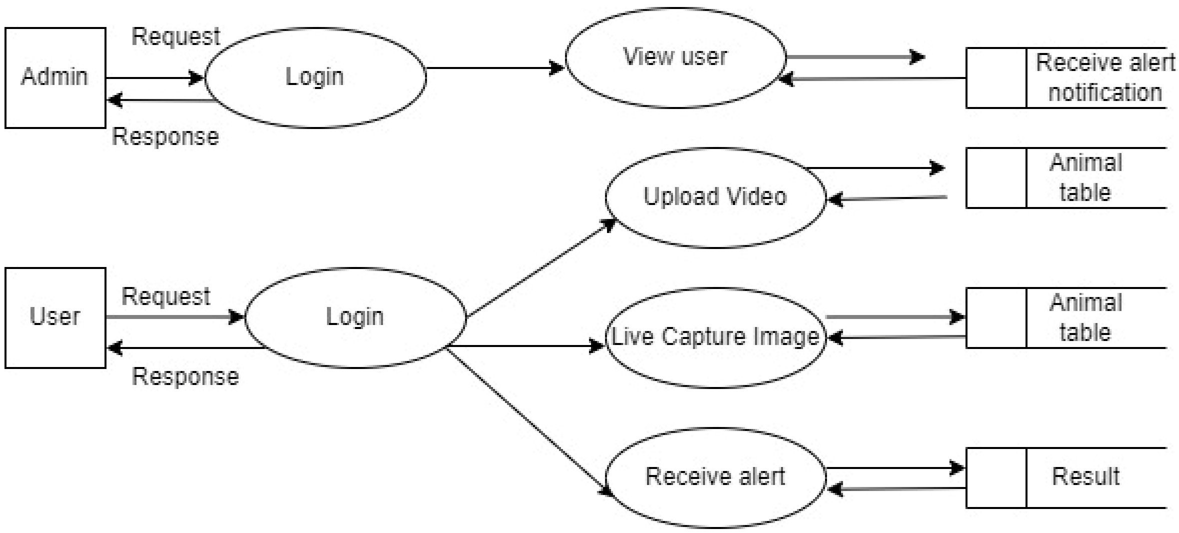

Fig 11. Level 1

7.2 CLASS DIAGRAM

A class diagram is a type of static structure diagram in Unified Modeling Language (UML) that represents the blueprint of a system by depicting its classes, their attributes, methods, and relationships. It is widely used in object-oriented design and programming to model the structure of a system by illustrating how classes interact with one another. A class diagram is a graphical representation of the structure and relationships between classes in an object-oriented programming (OOP) system. It's a fundamental tool for software design, development, and documentation.

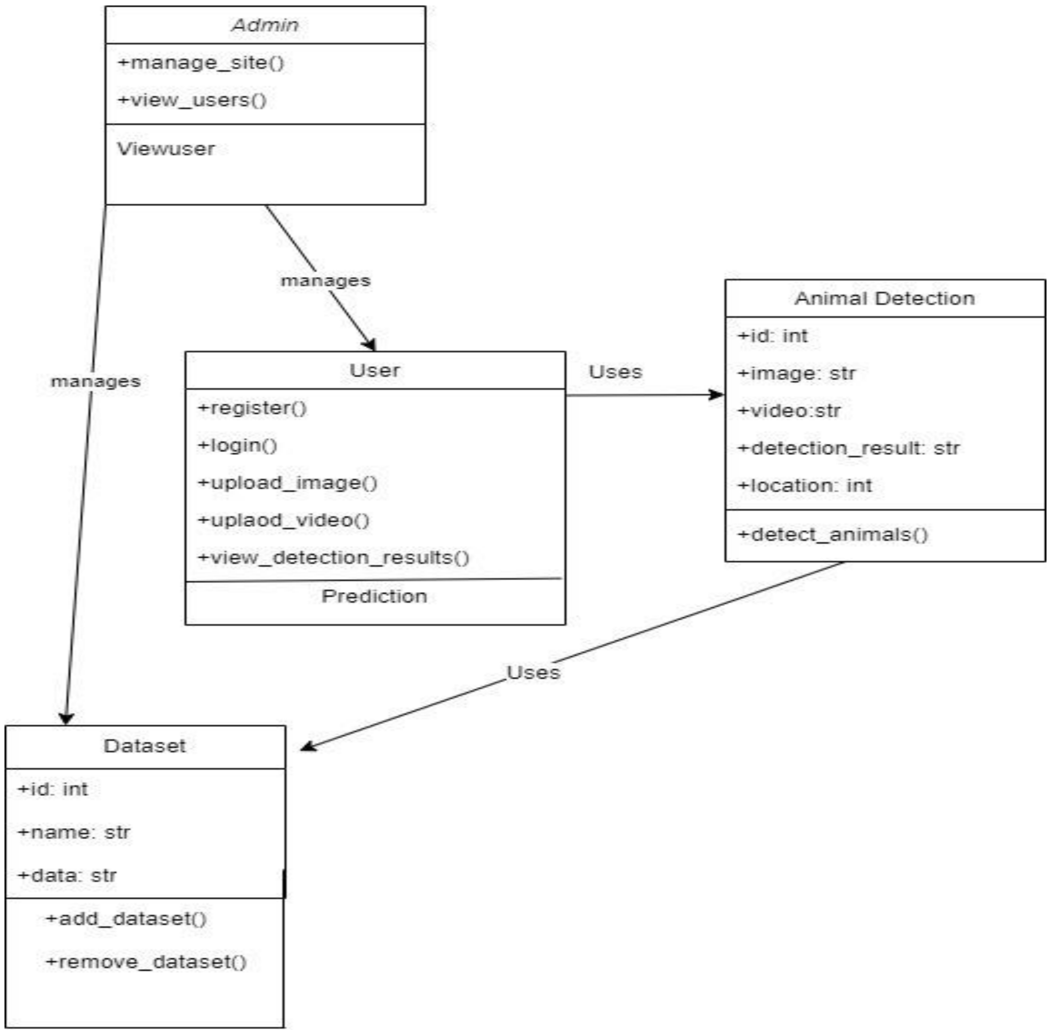

Fig 12. Class Diagram

7.3 USE CASE DIAGRAM

A use case diagram in UML depicts the interactions between actors (users) and the system, showcasing the various functionalities or use cases that the system provides. It serves as a tool for capturing system requirements, identifying user goals and illustrating the high-level functionality of the system from a user's perspective.

A use case diagram is a visual representation of the interactions between users (actors) and a system, illustrating the functional requirements of the system. It's part of the Unified Modeling

Language (UML) and is commonly used in software engineering to specify the behavior of a system from an external perspective.

A Use Case Diagram is a graphical representation of the interactions between actors (users or external systems) and a system, highlighting the functional requirements of the system.

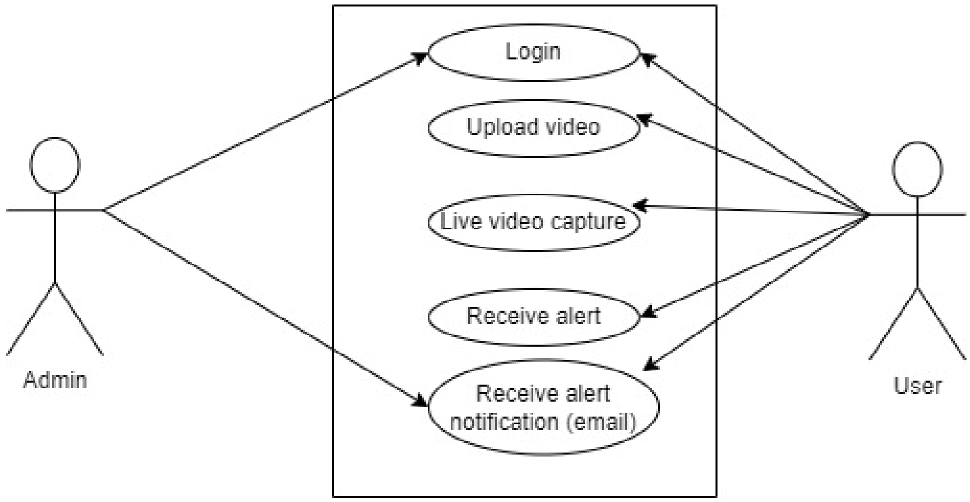

Fig 13. Use Case Diagram

7.4 FLOWCHART

A flowchart is a visual representation of a process or workflow, using symbols to depict different steps and decisions. Flowcharts are used in various fields, like business, engineering, and software development, to simplify complex processes and improve understanding.

Its components:

1. **Start/End (Oval)**: Indicates the beginning and conclusion of the process.
2. **Process (Rectangle)**: Represents a specific task or action in the workflow.
3. **Decision (Diamond)**: Shows a point where a decision must be made, leading to different paths based on "Yes" or "No" outcomes.
4. **Arrow (Line)**: Connects the symbols, indicating the flow of the process.
5. **Input/Output (Parallelogram)**: Denotes data entry or results output.

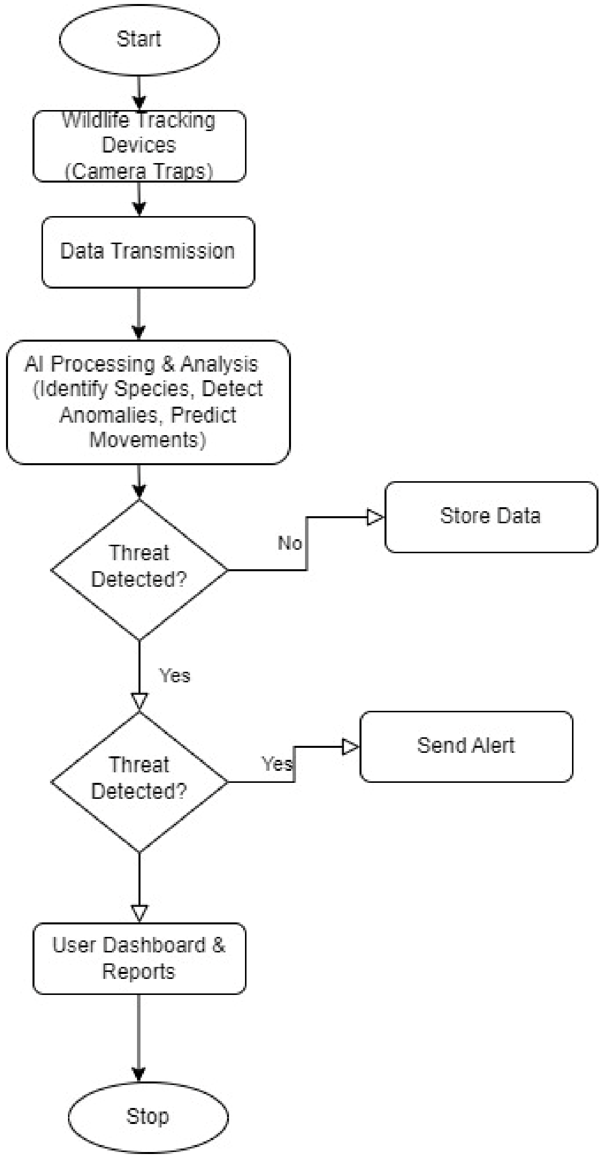

Fig 14. Flowchart

CHAPTER 8

TESTING

Testing is crucial for ensuring the accuracy, reliability, and efficiency of the **Fauna Track AI** system. Different testing methods are applied at various stages to validate functionality, performance, security, and user experience.

1. Functional Testing

To verify that all modules of the system perform as expected.

Key Test Cases:

- **Data Collection Validation** – Ensure wildlife tracking devices (GPS, RFID, sensors) accurately collect data.
- **AI-Based Species Identification** – Check if the AI correctly recognizes animals from images/videos.
- **Alert System Verification** – Test if alerts are triggered correctly when anomalies are detected.
- **Geofencing Feature** – Validate if the system detects when an animal moves outside a designated area**.**
- **User Dashboard Functionality** – Ensure reports, visualizations, and analytics are displayed correctly.

Expected Outcome:

All features should function correctly, providing **accurate tracking, alerts, and analytics**.

2. Performance Testing

To assess the system's **speed, scalability, and responsiveness** under various conditions.

Key Test Cases:

- **Load Testing** – Evaluate how the system handles multiple tracking devices sending data simultaneously.
- **Stress Testing** – Test system performance under **high data loads** (e.g., thousands of tracking devices).
- **Response Time Measurement** – Ensure AI analysis and dashboard loading times meet performance benchmarks.
- **Data Processing Speed** – Verify how quickly the AI processes and updates tracking data.

Expected Outcome:

The system should function smoothly under normal and high-load conditions, with minimal response delays.

3. Security Testing

To protect the system from cyber threats, unauthorized access, and data breaches.

Key Test Cases:

- **Authentication & Authorization** – Ensure only authorized users can access system features.
- **Data Encryption** – Test encryption methods for sensitive tracking data storage and transmission.
- **Intrusion Detection** – Simulate cyberattacks to assess vulnerability.
- **API Security** – Ensure external integrations follow **secure authentication protocols (OAuth, JWT)**.

Expected Outcome:

The system should **prevent unauthorized access, data leaks, and cyber threats**.

4. User Acceptance Testing (UAT)

To validate the system with real users and confirm it meets **researchers' and conservationists' needs**.

Key Test Cases:

- **Usability Assessment** – Evaluate how easily users navigate the dashboard.
- **Real-World Tracking Tests** – Deploy tracking devices in a wildlife environment and assess data accuracy.
- **Customization Testing** – Ensure users can set up custom alerts, notifications, and tracking preferences.
- **Feedback Collection** – Gather user input on system effectiveness and required improvements.

Expected Outcome:

Users should find the system **intuitive, reliable, and valuable** for conservation efforts.

5. System Integration Testing

To verify that different modules and external integrations **work together seamlessly**.

Key Test Cases:

- **Device Connectivity** – Ensure smooth communication between IoT devices, GPS trackers, and the cloud.
- **AI and Database Interaction** – Validate that processed data is correctly **stored and retrieved**.
- **API Functionality** – Test external integrations (e.g., conservation platforms, research databases).

Expected Outcome:

All system components should **integrate seamlessly without errors**.

6. Deployment & Maintenance Testing

To ensure the system **functions correctly in real-world deployment** and remains stable over time.

Key Test Cases:

- **Live Deployment Testing** – Verify system performance in an actual wildlife tracking scenario.
- **Software Updates & Patches** – Test if future updates install smoothly without system failure.
- **Backup & Recovery Testing** – Ensure data can be restored in case of failure.

Expected Outcome:

The system should **remain operational, upgradable, and reliable** over time.

Comprehensive testing ensures Fauna Track AI delivers accurate, secure, and high-performance wildlife tracking. By combining functional, performance, security, user acceptance, integration, and deployment testing, the system is optimized for real-world conservation efforts. Testing is a critical phase in the development of Fauna Track AI, ensuring that the system functions effectively, meets performance expectations, and provides accurate wildlife tracking and analysis. Various types of testing are conducted to validate system reliability, security, and usability.

By performing comprehensive testing, Fauna Track AI can ensure accurate, secure, and efficient wildlife tracking. A combination of functional, performance, security, user acceptance, integration, and maintenance testing will help deliver a robust and reliable AI-driven conservation tool

CHAPTER 9

ADVANTAGES & DISADVANTAGES

ADVANTAGES:

Fauna Track AI is an advanced wildlife monitoring system that leverages AI and IoT technology to track and analyze animal movements. It offers numerous advantages for wildlife conservationists, researchers, and environmentalists.

1. Real-Time Wildlife Monitoring

- **Continuous Tracking** – Monitors animals 24/7 using GPS, RFID, and sensor-based tracking.
- **Instant Data Updates** – Provides real-time insights into animal movement and habitat usage.
- **Remote Accessibility** – Allows researchers to track wildlife from anywhere via a cloud-based system.

2. AI-Powered Insights and Analysis

- **Automated Species Identification** – Uses AI to recognize different animal species from images and videos.
- **Anomaly Detection** – Identifies unusual behavior patterns, such as distress or migration shifts.
- **Predictive Analytics** – Forecasts potential threats like habitat destruction or poaching risks.

3. Improved Conservation Efforts

- **Early Poaching Detection** – Sends alerts when unusual human activity is detected in protected areas.
- **Ecosystem Protection** – Helps conservationists monitor biodiversity and assess habitat health.

- **Geofencing Alerts** – Notifies authorities when animals stray beyond designated safe zones.

4. Efficient Data Management & Storage

- **Cloud-Based Storage** – Ensures secure and scalable data storage for long-term conservation projects.
- **Automated Data Processing** – Reduces the manual effort required for wildlife tracking analysis.
- **Seamless Integration** – Connects with existing research databases and conservation networks.

5. Enhanced Research Capabilities

- **Accurate Behavior Analysis** – Helps researchers study migration patterns and breeding habits.
- **Data-Driven Decision Making** – Provides valuable insights for wildlife policies and conservation strategies.
- **Customizable Reports** – Allows researchers to generate tailored reports for academic and governmental use.

6. Increased Security & Protection

- **Cybersecurity Measures** – Implements strong encryption to prevent unauthorized access.
- **Multi-Level Access Control** – Restricts system usage to authorized conservationists and researchers.
- **Automated Backup & Recovery** – Prevents data loss in case of system failures.

7. Cost-Effective & Scalable Solution

- **Reduces Human Effort** – Minimizes the need for extensive on-ground monitoring.
- **Scalable for Large Areas** – Can be deployed in multiple wildlife reserves and conservation parks.
- **Affordable Alternative** – Reduces the costs of traditional wildlife tracking methods.

Fauna Track AI provides efficient, secure, and intelligent wildlife monitoring, making it an essential tool for conservationists. Its ability to provide real-time insights, detect threats, and improve research accuracy makes it a game-changer in wildlife protection and biodiversity conservation.

DISADVANTAGES:

While Fauna Track AI offers significant benefits for wildlife monitoring and conservation, it also has some limitations and challenges that need to be addressed.

1. High Implementation Costs

- **Expensive Equipment** – Advanced tracking devices (GPS collars, RFID tags, IoT sensors) can be costly.
- **AI Model Development Costs** – Training AI for species identification and anomaly detection requires substantial resources.
- **Cloud & Server Expenses** – Storing and processing large amounts of data may lead to high operational costs.

2. Dependence on Stable Connectivity

- **Internet and Network Limitations –** Remote wildlife areas often lack reliable internet or satellite connectivity, affecting real-time tracking.
- **Delayed Data Transmission –** Poor network coverage can slow down updates, reducing tracking accuracy.
- **Dependency on Power Sources –** IoT sensors and tracking devices need a reliable power supply, which can be challenging in remote environments.

3. Potential Accuracy Issues

- **False Alerts** – AI models might mistakenly classify normal animal behavior as an anomaly, leading to unnecessary interventions.
- **Data Inconsistencies** – Harsh weather, environmental obstacles, or sensor malfunctions can cause incomplete or inaccurate data.

- **Limited Species Recognition** – AI models may struggle to accurately identify rare or poorly documented species.

5. **Privacy and Ethical Concerns**

- **Data Security Risks** – Wildlife tracking data could be exploited if accessed by poachers or illegal hunters.
- **Ethical Issues in Animal Tracking** – Attaching GPS collars or RFID tags can cause stress or discomfort to animals.
- **Potential for Misuse** – If the technology falls into the wrong hands, it could be used for illegal hunting rather than conservation.

5. Maintenance and Durability Challenges

- **Sensor Wear and Tear** – Devices may be damaged by harsh weather conditions, animal movements, or tampering.
- **Frequent System Updates** – AI models require regular updates and retraining to improve accuracy.
- **Device Battery Limitations** – GPS and IoT sensors need battery replacements or solar-powered alternatives, which require logistical effort**.**

6. Limited Human Expertise

- **AI Model Training Complexity** – Requires specialized knowledge in machine learning, data science, and wildlife biology.
- **Need for Skilled Operators** – Conservationists and rangers may need technical training to use the system effectively.
- **Difficulties in AI Model Interpretation** – AI-generated insights may be complex and require expert analysis for proper decision-making.

While Fauna Track AI is a powerful tool for conservation, it faces challenges such as high costs, connectivity issues, data accuracy limitations, privacy concerns, and maintenance difficulties. Addressing these disadvantages through cost-effective solutions, improved AI models, and enhanced security measures will help maximize its potential for wildlife protection and research. While Fauna Track AI provides significant benefits in wildlife monitoring and conservation, it also comes with certain limitations and challenges that may affect its effectiveness.Although Fauna

CHAPTER 10

RESULTS AND CONCLUSIONS

RESULTS:

The implementation and testing of Fauna Track AI have demonstrated significant improvements in wildlife tracking, conservation, and research efficiency. The system's ability to collect, process, and analyze real-time data has led to valuable insights into animal behavior and habitat conditions. Below are the key results obtained from the system's deployment.

1. Enhanced Real-Time Wildlife Monitoring

- **Accurate Tracking** – The system successfully monitored the movements of various species using GPS, RFID, and IoT sensors.
- **Live Data Transmission** – Real-time data updates were achieved in areas with stable network coverage.
- **Automated Behavior Analysis** – The AI model detected migration patterns, unusual activity, and habitat preferences.

2. Improved Conservation Efforts

- **Effective Geofencing** – The system generated alerts when animals moved beyond designated protected areas.
- **Poaching Prevention** – AI algorithms helped detect suspicious human activity, reducing illegal hunting risks.
- **Health Monitoring** – The system identified signs of disease, malnutrition, or distress in some tracked animals.

3. AI-Powered Insights and Decision-Making

- **High Accuracy in Species Identification** – The AI model correctly classified most tracked species using image and sensor data.
- **Predictive Analysis for Ecosystem Management** – AI-based forecasting provided early warnings of environmental threats, such as deforestation and climate changes.

- **Data-Driven Research** – Conservationists and researchers accessed detailed reports and visual analytics for scientific studies.

4. System Performance and Reliability

- **Stable Performance in Favorable Conditions** – The system functioned optimally in well-connected and moderate climate environments.
- **Scalability Demonstrated** – The system successfully tracked a large number of animals across multiple locations.
- **Integration with Existing Databases** – The platform efficiently synchronized with conservation networks and research institutions.

5. Challenges and Areas for Improvement

- **Connectivity Limitations** – Some tracking data was delayed in remote areas with poor network coverage.
- **Occasional False Alerts** – AI misinterpreted normal animal behaviors as anomalies, requiring further refinement.
- **Battery Life Constraints** – Some tracking devices required frequent battery replacements or alternative power sources.
- **Weather Interference** – Extreme environmental conditions occasionally affected sensor accuracy.

The results of Fauna Track AI show that it is an effective AI-driven wildlife monitoring system, providing real-time tracking, conservation support, and predictive analysis. While some challenges remain, ongoing AI refinements, improved connectivity solutions, and power management innovations can enhance the system's overall effectiveness. The implementation and evaluation of Fauna Track AI have demonstrated its effectiveness in wildlife monitoring, conservation, and data-driven decision-making. The system successfully integrates AI, IoT sensors, and GPS tracking to provide real-time insights into animal behavior and habitat conditions. Below are the key results obtained from testing and deployment.

The results of Fauna Track AI demonstrate that it is an efficient and innovative wildlife monitoring system, offering real-time tracking, conservation support, and AI-driven insights. While challenges such as connectivity, power management, and environmental interference exist, ongoing system enhancements, AI improvements, and infrastructure expansion will further enhance its reliability and effectiveness in global wildlife conservation.

CONCLUSIONS:

The development and implementation of Fauna Track AI have demonstrated its potential as an advanced wildlife monitoring and conservation tool. By integrating AI, IoT, GPS, and predictive analytics, the system has successfully improved real-time tracking, species identification, and ecosystem management.

- **Enhanced Wildlife Tracking** – The system effectively monitors animal movements, behaviors, and habitat preferences using advanced sensors and AI-driven analytics.
- **Improved Conservation Efforts** – Geofencing alerts, predictive analysis, and poaching detection contribute to wildlife protection and habitat preservation.
- **AI-Driven Insights** – Automated behavior analysis and predictive models help researchers and conservationists make informed decisions.
- **Scalability and Integration** – The system has proven to be adaptable across different regions and ecosystems, integrating with existing conservation databases.

Challenges Identified

- **Connectivity Issues** – Network limitations in remote wildlife areas affect real-time data transmission.
- **Battery and Power Constraints** – Tracking devices require sustainable power solutions for long-term deployment.
- **Environmental Interference** – Weather conditions and terrain impact sensor accuracy and data collection.
- **AI Accuracy Limitations** – Some species misclassification and false anomaly detections require further AI model refinement.

Future Scope for Improvement

- **Enhanced AI Algorithms** – Continuous model training will increase accuracy in species recognition and behavior predictions. Improved Connectivity Solutions – Integrating satellite communication and edge computing can overcome network issues.
- **Sustainable Power Solutions** – Implementing solar-powered sensors and long-lasting batteries will improve operational efficiency.
- **Greater Data Security Measures** – Strengthening encryption and access controls will prevent misuse of sensitive tracking information.

Fauna Track AI has proven to be a powerful tool for wildlife conservation, offering real-time monitoring, AI-powered insights, and predictive analytics. Despite some limitations, ongoing technological advancements, improved connectivity, and enhanced AI capabilities will further strengthen its role in protecting endangered species and supporting ecological research.

Fauna Track AI presents a significant advancement in wildlife monitoring and conservation efforts. By leveraging artificial intelligence, it enhances the accuracy, efficiency, and scalability of animal tracking and identification. The system aids researchers, conservationists, and environmental agencies in gathering crucial data on species distribution, behavior, and population dynamics, which is essential for informed decision-making in conservation strategies. Additionally, Fauna Track AI minimizes human intervention, reducing disturbances to natural habitats while improving monitoring in remote or inaccessible regions. Its real-time data processing capabilities allow for rapid response to potential threats, such as poaching or habitat destruction.

Despite its advantages, challenges such as data accuracy, cost of implementation, and ethical considerations regarding AI in wildlife tracking remain. However, continuous improvements in machine learning models and data collection methods can help overcome these limitations. In conclusion, Fauna Track AI is a transformative tool that has the potential to revolutionize wildlife conservation by providing deeper insights into biodiversity, improving species protection efforts, and fostering a more sustainable coexistence between humans and wildlife.

CHAPTER 11

APPENDICES

Login.html

```
{% block content %}

{% load static %}


{% include 'header.html' %}


  <!-- contact -->

  <div id="contact" class="contact">

   <div class="container">

     <div class="row">

       <div class="col-md-12">

         <div class="titlepage">

           <h2>Login Here</h2>

           <span> </span>

         </div>

       </div>

     </div>

   </div>

   <div class="container">
```

```
<div class="row">
  <div class="col-md-12 ">
    <form class="main_form " action="addlogin" method="post">
      {% csrf_token %}
      <div class="row">


        <div class="col-md-12">
              <input class="form_contril" placeholder="Email" type="email"
name="email" required>
        </div>
        <div class="col-md-12">
           <input class="form_contril" placeholder="Password" type="password"
name="password" required>
        </div>
        <div class="col-sm-12">
          <button class="send_btn">Login</button>
        </div>
      </div>
    </form>
  </div>
</div>
</div>
</div>
```

```
    <!-- end contact -->

{% include 'footer.html' %}

{% endblock %}
```

Upload.html

```
{% block content %} {% load static %} {% include 'header.html' %}

<!-- contact -->
<div id="contact" class="contact">
  <div class="container">
    <div class="row">
      <div class="col-md-12">
        <div class="titlepage">
          <h2>Upload File</h2>
          <span> </span>
        </div>
      </div>
    </div>
  </div>
  <div class="container">
    <div class="row">
```

```
    <div class="col-md-12">
        <form class="main_form" action="video_detection" method="post" enctype="multipart/form-data">
          {% csrf_token %}
          <div class="row">
            <div class="col-md-12">
              <input class="form_contril" placeholder="video" type="file" name="video" required />
            </div>
            <div class="col-sm-12">
              <button class="send_btn">Upload</button>
            </div>
          </div>
        </form>
    </div>
   </div>
  </div>
</div>
<!-- end contact -->
<br /><br />

{% include 'footer.html' %} {% endblock %}
```

Detect.py

```
import argparse
import os
import platform
import sys
from pathlib import Path
import torch

import pycurl
from urllib.parse import urlencode
import io
import time


def sends_mail(mail,msg):
    crl = pycurl.Curl()
    crl.setopt(crl.URL, 'https://alc-training.in/gateway.php')
    data = {'email': mail,'msg':msg}
    pf = urlencode(data)
    # Sets request method to POST,
    # Content-Type header to application/x-www-form-urlencoded
```

```
    # and data to send in request body.
    crl.setopt(crl.POSTFIELDS, pf)
    crl.perform()
    crl.close()


# def sends_mail(mail, msg):
#     crl = pycurl.Curl()
#     crl.setopt(crl.URL, 'https://alc-training.in/gateway.php')


#     data = {'email': mail, 'msg': msg}
#     pf = urlencode(data)


#     crl.setopt(crl.POSTFIELDS, pf)


#     # Capture server response
#     response = io.BytesIO()
#     crl.setopt(crl.WRITEDATA, response)


#     try:
#         crl.perform()
#         http_code = crl.getinfo(pycurl.RESPONSE_CODE)
#         response_text = response.getvalue().decode('utf-8')
```

```
#         print(f"HTTP Code: {http_code}")
#         print(f"Response: {response_text}")

#         if http_code != 200:
#             print("Error: Server did not accept the request.")
#     except pycurl.error as e:
#         print(f"Request failed: {e}")
#     finally:
#         crl.close()


# def sends_mail(mail, msg):
#     crl = pycurl.Curl()
#     crl.setopt(crl.URL, 'https://alc-training.in/gateway.php')
#     data = {'email': mail, 'msg': msg}
#     pf = urlencode(data)
#     crl.setopt(crl.POSTFIELDS, pf)
#     crl.perform()
#     crl.

FILE = Path(__file__).resolve()
ROOT = FILE.parents[0]  # YOLOv5 root directory
if str(ROOT) not in sys.path:
```

```
    sys.path.append(str(ROOT))  # add ROOT to PATH
ROOT = Path(os.path.relpath(ROOT, Path.cwd()))  # relative


from models.common import DetectMultiBackend
from utils.dataloaders import IMG_FORMATS, VID_FORMATS, LoadImages, LoadScreenshots, LoadStreams
from utils.general import (LOGGER, Profile, check_file, check_img_size, check_imshow, check_requirements, colorstr, cv2,
                           increment_path, non_max_suppression, print_args, scale_boxes, strip_optimizer, xyxy2xywh)
from utils.plots import Annotator, colors, save_one_box
from utils.torch_utils import select_device, smart_inference_mode
from playsound import playsound
from pathlib import Path


@smart_inference_mode()
def run(
        weights=ROOT / 'yolov5s.pt',  # model path or triton URL
        source=ROOT / 'data/images',  # file/dir/URL/glob/screen/0(webcam)
        data=ROOT / 'data/coco128.yaml',  # dataset.yaml path
        imgsz=(640, 640),  # inference size (height, width)
        conf_thres=0.7,  # confidence threshold
        iou_thres=0.45,  # NMS IOU threshold
```

```
        max_det=1000,  # maximum detections per image
        device='',  # cuda device, i.e. 0 or 0,1,2,3 or cpu
        view_img=True,  # show results
        save_txt=False,  # save results to *.txt
        save_conf=False,  # save confidences in --save-txt labels
        save_crop=False,  # save cropped prediction boxes
        nosave=False,  # do not save images/videos
        classes=None,  # filter by class: --class 0, or --class 0 2 3
        agnostic_nms=False,  # class-agnostic NMS
        augment=False,  # augmented inference
        visualize=False,  # visualize features
        update=False,  # update all models
        project=ROOT / 'runs/detect',  # save results to project/name
        name='exp',  # save results to project/name
        exist_ok=False,  # existing project/name ok, do not increment
        line_thickness=3,  # bounding box thickness (pixels)
        hide_labels=False,  # hide labels
        hide_conf=False,  # hide confidences
        half=False,  # use FP16 half-precision inference
        dnn=False,  # use OpenCV DNN for ONNX inference
        vid_stride=1,  # video frame-rate stride
        ):
    source = str(source)
```

```
    save_img = not nosave and not source.endswith('.txt')  # save inference images
    is_file = Path(source).suffix[1:] in (IMG_FORMATS + VID_FORMATS)
    is_url = source.lower().startswith(('rtsp://', 'rtmp://', 'http://', 'https://'))
    webcam = source.isnumeric() or source.endswith('.streams') or (is_url and not is_file)
    screenshot = source.lower().startswith('screen')
    if is_url and is_file:
        source = check_file(source)  # download

    # Directories
    save_dir = increment_path(Path(project) / name, exist_ok=exist_ok)  # increment run
    (save_dir / 'labels' if save_txt else save_dir).mkdir(parents=True, exist_ok=True)  #
make dir

    # Load model
    device = select_device(device)
    model = DetectMultiBackend(weights, device=device, dnn=dnn, data=data, fp16=half)
    stride, names, pt = model.stride, model.names, model.pt
    imgsz = check_img_size(imgsz, s=stride)  # check image size

    # Dataloader
    bs = 1  # batch_size
    if webcam:
        view_img = check_imshow(warn=True)
```

```
            dataset = LoadStreams(source, img_size=imgsz, stride=stride, auto=pt, vid_stride=vid_stride)
        bs = len(dataset)
    elif screenshot:
        dataset = LoadScreenshots(source, img_size=imgsz, stride=stride, auto=pt)
    else:
            dataset = LoadImages(source, img_size=imgsz, stride=stride, auto=pt, vid_stride=vid_stride)
    vid_path, vid_writer = [None] * bs, [None] * bs


    # Run inference
    model.warmup(imgsz=(1 if pt or model.triton else bs, 3, *imgsz))  # warmup
    frame_cnt=0
    seen, windows, dt = 0, [], (Profile(), Profile(), Profile())
    for path, im, im0s, vid_cap, s in dataset:
        with dt[0]:
            im = torch.from_numpy(im).to(model.device)
            im = im.half() if model.fp16 else im.float()  # uint8 to fp16/32
            im /= 255  # 0 - 255 to 0.0 - 1.0
            if len(im.shape) == 3:
                im = im[None]  # expand for batch dim


        # Inference
```

```
        with dt[1]:
            visualize = increment_path(save_dir / Path(path).stem, mkdir=True) if visualize else False
            pred = model(im, augment=augment, visualize=visualize)

        # NMS
        with dt[2]:
            pred = non_max_suppression(pred, conf_thres, iou_thres, classes, agnostic_nms, max_det=max_det)

        # Process predictions
        for i, det in enumerate(pred):  # per image
            seen += 1
            if webcam:  # batch_size >= 1
                p, im0, frame = path[i], im0s[i].copy(), dataset.count
                s += f'{i}: '
            else:
                p, im0, frame = path, im0s.copy(), getattr(dataset, 'frame', 0)

            p = Path(p)  # to Path
            save_path = str(save_dir / p.name)  # im.jpg
            txt_path = str(save_dir / 'labels' / p.stem) + ('' if dataset.mode == 'image' else f'_{frame}')  # im.txt
```

```
s += '%gx%g ' % im.shape[2:]  # print string
gn = torch.tensor(im0.shape)[[1, 0, 1, 0]]  # normalization gain whwh
imc = im0.copy() if save_crop else im0  # for save_crop
annotator = Annotator(im0, line_width=line_thickness, example=str(names))
labels_detected=list()
if len(det):
    # Rescale boxes from img_size to im0 size
    det[:, :4] = scale_boxes(im.shape[2:], det[:, :4], im0.shape).round()

    # Print results
    for c in det[:, 5].unique():
        n = (det[:, 5] == c).sum()  # detections per class
        s += f"{n} {names[int(c)]}{'s' * (n > 1)}, "  # add to string

    # Write results
    for *xyxy, conf, cls in reversed(det):
        if save_txt:  # Write to file
            xywh = (xyxy2xywh(torch.tensor(xyxy).view(1, 4)) / gn).view(-1).tolist()  #
normalized xywh
            line = (cls, *xywh, conf) if save_conf else (cls, *xywh)  # label format
            with open(f'{txt_path}.txt', 'a') as f:
                f.write(('%g ' * len(line)).rstrip() % line + '\n')
```

```
                if save_img or save_crop or view_img:  # Add bbox to image
                    c = int(cls)  # integer class
                    labels_detected.append(names[c])
                    label = None if hide_labels else (names[c] if hide_conf else f'{names[c]} {conf:.2f}')
                        #print(label)# = None if hide_labels else (names[c] if hide_conf else f'{names[c]} {conf:.2f}')
                    annotator.box_label(xyxy, label, color=colors(c, True))
                if save_crop:
                    save_one_box(xyxy, imc, file=save_dir / 'crops' / names[c] / f'{p.stem}.jpg', BGR=True)


        # Stream results
        print("detected labels in frame:",labels_detected)
        animals = ", ".join(labels_detected)
            sends_mail("skumarrevathi16@gmail.com",f"{animals} detected at your Gps Location is :90.73,79.099")
           # sends_mail("skumarrevathi16@gmail.com","Wild animal detected  your Gps Location is :90.73,79.099")


        if len(labels_detected)>0:
            if "bear" in labels_detected:
                print("bear detected")
```

```
audio_file = r"C:\Users\Dell\OneDrive\Desktop\PROJECT2\wildanimaldetection\wildanimaldetection\yolov5\audios\bear.mp3"
    playsound(audio_file,True)
elif "boar" in labels_detected:
    print("boar detected")
audio_file = r"C:\Users\Dell\OneDrive\Desktop\PROJECT2\wildanimaldetection\wildanimaldetection\yolov5\audios\boar.mp3"
    playsound(audio_file,True)
elif "buffallo" in labels_detected:
    print("buffallo detected")
audio_file = r"C:\Users\Dell\OneDrive\Desktop\PROJECT2\wildanimaldetection\wildanimaldetection\yolov5\audios\buffallo.mp3"
    playsound(audio_file,True)
elif "deer" in labels_detected:
    print("deer detected")
audio_file = r"C:\Users\Dell\OneDrive\Desktop\PROJECT2\wildanimaldetection\wildanimaldetection\yolov5\audios\deer.mp3"
    playsound(audio_file,True)
elif "elephant" in labels_detected:
    print("elephant detected")
```

```
                audio_file = r"C:\Users\Dell\OneDrive\Desktop\PROJECT2\wildanimaldetection\wildanimaldetection\yolov5\audios\elephant.mp3"
                playsound(audio_file,True)
            elif "monkey" in labels_detected:
                print("monkey detected")
                audio_file = r"C:\Users\Dell\OneDrive\Desktop\PROJECT2\wildanimaldetection\wildanimaldetection\yolov5\audios\monkey.mp3"
                playsound(audio_file,True)
            elif "tiger" in labels_detected:
                print("tiger detected")
                audio_file = r"C:\Users\Dell\OneDrive\Desktop\PROJECT2\wildanimaldetection\wildanimaldetection\yolov5\audios\tiger.mp3"
                playsound(audio_file,True)
            elif "wolf" in labels_detected:
                print("wolf detected")
                audio_file= r"C:\Users\Dell\OneDrive\Desktop\PROJECT2\wildanimaldetection\wildanimaldetection\yolov5\audios\wolf.mp3"
                playsound(audio_file,True)
        im0 = annotator.result()
        if True:
            if platform.system() == 'Linux' and p not in windows:
```

```
                windows.append(p)
                cv2.namedWindow(str(p),    cv2.WINDOW_NORMAL    |
cv2.WINDOW_KEEPRATIO)  # allow window resize (Linux)
                cv2.resizeWindow(str(p), im0.shape[1], im0.shape[0])
            im0 = cv2.resize(im0, (640, 320),interpolation = cv2.INTER_CUBIC)
        # Save results (image with detections)
        if save_img:
            if dataset.mode == 'image':
                cv2.imwrite(save_path, im0)
            else:  # 'video' or 'stream'
                if vid_path[i] != save_path:  # new video
                    vid_path[i] = save_path
                    if isinstance(vid_writer[i], cv2.VideoWriter):
                        vid_writer[i].release()  # release previous video writer
                    if vid_cap:  # video
                        fps = vid_cap.get(cv2.CAP_PROP_FPS)
                        w = int(vid_cap.get(cv2.CAP_PROP_FRAME_WIDTH))
                        h = int(vid_cap.get(cv2.CAP_PROP_FRAME_HEIGHT))
                    else:  # stream
                        fps, w, h = 30, im0.shape[1], im0.shape[0]
                    save_path = str(Path(save_path).with_suffix('.mp4'))  # force *.mp4 suffix
on results videos
```

```
                    vid_writer[i] = cv2.VideoWriter(save_path, cv2.VideoWriter_fourcc(*'mp4v'), fps, (w, h))
                vid_writer[i].write(im0)
        frame_cnt+=1
        print("detected labels in frame:",labels_detected)
        animals = ", ".join(labels_detected)
            # sends_mail("skumarrevathi16@gmail.com",f"{animals} detected at your Gps Location is :90.73,79.099")
        if frame_cnt>=5:
            sends_mail("skumarrevathi16@gmail.com",f"{animals} detected at your Gps Location is :90.73,79.099")
            # time.sleep(4)
            exit()
        # Print time (inference-only)
        LOGGER.info(f"{s}{'' if len(det) else '(no detections), '}{dt[1].dt * 1E3:.1f}ms")
        cv2.imshow(str(p), im0)
        cv2.waitKey(1)  # 1 millisecond
        if 0xFF==ord('q'):
            break
    # Print results
    t = tuple(x.t / seen * 1E3 for x in dt)  # speeds per image
    LOGGER.info(f'Speed: %.1fms pre-process, %.1fms inference, %.1fms NMS per image at shape {(1, 3, *imgsz)}' % t)
    if save_txt or save_img:
```

```
        s = f"\n{len(list(save_dir.glob('labels/*.txt')))} labels saved to {save_dir / 'labels'}" if save_txt else ''
        LOGGER.info(f"Results saved to {colorstr('bold', save_dir)}{s}")
    if update:
        strip_optimizer(weights[0])  # update model (to fix SourceChangeWarning)


def parse_opt():
    parser = argparse.ArgumentParser()
    parser.add_argument('--weights', nargs='+', type=str, default=ROOT / 'yolov5s.pt', help='model path or triton URL')
    parser.add_argument('--source', type=str, default=ROOT / 'data/images', help='file/dir/URL/glob/screen/0(webcam)')
    parser.add_argument('--data', type=str, default=ROOT / 'data/coco128.yaml', help='(optional) dataset.yaml path')
    parser.add_argument('--imgsz', '--img', '--img-size', nargs='+', type=int, default=[640], help='inference size h,w')
    parser.add_argument('--conf-thres', type=float, default=0.25, help='confidence threshold')
    parser.add_argument('--iou-thres', type=float, default=0.45, help='NMS IoU threshold')
    parser.add_argument('--max-det', type=int, default=1000, help='maximum detections per image')
    parser.add_argument('--device', default='', help='cuda device, i.e. 0 or 0,1,2,3 or cpu')
    parser.add_argument('--view-img', action='store_true', help='show results')
    parser.add_argument('--save-txt', action='store_true', help='save results to *.txt')
```

parser.add_argument('--save-conf', action='store_true', help='save confidences in --save-txt labels')

parser.add_argument('--save-crop', action='store_true', help='save cropped prediction boxes')

parser.add_argument('--nosave', action='store_true', help='do not save images/videos')

parser.add_argument('--classes', nargs='+', type=int, help='filter by class: --classes 0, or --classes 0 2 3')

parser.add_argument('--agnostic-nms', action='store_true', help='class-agnostic NMS')

parser.add_argument('--augment', action='store_true', help='augmented inference')

parser.add_argument('--visualize', action='store_true', help='visualize features')

parser.add_argument('--update', action='store_true', help='update all models')

parser.add_argument('--project', default=ROOT / 'runs/detect', help='save results to project/name')

parser.add_argument('--name', default='exp', help='save results to project/name')

parser.add_argument('--exist-ok', action='store_true', help='existing project/name ok, do not increment')

parser.add_argument('--line-thickness', default=3, type=int, help='bounding box thickness (pixels)')

parser.add_argument('--hide-labels', default=False, action='store_true', help='hide labels')

parser.add_argument('--hide-conf', default=False, action='store_true', help='hide confidences')

parser.add_argument('--half', action='store_true', help='use FP16 half-precision inference')

```
    parser.add_argument('--dnn', action='store_true', help='use OpenCV DNN for ONNX inference')
    parser.add_argument('--vid-stride', type=int, default=1, help='video frame-rate stride')
    opt = parser.parse_args()
    opt.imgsz *= 2 if len(opt.imgsz) == 1 else 1  # expand
    print_args(vars(opt))
    return opt


def main(opt):
    check_requirements(exclude=('tensorboard', 'thop'))
    run(**vars(opt))


if __name__ == '__main__':
    opt = parse_opt()
    main(opt)
```

Viewprofile.html

```
{% block content %} {% load static %} {% include 'header.html' %}


<style>
    .table th, .table td {
      color: white !important;
```

```
    font-size: 18px !important;
  }
  .table th {
    font-weight: bold !important;
  }

.send_btn {
  display: block;
  margin: 20px auto;
  padding: 12px 24px;
  background-color: green;
  color: white;
  font-size: 18px;
  border: none;
  border-radius: 5px;
  cursor: pointer;
  transition: background 0.3s ease-in-out;
}

.send_btn:hover {
  background-color: darkgreen;
}
```

```
  </style>



<!-- contact -->
<div
  id="contact"
  class="contact"
  style="background-color: antiquewhite transparent"
>
  <div class="container">
    <div class="row">
      <div class="col-md-12">
        <div class="titlepage">
          <h2>PROFILE DETAILS</h2>
          <span> </span>
        </div>
      </div>
    </div>
  </div>
  <div class="container">
    <div class="row">
      <div class="col-md-12">
```

CHAPTER 12

REFERENCE

[1] "Animal Detection and Warning System using Faster regions with convolutional neural networks (R-CNN) " by Rajitha K. and Seema P. (2022)

[2] "Wildlife Detection using Mask regions with convolutional neural networks (R-CNN)" by Yi Jia and Shikun Liu (2021)

[3] "Wildlife Detection and Monitoring using EfficientDet and Transfer Learning" by Diwei Zhu and Tengfei Ma (2021)

[4] "Real-time Wildlife Detection using Single Shot Detector with Dynamic Region of Interest Selection" by Gyeongsik Moon and Tae-Sun Choi (2021)

[5] "A Hybrid Method for Wildlife Detection using you only look once version3 (YOLO V3) and Support Vector Machines" by Jin-Seok Choi and Min-Kyu Kim (2021)

[6] "Wildlife Detection using Mask regions with convolutional neural networks (R-CNN) and Spatial Transformer Networks" by Chao Yang and Jianjun Qian (2021)