Introduction

The task is to ==identify frames== in a video where a staff member (with a name tag) is present. Additionally, the task aims to ==locate the x and y coordinates== of the staff member when they are present in the clip. This document outlines the solution approach, including assumptions, methodology, tools, and the code implementation.
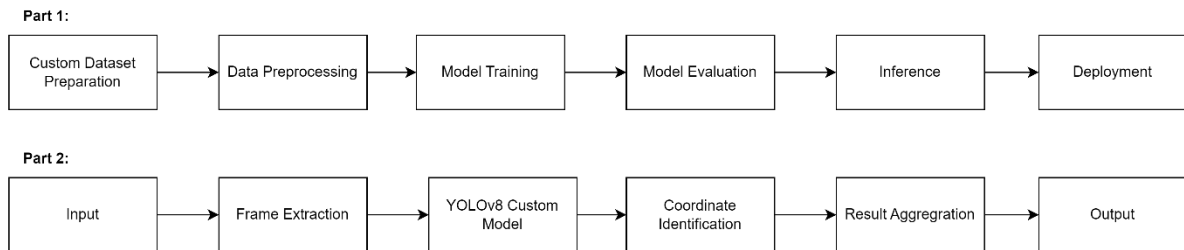
Assumptions

1. **Frame rate**: The frame rate of the video is assumed to be consistent.
2. **Resolution and quality**: The video resolution and quality are sufficient for detecting the staff name tag.
3. **Staff xy coordinates**: The x and y coordinates of the staff are assumed to be the ==center points of the bounding box==.
4. **Movement**: The staff member moves at a moderate pace that allows for clear frames where the name tag is visible.
5. **Staff identification**: The model is assumed to detect the ==staff and staff tag (two classes)==. The staff will be detected only if the staff tag is visible.

Methodology

1. **Custom Dataset Preparation**: Upload the video clip and annotate the dataset using Roboflow.
2. **Data Preprocessing**: Apply preprocessing techniques such as resizing, flipping, rotation, and brightness augmentation to enhance detection.
3. **Model Training**: Load a custom dataset from Roboflow and train the YOLOv8 model using Colab Notebook.
4. **Model Validation**: Validate the custom model's performance using the validation set.
5. **Inference**: Test the custom model using a test set and sample video.
6. **Deployment**: Deploy the custom model on Roboflow and download it for later use.
7. **Frame Extraction**: Load a video clip and extract individual frames from the video to analyse each frame for the presence of the staff and staff tag.
8. **Coordinate Identification**: Once a staff is detected, identify the bounding box around the staff and compute the x and y coordinates of the detected staff.
9. **Result Aggregation**: Compile the results to list frames where the staff is present and their corresponding coordinates in a text file.

System Design

Part 1:

Custom Dataset Preparation → Data Preprocessing → Model Training → Model Evaluation → Inference → Deployment

Part 2:

Input → Frame Extraction → YOLOv8 Custom Model → Coordinate Identification → Result Aggregation → Output

Tools and Libraries

- Python: Programming language used for implementation.
- OpenCV: For image processing and frame extraction.
- Roboflow: For dataset preparation and deployment.
- Google Colab: For training the custom model.
- Ultralytics: For implementing YOLOv8 models.

Outputs

- sample.avi: Output video from inference with the custom model in Colab Notebook.
- sample.mp4_out.mp4: Output video with the number of detected staff.
- staff_coordinates.txt: Text file with frame number and xy coordinates of the staff.

Repository Link

https://github.com/Hm-08/Staff-Identification

Conclusion

This solution provides a systematic approach to identifying staff presence in video clips by detecting staff name tags and locating their coordinates. The implementation leverages computer vision techniques and can be further refined with more sophisticated models (e.g. YOLOv9) or a larger dataset for higher accuracy.