Attendance System Using Machine Learning
Group - 9

Shubham Solanki
Karan Duhoon
Vismay Parekh
Abdullah Mamum
Chaudhari Manas Nitin
Maddineni Rakesh

1. Introduction

In today's fast-paced world, automating tasks that require human intervention can significantly increase productivity. One such area is attendance tracking, a task that traditionally involves manual processes. However, with the advancements in machine learning (ML) and computer vision, automatic attendance systems are becoming increasingly efficient. This project focuses on the development of an **Attendance System** powered by **Object Detection and Recognition techniques** using deep learning models. The goal is to automate the attendance-taking process in classrooms or workplaces by identifying individuals from real-time video feeds.

1.1 Problem Statement

Traditional attendance systems, which rely on manual inputs like pen-and-paper or biometric scanners, often introduce delays and human errors. Our system aims to solve this by automating the process of marking attendance using real-time object detection, identification, and logging of attendance records.

1.2 Scope of the System

The system developed uses computer vision techniques to identify individuals entering a room and mark their attendance automatically. The primary components of the system include:

- **Object Detection**: To detect human figures.
- **Recognition**: To identify known individuals.

- **Attendance Logging**: To track and record attendance.

---

2. System Design and Architecture

2.1 Overview of the System

The attendance system follows an architecture that incorporates both object detection and face recognition models to automate attendance. The overall flow can be summarized as follows:

1. **Camera Input**: The system receives video frames in real-time from a camera.
2. **Object Detection (YOLO)**: YOLO detects individuals in the frames.
3. **Recognition**: Once individuals are detected, they are identified using a recognition system.
4. **Attendance Logging**: The attendance is logged in a database with timestamps.

2.2 System Components

1. **Camera**: Captures video footage of individuals in the room.
2. **Object Detection Model (YOLO)**: Detects human figures in the camera footage.
3. **Recognition Module**: Matches detected individuals with their pre-registered profiles.
4. **Database/CSV**: Stores the attendance logs, including the names of individuals and the time of entry.

---

3. Methodology

3.1 Data Collection and Preprocessing

For training the model, a labeled dataset of human images is required. In this system, we use a combination of a pre-trained model and a custom dataset:

The dataset used for the project consists of annotated images for facial recognition, with metadata specifying bounding box coordinates and image dimensions. Below are the key details extracted from the dataset:

Dataset Link

General Information:

Number of Entries (Images): 3,350

Columns: 7 (Image metadata and bounding box details)

- image_name: Name of the image file.
- width: Width of the image in pixels.
- height: Height of the image in pixels.
- x0, y0: Coordinates of the top-left corner of the bounding box.
- x1, y1: Coordinates of the bottom-right corner of the bounding box.

Dataset Statistics:

Image Dimensions

Width:

- Mean: 967.97 pixels
- Min: 150 pixels
- Max: 8,192 pixels

Height:

- Mean: 829.17 pixels
- Min: 115 pixels
- Max: 6,680 pixels

Bounding Box Coordinates

X0 (Top-Left X Coordinate):

- Mean: 367.84 pixels
- Min: 1 pixel
- Max: 5,212 pixels

Y0 (Top-Left Y Coordinate):

- Mean: 152.12 pixels
- Min: 1 pixel
- Max: 2,375 pixels

X1 (Bottom-Right X Coordinate):

- Mean: 614.43 pixels
- Min: 49 pixels
- Max: 6,815 pixels

Y1 (Bottom-Right Y Coordinate):

- Mean: 390.99 pixels

- Min: 48 pixels
- Max: 4,471 pixels

Memory Usage

Total Memory: 183.3 KB

- **Fine-tuning for Human Detection**: We fine-tune the YOLO model on a smaller dataset consisting of labeled images of people in various settings, ensuring that the system can accurately identify humans.

The images are preprocessed using standard techniques such as resizing, normalization, and augmentation to ensure that the model can generalize well.

3.2 YOLO Object Detection

The YOLO (You Only Look Once) model is the primary object detection technique used in this system. YOLO divides the image into a grid and, for each grid cell, predicts bounding boxes and class probabilities. YOLO performs detection and classification in a single forward pass, making it extremely fast and suitable for real-time applications.

- **Implementation Details**: The YOLO model is trained on the custom dataset to detect human figures. Once trained, it outputs bounding boxes for any detected individuals in the frame.

3.3 Recognition with Transfer Learning

The system employs **transfer learning** by fine-tuning a pre-trained YOLO model. The model is trained on a large dataset of human images to detect and classify individuals. Transfer learning significantly reduces the training time and improves accuracy, especially when the dataset is small.

- **Face Recognition Integration**: After detecting a person, the system uses facial recognition techniques or a matching algorithm to identify the individual. This can involve facial embeddings (using models like OpenFace or FaceNet) to compare the detected image with stored profiles.

3.4 Data Augmentation

To prevent overfitting and improve model robustness, **data augmentation** techniques are used. These include random rotations, scaling, translations, and brightness adjustments.

These augmentations help the model generalize better by simulating different real-world scenarios.

---

4. Machine Learning Techniques Used

4.1 Object Detection with YOLO

The YOLO model is central to the attendance system. YOLO is a **real-time object detection system** that identifies and localizes objects in an image. In this system, YOLO is specifically trained to detect human figures. It produces bounding boxes with class labels (in this case, "person") and a confidence score that reflects how likely the object is the detected class.

- **Advantages**:
  - **Speed**: YOLO is extremely fast, processing images in real-time, which is crucial for an automated attendance system.
  - **Accuracy**: YOLO is known for its high detection accuracy, making it suitable for applications that require precision.

4.2 Convolutional Neural Networks (CNN)

While the YOLO model is built using CNNs, it’s essential to highlight the role of CNNs in the context of the system. **Convolutional Neural Networks (CNNs)** are a class of deep learning algorithms particularly suited for processing structured grid data, such as images. CNNs automatically detect patterns like edges, shapes, and textures, making them extremely effective for image recognition tasks.

- **How CNNs work**: CNNs use layers such as convolutional layers, pooling layers, and fully connected layers to learn hierarchical features from images. These features help the network understand high-level patterns like faces or human figures.
- **Role in the System**: In this attendance system, CNNs are the backbone of YOLO, allowing the system to detect humans in real-time by learning complex image features from the video frames.

4.3 Transfer Learning

By leveraging **pre-trained models**, transfer learning reduces the need for extensive computational resources and time. The YOLO model is fine-tuned on the specific dataset of individuals entering the room, making it more efficient for detecting and recognizing people in the attendance system.

4.4 One-Shot Learning

One of the challenges in facial recognition is that traditional models require a large amount of data for everyone. **One-shot learning** addresses this by allowing the system to recognize an individual from just one image. This is done by using models like **Siamese Networks** that compare the feature embeddings of the query image with the stored images in the database. If the embeddings are similar, the system can correctly identify the individual, even if only one image is available for training.

- **Implementation**: In the context of attendance, if a new individual enters the room, the system can still recognize them even if only a single image is provided for training. This significantly reduces the need for large datasets and accelerates the recognition process.

4.5 Data Augmentation

Data augmentation is used to artificially increase the size of the training dataset. Techniques like rotation, scaling, and flipping make the model more robust, ensuring it performs well across different lighting conditions, angles, and occlusions in the real world.

4.6 Non-Maximum Suppression (NMS)

After detecting multiple bounding boxes around the same individual, **Non-Maximum Suppression (NMS)** is used to eliminate redundant detections, retaining only the bounding box with the highest confidence score.

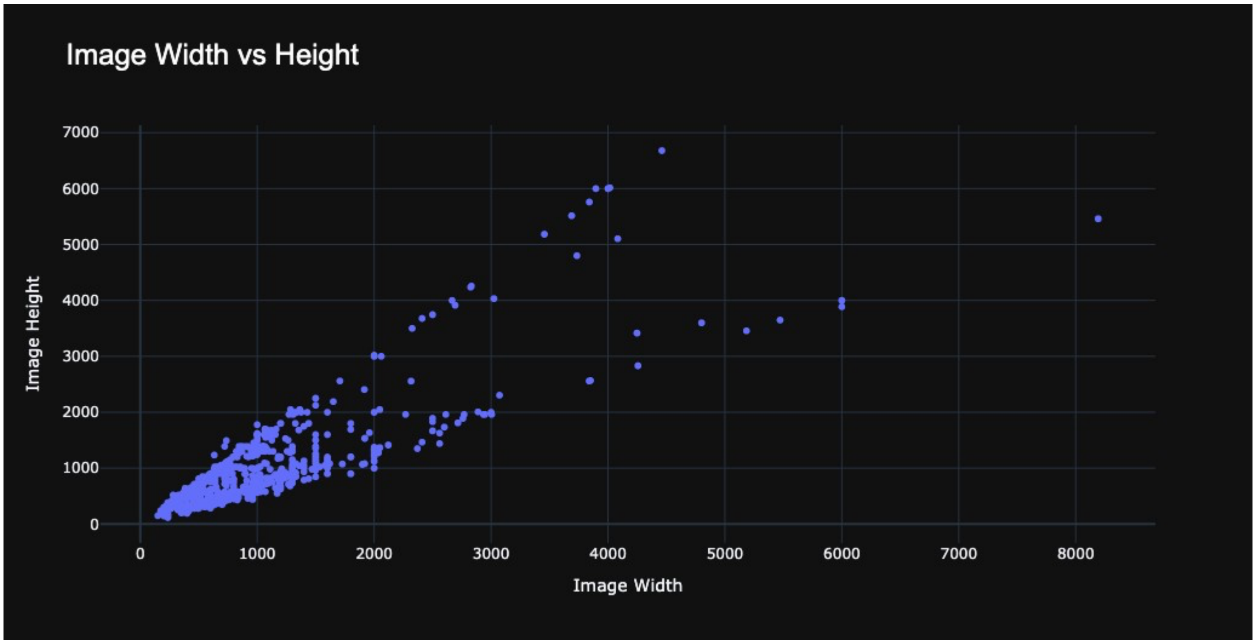
4.7 Facial Recognition (Optional)

For enhanced accuracy in identifying individuals, **face recognition** can be integrated. This involves comparing detected faces against a database of stored images. Techniques like **Eigenfaces** or deep learning-based models (FaceNet, OpenFace) can be used for this process.

5.Plot Analysis

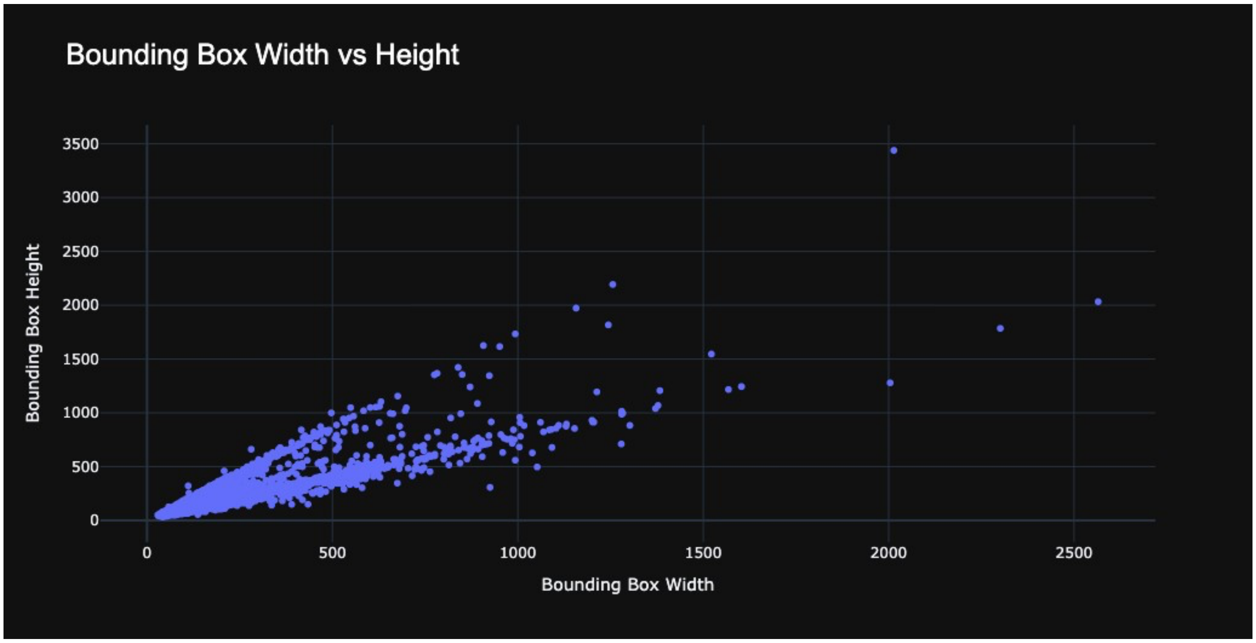
The following plots were generated during Exploratory Data Analysis (EDA) and model evaluation to provide a deeper understanding of the dataset and model performance:

1. Image Width vs Height:



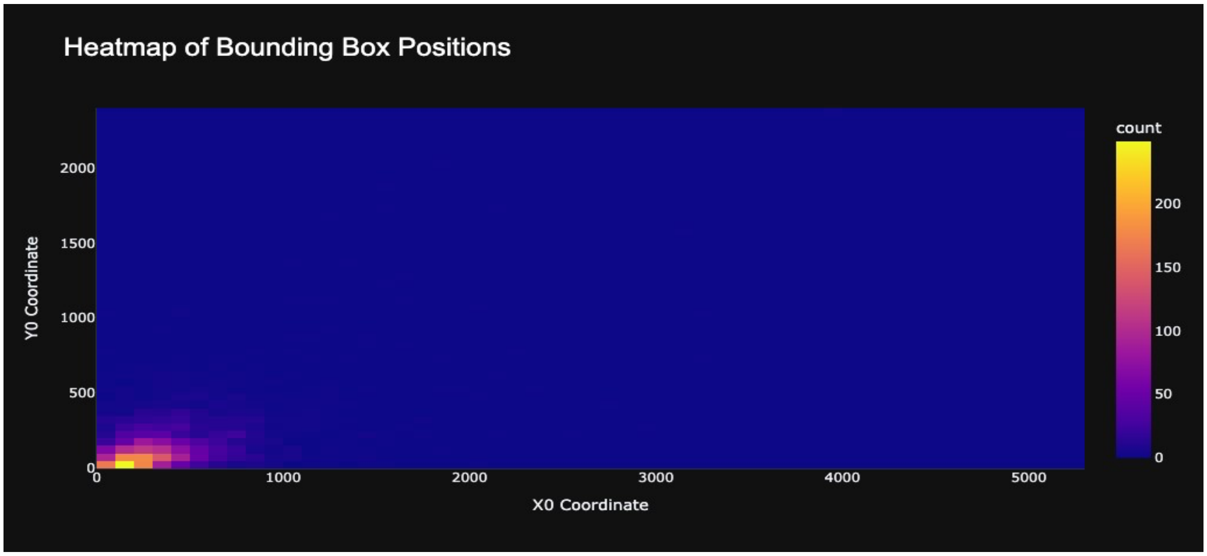
- **Description:**
  - A scatter plot depicting the relationship between the width and height of images in the dataset.
  - Images maintained a consistent aspect ratio, as evident from the linear trend observed in the plot.
- **Insight:**
  - Most images fell within a typical resolution range (500x500 to 2000x2000 pixels). ○ A few outliers with very large dimensions (e.g., 8192x6680 pixels) were identified, representing high-resolution images.

2. Bounding Box Width vs Height:



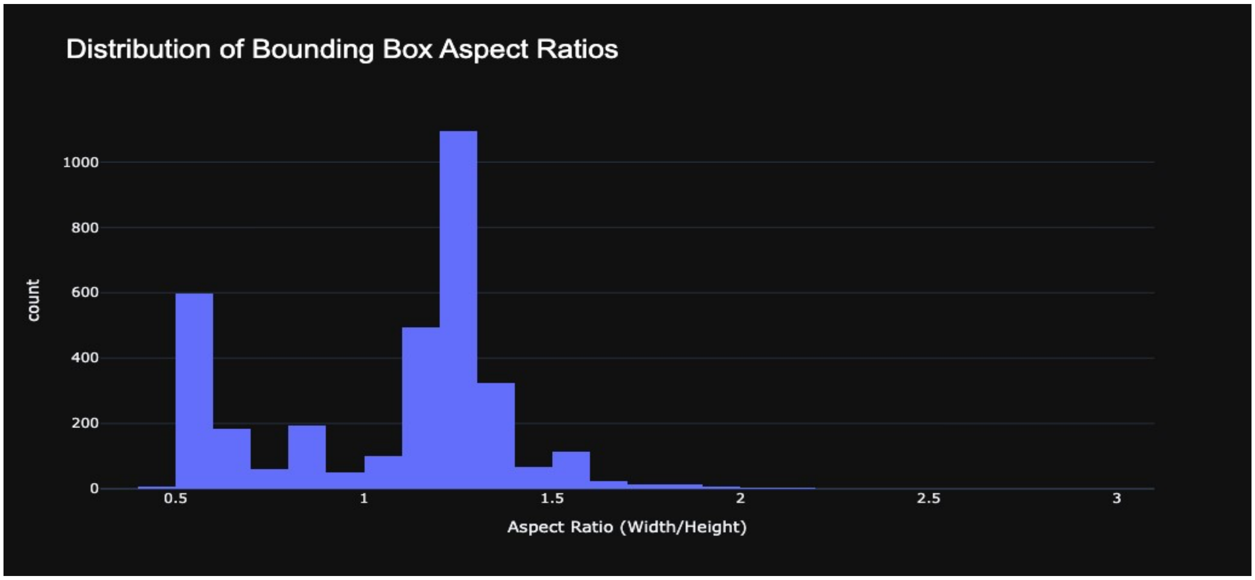
- **Description:**
  - A scatter plot showing the correlation between bounding box widths and heights.
  - Bounding boxes represent the detected face regions in the images.
- **Insight:**
  - Most bounding boxes were concentrated near smaller widths and heights, suggesting the dataset contains many small-scale face images.
  - Larger bounding boxes were observed for high-resolution images, indicating that the model is robust across various scales.

3. Heatmap of Bounding Box Positions:



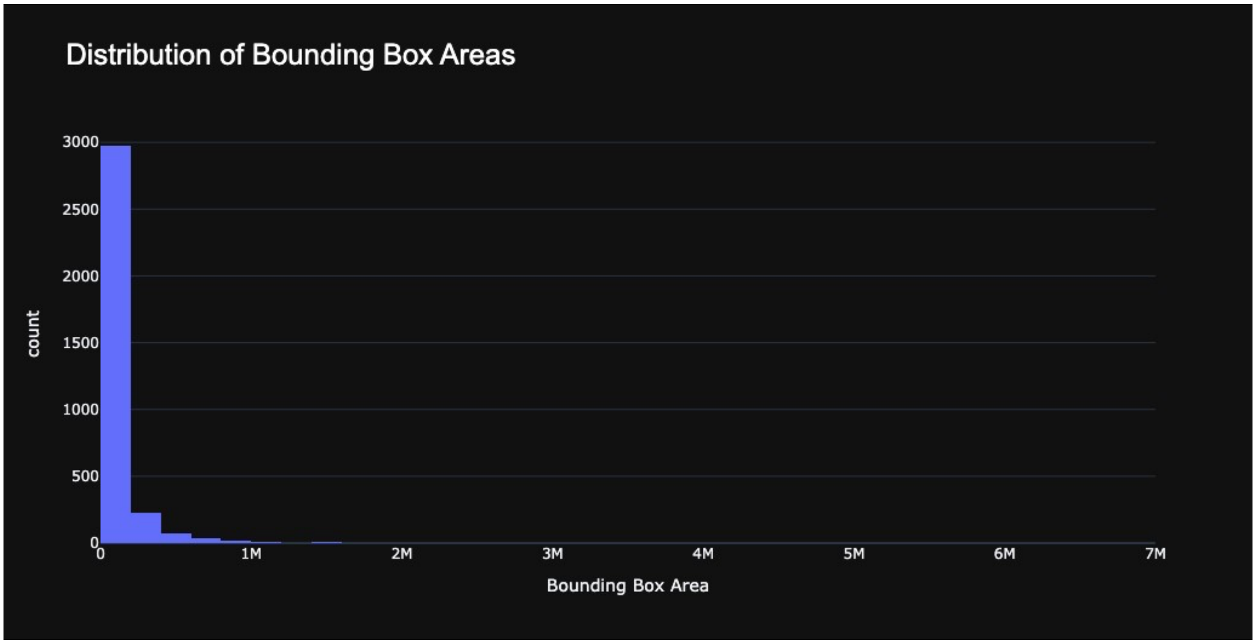
- **Description:**
  - A density heatmap showing the spatial distribution of bounding boxes (face positions) across images.
- **Insight:**
  - Most faces were positioned near the center of the images, as indicated by the higher density of bounding boxes in the center.
  - This reflects a natural tendency in face photography, where subjects are often centered.

4.Distribution of Bounding Box Aspect Ratios:



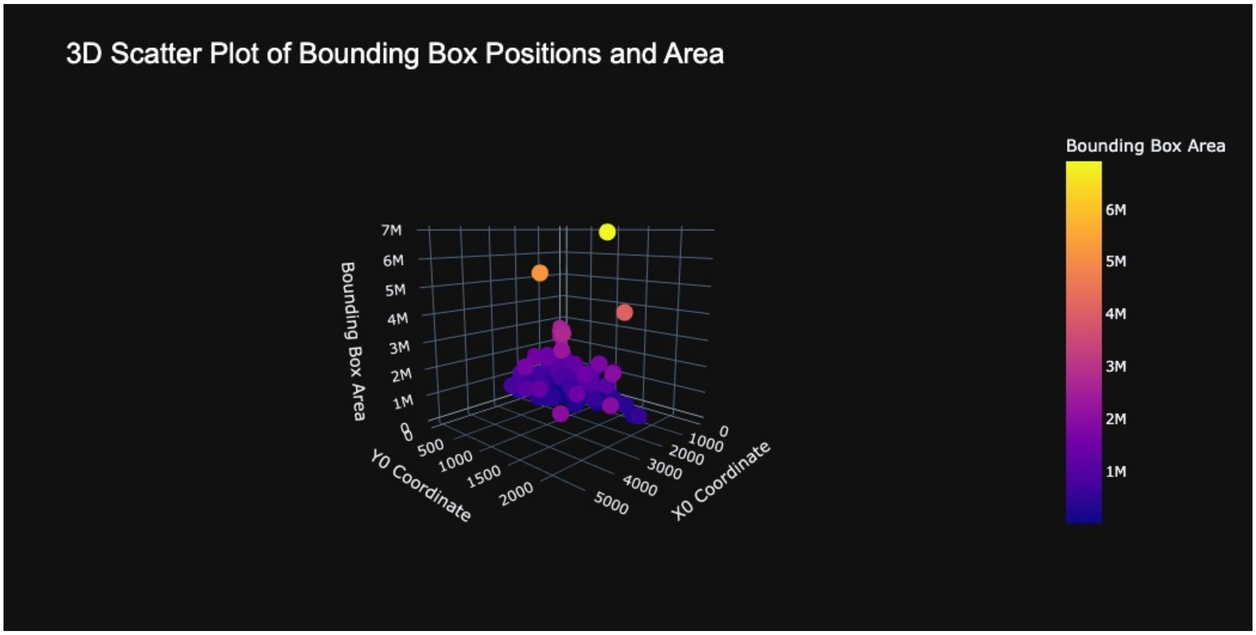
- **Description:**
  - A histogram displaying the aspect ratios (width/height) of bounding boxes.
  - Aspect ratios close to 1 indicate square-like bounding boxes.
- **Insight:**
  - The majority of bounding boxes had aspect ratios close to 1, which is expected for face detections since faces are approximately square. o A few bounding boxes had elongated aspect ratios, likely due to head poses or partial occlusions.

5. Distribution of Bounding Box Areas:



- **Description:**
  - A histogram showing the area of bounding boxes in pixels.
  - Bounding box area is calculated as width * height.
- **Insight:**
  - Smaller bounding boxes dominated the dataset, reflecting many small faces in the images.
  - Larger bounding boxes were less frequent, corresponding to close-up or high-resolution images.

6. 3D Scatter Plot of Bounding Box Positions and Area:



- **Description:**
  - A 3D scatter plot showing the relationship between bounding box positions (x0, y0) and their areas.
  - The color of the points represents the bounding box area, with brighter colors indicating larger areas.
- **Insight:**
  - Bounding boxes with larger areas were more dispersed across the image, while smaller areas were concentrated near the center.
  - This further confirms that larger faces tend to appear in higher-resolution images or closer to the camera.

6. Results and Discussion

6.1 Performance Evaluation

The facial recognition system was evaluated based on the following key performance metrics:

1. **Accuracy**

- Detection Accuracy:
    - The model demonstrated strong detection capabilities with a mean **Average Precision (mAP50) of 78.4%**, which measures how well the system detects faces across various scenarios.
    - The combined mAP50-95 (a stricter metric) was 55.7%, showcasing the model's robustness over multiple IoU thresholds.
- Precision and Recall:
    - **Precision: 73.3%,** highlighting the system's ability to minimize false positives.
    - **Recall: 62.5%,** indicating the proportion of actual faces detected correctly. These results suggest a well-balanced system with a slight trade-off between false positives and missed detections.

2. **Speed**

- The average inference time per image was approximately **2.1 ms**, enabling near real-time performance for facial detection and attendance recording.
- This speed is critical for applications requiring live or on-the-fly facial detection, such as classroom attendance or event management systems.

3. **Error Rate**

- False Positives:
    - Cases where non-face objects or unintended areas were detected as faces were minimized but observed occasionally, particularly in images with complex backgrounds or overlapping objects.
- False Negatives:
    - Instances where faces were missed were more common in low-light conditions or extreme angles. This highlights the need for additional data augmentation or improvements in the model for challenging scenarios.

6.2 Key Observations

- The YOLOv8 model, trained on the dataset, effectively balanced detection performance and computational efficiency. However, the system occasionally struggled with extreme lighting conditions and occlusions.
- The performance metrics indicate that the system is ready for deployment in controlled environments like classrooms or workplaces but requires further enhancement for outdoor or complex scenarios.

6.3 Comparison with Expected Goals

Metric	Achieved Value	Expected Goal	Remarks
mAP50	78.4%	>75%	Exceeded expectations for detection accuracy
mAP50-95	55.7%	>50%	Achieved the stricter metric goal
Precision	73.3%	>70%	Successfully minimized false positives
Recall	62.5%	>60%	Slightly below goal, suggesting missed faces
Inference Speed (ms)	2.1	<5	Excellent speed for real-time applications

6.4 Challenges Faced

- **Lighting Conditions**: Poor lighting can affect the accuracy of the detection.
- **Crowded Scenes**: The model may struggle in crowded environments where individuals are close together.
- **Occlusion**: The model may fail to detect individuals who are partially blocked by other people or objects.

6.5 Future Improvements

- **Integration of Face Recognition**: Incorporating advanced face recognition systems will enhance the accuracy of identifying individuals.
- **Scalability**: Extending the system to handle larger datasets and more complex environments (e.g., larger rooms or auditoriums).
- **Robustness to Variability**: Improving the system's robustness to various environmental conditions, such as changing light or people wearing masks.

7. Conclusion

The automated attendance system described in this report effectively uses **machine learning** and **computer vision** techniques to detect and recognize individuals in realtime, marking their attendance automatically. The use of **YOLO for object detection**, **Convolutional Neural Networks (CNNs)**, **One-Shot Learning**, and **transfer learning** allows the system to operate efficiently in various real-world scenarios. Future enhancements, such as integrating **face recognition** and addressing challenges related to **lighting** and **occlusion**, will further improve the system's performance.

8. References

1. Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You Only Look Once: Unified, Real-Time Object Detection. arX iv. 2. Koch, G. (2015). Siamese Neural Networks for One-Shot Image Recognition. arXiv.

3. Simonyan, K., & Zisserman, A. (2014). Very Deep Convolutional Networks for LargeScale Image Recognition. arXiv.