Wildlife Monitoring System using Camera Traps

# 📌 Project Overview

The Wildlife Monitoring System using Camera Traps is a computer vision project that leverages the YOLOv5 object detection model to identify and monitor wildlife species from camera trap images. This system allows conservationists and researchers to automate the identification process of animals captured on remote camera traps, saving both time and resources.

# 🛠 Technologies Used

- Python  
- Google Colab  
- YOLOv5 (You Only Look Once version 5)  
- PyTorch  
- Matplotlib  
- PIL (Python Imaging Library)

# 📂 Project Structure

- Cloning YOLOv5 Repository: Downloads the YOLOv5 model and dependencies.  
- Installing Requirements: Installs necessary Python libraries.  
- Loading Pre-trained Model: Loads a small and fast YOLOv5s model.  
- Image Uploading: Uses Google Colab’s upload interface to allow users to input camera trap images.  
- Object Detection: Performs detection on the uploaded image.  
- Results Visualization & Saving: Displays and saves the detection results.

# 🚀 How It Works

## Step 1: Clone the YOLOv5 Repository

!git clone https://github.com/ultralytics/yolov5

## Step 2: Install Required Packages

!pip install -r requirements.txt

## Step 3: Import Required Libraries

import torch  
from matplotlib import pyplot as plt  
import os  
from PIL import Image

## Step 4: Load Pre-trained YOLOv5 Model

model = torch.hub.load('ultralytics/yolov5', 'yolov5s')

## Step 5: Upload Camera Trap Images

from google.colab import files  
uploaded = files.upload()

## Step 6: Run Detection

results = model(img\_path)  
results.print()  
results.show()  
results.save()

# 📷 Output

The detected image will be shown with bounding boxes indicating identified wildlife species. Output images are saved under yolov5/runs/detect/exp.



# ✅ Advantages

- Easy integration with Google Colab.  
- Fast and accurate detection using YOLOv5.  
- Simplified workflow for non-technical users via Colab’s UI.

# 🔄 Future Improvements

- Support for batch image processing.  
- Integration with real-time video feeds.  
- Exporting detection data to CSV for further analysis.  
- Custom training on wildlife-specific datasets.