IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 5, Issue 3, October 2025
Impact Factor: 7.67

AI Smart Bus Stop Crowd Detection System

Ankita Vikas Kurkute[1], Vedanti Nitin Ballal[2], Sanskruti Bhimraj Bhogade[3], Ankita Vijay Gadekar[4], Mrs. V. M. Khanapure[5]
Department of Information Technology[1-5]
Government Polytechnic, Pune, India

Abstract: *Urban public transport systems face growing pressure from increasing passenger volumes and unpredictable demand patterns, leading to overcrowded stops, inefficient fleet allocation, and reduced passenger comfort. The AI Smart Bus Stop Crowd Detection System integrates computer vision, time-series forecasting and conversational AI to provide real-time crowd monitoring, arrival-time prediction and a passenger-oriented query interface. The system uses camera feeds processed by a fine-tuned YOLOv8 model (person detection) running on edge or cloud GPUs to estimate instantaneous occupancy. A buffering layer with simple smoothing reduces frame-to-frame jitter. A Prophet- based forecasting module fuses historical bus GPS traces and live counts to produce robust ETA estimates.*

Keywords: Crowd detection, YOLOv8, real-time monitoring, bus arrival prediction, Prophet, Flask API, React dashboard, LLM chatbot, smart transit

I. INTRODUCTION

In rapidly urbanizing cities, public transportation systems play a crucial role in ensuring affordable and efficient mobility for millions of commuters. However, the increasing dependency on public buses has also introduced serious operational challenges such as overcrowded bus stops, inconsistent schedules, and passenger dissatisfaction during peak hours. As cities like Pune, Mumbai, and Bengaluru experience significant daily commuter volumes, manual observation or schedule-based management is no longer sufficient to maintain smooth transport operations.

Traditionally, most public transport systems rely on Global Positioning System (GPS) data to track bus movement and estimate arrival times. While GPS tracking improves basic visibility, it fails to capture real-time ground-level passenger conditions, especially the density of commuters waiting at each bus stop. Consequently, buses often arrive either overcrowded or underutilized, leading to inefficient resource allocation, longer waiting times, and negative commuter experiences. Furthermore, unpredictable events such as weather changes, roadblocks, or festival crowds can cause significant variation in demand that static systems cannot adapt to.

With the rapid advancement of Artificial Intelligence (AI), Computer Vision (CV), and Internet of Things (IoT) technologies, it has become feasible to develop intelligent systems that monitor, analyze, and predict real-world phenomena in real time. One promising application is crowd detection and analysis at public transport stops, which can offer critical insights for both commuters and transport authorities. Integrating such a system into smart-city infrastructure can revolutionize how bus transport is managed—transitioning from reactive management to data- driven, proactive decision-making.

The AI Smart Bus Stop Crowd Detection System addresses this gap by combining deep learning-based object detection, time-series forecasting, and conversational AI. The system is designed to monitor real-time crowd density at bus stops through live camera feeds, process the video frames using an optimized YOLOv8 model for person detection, and estimate bus arrival times using the Prophet forecasting algorithm. The processed information is then displayed on a web dashboard for administrators and made accessible to passengers via a multilingual AI chatbot. This integration transforms an ordinary bus stop into a smart, connected, and self-aware node in the city's transportation ecosystem.

A. Problem Statement

Key problems addressed:

• Lack of real-time, stop-level crowd information for commuters.

• Inability of transport operators to adapt fleet allocation based on instantaneous demand.
• Existing sensor approaches (pressure mats, turnstiles) are expensive or prone to maintenance; naïve vision solutions are sensitive to lighting, occlusion, and weather.

II. LITERATURE REVIEW

A. Overview of Existing Research

Over the past three years (2022–2025), rapid progress in computer vision and AI-based crowd analysis has transformed how urban environments can be monitored. Researchers have increasingly focused on real-time human detection, crowd density estimation, and smart public transport applications. However, despite substantial work on visual crowd counting and bus arrival prediction, very few systems integrate these capabilities into a unified, real-time deployment pipeline suitable for city-scale implementation. This section reviews major contributions from the literature and highlights the specific advancements this project builds upon.

B. YOLO-Based Object Detection and Crowd Counting

The YOLO (You Only Look Once) family of object detection models has consistently dominated real-time computer vision applications since its inception. Recent research continues to refine YOLO architectures for crowd detection and density analysis.
• Gündüz (2023) introduced an enhanced YOLO-based real-time crowd detection mechanism that counts people within a predefined Region of Interest (ROI). Their system achieved high detection accuracy with a mean average precision (mAP) above 0.85, while maintaining processing speeds above 30 FPS, making it suitable for surveillance and public monitoring scenarios. However, their evaluation was limited to indoor datasets and lacked outdoor adaptation to varying lighting and weather conditions.
• Alhawsawi et al. (2024) proposed an Enhanced YOLOv8 model incorporating context enrichment layers to improve detection in crowded, low-resolution environments. Their method reduced false positives in dense scenes by over 12% compared to the baseline YOLOv8. Yet, their model remained computationally heavy, which limited deployment on low-cost edge devices.
• Yan (2024) further optimized YOLOv8 for dense crowd scenarios by integrating a multi-scale attention mechanism, achieving higher precision on urban surveillance datasets. While effective, this method focused on dense events like concerts or rallies, not dynamic commuter environments such as bus stops.
These studies confirm that YOLO models are the leading approach for fast and accurate person detection. Our project builds on this foundation by fine-tuning YOLOv8 specifically for outdoor bus-stop environments, adding ROI masking to ignore nearby sidewalks, and using temporal smoothing for stability — all optimized for low-cost GPU or Jetson deployment.

C. Domain-Specific Research in Bus Stop Monitoring

Efforts to apply AI to public transportation monitoring have begun emerging recently. The BusStopCV Project (University of Washington, 2023) is one of the earliest prototypes designed for real-time bus stop analysis. It fine-tuned YOLOv5 for bus stop imagery to detect accessibility features (shelters, benches, passengers) and validated the concept of visual monitoring in public transit systems. However, BusStopCV mainly served as a proof of concept — it lacked predictive analytics, real-time dashboards, and integration with passenger communication tools.
Similarly, Li et al. (2022) proposed a Vision-based Passenger Counting System for public buses using CNN architectures. Their approach achieved about 92% accuracy but required dedicated sensors and high-end GPUs, making it cost-prohibitive for large-scale use.
Other 2023 studies in Transportation Research Part C and IEEE Access explored combining GPS data and IoT sensors for passenger load prediction. While these methods improved fleet management, they ignored visual data — limiting adaptability to unstructured, real-world situations.
In contrast, our system integrates camera-based crowd detection with arrival prediction and user interaction — bridging the operational and commuter-level information gap.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 5, Issue 3, October 2025
Impact Factor: 7.67

D. Forecasting and Predictive Modeling for Transport

While vision systems detect current crowd states, forecasting algorithms predict future bus arrivals or passenger load trends.

• Gao (2023) proposed an improved Prophet model for short-term traffic flow prediction, demonstrating better stability in non-linear time series than ARIMA models.

• Jeong et al. (2024) introduced a Transformer Encoder for Bus ETA prediction, which reduced mean absolute error by 10% compared to baseline LSTM models but required large datasets and high computation.

Your project strategically chooses Prophet for its balance between accuracy and interpretability. Prophet's modular nature enables incremental learning from live data, making it ideal for urban use where real-time re-training and lightweight computation are essential.

E. Alternative Sensing and Hybrid Models

Recent literature also explores multimodal approaches combining Bluetooth Low Energy (BLE), Wi-Fi, or infrared sensors with AI for privacy-preserving crowd estimation.

• Ikenaga et al. (2025) used BLE signal patterns to estimate subjective crowdedness with 88% correlation to actual human counts.

• MDPI Sensors (2024) presented hybrid frameworks fusing vision and proximity sensing to improve robustness under occlusion or low-light conditions.

Although these methods address some vision challenges, they require dedicated hardware and complex calibration. Your system remains cost-efficient by relying solely on standard IP cameras, making it practical for immediate urban deployment, with future scope for optional BLE integration.

F. Identified Research Gaps

After reviewing the recent literature (2022–2025), several key research gaps emerge:

Observed Gap	How This Project Overcomes It
Lack of outdoor domain adaptation for crowd detection models	Fine-tuned YOLOv8 model trained with bus stop imagery and ROI masking for outdoor conditions.
Separation between detection and prediction modules	Integrated pipeline combining YOLOv8 detection + Prophet forecasting for end-to-end intelligence.
Absence of user-facing interfaces	Introduces multilingual LLM chatbot and live dashboard for passengers and operators.
High deployment cost in prior works	Utilizes open-source frameworks and low-cost edge hardware for scalable implementation.
Limited real-time validation	Designed for continuous operation with low latency and real-time updates via Flask API.

G. Summary of Literature Insights

The reviewed literature demonstrates significant progress in vision-based crowd counting and transport analytics but highlights a lack of holistic integration between these technologies. This project contributes a unique interdisciplinary approach combining computer vision, predictive modeling, and AI-driven communication within a single framework. It transforms traditional bus stops into intelligent, adaptive, and interactive smart nodes — representing a tangible step toward data-driven public transport under the Smart City initiative.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 5, Issue 3, October 2025
Impact Factor: 7.67

III. METHODOLOGY

A. System Overview

The pipeline has three logical layers:

1. Sensing & Inference (Edge/Cloud)

• IP Camera → RTSP/HTTP stream → Frame capture (OpenCV).

• Inference engine runs YOLOv8 (PyTorch/Ultralytics) to detect person class; bounding boxes + confidence scores returned per frame.

• Lightweight tracker (e.g., Deep SORT or simple centroid tracking) can be used to reduce double- counts across frames.

2. Server & Prediction Layer

• Flask REST API receives per-frame counts, aggregates buffered counts per time window, and stores them in MongoDB/PostgreSQL.

• Prophet (Facebook/Meta Prophet) or comparable time-series model is trained on historical GPS arrival traces and observed dwell/crowd patterns to predict next bus ETA for each stop.

3. Presentation & Interaction

• React.js + WebSocket dashboard for live visualization (map, stop list, heatmaps, alerts).

• Multilingual LLM chatbot (Llama 3 or fine-tuned GPT) exposed via Node.js proxy to handle NLU and compose API calls for real-time answers.

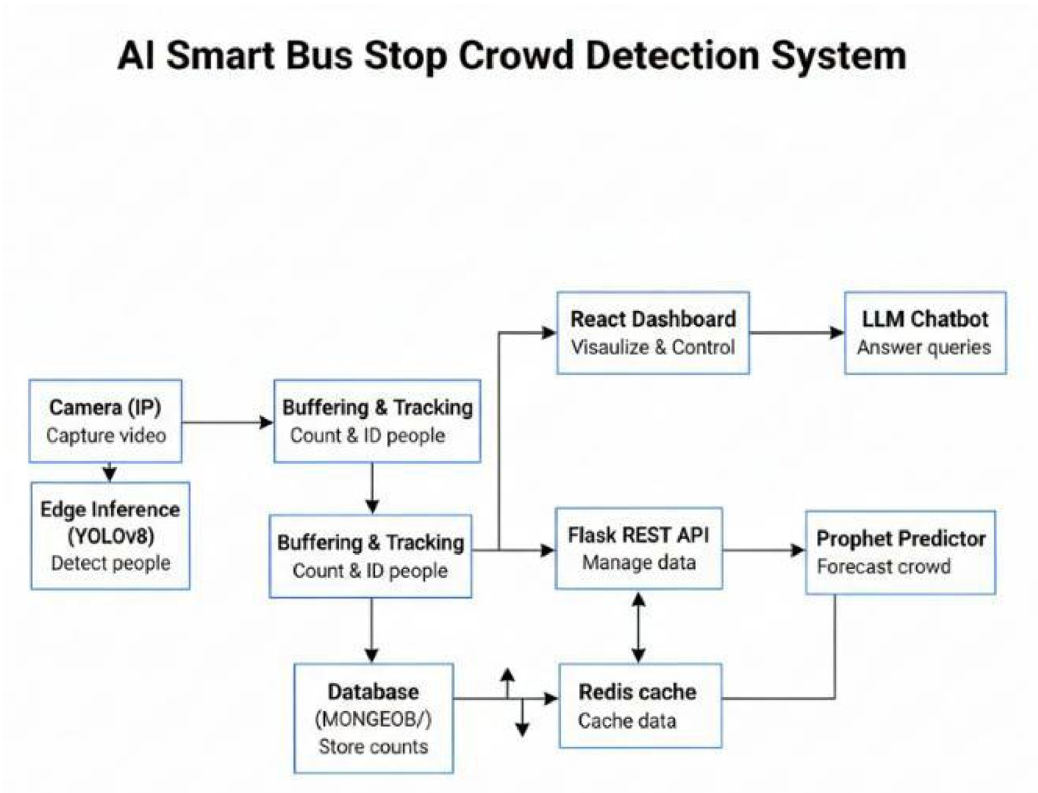

Figure A — System Architecture Diagram

DOI: 10.48175/IJARSCT-29377

ISSN 2581-9429 IJARSCT

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 5, Issue 3, October 2025
Impact Factor: 7.67

B. Data Flow

• Frame captured → preprocessed (resize, normalize) → YOLO detection → person_count.
• Buffer: short rolling window (e.g., last 10 samples) → median smoothing → persisted count with timestamp.
• Prediction: recent counts + historical schedule/GPS features → ETA output.
• API serves JSON: {stop_id, timestamp, crowd_level, count, next_bus_eta}.

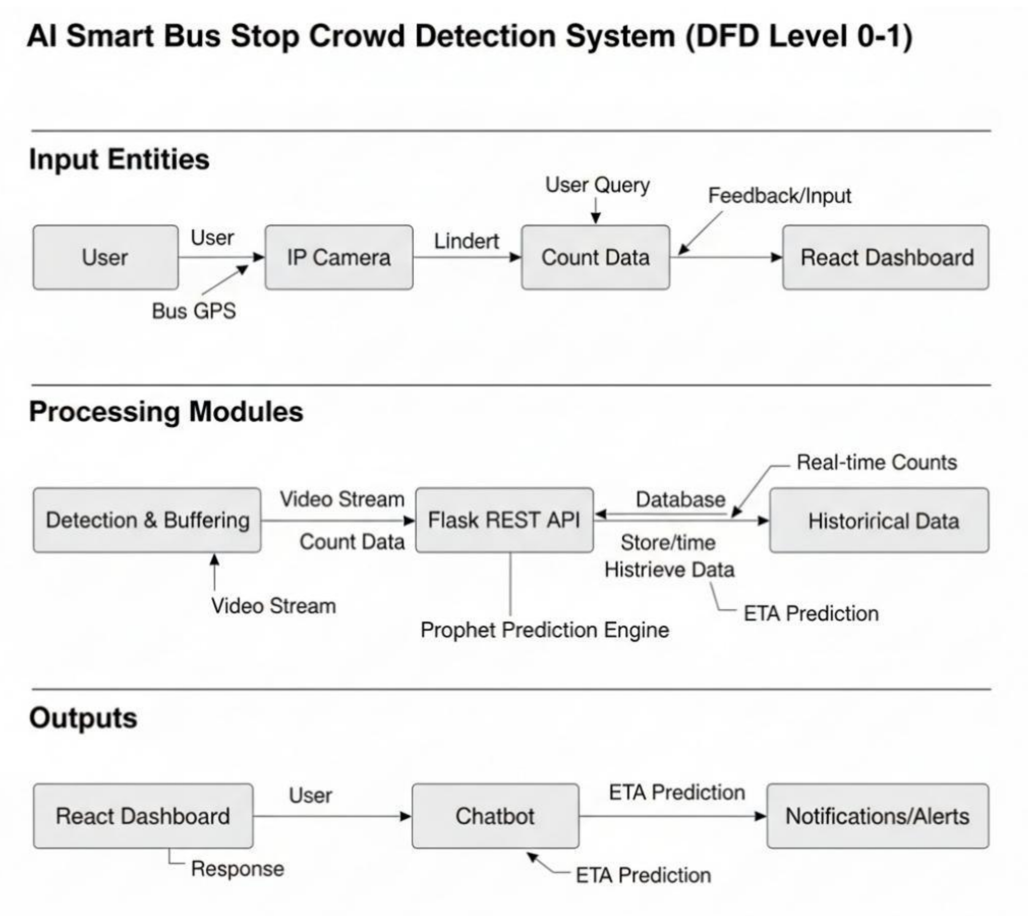

Fig – Data Flow Diagram

IV. MODULE DESCRIPTION

A. Camera & Preprocessing Module

• Responsibilities: frame acquisition, lens distortion correction, ROI masking (ignore sidewalks beyond stop).
• Tools: OpenCV, FFmpeg for stream handling.

B. Detection Module

• Responsibilities: run YOLOv8 inference, filter detections by confidence and ROI, output per- frame count.
• Model details: fine-tuned YOLOv8 (transfer learning from COCO; additional bus-stop dataset with augmentation for low light and rain).

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 5, Issue 3, October 2025
Impact Factor: 7.67

C. Buffering & Alert Module

• Responsibilities: average/median smoothing across window; threshold checks to raise overcrowding alerts; CSV / DB logging.

• Alerting: HTTP webhook / SMS / push notifications.

D. Prediction Module

• Responsibilities: ETA prediction using Prophet; fuse live GPS pings and historic dwell patterns; return probabilistic ETA intervals.

E. API & Data Management

• Responsibilities: expose REST endpoints (/live, /history, /predict, /auth), caching with Redis, secure storage.

F. Dashboard & Analytics

• Responsibilities: live map, stop-level cards, historical charts, exportable reports (CSV/PDF).

G. Chatbot Module

• Responsibilities: natural language understanding, intent extraction, query to API, multilingual responses, optional voice output.

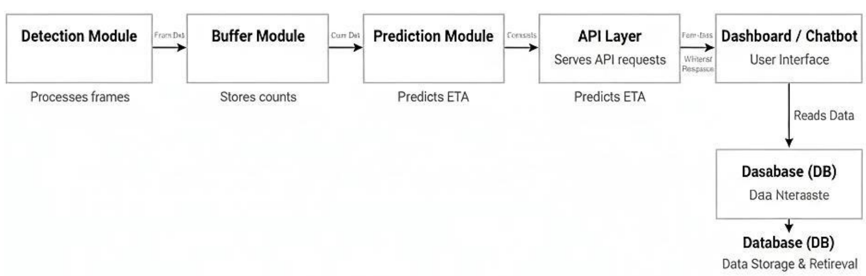

FIGURE - MODULE INTERACTION FLOW

V. ALGORITHMS & MODELS USED

A. Object Detection: YOLOv8

• Why YOLOv8: state-of-the-art speed/accuracy for real-time detection; well-supported tooling (Ultralytics).

• Training strategy: start from COCO weights → fine-tune on custom bus-stop dataset (annotations for person), include augmentation (brightness, blur, rain overlays) to improve robustness.

B. Tracking / De-duplication

• Lightweight centroid tracker or Deep SORT prevents counting same person multiple times across frames; uses IoU/centroid distance for association.

C. ETA Forecasting: Prophet

• Inputs: historical arrival timestamps, live GPS timestamps from buses, recent dwell times and crowd counts.

• Output: predicted ETA for next bus with uncertainty interval. Prophet is robust to seasonalities and missing data and is quick to retrain on new historical logs.

DOI: 10.48175/IJARSCT-29377

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 5, Issue 3, October 2025
Impact Factor: 7.67

D. Chatbot: LLM (fine-tuned)

• Uses a fine-tuned Llama3/GPT model (or cloud LLM) to interpret natural language, map intents (e.g., "Is stop 5 crowded?" → fetch /live?stop_id=5), and generate concise replies. Multilingual translation layers are added to accept multiple languages.

E. Thresholding & Alerts

1. Crowd levels classified as: Low (0–10), Medium (11–30), High (31+) — thresholds configurable per stop. Alerts issued when buffered average > threshold for a sustained duration.

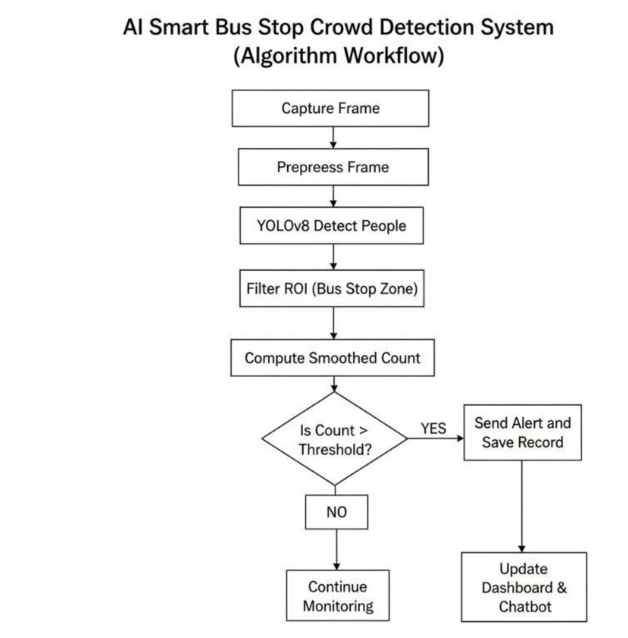

FIG- FLOW CHART

VI. HARDWARE & SOFTWARE REQUIREMENTS

A. Software

• OS: Linux (Ubuntu 20.04+) recommended for server; Windows/Mac for dev.
• Backend: Python 3.9+, Flask, PyTorch (Ultralytics), Prophet (fbprophet / prophet), OpenCV.
• Frontend: React.js, TypeScript, WebSocket support.
• DB/Cache: MongoDB/PostgreSQL; Redis for caching.
• Containerization: Docker, docker-compose.
• Optional: Node.js for chatbot proxy and websocket server.

B. Hardware (per small pilot stop)

• Edge inference: NVIDIA Jetson Nano / Xavier (for low power) or GPU server with NVIDIA RTX 3060+ for cloud inference.
• Recommended server: Intel Xeon / AMD Ryzen9, 16–32 GB RAM, 500 GB SSD, GPU (RTX 3060+).
• Network: stable broadband (10–100 Mbps) per site; secure VPN preferred.

VII. ADVANTAGES & LIMITATIONS

A. Advantages

• Real-time, stop-level visibility improves passenger decision making and operator response.
• Cost-effective vs. dedicated sensors — uses affordable IP cameras and software scaling.
• Accessible via multilingual chatbot and simple dashboard.
• Modular & scalable (Docker + cloud services).

B. Limitations

• Weather sensitivity: heavy rain/fog reduces visual accuracy (mitigated by augmentation and multimodal fusion).
• Privacy concerns: camera usage must follow local laws and apply privacy-preserving measures (no face storage, anonymization, encrypted channels).
• Connectivity dependence: real-time features require reliable network connectivity; local caching partly mitigates this.

VIII. APPLICATIONS

• Real-time commuter information at bus stops (crowd levels + ETA).
• Transport operator dashboards for dynamic fleet reallocation.
• Urban planning — historical analytics for peak times and route adjustments.
• Special events: crowd management and safety monitoring.

IX. FUTURE SCOPE

• Edge-optimized model (quantized YOLOv8 or YoLoV-lite) for cheaper inference.
• Multimodal fusion with BLE/Wi-Fi counts and privacy-preserving sensors to improve accuracy in adverse weather. (Yuki Matsuda Portfolio)
• Reinforcement-learning scheduling to dynamically adjust bus frequencies based on predicted demand.
• Privacy-preserving analytics: on-device aggregation that shares only counts, not raw frames.
• Integration with V2I for bus-to-stop communications and automated stop management.

X. TESTING & EVALUATION

A. Testing Strategy

• Unit tests for API endpoints; integration tests between detection and API; field tests at selected stops across different times and weather conditions.
• Performance metrics: detection precision/recall, MAE in counts, prediction MAE for ETA, system latency (end-to-end), user satisfaction (chatbot).
• Pilot results (project): the original pilot reported detection accuracy ≈ 95% under test conditions and strong user satisfaction for chatbot responses (project team data).

B. Sample Test Case (functional)

• Case: Camera frames with 0–60 people; evaluate buffered count vs. manual ground truth over 30 min.
Metric: MAE $\leqslant$ 3 for counts under typical daylight; detection precision > 90% preferred.

XI. CONCLUSION

The AI Smart Bus Stop Crowd Detection System demonstrates a practical route from modern computer vision research to operational public transport tools: reliable per-stop crowd estimation, ETA forecasting, and user-friendly interfaces. By combining a fine-tuned YOLOv8 detector, smoothing and tracking, Prophet forecasting and an LLM-backed chatbot, the system equips passengers and operators with actionable, real-time insights while keeping deployment cost and complexity manageable. With careful attention to privacy, weather robustness, and edge/cloud tradeoffs, the system is well positioned as a building block for smart-city mobility platforms.

XII. AKNOWLEDGEMENT

We sincerely thank Mrs. V. M. Khanapure for her mentorship and continuous guidance throughout this project. Appreciation is extended to the Department of Information Technology, Government Polytechnic, Pune, and to Dr. D. N. Rewadkar and Dr. R. K. Patil for their institutional support. We also acknowledge the assistance of laboratory staff and student volunteers for their help during experimentation and validation.

REFERENCES

[1]. M. Ş. Gündüz, "A new YOLO-based method for real-time crowd detection," Peer-Reviewed Journal / Sensors, 2023. (PMC)

[2]. BusStopCV — Makeability Lab, University of Washington. "BusStopCV: crowd + AI web prototype for bus stop features," project demo and evaluation. (makeabilitylab.cs.washington.edu)

[3]. N. Alhawsawi et al., "Enhanced YOLOv8-Based Model with Context Enrichment for Crowd Counting," Remote Sensing (MDPI), 2024. (MDPI)

[4]. T. Ikenaga et al., "Using BLE Signals to Estimate Objective and Subjective Crowdedness," IEEE Access / Journal (2025). (Yuki Matsuda Portfolio)

[5]. "Automatic Passenger Counting System on Public Buses Using CNN YOLOv8," research preprint /conference paper (2024). (ResearchGate).

[6]. Malik, M., Sharma, M., & Chopra, N. (2022). Crowd Counting and Detection. International Journal of Recent Advances in Multidisciplinary Topics, 3(5), 49-51. journals.ijramt.com

[7]. Nagaraj, N., Gururaj, H.-L., Harish Swathi, B., & Hu, Y.-C. (2022). Passenger flow prediction in bus transportation system using deep learning. Multimedia Tools and Applications, 2022. OUCI

[8]. Deepthi, V. S., Chavan, N. V., Shanbhag, S., & Dandappala, S. D. (2022). Crowd Density Estimation and Location Prediction in Public Transport System. IJERT, 11(07), July 2022. IJERT

[9]. Prastyo, A. D., Andzani Minhalina, S., Agung, S., Nirwana Bintang, D., Yordi Septian, M., Giri, E. P. (2024). Automatic Passenger Counting System on Public Buses Using CNN YOLOv8 Model for Passenger Capacity Optimization. International Journal of Information Engineering and Science. international.arteii.or.id