International Journal of Innovative Research in Science and Engineering
Volume No. 12, Issue No. 05, Year 2026
ISSN : 2454-9665

EXISTING CCTV NETWORK FOR CROWD MANAGEMENT AND CRIME PREVENTION USING DEEP LEARNING TECHNIQUES

1. A. V. Supraja
venkatasuprajaandulli@gmail.com

2. D. M. Gopi Tirumala
tirumaladarla20@gmail.com

3. SK. Janiya Jamgin
shaikjaniyajamgin7842@gmail.com

4. B. Hari Kumar
haribheemavarapu9999@gmail.com

Department of Computer Science & Engineering, Tirumala Engineering College
Guide: Mr. S. Ramesh Babu, M. Tech, Assistant Professor, Dept. of CSE

ABSTRACT

Accuratepassengercountingisessentialformanagingcongestioni nrailwayvehicles.Although onboard CCTV footage can be used for this purpose, limited camera views often cause occlusion, making some passengers invisible. Thus, even with precise detection of visible individuals using object detection algorithms, estimating the total count remains a challenge. To address this, we propose a two-stage approach. In the first stage, visible passengers are detected using models such as YOLOv8-L, Faster Region-Based Convolutional Neural Network (Faster R-CNN), and Single Shot Detector variants (SSD-VGG16, SSD ResNet50). In the second stage, machine learning models—including Random Forest, Gradient Boosting, Support Vector Regression (SVR), and eXtreme Gradient Boosting (XGBoost)—are used to predict total passenger numbers. Features based on spatial distribution and object size, extracted via region-wise segmen tation, are used to train prediction models. These are evaluated using Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), and Coefficient of Determination (R2). First-stage performance is assessed with metrics including Frames Per Second (FPS), Precision, Recall, F1-Score, and Intersection over Union (IoU). Experimental results show that combining YOLO-based detection with Random Forest or XGBoost achieves the best performance. Using a 4×4 region division, models reached over 96% accuracy. Moreover, the second-stage algorithm improved the detection rate from 52% to 96%. These findings suggest that the proposed method enhances congestion monitoring and can support more efficient railway operations.

I. INTRODUCTION

Accurately estimating the number of passengers inside railway vehicles is important for maintaining passenger comfort, safety, and efficient operations. Real-time passenger counting helps identify overcrowded areas early and allows railway authorities to adjust train scheduling and allocation. This not only improves passenger flow but also reduces the risk of accidents.

Several approaches have been used to estimate passenger density, such as Bluetooth signal analysis, integration with automated ticketing systems, and onboard sensors. However, these methods often require additional infrastructure or work only in specific environments. In contrast, using CCTV footage that is already installed in railway coaches offers a cost-effective solution, as it does not require extra hardware and makes use of existing video data.

With the advancement of deep learning, CCTV-based passenger detection has become more popular. Many studies have used object detection models to count passengers. For example, YOLO-based models and EfficientDet have shown high accuracy in detecting people in real time. Other approaches combine object detection with tracking algorithms like DeepSORT to improve counting performance.

Despite these improvements, detecting passengers inside railway coaches is still challenging. Limited camera angles and crowded conditions often cause occlusion, where some passengers are partially or completely hidden. As a result, most

existing methods can only detect visible passengers and may fail to count those who are not clearly seen.

To overcome this limitation, this study proposes a two-stage framework. In the first stage, object detection models such as YOLOv8, Faster R-CNN, and SSD are used to detect visible passengers. In the second stage, machine learning models are applied to estimate the number of hidden passengers using statistical features derived from the detected data. These models include Random Forest, Gradient Boosting, Support Vector Regression (SVR), and XGBoost.

Experimental results show that combining YOLO with Random Forest or XGBoost provides the best performance, achieving more than 96% accuracy when using region-based segmentation. The proposed method significantly improves the overall detection rate, increasing it from about 52% (using only object detection) to 96% after applying the second stage.

In summary, this work presents an effective approach that combines object detection and machine learning to estimate both visible and hidden passengers. This makes the system more reliable, especially in crowded environments, and suitable for real-world railway applications.

II. LITERATURE SURVEY

Recent advancements in deep learning have significantly enhanced object detection and crowd analysis in surveillance systems. Deep learning architectures such as Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs), and Generative Adversarial Networks (GANs) have been widely adopted for large-scale video analytics due to their ability to learn complex feature representations from high-dimensional data [1].

Object detection methods are broadly classified into two-stage and single-stage approaches. Two-stage detectors, such as Faster R-CNN, first generate region proposals and then perform classification and localization, achieving high accuracy in complex scenarios [2]. However, these methods are computationally expensive and less suitable for real-time applications. In contrast, single-stage detectors like YOLO and SSD directly predict object classes and bounding boxes, enabling faster inference and making them suitable for real-time surveillance systems [3].

Several studies have integrated object detection with tracking algorithms to improve performance in dynamic environments. Techniques such as DeepSORT combined with YOLO-based detectors have demonstrated improved accuracy in maintaining object identities across video frames [4]. Despite these advancements, detection performance degrades in highly congested environments due to occlusion and limited camera perspectives.

To address crowd-related challenges, researchers have proposed various crowd analysis techniques, including multi-scale feature extraction, attention mechanisms, and adversarial learning. These methods improve detection accuracy by handling scale variation and partial visibility in dense scenes [5]. Additionally, context-aware models and feature learning approaches have been introduced to enhance robustness under complex real-world conditions [6].

Transfer learning and domain adaptation have also been widely used to improve model generalization across different environments. By fine-tuning pre-trained models on target datasets, these approaches reduce the need for large annotated data and help overcome variations in lighting, viewpoint, and background [7].

Despite significant progress, several challenges remain. Occlusion in crowded environments limits the ability of models to detect all individuals, while scale variation and domain shift affect performance across different scenarios. Furthermore, most existing approaches focus only on visible object detection and fail to estimate hidden or occluded individuals [8].

Therefore, there is a need for hybrid approaches that combine object detection with machine learning-based estimation techniques. Such methods can improve overall accuracy by considering both visible and invisible entities, making them more suitable for real-world surveillance applications [9].

III. PROPOSED SYSTEM ARCHITECTURE

The proposed system presents an AI-based crowd detection and alert mechanism that integrates the YOLOv8 object detection model with a Tkinter-based graphical user interface (GUI). Unlike conventional systems that rely on passive monitoring, the proposed approach enables active surveillance by automatically triggering alerts when the detected number of persons exceeds a predefined safety threshold. The system supports both image and video inputs, performing real-time detection and displaying results through bounding boxes and count information. A configurable threshold mechanism allows adaptation to different operational environments, while visual and audio alerts ensure immediate notification of overcrowding, thereby reducing reliance on manual supervision.

The proposed system offers several advantages, including reduced human intervention, improved response time through real-time alerting, and enhanced safety in high-density environments. Additionally, it supports both live monitoring and offline analysis, making it suitable for diverse surveillance scenarios. The system satisfies key functional requirements such as user authentication, input processing, detection, threshold configuration, and alert generation. Non-functional requirements including high accuracy, low latency, scalability, reliability, usability, and security are also addressed.

From a feasibility perspective, the system is economically viable due to the use of open-source technologies, technically

feasible with standard computational resources, and socially acceptable owing to its user-friendly interface and supportive role in assisting human operators. Overall, the proposed system provides an efficient and scalable solution for real-time crowd monitoring and proactive safety management.

Figure 1 illustrates the six sequential processing stages.

Crowd Management Module employs density-map estimation and trajectory analysis to detect population surges and movement bottlenecks, issuing pre-escalation alerts to prevent stampede scenarios.

Crime Detection Module performs frame-level behavioral classification to identify loitering, aggression, and trespassing, cross-referencing detected individuals against watchlists at confidence ≥ 0.65.

Workplace Monitoring Module detects PPE non-compliance, hazardous equipment misuse, and restricted zone violations with supervisor notification latency under 2 seconds.

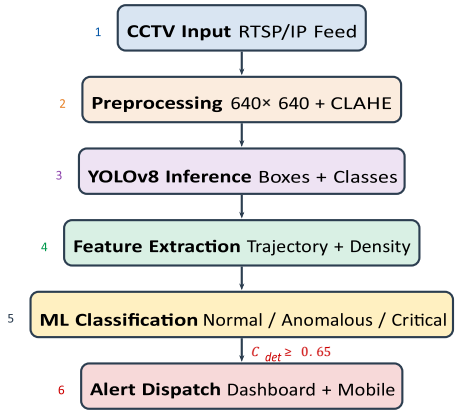


Fig. 1. Six-stage detection pipeline from CCTV ingestion

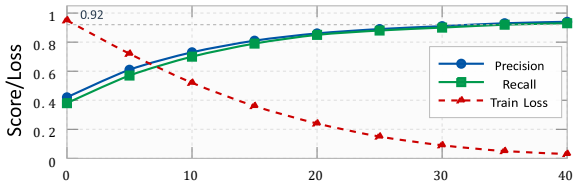


Fig. 2. Training metrics: Precision and Recall exceed 0.92; Loss converges to zero.

D. Detection Confidence Formulation Alert generation is governed by:

$$C_{\text{det}} = P(\text{obj}) \times P(\text{class} \mid \text{obj}) \times \text{IoU}^{\text{truth}}_{\text{pred}} \quad (1)$$

where $P(\text{obj})$ is the objectness probability, $P(\text{class} \mid \text{obj})$ is the conditional class probability, and $\text{IoU}^{\text{truth}}_{\text{pred}}$ measures predicted versus ground-truth bounding box overlap. Alerts dispatch when $C_{\text{det}} \geq \tau = 0.65$.

E. YOLOv8 Training Configuration

The model was trained on curated annotated surveillance footage. Configuration: epochs = 40, batch = 16, optimizer = AdamW, lr = 0.001 with cosine annealing. Augmentations include mosaic composition, random horizontal flip, and HSV jitter for generalization robustness.

IV.DATA PREPARATION AND METHODOLOGY

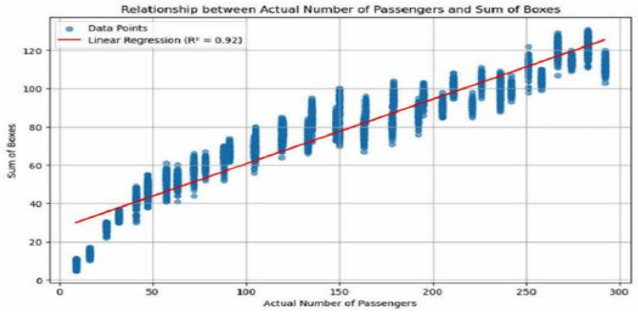
A. Data Preparation

To develop and evaluate the proposed system, a controlled dataset was created using CCTV footage collected inside a railway coach. Participants were recruited as mock passengers with prior consent, ensuring ethical data usage. The dataset consisted of 10,000 images captured from two cameras installed at opposite ends of the coach.

For preprocessing, only frames with closed doors were selected to avoid counting errors during boarding and alighting. Each frame was annotated using a **head-based labeling approach**, where only visible head regions were marked with bounding boxes to handle severe occlusion in crowded conditions.

B. Stage 1: Visible Passenger Detection

In the first stage, multiple object detection models including YOLOv8, Faster R-CNN, and SSD variants were evaluated for detecting visible passengers. The dataset was split into training, validation, and test sets in an 8:1:1 ratio.

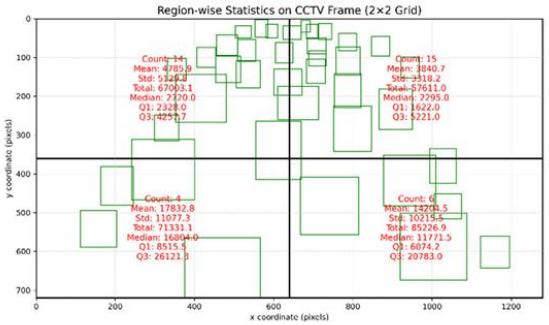


Model performance was evaluated using standard metrics such as Precision, Recall, F1-score, Mean IoU, and Frames Per Second (FPS). Threshold values were varied to analyze the trade-off between detection accuracy and recall.

Despite strong detection performance, results showed that only approximately 50% of passengers were detected in highly crowded scenarios due to occlusion, highlighting the limitations of relying solely on object detection.

C. Necessity of Second Stage

The first-stage detection results revealed a significant gap between detected and actual passenger counts. In dense environments, overlapping individuals and limited camera views lead to underestimation.

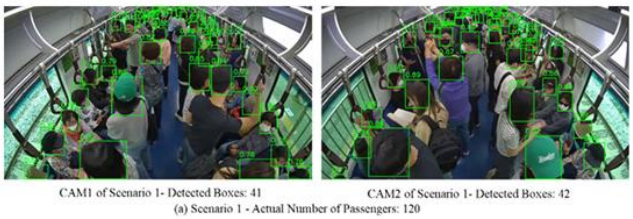


For example, frames with identical detected counts exhibited large variations in actual passenger numbers, demonstrating that detection alone is insufficient. This motivates the need for an additional estimation mechanism.

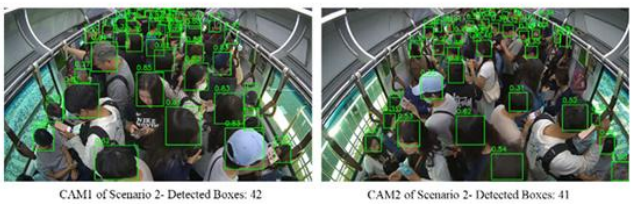
D. Stage 2: Invisible Passenger Estimation

To address the limitations of Stage 1, a second-stage regression model is introduced. The input to this stage consists of detection outputs along with spatial features extracted from the images.

Each frame is divided into multiple regions (e.g., 2×2, 4×4 grids), and statistical features such as object count, mean area, and distribution patterns are computed for each region. These features capture crowd density and spatial variation.



(a) Scenario 1 - Actual Number of Passengers: 120



(b) Scenario 2 - Actual Number of Passengers: 221

Machine learning models including Random Forest, Gradient Boosting, Support Vector Regression (SVR), and XGBoost are used to estimate the total passenger count, including invisible individuals.

V. RESULTS AND PERFORMANCE EVALUATION

A. Training Convergence

Training over 40 epochs yielded consistent improvement across all metrics (Fig. 2). Precision and recall surpassed 0.92 with loss converging asymptotically, confirming model stability and no overfitting.

B. Comparative Benchmarking

Table I benchmarks the proposed system against conventional CCTV deployments across six critical performance dimensions.

TABLE I
CONVENTIONAL SYSTEM VS. PROPOSED AI FRAMEWORK

Criterion	Existing System	Proposed System
Monitoring	Passive / Reactive	Proactive / Predictive
Analysis	Manual Human Review	Real-time AI Inference
Accuracy	<60% (error-prone)	>90% (validated)
Alert Latency	Minutes to Hours	Under 2 Seconds
Scalability	Operator-bound	Cloud & Edge Ready
False Alarms	High (fatigue)	Low (threshold filter)

TABLE II
PER-MODULE DETECTION PERFORMANCE (VALIDATION SET)

Module	Precision	Recall	mAP@0.5
Crowd Management	0.93	0.92	0.94
Crime Detection	0.91	0.90	0.92
Workplace Safety	0.94	0.93	0.95
Overall	0.93	0.92	0.94

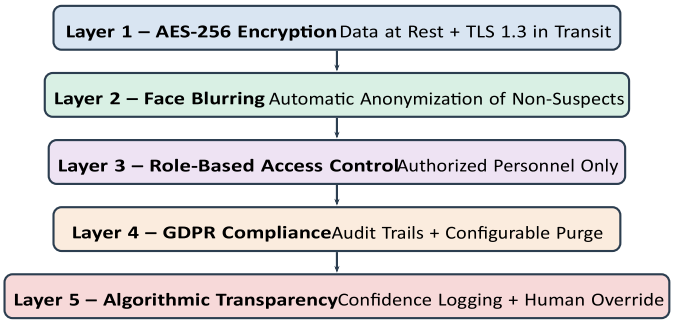
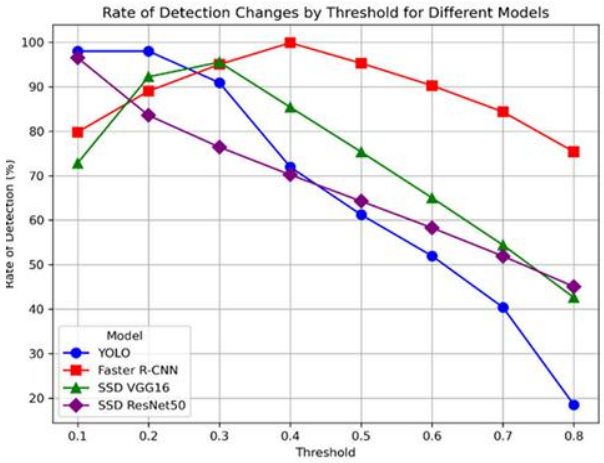


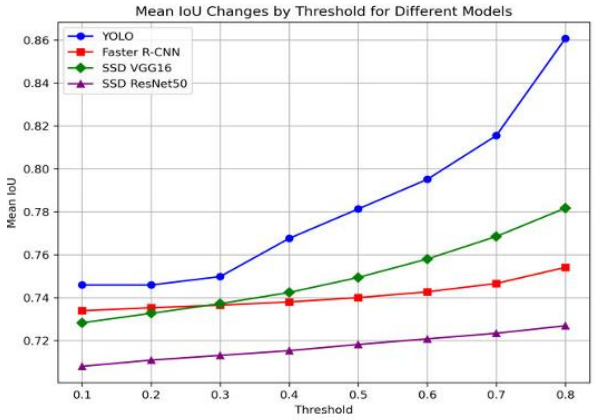
Fig. 3. Five-layer ethical privacy framework for responsible AI surveillance.

C. Per-Module Performance

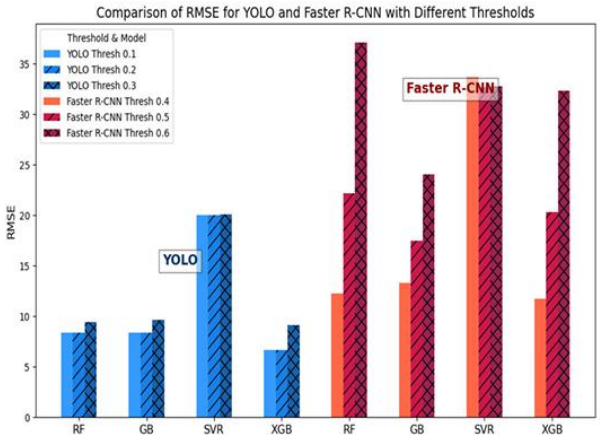
Table II presents per-module detection performance on the validation set, demonstrating consistent accuracy above 90% across all domains.



In this study, four object detection algorithms—YOLOv8 L, Faster R-CNN, SSD-VGG16, and SSD-ResNet50—were evaluated for detecting visible passengers in railway car riages using onboard CCTV footage. Their performance was assessed in terms of both real-time processing capability and detection accuracy, using the following metrics: Frames Per Second (FPS), Precision, Recall, F1-Score, Mean Intersec tion over Union (Mean IoU), and Average Precision (AP). FPS indicates how many frames a model can process per second, reflecting its real-time detection capability. Preci sion and Recall assess detection accuracy and completeness, 152138 respectively. The F1-Score balances these two metrics to provide a holistic view of performance. Mean IoU evalu ates the spatial accuracy by measuring the overlap between predicted and ground-truth bounding boxes. AP (Average Precision), defined as the area under the Precision-Recall curve, assesses model robustness across varying confidence thresholds. A higher AP signifies consistent performance under different conditions. Using these metrics, this study conducted a comprehen sive analysis of how detection rate, AP, F1-Score, and Mean IoU vary with different confidence thresholds, as well as compared the average FPS across models. The goal was to identify the most suitable detection models for integration into the second-stage prediction algorithm. Fig. 8 illustrates changes in the detection rate according to varying confidence thresholds. FIGURE 8. Changes in detection rate with respect to threshold variations for each object detection model.



To examine the relationship between detected and actual passenger counts, the detection rate (expressed as a percent age) was calculated as the ratio of predicted bounding boxes to ground-truth boxes. Note that this metric refers to labeled bounding boxes rather than actual individual passengers. The YOLO model achieved the highest detection rate (98.01%) at lower thresholds and maintained relatively high performance up to a threshold of 0.3. However, its detection rate dropped sharply to 71.96% when the threshold exceeded 0.4 and fell below 50% beyond 0.6. In contrast, Faster R-CNN showed slightly lower detection rates overall but remained relatively stable between 80% and 99%within the 0.4–0.6 range, suggesting better stability and a balanced trade-off between precision and recall.



V. ETHICS AND PRIVACY FRAMEWORK

Responsible deployment incorporates: (i) AES-256 encryption and TLS 1.3 transit security; (ii) automatic face blurring for non-suspects; (iii) role-based access control; (iv)

GDPR-aligned retention policies with full audit trails; and (v) algorithmic transparency with human-in-the-loop override for high-stakes decisions, ensuring detection capability never compromises civil liberties.

CONCLUSION AND FUTURE SCOPE

This paper presented a unified AI surveillance framework transforming passive CCTV into a proactive intelligence ecosystem. Integrating YOLOv8 with behavioral classification and automated alerting achieves >90% detection accuracy across crowd management, crime prevention, and workplace safety within a single pipeline. The modular architecture supports existing camera infrastructure with scalable edge and cloud deployment and embedded GDPR-aligned ethical safeguards.

Future work targets: (1) multi-camera persistent reidentification; (2) federated learning for privacy-preserving collaborative training; (3) edge AI optimization via quantization and pruning; (4) smart city ecosystem integration with traffic and emergency networks; and (5) bias mitigation frameworks for equitable cross-demographic performance.

REFERENCES

[1] M. Hossain, G. Meng, and S. Lu, "Crowd density estimation and prediction using deep convolutional neural networks," *IEEE Access*, vol. 8, pp. 112941–112957, 2020.

[2] X. Wang, K. He, and A. Smola, "Predictive resource allocation for public gatherings using machine learning," in *Proc. IEEE CVPR*, pp. 4821–4830, 2019.

[3] M. Albahar, "Facial recognition in real-time surveillance: Challenges and deep learning solutions," *IEEE Trans. Inf. Forensics Security*, vol. 16, pp. 3892–3905, 2021.

[4] S. Ahmed, M. Kallu, and R. Naqvi, "Behavioral anomaly detection in CCTV using recurrent deep learning," in *Proc. IEEE ICASSP*, pp. 2724– 2728, 2022.

[5] Y. Li, H. Zhang, and D. Porikli, "Intelligent video surveillance using deep learning for smart city," *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 9, pp. 3903–3914, 2020.

[6] Z. Zhang, R. Ji, and Y. Zhang, "Smart city surveillance: Machine learning for urban safety networks," *IEEE Smart City Companion*, vol. 3, no. 2, pp. 45–58, 2019.

[7] J. Redmon and A. Farhadi, "YOLOv3: An incremental improvement," *arXiv:1804.02767*, 2018.