Amey Chhaya : B22ES011
Devansh Panchal : B22CS021

HOTSPOT IDENTIFICATION

THE URGE TO TAKE A PLUNGE INTO THE REALMS OF THAR..

The Thar Desert, despite its arid conditions, harbors a unique biodiversity hotspot characterized by the presence of specialized flora and fauna adapted to extreme water scarcity and high temperatures. This hotspot showcases a variety of xerophytic plant species, such as succulents and drought-resistant grasses, which have evolved to thrive in the desert environment. Additionally, the region supports a diverse range of animal life, including reptiles like the Indian spiny-tailed lizard and various avian species adapted to desert conditions. The conservation of this biodiversity hotspot in the Thar Desert is crucial, as it represents a specialized ecological niche and contributes to the overall understanding of adaptations to arid environments.

THE GIST OF OUR PROJECT

- The goal is to create a Drone-Based Hotspot Identification system, which can be implemented in the arid regions of the Thar.
- The drone we use is a Q4i UAV Drone.
- In order to identify a Hotspot, we need to identify certain flora and fauna across the region. Hence, we have tried to find a way by which we can identify our galloping friend, The Indian Gazelle, or better known as the "Chinkara".
- The Real-time recorded video of the drone's camera shall be employed as our working dataset.
- We will then use machine Learning Algorithms to iterate through the frames and try to pinpoint the locations of the chinkara.

The Q4i UAV Drone

The Q4i UAV drone is a cutting-edge unmanned aerial vehicle characterized by its compact design and advanced technical features. Equipped with a high-resolution camera boasting a 4K Ultra HD capability, the drone ensures crisp and detailed aerial imagery. Its impressive flight performance is facilitated by a precise GPS system, allowing for accurate navigation and stable hovering. The Q4i incorporates intelligent obstacle avoidance sensors, enhancing its safety and maneuverability during flight. With a maximum flight time of 30 minutes and a range of 5 kilometers, this drone is ideal for various applications, including surveillance, mapping, and aerial photography. Additionally, the Q4i supports real-time video streaming and boasts a user-friendly interface, making it a versatile and user-centric UAV solution in the rapidly evolving drone technology landscape.

The Main Problem

- We cannot employ any kind of additional sensors or the cameras on the drone as the compactness and the balance of the drone would be compromised.
- The Chinkara can be spotted by only a keen observer through a video due to the color of it’s body, the chinkara is always highly camouflaged by it’s surroundings.
- This would result in a substantial decrement in the accuracy of the correct identifications, sometime going as low as the natural probabilistic measures.
- The limitations in resolution as well as detail also hinder our results.

GRABBING THE PROBLEM BY IT’S TAIL!

Identifying a chinkara through overhead drone shot videos can be challenging due to the limited resolution and details at a distance. However, you can still focus on certain features and behaviors to increase the likelihood of accurate identification:

1. Size and Proportion:

- Chinkaras are relatively small, so look for small-sized animals with a slender build and long legs. Compare their size relative to other objects or animals in the video.

2. Coloration:

- Chinkaras have a sandy or light brown coat that helps them blend into their arid environment. Look for animals with a similar coloration, especially when contrasted against the surroundings.

3. Movement Patterns:

- Chinkaras are known for their swift and graceful movements. Observe the behavior of the animals in the video. Chinkaras often exhibit bounding or leaping motions.

4. Group Dynamics:

- Chinkaras are social animals and are often found in small groups. If you see multiple individuals moving together, it could be an indication of chinkaras.

5. Habitat:

- Consider the location and terrain shown in the drone videos. Chinkaras are adapted to arid and semi-arid environments, so if the video is captured in such a habitat, it increases the likelihood of spotting chinkaras.

6. Zoom-In on Details:

- If the drone footage allows for zooming in, focus on details such as the shape of the horns. Both male and female chinkaras have short, straight horns with a slight backward curve.

7. Tail Characteristics:

- While the tail may not be the most distinguishing feature, observe the tail's general characteristics. Chinkaras typically have slender and inconspicuous tails.

The proposed solution

YOLO, which stands for "You Only Look Once," is a real-time object detection algorithm that can detect multiple objects in an image or video frame in a single forward pass. YOLO is known for its speed and efficiency and has been widely used in various computer vision applications.

1. Importing Libraries and Load YOLO Model

```python
import cv2
import numpy as np

# Load pre-trained YOLO model and classes
net = cv2.dnn.readNet("yolov3.weights", "yolov3.cfg")
classes = []
with open("coco.names", "r") as f:
    classes = [line.strip() for line in f.readlines()]

layer_names = net.getUnconnectedOutLayersNames()
```

2. Initializing Video Capture

```python
# Load the drone video
video_path = "path/to/your/drone_video.mp4"
cap = cv2.VideoCapture(video_path)
```

3. Processing Video Frames

```
while True:
    ret, frame = cap.read()
    if not ret:
        break

    height, width, _ = frame.shape
```

4. Converting Frame to Blob and Perform Detection

```
# Convert the frame to a blob
blob = cv2.dnn.blobFromImage(frame, 0.00392, (416, 416), (0, 0, 0), True, crop=False)
net.setInput(blob)
outs = net.forward(layer_names)
```

5. Filtering out Chinkara Detection

```
class_ids = []
confidences = []
boxes = []

for out in outs:
    for detection in out:
        scores = detection[5:]
        class_id = np.argmax(scores)
        confidence = scores[class_id]
        if confidence > 0.5 and classes[class_id] == "chinkara":
            center_x, center_y, w, h = (detection[0:4] * np.array([width, height, width, height])).astype('int')
            x, y = int(center_x - w / 2), int(center_y - h / 2)
            class_ids.append(class_id)
            confidences.append(float(confidence))
            boxes.append([x, y, w, h])
```

6. Drawing Bounding Boxes

```
indices = cv2.dnn.NMSBoxes(boxes, confidences, 0.5, 0.4)

for i in indices:
    i = i[0]
    box = boxes[i]
    x, y, w, h = box
    cv2.rectangle(frame, (x, y), (x + w, y + h), (0, 255, 0), 2)
```

7. Displaying Frame and Cleanup

```
    cv2.imshow("Chinkara Detection", frame)
    if cv2.waitKey(1) & 0xFF == ord('q'):
        break

cap.release()
cv2.destroyAllWindows()
```

Why would the code work?

- The model is trained on a robust dataset.
- Separate classes can be simply added for different species of plants and animals, and it will still work with the same efficiency.
- If specifications are required, then we can simply train the ML model on the required dataset.

What Tinkering might be fruitful?

- We can try and attach a small chip to the drone(Or a GSM Modem) with the drone, which would tell us the real time Geographic coordinates of the drone.
- The geographic coordinates of a drone for a particular frame can be studied with respect to the density of flora and fauna which is to be detected using advanced ML algorithms.
- That study can govern the direction of the drone when we get it’s controls to be automised.

Constructive Criticism, please!


Thank you