`

Object Detection YOLOv8

Overview

I Implemented object detection using YOLOv8 to detect persons and various PPE items using a custom-trained YOLOv8 model. It includes:

- Converting XML annotations from the PascalVOC format to YOLO format.
- Training a YOLOv8 model on a custom dataset to detect Person
- Training another YOLOv8 model on custom dataset to detect PPE .
- Inference Pipeline
- Making predictions and visualizing the results with bounding boxes.

WORKING FLOW

➢ Converting PascalVOC Annotations to YOLO Format

The dataset annotations provided in PascalVOC XML format need to be converted to YOLO format for training the YOLOv8 model. The conversion ensures that the annotations are in the required format for YOLO, where each line in the .txt file corresponds to an object in the image with normalized bounding box coordinates.

1. **Parsing XML Annotations** - Extracted object class names and bounding box coordinates (xmin, ymin, xmax, ymax).
2. **Bounding Box Processing** -Convertsed bounding box cordinates into YOLO format .
3. **Handling Image Dimensions** -Retrieved image dimensions to normalize bounding box coordinates. ,used OpenCV (`cv2.imread()`) to get image height and width.
4. **Generating YOLO Annotation Files** - Created a `.txt` file for each XML annotation file in the YOLO format. provided class index and normalized bounding box coordinates.
5. **Managing Class Labels** -Maintains a class list and assigns a numerical class ID.

NOTE : Saved the executed code in PascalVOC_To_Yolo .py file

`

➢ YOLOv8 Model - Person Detection

1. Once the dataset is prepared, the YOLOv8 model is trained on custom dataset.
2. The dataset should included images and corresponding YOLO-format labels.
3. Split the dataset into Test ,Train ,validation set
4. Trained the model for over 140 epochs , handled loading images and annotations and saving the best weights **(best.pt YOLOv8 model)** for later inference.
5. The model supports detection only for **PERSON** class

NOTE : Trained model best weights saved as best_Person.pt

➢ YOLOv8 Model - PPE Detection

1. The model supports detection on cropped person images of classes -hard-hat, gloves, mask, glasses, boots, vest, ppe-suit, ear-protector, safety-harness
2. The dataset should included images and corresponding YOLO-format labels.
3. Split the dataset into Test ,Train ,validation set
4. Trained the model for over 140 epochs , handled loading images and annotations and saving the best weights **(best.pt YOLOv8 model)** for later inference.

NOTE : Trained model best weights saved as best_PPE.pt

➢ Inference and Prediction

1. I implemented a pipeline that takes an image directory as input, performs detection using both YOLOv8 models, and saves the results in another directory.
2. First, I applied the person detection model on full images to identify and extract bounding boxes for persons.
3. Next, I crop these detected person regions and pass them through the PPE detection model. To ensure proper alignment, I converted the PPE detections from cropped images back to full-image coordinates.

`

4. Finally, I visualized the results by drawing bounding boxes and confidence scores using cv2.rectangle() and cv2.putText(),
5. Saved the processed images in the output directory in two folders **annotated** and **cropped**

NOTE : saved as inference.py

Results

The project demonstrates the effectiveness of YOLOv8 for detecting persons and PPE items.

Results include high-confidence detections for various safety equipment in real-world images.