Applied Computer Vision Intern - Assignment

Information:

You will find a dataset directory named "Datasets.zip" which have images along with annotations for the following classes:
Person, hard-hat, gloves, mask, glasses, boots, vest, ppe-suit, ear-protector, safety-harness

Inside the dataset, you will find two directories, namely **images** and **annotations**. Annotations are in PascalVOC format which needs to be converted into **yolov8** format for training the "Object Detection" model.
Along with these two directories, we have also provided classes.txt for class mapping

Important Instructions:

- You have to train a person detection model on the whole image.
- For PPE detection, please train another model on cropped images after cropping the person's bounding box.
- Please make suitable assumptions regarding class filtering/balancing imbalance classes etc. for reaching an optimized solution).
- You can drop some classes as well if you are seeing inconsistent results but atleast 5 classes model must be trained for ppe detection.
- Please zip everything and share in the email. Please refer to the **submission** section for instructions on submission.
- Dataset: https://drive.google.com/file/d/1myGjrJZSWPT6LYOshF9gfikyXaTCBUWb/view?usp=sharing

Problem Statement

1. Write a python script to convert the annotations from PascalVOC format to yolov8 format.
2. Train yolov8 object detection model for person detection (https://docs.ultralytics.com/).
3. Train another yolov8 object detection model for **PPE detection on cropped person images.** (hard-hat, gloves, mask, glasses, boots, vest, ppe-suit, ear-protector, safety-harness). ***You have to implement a logic to convert ppe annotations from full images(having multiple***

person) to cropped single person image. For eg: If in one full image you have 3 person, then for ppe detection model training, you'll have 3 images, one crop for each person.

4. Write the flow which will take an image directory as input, perform inference through both the models and save them in another directory (inference.py). ***Weights from question 2 will be inferred on full image and weights from question 3 will be inferred on cropped images from the output of question 2 weights. While saving, you have to find a logic to convert the predictions from cropped person images to full images***
5. For drawing the predicted bounding boxes and confidence, please use opencv's **cv2.rectangle()** and **cv2.puttext()** and **NOT** yolo's inbuilt function for drawing.
6. Report containing the approaches, learning and evaluation metrics in pdf format.

Submission:

Question 1: Please name your script "pascalVOC_to_yolo.py" which will take two paths, fist path is base input directory path and second path is output directory where yolov8 annotations will be saved. Please use python's **argparse** library (https://docs.python.org/3/library/argparse.html) as we will be running the script with the command line.

Question 2 and 3: Put weights files in the "**weights**" named directory.

Question 4 and 5: Please name your script "inference.py" and use **argparse** library. This script will take the following 4 arguments: input_dir, output_dir, person_det_model and ppe_detection_model. Use opencv to draw the bounding boxes and put texts.

Question 6: Report must contain the logic used in question 2

Final Notes:

- Please follow best practices of model training and scripting to achieve the best trade-offs between speed and accuracy.
- Whole image and cropped image sample have been put in "dataset.zip"

Ensure:

- Adherence to submission guidelines.
- All scripts run without errors.

Failure to comply with these requirements will result in the rejection of your application.