**Wildlife Detection and Deterrence Using CRNN-Based Analysis for Agricultural Protection**

This project presents a real-time wildlife detection and deterrence system developed to mitigate human-wildlife conflict in agricultural areas. The system integrates CRNN-based image classification, audio verification using MFCC features, distance measurement using ultrasonic sensors, and deterrence mechanisms to alert and protect farmlands.

**Objective**

To design and implement a practical, cost-effective, and eco-friendly AI-driven solution that detects wild animals approaching agricultural fields and triggers automated alerts to safeguard crops.

**Features**

* Animal detection using a Convolutional Recurrent Neural Network (CRNN) model on captured images.
* Audio confirmation using MFCC-based similarity matching between live and reference recordings.
* Proximity sensing using HC-SR04 ultrasonic sensor to confirm closeness of detected animals.
* Buzzer-based deterrence triggered when the presence of wildlife is confirmed.
* Remote monitoring capability using RealVNC for live feed access and system control.

**Hardware and Software Components**

**Hardware:**

* Raspberry Pi (central controller)
* Camera Module (image capture)
* Ultrasonic Sensor (HC-SR04)
* Buzzer (alert mechanism)
* Microphone (audio input)

**Software and Libraries:**

* Python 3.6+
* PyTorch (deep learning framework)
* OpenCV (image processing)
* Librosa, Sounddevice, SciPy (audio processing)
* NumPy, PIL (data handling and image transformation)

**Model Description**

The CRNN model combines convolutional layers for spatial feature extraction with recurrent layers (LSTM) for sequential feature learning. It is trained to classify images into one of eight wildlife categories: bear, elephant, fox, hyena, leopard, lion, pig, and tiger.

**System Workflow**

1. The Raspberry Pi captures an image using the camera module.
2. The CRNN model processes the image to identify potential animal presence.
3. If detection confidence is high, the system records audio for a fixed duration.
4. MFCC features are extracted from the audio and compared with reference features.
5. If similarity exceeds a defined threshold, the ultrasonic sensor measures the animal's proximity.
6. If the animal is within a critical distance (e.g., less than 10 cm), the buzzer is activated to deter the animal.
7. All activity can be remotely monitored using RealVNC.

**File Structure**

* wildlife\_detector.py – Main Python script
* animalscrnn\_model.pth – Trained CRNN model weights
* \*.wav – Reference audio files for animal sound comparison
* README.md – Project documentation

**Future Enhancements**

* Integration of GSM/SMS alert system
* Extension to night vision support
* Expansion to additional animal species
* Solar-powered deployment for remote areas