

Programming for Artificial Intelligence



SUPERIOR UNIVERSITY

Name:

Mahak Farhan

Roll no:

068

Class:

BSAI

Section:

4B

Course:

Programming for Artificial Intelligence

Submitted to:

Sir Rasikh Ali

Lab 6

Animal Herd Detection

Objectives

- To develop an automated system that detects only animal herds.
- To create a simple and interactive web application using Flask.
- To utilize YOLOv8 for efficient and accurate animal detection.
- Users upload an image through the web interface.
- The uploaded image is passed to the YOLOv8 model for object detection.
- The detected image is displayed on the result page.

Tools & Technologies

- **Python:** Main programming language for implementation.
- **Flask:** Web framework for creating the application.
- **YOLOv8:** Deep learning model used for object detection.
- **OpenCV:** Image processing library.
- **HTML, CSS:** For designing the web interface.

Code

App6.py

```
from flask import Flask, render_template, request, send_from_directory
from ultralytics import YOLO
import cv2
import os

app = Flask(__name__)

# Define folders for uploads and results
UPLOAD_FOLDER = "main/uploads"
RESULT_FOLDER = "main/results"
os.makedirs(UPLOAD_FOLDER, exist_ok=True)
os.makedirs(RESULT_FOLDER, exist_ok=True)
```

Programming for Artificial Intelligence

```
# Load the YOLOv8 model
model = YOLO("yolov8n.pt") # Using YOLOv8 Nano (fast & lightweight)

# List of COCO animal class IDs
ANIMAL_CLASSES = [16, 17, 18, 19, 20, 21, 22, 23, 24, 25]

@app.route("/")
def index():
    return render_template("index.html")

@app.route("/detect", methods=["POST"])
def detect():
    if "file" not in request.files:
        return "No file uploaded", 400

    file = request.files["file"]
    if file.filename == "":
        return "No file selected", 400

    # Save the uploaded image
    file_path = os.path.join(UPLOAD_FOLDER, file.filename)
    result_path = os.path.join(RESULT_FOLDER, "result_" + file.filename)
    file.save(file_path)

    # Perform detection
    results = model(file_path)
    boxes = results[0].boxes
    filtered_boxes = []

    # Filter only animal detections
    for box in boxes:
        cls_id = int(box.cls) # Class ID of detected object
        if cls_id in ANIMAL_CLASSES:
            filtered_boxes.append(box)

    # If no animals were detected, return a message
    if not filtered_boxes:
        return "No animal herd detected.", 200

    # Save filtered detection results
    results[0].save(filename=result_path)

    return render_template("result.html", filename="result_" + file.filename)
```

Programming for Artificial Intelligence

```
@app.route("/static/results/<filename>")
def result_file(filename):
    return send_from_directory(RESULT_FOLDER, filename)

if __name__ == "__main__":
    app.run(debug=True)
```

Templates\index.html

```
<!DOCTYPE html>
<html lang="en">
<head>
  <title>Animal Herd Detection</title>
  <style>
    body {
      text-align: center;
      font-family: Arial, sans-serif;
      background-color: #f4f4f4;
      margin: 0;
      padding: 20px;
    }

    .container {
      background: white;
      padding: 30px;
      border-radius: 10px;
      box-shadow: 0 4px 8px rgba(0, 0, 0, 0.2);
      max-width: 400px;
      margin: auto;
    }

    h2 {
      color: #333;
      font-size: 24px;
      margin-bottom: 20px;
    }

    form {
      display: flex;
      flex-direction: column;
      align-items: center;
    }
```

Programming for Artificial Intelligence

```
input[type="file"] {
  padding: 10px;
  border: 2px solid #3498db;
  border-radius: 5px;
  background-color: #fff;
  cursor: pointer;
  width: 100%;
}

.btn {
  display: inline-block;
  margin-top: 15px;
  padding: 12px 20px;
  font-size: 16px;
  font-weight: bold;
  color: white;
  background-color: #3498db;
  border: none;
  border-radius: 5px;
  cursor: pointer;
  transition: 0.3s;
  width: 100%;
}

.btn:hover {
  background-color: #217dbb;
}
</style>
</head>
<body>

<div class="container">
  <h2>Upload an Image</h2>
  <form action="/detect" method="post" enctype="multipart/form-data">
    <input type="file" name="file" required>
    <button type="submit" class="btn">Proceed</button>
  </form>
</div>

</body>
</html>
```

Templates\result.html

Programming for Artificial Intelligence

```
<!DOCTYPE html>
<html lang="en">
<head>
  <title>Detection Result</title>
  <style>
    body {
      text-align: center;
      font-family: Arial, sans-serif;
      background-color: #f4f4f4;
      margin: 0;
      padding: 20px;
    }

    .container {
      background: white;
      padding: 20px;
      border-radius: 10px;
      box-shadow: 0 4px 8px rgba(0, 0, 0, 0.2);
      max-width: 600px;
      margin: auto;
    }

    h2 {
      color: #333;
      font-size: 24px;
      margin-bottom: 20px;
    }

    img {
      width: 100%;
      max-width: 500px;
      border-radius: 8px;
      box-shadow: 0 2px 8px rgba(0, 0, 0, 0.3);
    }

    .btn {
      display: inline-block;
      margin-top: 20px;
      padding: 12px 20px;
      font-size: 16px;
      font-weight: bold;
      color: white;
      background-color: #3498db;
      border: none;
    }
```

Programming for Artificial Intelligence

```
border-radius: 5px;
text-decoration: none;
transition: 0.3s;
}

.btn:hover {
background-color: #217dbb;
}
</style>
</head>
<body>

<div class="container">
  <h2>Detection Result</h2>
  
  <br>
  <a href="/" class="btn">Upload Another Image</a>
</div>

</body>
</html>
```

Output:

Programming for Artificial Intelligence

