Infosys Springboard Virtual Internship 6.0

Crowd Count

Real-Time People Counting and Crowd Monitoring System

By: Shanta Angadageri
Mentor: Asma
Batch: 3
Start Date: 22 September 2025
End Date: 21 November 2025

CONTENTS

- ▶ Introduction
- ▶ Problem Statement
- ▶ Objectives
- ▶ Proposed Model
- ▶ Tools and Technologies
- ▶ Results
- ▶ Applications
- ▶ Future Enhancement
- ▶ Conclusion

INTRODUCTION

This project is a real-time crowd and video analysis web application built using Flask and advanced computer vision models such as YOLO combined with ByteTrack for tracking individuals. It enables users to securely upload videos and images for detailed crowd monitoring, providing live video streams annotated with detection data, zone-based counting, and predictive alerts. The system also features dynamic data visualization through charts, heatmaps, and downloadable reports, delivering actionable insights to enhance public safety and operational efficiency in crowded environments.

PROBLEM STATEMENT

Managing and ensuring safety in crowded public spaces is a complex challenge, especially with increasing populations worldwide. Traditional manual monitoring methods are labor-intensive, error-prone, and often ineffective in real-time scenarios. There is a critical need for an automated, intelligent system that can accurately detect, track, and analyze crowd density and movement in real time to prevent overcrowding, reduce risks, and enable timely interventions. This project addresses these challenges by combining computer vision with web technologies to provide an efficient, scalable crowd analysis and alert system.

OBJECTIVES

▶ To Detection and Tracking using YOLOv8 model with ByteTrack for real-time detection and tracking of people in uploaded videos and webcam streams.

▶ To enable zone-based crowd analysis, tracking density and movement within defined regions and provide timely alerts

▶ To Build a secure role-based access control for the admin that allows for user management.

▶ Provide dynamic visualizations including bar charts, line charts, and heatmaps that illustrate crowd trends over time, empowering users to analyze and understand crowd behavior.

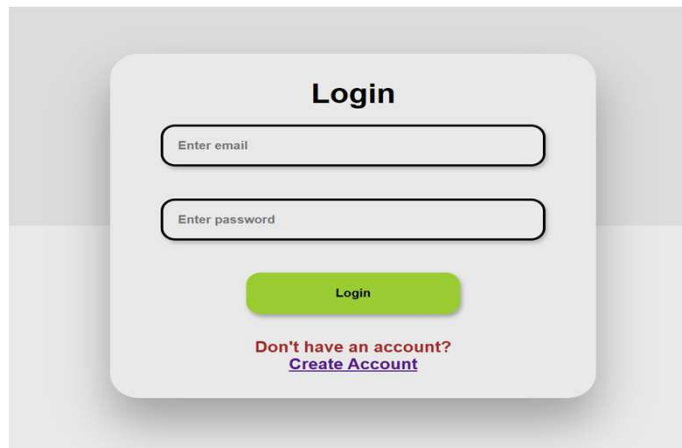▶ Allow users to download CSV files containing detailed crowd analysis
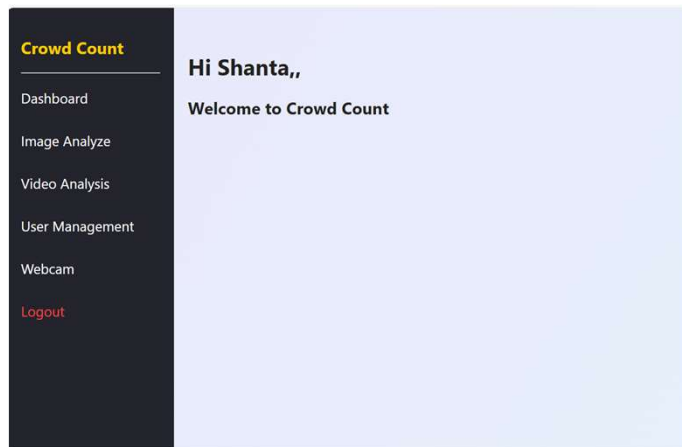
PROPOSED MODEL
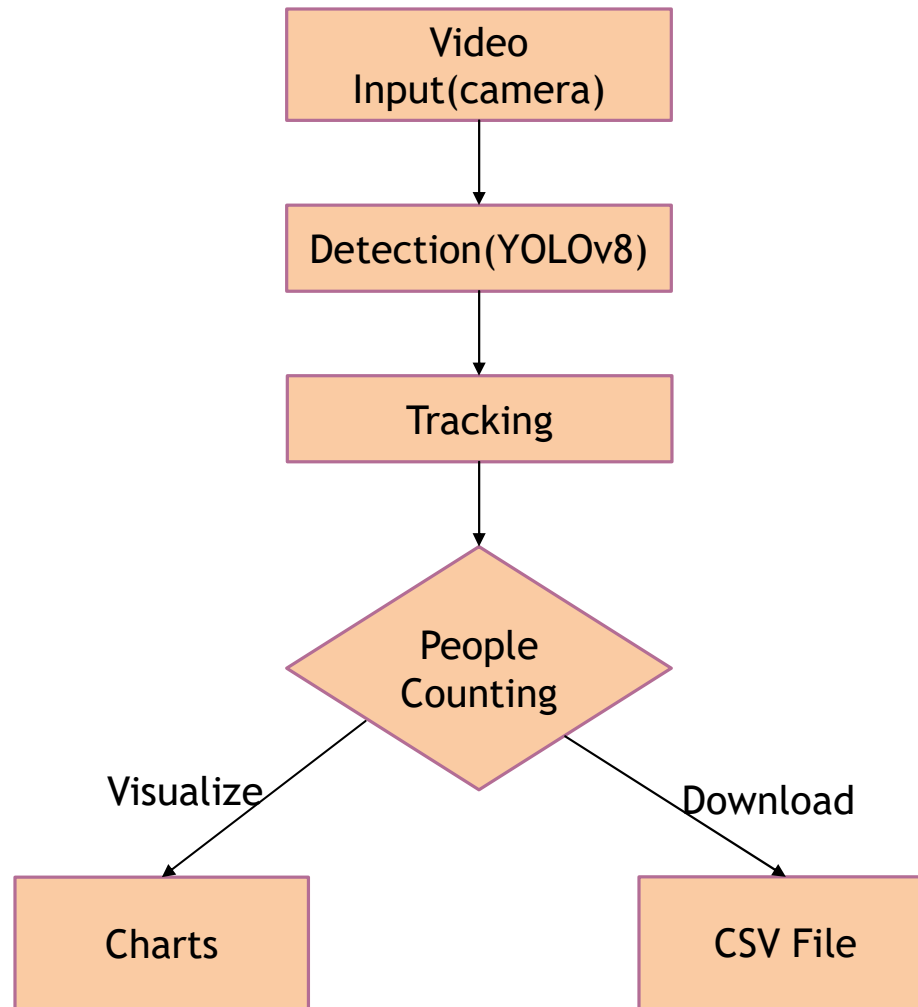
TOOLS AND TECHNOLOGIES

Component	Technology	Role in the Project
Web Framework	Flask	Main backend web server for routing, authentication, and UI logic
Database	SQLite	Secure storage of user credentials, roles, and sessions
Authentication	JWT (PyJWT)	User authentication; role-based access; secure session management
Object Detection	YOLOv8 (Ultralytics)	The Deep Learning Model used to Detects people in video and image frames
Object Tracking	ByteTrack	Tracks detected people across frames for consistent zone counting
Video/Image Input Processing	OpenCV (cv2)	Reads video/image files, processes frames, draws bounding boxes

Component	Technology	Role in the Project
Data Visualization	Matplotlib, Pandas	Generates charts, graphs, heatmaps, and CSV reports
Front-End	HTML, CSS, javascript	Provide structure and styling for web pages
Real-Time Video Streaming	OpenCV, Flask (Response)	Provides live object detection video streams in the browser
Live & Historic Analytics	Python + Visualization	Enables live zone counts, population trends, and report downloads
User Management	Flask, SQLite	Registration, login, user-role checks, and admin features

Milestone1

Milestone2

If Detections Exist

- Track_id:The unique ID for the person (remains consistent as long as person is visible).
- Bouning box: x1, y1, x2, y2
- Center:cx=(x1+x2)/2, cy=(y1+y2)/2 for zone determination, visualization, and tracking
- Add track_id to the video-wide set of unique IDs.
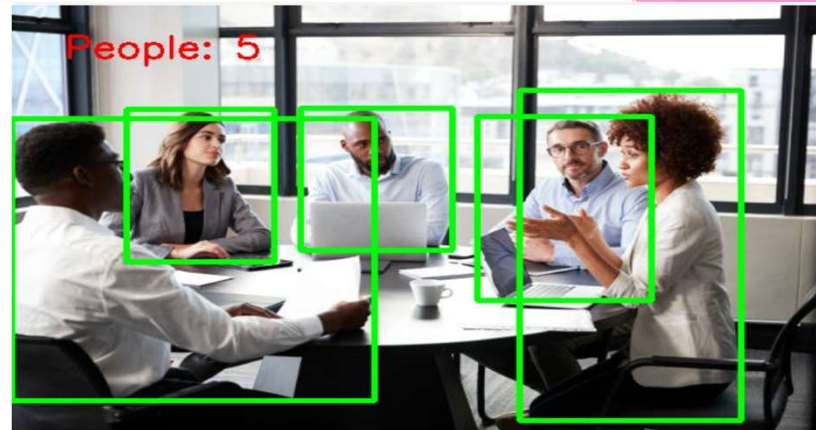
People: 5

Detected People: 5

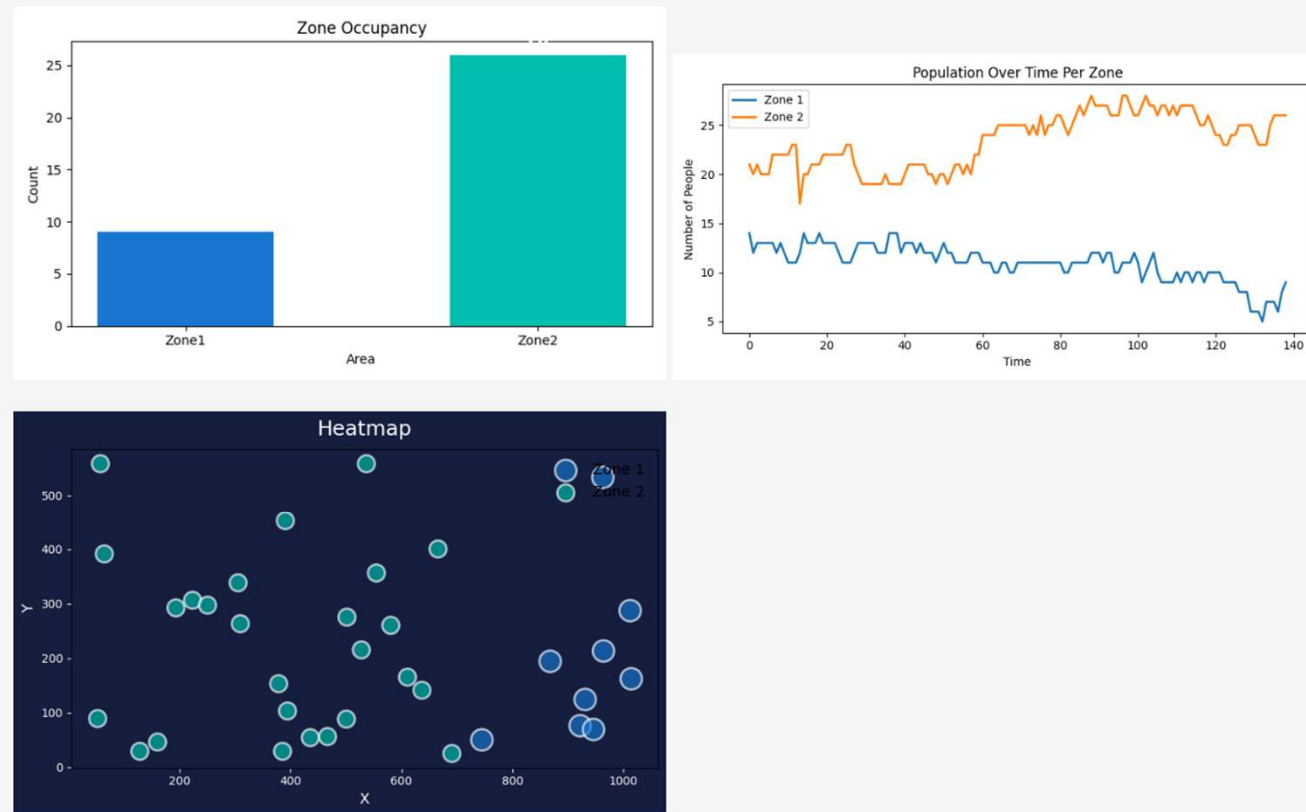Streaming Analyzed Video:

Zone 1: 2
Total: 11

Milestone3

- Zone 1: If center is inside the defined rectangular zone, add to zone 1 set/list, draw bounding box in yellow.
- Zone 2: Otherwise, add to zone 2, draw bounding box in magenta.
- Label each tracked person with their unique track ID
- If zone 1 is too crowded, add a big red alert at the bottom of the frame.

Visualizations

Milestone4

APPLICATIONS

▶ **Retail and Commercial Environments**: Analyze visitor density in malls, stores, and commercial venues to improve customer experience, optimize staffing, and reduce bottlenecks. Manage queues and adjust entry/exit flows based on live occupancy data.

▶ **Public Safety and Event Management:** Monitor crowd density at large gatherings such as concerts, sporting events, and religious festivals to prevent overcrowding and stampedes. Detect unusual or risky behaviors and generate real-time alerts for security teams

▶ **Transport Hubs and City Spaces:** Track crowd movement and occupancy at airports, train stations, and bus terminals for efficient resource allocation and congestion prevention Assist city planners in analyzing pedestrian flow and optimizing urban design for safer and more accessible public spaces.

▶ **Threat Detection and Security:** Identify potential threats like overcrowding, stampedes, and abnormal crowd movements using AI and machine learning models. Facilitate quick incident response by integrating alerts and logs with emergency services systems.

FUTURE ENHANCEMENT

▶ Sensor Fusion and IoT Integration: Connect multiple sensor types—such as cameras, LiDAR, thermal imaging, Wi-Fi, and RFID—to deliver more accurate crowd analysis, location tracking, and environmental awareness in real time.

▶ Behavior Analysis Beyond Counting: Enhance crowd analytics to recognize specific behaviors, sentiments, or anomalies (e.g., panic, aggression, distress) by integrating deep learning models capable of complex activity recognition.

▶ Edge Computing & Real-Time Response: Employ edge computing with 5G connectivity to process video and sensor data directly at the source, reducing latency and enabling instant alerts and interventions during large-scale events.

▶ AI-Powered Predictive Analytics: Integrate advanced AI models that use historical and real-time data to forecast crowd behaviors, movement patterns, and potential risks, enabling proactive safety measures and optimized event planning.

CONCLUSION

This project successfully demonstrates an automated, real-time crowd analysis system utilizing deep learning and computer vision techniques such as YOLO and ByteTrack. The system achieves accurate detection, tracking, and zone-based counting of individuals in live video, webcam, and image feeds. By integrating a secure Flask web interface, users are able to upload media, view live analytics, generate heatmaps, and download detailed CSV reports—all contributing to enhanced crowd management and public safety. The clear results and flexible architecture highlight the project's usefulness across various applications, including event security, urban planning, and real-time monitoring. With continued refinement and advanced analytics, this work lays a strong foundation for future enhancements and broader deployment in smart environments.

THANK YOU