Internship Project Report

Animal Detection System

Internship Project Task 2

Submitted by:

Tarrush Saxena

December 2025

Internship Project Report

Chapter 1 : Introduction

In recent years, the intersection of wildlife conservation and technology has become increasingly critical. This project, the 'Animal Detection System', leverages state-of-the-art computer vision techiques to identify and classify various animal species in real-time. By utilizing deep learning models, specifically the YOLO (You Only Look Once) architecture, the system provides accurate and efficient detection capabilities suitable for applications in wildlife monitoring, road safety (preventing animal-vehicle collisions), and agricultural protection.

Chapter 2 : Project Objectives

The primary objectives of this internship project were:

- To design and implement a robust machine learning model capable of detecting multiple animal classes.
- To develop a user-friendly Graphical User Interface (GUI) for easy interaction with the system.
- To ensure real-time performance on standard computing hardware.
- To analyze the model's performance using standard metrics such as Precision, Recall, and Confusion Matrices.

Chapter 3 : Methodology

3.1 Technologies Used

- Programming Language: Python
- Deep Learning Framework: YOLOv8 / YOLO11 (Ultralytics)
- Data Manipulation: Pandas, NumPy
- Visualization: Matplotlib, Seaborn
- GUI Framework: Tkinter

3.2 System Architecture

The system follows a modular architecture. The core detection engine is powered by a pre-trained YOLO model, fine-tuned on a specific animal dataset. Input is gathered via a live camera feed or video file. Frames are processed individually, where the model predicts bounding boxes and class probabilities. The results are then rendered onto the interface, highlighting detected animals with labels and confidence scores.

Chapter 4 : Implementation Details

The project structure is organized into separate modules for data handling, model training, and the graphical interface. Key components include:

- main.py: The entry point of the application, initializing the GUI.
- train.py: A script dedicated to training the YOLO model on the custom dataset, handling data augmentation and validation.

- animal_gui.py: Manages the frontend logic, allowing users to start/stop detection and view logs.

Training was conducted over multiple epochs to minimize loss and maximize mean Average Precision (mAP). The dataset was split into training, validation, and testing sets to ensure robust evaluation.

Chapter 5 : Results

The deployed model demonstrated high efficacy in detecting target species. Evaluation metrics indicated a high degree of precision, meaning few false positives were generated. The system achieved a real-time frame rate (FPS) sufficient for live monitoring tasks.

Visual outputs, such as confusion matrices generated during the evaluation phase, confirmed that the model correctly distinguishes between similar animal classes.

Chapter 6 : Conclusion

This internship successfully delivered a functional Animal Detection System. The project not only met the technical requirements set forth in Task 2 but also provided valuable insights into the practical challenges of deploying AI models.Future enhancements could include integrating edge computing devices for remote monitoring and expanding the dataset to include more species.