The University of Mirpurkhas
Face Recognition Project

Object Detection Libraries in Python

Yolov8 – Object Detection:

This library can help to detect object in given frame. it will use in python and it helps to inform the compiler which object is front of the camera.

In Face Recognition project we will use it for counting the employees in given frame.

First Camera setup (main gate Camera): first generate the code given frame and creating object detection. After that we will connect with **CCTV camera IP Camera** for detection. When group enter in our given frame after that it will count and insert the counting as a format data in excel file.

Object	In	out	off
Person 1	**12:00 PM**	**12:30 PM**	**02:20 PM**
Person 2	**12:00 PM**	**01:33 PM**	**1:20 PM**

Fast API – Data Service:

This library makes easier way to data shifting and location file. When we have to save video recording into database in this case, it will use there. Simple and easy to use it. The code will be short and just create API information.

Example:

🔌 **API Endpoints:**

✦ **POST /api/track/video - Upload and process video files.**

✦ **POST /api/track/camera - Track from live camera with duration control.**

✦ **GET /api/download/{filename} - Download processed output.**

✦ **DELETE /api/cleanup - Remove temporary files.**

Unicorn – Data Server (with FastAPI)

A high-performance ASGI server that runs Python web apps like FastAPI.

Without a server, your FastAPI code can't listen for web requests.

Uvicorn is the tool that opens the port and handles incoming traffic.

How Uvicorn fits into our project?

we can use FastAPI + Uvicorn in a face-recognition project by turning your Python code into API endpoints.

Instead of running everything in a local script, you expose features like uploading an image, running recognition, or starting a camera session through HTTP requests.

Our Targets:
Camera Setup – Install System – Counting People – Auto Timing – Data Insert – Excel file (currently) – Video Recording.
Features:
Video file processing - Adjustable confidence thresholds - Downloadable processed videos – Real time visualization with bounding boxes and IDs - live camera feed tracking (**IP Camera or Web Cam**).

What I think to do?
I believe, we work harder only then we will be able to complete this project otherwise not. When people in our given border line then our program recognize their faces with their given Data like **name, time, ID number**. In addition, when people go out or in then our program recognize them, in current file with timing that data must be inserted into a file when they crossed the given border.
On the other hand, every class camera will be installed by University of Mirpurkhas. Then we create a **dashboard**, in which we will provide the timing option on dashboard according our classes timing (simplest way to collect timing attendance) of each person.
Sample of dashboard:
Camera 1 - Timing (09:00 – 14:20) Computer Science Area
Camera 2 - Timing (10:00 – 14:20) IT Department Area
Camera 3 - Timing (08:00 – 14:20) English Department Area
Camera 4 - Timing (12:00 – 14:20) Commerce Department Area

Real-Time People Tracking & Counting System Project:
His Repository on GitHub (for analyzing or checking management): https://lnkd.in/ehm4K8DM
He generated project for tracking people with deep machine learning. After completing the project, he uploaded his work on GitHub.

Researched By Muhammad Bin Mansoor