# 🐾 Animal-Human Detection using YOLOv8l

This project demonstrates a custom object detection model trained using [Ultralytics YOLOv8 Large (YOLOv8l)](https://docs.ultralytics.com/models/yolov8/) to detect two classes: \*\*animals\*\* and \*\*humans\*\*. The model was trained on a manually annotated dataset using \*\*Labelbox\*\*, converted to YOLOv8 format, and trained on a Colab environment.

---

## 🔍 Project Overview

- \*\*Model\*\*: YOLOv8 Large (`yolov8l.pt`)

- \*\*Classes\*\*: `['animals', 'humans']`

- \*\*Framework\*\*: PyTorch with Ultralytics YOLOv8

- \*\*Dataset Size\*\*: 72 total images

- \*\*Annotations\*\*: Created in Labelbox → exported as JSON → converted to YOLOv8 format

- \*\*Environment\*\*: Google Colab with Tesla T4 GPU

- \*\*Training Duration\*\*: 20 epochs

---

## 📁 Dataset Preparation

1. Annotations were created using [Labelbox](https://labelbox.com).

2. Exported JSON files were converted to YOLOv8 format.

3. Images and labels were split into `train`, `val`, and `test` using a Python script (`train-test-val-split.py`).

Directory structure after processing:

/dataset\_2/  
├── train/  
│ ├── images/  
│ └── labels/  
├── val/  
│ ├── images/  
│ └── labels/  
└── dataset.yaml

---

## 🧠 Training Configuration

- \*\*Model Used\*\*: YOLOv8 Large (`yolov8l.pt`)

- \*\*Image Size\*\*: 928×928

- \*\*Batch Size\*\*: 4

- \*\*Epochs\*\*: 20

- \*\*Optimizer\*\*: SGD (lr=0.01, momentum=0.937)

- \*\*AMP\*\*: Enabled

- \*\*Device\*\*: CUDA (Tesla T4)

### YOLOv8 Training Command

```bash

yolo detect train \

model=yolov8l.pt \

data=/content/dataset\_2/dataset.yaml \

imgsz=900 \

epochs=20 \

batch=4 \

name=my-yolov8-project-colab-16 \

device=0

## 📊 Evaluation Results

| **Class** | **Precision** | **Recall** | **mAP@0.5** | **mAP@0.5:0.95** |
| --- | --- | --- | --- | --- |
| animals | 0.994 | 0.957 | 0.956 | 0.720 |
| humans | 0.960 | 0.974 | 0.991 | 0.687 |
| **Overall** | **0.977** | **0.965** | **0.974** | **0.703** |

* **Inference Time**: ~67.2ms per image
* **Postprocessing Time**: ~3.4ms per image

## 🚀 How to Run

1. Clone the repository:

git clone https://github.com/yourusername/animal-human-detection-yolov8.git

cd animal-human-detection-yolov8

1. Install dependencies:

pip install ultralytics

1. Download or prepare your dataset in YOLOv8 format.
2. Train the model (customize paths as needed):

yolo detect train \

model=yolov8l.pt \

data=path/to/dataset.yaml \

imgsz=928 \

epochs=20 \

batch=4

## 📥 Downloads

* 📦 [Trained Weights](https://github.com/yourusername/animal-human-detection-yolov8/releases) (Upload .pt model here after training)
* 📁 [Dataset (YOLO Format)](https://github.com/yourusername/animal-human-detection-yolov8/releases) (Optional)
* 📓 [Colab Notebook](https://chatgpt.com/c/link-to-colab-if-public)

## 📌 Notes

* Model automatically adjusted image size from 900 to 928 to fit stride 32 requirement.
* Albumentations used: Blur, MedianBlur, ToGray, CLAHE.

## 📷 Demo

Coming soon: sample detection images and inference script.

## 🧑‍💻 Contributors

* **Your Name** - Developer & Trainer
* YOLOv8 by [Ultralytics](https://github.com/ultralytics/ultralytics)

## 📄 License

This project is licensed under the MIT License - see the [LICENSE](https://chatgpt.com/c/LICENSE) file for details.