**Wildlife Conservation with Image Recognition**

## **Abstract**

Wildlife conservation is critical for maintaining ecological balance and biodiversity. This project proposes a **Wildlife Conservation System** that utilizes **image recognition** powered by **machine learning** and **computer vision** to identify and monitor wildlife species from uploaded images. The system is built using **Python Django** as the backend framework and integrates image recognition models such as **Convolutional Neural Networks (CNNs)** trained using datasets like **ImageNet** or wildlife-specific datasets. Users can upload images of animals, and the system identifies the species and provides relevant conservation information. This application aids researchers, wildlife organizations, and enthusiasts in monitoring wildlife and implementing conservation strategies efficiently.

## **Synopsis**

### **Introduction**

Wildlife conservation involves the protection and monitoring of animal species to prevent extinction. With advancements in computer vision and deep learning, **image recognition** technologies have become valuable tools for identifying animals, tracking their populations, and studying their behavior. This project builds a **Django-based web application** that leverages image recognition to automate wildlife species identification and provide conservation insights.

### **Objectives**

* To identify wildlife species from uploaded images using deep learning models.
* To assist conservationists, researchers, and wildlife enthusiasts in monitoring animal populations.
* To provide species-specific information, including conservation status and habitat details.
* To build a scalable and user-friendly web platform for wildlife monitoring.

### **Scope**

The system will:

* Allow users to upload images of animals for identification.
* Use pre-trained deep learning models to classify species and display results.
* Provide additional conservation data (habitat, population trends, and status).
* Support wildlife research, educational purposes, and conservation planning.

## **Existing System**

Currently, wildlife conservation relies on:

1. **Manual Identification**:
   1. Researchers manually identify animals from images or field data, which is time-consuming and prone to errors.
2. **Non-Automated Image Recognition**:
   1. Some systems use basic image processing tools without advanced deep learning models, leading to limited accuracy.
3. **Conventional Surveys**:
   1. Wildlife monitoring relies on physical surveys, camera traps, and tagging, which are resource-intensive.

### **Limitations of the Existing System:**

* **Manual Processes**: High dependency on human expertise leads to inefficiency.
* **Limited Accuracy**: Non-deep learning models lack precision in species identification.
* **Scalability Issues**: Traditional methods cannot handle large datasets or wide geographical areas.

## **Proposed System**

The **Wildlife Conservation with Image Recognition** system automates species identification using deep learning models integrated into a Django-based web application. It enhances wildlife monitoring and provides actionable insights to researchers and conservationists.

### **Features**

1. **Image Upload and Processing**:
   1. Users can upload wildlife images in formats like **JPG, PNG, or JPEG**.
   2. Uploaded images are preprocessed for model input (resizing, noise reduction, normalization).
2. **Species Identification**:
   1. Integrate a **Convolutional Neural Network (CNN)** model (e.g., ResNet, InceptionV3, or custom-trained models) to classify animals.
   2. Return results with the **species name**, confidence score, and additional details.
3. **Conservation Data Display**:
   1. Provide species-specific information, such as:
      1. Scientific name
      2. Habitat
      3. Conservation status (e.g., Critically Endangered, Endangered, Vulnerable)
      4. Population trends
4. **Interactive Dashboard**:
   1. Admins can view:
      1. Total uploads by species
      2. Classification accuracy metrics
      3. Visual reports on wildlife population trends.
5. **User Authentication and Management**:
   1. Role-based access for researchers, admins, and general users.

### **Technologies Used**

* **Backend**: Python Django
* **Frontend**: HTML, CSS, Bootstrap, JavaScript
* **Image Processing**: OpenCV, Pillow
* **Machine Learning/Deep Learning**: TensorFlow/Keras, PyTorch
* **Pre-trained Models**: ResNet, InceptionV3, VGG16, or a custom-trained CNN model
* **Database**: SQLite/PostgreSQL
* **Visualization**: Matplotlib, Plotly
* **APIs**: Integration with third-party APIs for additional wildlife data (e.g., IUCN Red List API).

## **Workflow**

1. **User Uploads an Image** →
2. **Image Preprocessing** →
3. **Image Classification (Model Prediction)** →
4. **Species Identification and Data Retrieval** →
5. **Display Results (Species Name, Conservation Data)** →
6. **Store Data for Monitoring and Analytics**

## **Proposed Model**

The classification model will use **CNN-based architectures**:

* **Pre-trained Models**: ResNet50, VGG16, or InceptionV3 (fine-tuned on wildlife datasets).
* **Training Dataset**: Public datasets like **iNaturalist**, **ImageNet**, or custom datasets with labeled wildlife images.

### **Modules:**

### **1. Image Upload and Preprocessing**

* **Description**: This module allows users to upload images of animals in formats like JPG, PNG, or JPEG. It also preprocesses the images to prepare them for classification.
* **Features**:
  + **Upload Interface**: Provides a user-friendly interface for uploading wildlife images.
  + **Image Preprocessing**:
    - **Resizing**: Scales images to the input size required by the model.
    - **Noise Reduction**: Removes unwanted artifacts for cleaner analysis.
    - **Normalization**: Scales pixel values for better model performance.

### **2. Species Identification**

* **Description**: This is the core module, where the uploaded images are processed through a deep learning model to classify the animal species.
* **Features**:
  + **CNN Integration**: Uses pre-trained models like ResNet, VGG16, or InceptionV3 fine-tuned for wildlife datasets.
  + **Output**:
    - Predicted species name.
    - Confidence score of the classification.
    - Additional species-specific information.
* **Dataset**: Trained on wildlife datasets like iNaturalist or ImageNet for high accuracy.

### **3. Conservation Data Display**

* **Description**: Provides detailed information about the identified species, aiding users in understanding its conservation needs.
* **Features**:
  + **Species Information**:
    - Scientific name.
    - Natural habitat details.
    - Conservation status (e.g., Endangered, Vulnerable).
    - Population trends.
  + **API Integration**: Pulls real-time data from external sources like the IUCN Red List API for accurate and up-to-date information.

### **4. Interactive Dashboard**

* **Description**: Designed for admin users to monitor system usage, analyze classification metrics, and view trends in wildlife data.
* **Features**:
  + **Reports**:
    - Total uploads categorized by species.
    - Metrics like classification accuracy and confidence scores.
    - Population trends visualized through graphs.
  + **Visualization**: Uses tools like Matplotlib and Plotly for dynamic charts and graphs.

### **5. User Authentication and Management**

* **Description**: Ensures secure access to the system and allows role-based functionality.
* **Features**:
  + **User Roles**:
    - **Researchers**: Upload images and access detailed conservation data.
    - **Admins**: View system reports, manage users, and monitor uploads.
    - **General Users**: Limited access for species identification and viewing basic data.
  + **Authentication**:
    - Registration and login system.
    - Password-protected accounts.

## **Conclusion**

The **Wildlife Conservation with Image Recognition System** provides an efficient, scalable solution for identifying and monitoring wildlife species. By leveraging deep learning and Django, the system automates species recognition, reducing reliance on manual processes. It delivers accurate predictions and provides critical conservation insights, assisting researchers and organizations in protecting endangered species and preserving biodiversity. This project bridges technology and conservation, offering a valuable tool for wildlife management.