Task: Human Pose Estimation Using a Custom YOLO-Pose Model

Objective of the Task

The objective of this task is to help students gain practical experience in:

- Understanding human pose estimation and keypoint detection.
- Preparing and training a custom dataset using the YOLO-Pose framework.
- Performing model training, validation, and inference.
- Deploying a trained model on a live camera feed.
- Visualizing and interpreting pose key points in real time.

Instructions for Performing the Task

1. Environment Setup

Set up a Google Colab or Jupyter Notebook environment and install the required libraries:

- ultralytics
- opencv-python
- torch
- torchvision
- matplotlib

2. Dataset Preparation

- Select or create a dataset suitable for human pose estimation.
- The dataset should contain annotated keypoints for the target poses.
- Organize the dataset according to the YOLO-Pose format.
- Split the dataset into training and validation sets.

3. Model Training

- Train a custom YOLO-Pose model on the prepared dataset.
- You may use any YOLO-Pose variant (e.g., YOLOv8n-pose).
- Record the training configuration, including:
    - Number of epochs
    - Batch size
    - Image size
    - Training duration

4. Model Validation

- Evaluate the trained model on the validation dataset.
- Report key metrics such as:
    - mAP (Mean Average Precision)
    - Precision
    - Recall

5. Live Camera Inference

Using a webcam (laptop webcam or external camera):

- Load the trained YOLO-Pose model.
- Perform real-time pose estimation on the camera feed.
- Display:
    - Detected person(s)
    - Pose keypoints
    - Skeleton connections

The inference should run continuously and update the pose visualization in real time.

6. Output Recording

Record a short demonstration video (30–60 seconds) showing:

- The live camera feed.
- Real-time pose detection.
- Multiple poses or movements being detected successfully.

Requirements for Submission

Students must submit:

1. **Training Notebook (.ipynb)**
    - All cells executed.
    - Training and inference code included.
2. **Dataset Details**
    - Dataset source or description.
    - Number of training and validation samples.
3. **Trained Model**
    - Best model weights (`best.pt`).
4. **Demonstration Video**
    - 30–60 second recording of live webcam inference.