**System Design Workflow for Staff Detection via Name Tag Recognition**

**GitHub repository:** <https://github.com/EASONTAN03/FootFallCam_staff_detect> for running the whole project.

# Detection Workflow (Idea for local deployment)

1. *Video Input*
   * Read video file, extract frames at fixed intervals.
2. *Staff Detection*
   * Detect objects <staff, staff\_tag> using a pre-trained YOLO model.
   * If any staff has a staff tag, flag frame as containing staff.
3. *Coordinate Extraction*
   * Record bounding box as coordinates of the staff’s bounding box.
   * Save staff presence frame.
4. *Output and Reporting*
   * To run with a different test video/image.

# Step 1: Modelling

**Data exploration**

The raw input data for this analysis consists of frames extracted from a 25 fps video (sample.mp4). Each frame undergoes preprocessing, including auto-orientation correction with EXIF stripping and resizing to 640×640 pixels using stretch interpolation. The dataset is then split into training, validation, and testing sets in a ratio of 0.8, 0.1, 0.1. To enhance model generalization, data augmentation is applied for training dataset to generate three versions of each image, incorporating a 50% chance of horizontal flipping, random rotation between -15° to +15°, and salt-and-pepper noise affecting 1.01% of pixels.

A group of graphs showing the number of data

AI-generated content may be incorrect.**Training configuration**: epochs=60, resolution=640x640. Below are the training results.

A graph of a graph

AI-generated content may be incorrect.A diagram of a graph

AI-generated content may be incorrect.

The object detection model trained for 60 epochs demonstrates strong performance and good convergence. Training losses (box, classification, DFL) consistently decreased, while precision and recall quickly approached 1.0, indicating effective learning. Validation results, though showing some fluctuations, also exhibited positive trends with decreasing losses and increasing mAP scores, culminating in an excellent mAP50 near 1.0 and a respectable mAP50-95 of 0.5-0.6. This suggests the model is generalizing well to unseen data, with minimal signs of overfitting by epoch 60.

# Step 2: Deployment

Demo of detecting staff with video.

[](https://www.youtube.com/embed/6_Xb9QUhnJ8?feature=oembed)

The outputs will be the real time video visualisation and detected staff was cropped in the bounding box of staff with image file name <#frame\_[bouding\_box\_metrics]>.

**Future works:** Enabling real-time video stream processing on edge devices, Name tag detection using OCR, Monitoring dashboard and Automated attendance tracking.