***Animal Herd Detection Application***

***Overview***

*This application allows users to upload images, which are then processed using a YOLO-based object detection model to identify groups of animals (herds). The application is built with Flask and utilizes OpenCV for image processing.*

***Project Structure***

* *app.py: Main Flask application handling image uploads, processing, and displaying results.*
* *templates/: Contains HTML files for different pages (index, upload, and result pages).*
* *static/: Contains styles (CSS) and JavaScript (JS) for the frontend.*
* *uploads/: Stores uploaded images temporarily.*
* *results/: Stores processed images with detected animals.*
* *project/: Contains YOLO model configuration, weights, and class names.*

***Dependencies***

*Located in requirements.txt:*

* *Flask: Web framework for handling requests.*
* *opencv-python: OpenCV library for image processing.*
* *numpy: Used for numerical operations in image processing.*

***Install dependencies using:***

*pip install -r requirements.txt*

***app.py (Flask Backend)***

***Key Features:***

1. ***Flask Initialization:***
   * *Creates an app instance and configures necessary directories (uploads/ and results/).*
   * *Defines a secret key for session management.*
2. ***YOLO Model Loading:***
   * *Loads YOLO model using configuration (yolov4.cfg), weights (yolov4.weights), and class labels (coco.names).*
3. ***Animal Detection Function (detect\_animals):***
   * *Loads the uploaded image and processes it using YOLO.*
   * *Extracts bounding boxes for detected animals.*
   * *Determines whether individual animals or herds (3 or more) are present.*
   * *Saves the processed image and returns detection results.*
4. ***Routes:***
   * */: Home page.*
   * */upload: Upload page where users submit images.*
   * */results/<filename>: Displays the detection results.*
   * */uploads/<filename> & /results\_img/<filename>: Serves stored images.*

***Run the Flask app using:***

*python app.py*

***Frontend (HTML, CSS, JS)***

***HTML Files (Templates)***

* *index.html: Home page with an upload link.*
* *upload.html: Upload form with image preview.*
* *result.html: Displays detection results with an annotated image.*

***CSS (style.css)***

* *Provides a visually appealing UI with a gradient background.*
* *Styles buttons, text, and uploaded images.*

***JavaScript (script.js)***

* *Handles file preview before uploading.*

***Workflow***

1. *User uploads an image via upload.html.*
2. *Backend processes the image using YOLO.*
3. *Detection results are stored and displayed on result.html.*
4. *User sees the identified animals and herd classification.*

***Enhancements***

* *Replace YOLOv4 with YOLOv8 for improved accuracy.*
* *Enable real-time video processing.*
* *Store results in a database for historical tracking.*

*This application effectively detects and classifies animal herds, making it useful for wildlife monitoring and livestock management.*