IJARETY

International Journal of Advanced Research in Education and TechnologY (IJARETY)

Volume 12, Issue 4, July-August 2025

Impact Factor: 8.152

www.ijarety.in editor.ijarety@gmail.com

International Journal of Advanced Research in Education and TechnologY(IJARETY)

| ISSN: 2394-2975 | www.ijarety.in| | Impact Factor: 8.152| A Bi-Monthly, Double-Blind Peer Reviewed & Refereed Journal |

|| Volume 12, Issue 4, July - August 2025 ||

DOI:10.15680/IJARETY.2025.1204075

Real-Time Crowd Analytics using Deep Object Detection Models

Ayan Ahamed, Balaji, Bharath Kumar Y

Department of Computer Application, CMR Institute of Technology, Bengaluru, India

ABSTRACT: Efficient crowd analytics is essential for public safety, event monitoring, and urban planning, where timely detection and density estimation can mitigate risks and improve resource allocation. Traditional methods often fail under real-world conditions involving occlusion, varying illumination, and perspective distortion. Recent advances in deep learning-based object detection provide scalable, real-time solutions for such challenges. This study presents a comparative evaluation of three state-of-the-art architectures—Faster R-CNN, YOLOv4, and Single Shot Multi-box Detector (SSD)—for real-time crowd analytics. Experiments were conducted on the ShanghaiTech Crowd Counting Dataset and a custom in-house dataset featuring diverse densities, lighting conditions, and motion patterns. Models were fine-tuned from COCO-pre trained weights, with pre-processing, data augmentation, and anchor box optimization tailored for dense scenes. Performance was assessed using mean average precision (mAP), precision, recall, F1-score, mean absolute error (MAE), mean squared error (MSE), and inference speed (FPS).Results indicate that YOLOv4 achieved the best overall performance, exceeding 90% mAP with the lowest MAE/MSE and the highest FPS (>40), making it ideal for high-density, real-time applications. SSD provided a strong balance between accuracy and speed, while Faster R-CNN offered high precision but lower recall and speed, making it better suited for offline analytics. These findings underscore the importance of model selection based on operational constraints and highlight avenues for improvement, including lightweight architectures, attention mechanisms, and spatial-temporal modelling for robust deployment in smart city and public safety systems.

KEYWORDS: Crowd Analytics, Deep Learning, YOLOv4, SSD, Faster R-CNN, Real-Time Detection, Urban Safety.

I. INTRODUCTION

The rapid urbanization of the 21st century has led to an unprecedented concentration of people in metropolitan areas, resulting in frequent high-density gatherings in venues such as transport hubs, sports arenas, concert venues, and religious events. In these scenarios, real-time crowd analytics has become an essential tool for ensuring public safety, optimizing event management, and enabling efficient urban planning. The ability to accurately estimate crowd size, monitor movement patterns, and detect anomalies in real time is not only critical for disaster prevention but also for improving the overall quality of public infrastructure and services [1], [2].

Traditional crowd counting and monitoring approaches—such as manual counting, visual inspection, and classical computer vision methods based on handcrafted features—struggle to cope with the challenges of real-world environments. Factors such as occlusion, perspective distortion, illumination variation, and non-uniform crowd distribution significantly degrade their accuracy and reliability [3], [4]. Moreover, these conventional methods lack scalability and adaptability to diverse crowd scenes, making them unsuitable for large-scale, dynamic deployments in smart city ecosystems.

The evolution of deep learning and computer vision has significantly transformed crowd analytics. State-of-the-art object detection architectures, including Faster R-CNN [5], You Only Look Once (YOLO) [6], and Single Shot Multi-box Detector (SSD) [7], have demonstrated remarkable success in detecting and localizing individuals in both still images and live video feeds. By leveraging convolutional neural networks (CNNs) for hierarchical feature extraction, these models outperform traditional density regression or background subtraction methods by providing precise bounding boxes and confidence scores for each detected individual. This feature-rich output facilitates downstream tasks such as behavior recognition, anomaly detection, and spatiotemporal pattern analysis [8], [9].

Real-time performance is particularly important for operational scenarios such as crowd control during public events, emergency evacuation planning, and traffic management. Models like YOLO, with its single-stage detection pipeline, excel in achieving high-speed inference without significantly compromising accuracy, making them ideal for latency-sensitive applications [6], [10]. SSD strikes a balance between computational cost and detection performance, while

International Journal of Advanced Research in Education and TechnologY(IJARETY)

| ISSN: 2394-2975 | www.ijarety.in| | Impact Factor: 8.152| A Bi-Monthly, Double-Blind Peer Reviewed & Refereed Journal |

|| Volume 12, Issue 4, July - August 2025 ||

DOI:10.15680/IJARETY.2025.1204075

Faster R-CNN, although computationally heavier, delivers high precision in controlled or moderately dense settings [5], [11].

Despite these advances, critical challenges persist. Extreme crowd densities cause severe occlusions, overlapping human silhouettes, and scale variations, which negatively affect detection accuracy [12]. Low-light conditions, environmental noise, and camera motion further degrade performance. Additionally, the lack of large, diverse, and well-annotated datasets limits the generalizability of models across different geographic, cultural, and infrastructural contexts [13]. Researchers have explored domain adaptation, synthetic data generation, and attention mechanisms to address these challenges, yet real-world deployment at scale remains a non-trivial problem [14], [15].

In this research, we present a comparative evaluation of Faster R-CNN, YOLO, and SSD for real-time crowd analytics using both the ShanghaiTech dataset [16] and a custom in-house dataset collected from public gatherings, urban streets, and event venues. The study assesses each model's performance across key evaluation metrics such as Mean Absolute Error (MAE), Mean Squared Error (MSE), Precision, Recall, and F1-score. Our analysis combines quantitative benchmarking with qualitative visual assessments to provide a comprehensive understanding of each model's strengths, limitations, and suitability for real-world applications.

The insights from this work aim to bridge the gap between theoretical advances in deep object detection and practical crowd monitoring needs, offering actionable recommendations for deploying robust, real-time crowd analytics systems in smart cities, transportation hubs, and high-density public events.

II. LITERATURE REVIEW

Crowd counting and analytics have been an active research area for over two decades due to their applications in public safety, transportation management, event organization, and smart city development. Early approaches primarily relied on manual observation and classical image processing techniques using handcrafted features such as edge detection, texture descriptors, and motion analysis [1], [2]. While these methods were computationally inexpensive, they performed poorly under real-world challenges such as dense crowds, occlusion, illumination changes, and non-uniform crowd distribution.

1. Early Density Estimation and Regression-Based Approaches

Lempitsky and Zisserman [3] pioneered a density estimation method that learned a mapping between local features and crowd density maps. Chan et al. [4] proposed a method based on multi-column cell histograms for privacy-preserving crowd counting without explicit detection. These early methods formed the basis for non-detection-based crowd analytics but suffered from limited scalability and robustness under occlusion and perspective distortion. Later works such as Idrees et al. [5] integrated multiple visual cues, including texture, edge features, and motion patterns, to improve counting accuracy in complex scenes.

2. Traditional Machine Learning-Based Object Detection

Before the deep learning era, object detection for crowd analysis was often implemented using Haar cascades, HOG descriptors, and SVM classifiers [6]. While these models could detect individuals in low-density settings, their performance dropped significantly in dense and cluttered environments. The inability to handle significant scale variation and occlusion hindered their deployment in real-time crowd monitoring systems.

3. Deep Learning-Based Crowd Counting

The introduction of Convolutional Neural Networks (CNNs) revolutionized the field. Deep learning approaches can be broadly categorized into:

Density Map Regression Networks – Zhang et al. [7] introduced the Multi-Column Convolutional Neural Network (MCNN), which used multiple receptive fields to handle scale variations. Subsequent works like CSRNet [8] improved accuracy using dilated convolutions to preserve spatial information.

Object Detection-Based Methods – Models such as Faster R-CNN [9], YOLO [10], and SSD [11] have been applied to crowd counting tasks due to their ability to detect and localize individuals. Faster R-CNN offers high accuracy by combining a Region Proposal Network (RPN) with a detection network, while YOLO's single-stage architecture enables real-time performance. SSD balances accuracy and speed by detecting objects at multiple scales.

| ISSN: 2394-2975 | www.ijarety.in| | Impact Factor: 8.152| A Bi-Monthly, Double-Blind Peer Reviewed & Refereed Journal |

|| Volume 12, Issue 4, July - August 2025 ||

DOI:10.15680/IJARETY.2025.1204075

4. Comparative Evaluations of Detection Models

Several studies have benchmarked these models for crowd analytics. Wang et al. [12] found YOLO to outperform Faster R-CNN in high-speed crowd surveillance, though Faster R-CNN retained an advantage in precise localization under moderate crowd densities. Ma et al. [13] proposed an Adaptive SSD variant to improve performance in occlusion-heavy scenes. However, both YOLO and SSD face challenges in extremely dense crowds where individuals occupy minimal pixel space.

5. Handling Occlusion, Perspective, and Scale

Occlusion and scale variation remain key challenges. Attention-based mechanisms such as the Scale-Aware Attention Network (SAANet) [14] dynamically focus on informative features, improving detection under crowded conditions. Graph-based spatial reasoning [15] and transformer-based architectures [16] have also been proposed to model contextual relationships between individuals, enhancing robustness in dense scenes.

6. Dataset Limitations and Domain Adaptation

A major bottleneck in deep learning-based crowd counting is the lack of diverse, large-scale annotated datasets. The ShanghaiTech dataset [17] and UCF-QNRF [18] are widely used benchmarks, but models trained on these datasets often struggle to generalize to new environments. To address this, researchers have explored domain adaptation techniques [19], synthetic dataset generation [20], and data augmentation strategies to improve cross-scene performance.

7. Real-Time and Edge Deployment

Real-time crowd analytics in resource-constrained environments requires model optimization. Techniques such as quantization, model pruning, and knowledge distillation have been applied to YOLO and SSD for deployment on edge devices [21]. Lightweight architectures such as YOLOv5-Nano and MobileNet-SSD offer feasible trade-offs for on-site crowd monitoring with reduced latency.

8. Explain ability and Ethical Considerations

As crowd analytics is increasingly used in public safety and surveillance, explainable AI (XAI) approaches are gaining importance. Zhang et al. [22] proposed explainable density maps and attention heat maps to visualize model decision-making, enhancing transparency and trust. Ethical frameworks addressing privacy preservation and data security are also becoming critical for real-world adoption.

The literature shows a clear trajectory from handcrafted feature-based methods to deep learning-based architectures that excel in accuracy and robustness. While YOLO, SSD, and Faster R-CNN are the leading object detection models in real-time crowd analytics, their effectiveness depends on environmental conditions, density levels, and computational constraints. Current research trends emphasize occlusion handling, cross-domain adaptation, and lightweight deployment, pointing toward an integrated future where accuracy, speed, and ethical compliance are balanced for optimal crowd monitoring solutions.

III. METHODOLOGY

The proposed study adopts a comparative experimental framework to evaluate the performance of three state-of-the-art deep object detection models—Faster R-CNN [1], You Only Look Once (YOLO) [2], and Single Shot Multibox Detector (SSD) [3]—for real-time crowd analytics. The process begins with dataset selection and preparation, where two diverse datasets are employed. The first is the benchmark ShanghaiTech Crowd Counting Dataset [4], which contains high-resolution images with significant scale variation, occlusion, and perspective distortion. Part A comprises images from internet sources representing high-density crowds, while Part B contains urban street scenes depicting moderate densities. The second dataset is a custom in-house collection captured via high-definition CCTV and handheld cameras from urban streets, event venues, and public gatherings. This dataset introduces variation in lighting conditions (day, night, indoor, outdoor), multiple crowd densities, and dynamic crowd motion, with annotations stored in Pascal VOC XML and COCO JSON formats. Using both datasets ensures that the evaluation encompasses both controlled benchmarks and unconstrained real-world conditions.

In the data pre-processing phase, images are resized to match each model's architectural input (416×416 px for YOLO, 300×300 px for SSD, and 600×600 px for Faster R-CNN) and normalized to a pixel range of [0, 1] to maintain consistent gradient magnitudes during training. Data augmentation techniques—including horizontal flipping, random brightness and contrast adjustments, Gaussian noise injection, and random cropping while preserving aspect ratio—are

DOI:10.15680/IJARETY.2025.1204075

applied to improve model generalization. Annotation integrity is verified to ensure bounding box consistency. Preprocessing is implemented using OpenCV and Albumentations for reproducibility and efficiency.

The model architecture and adaptation stage tunes each detector for dense crowd scenarios. Faster R-CNN uses a ResNet-50 backbone with a Feature Pyramid Network (FPN), a Region Proposal Network generating ~300 proposals per image, and anchor sizes optimized for small object detection, trained in PyTorch Detectron2. YOLOv4 adopts a CSPDarknet-53 backbone with three-scale detection heads, anchor boxes from k-means clustering, and CIoU plus BCE loss, trained using the Darknet framework. SSD employs a VGG-16 backbone with multi-scale detection from several feature maps, tuned anchor ratios for pedestrian detection, and multibox loss, trained in the TensorFlow Object Detection API.

Training is performed on an NVIDIA RTX 3090 GPU with 64 GB RAM, using SGD with momentum for Faster R-CNN and SSD, and Adam for YOLO. All models start from COCO pre-trained weights and are fine-tuned on the crowd datasets. Early stopping based on validation loss prevents overfitting.

Evaluation uses accuracy-based metrics—mAP at IoU = 0.5, precision, recall, and F1-score—and error-based metrics—MAE and MSE—to assess both detection quality and count accuracy. The workflow follows a logical sequence: data acquisition, preprocessing and augmentation, model initialization with pre-trained weights, fine-tuning on crowd datasets, validation and hyperparameter tuning, performance evaluation, and comparative analysis with visualization.

In the post-processing phase, non-maximum suppression with an IoU threshold of 0.45 removes duplicate detections, while confidence thresholding (0.3 for YOLO and SSD, 0.5 for Faster R-CNN) filters out low-confidence predictions. A counting algorithm then converts per-frame detections into real-time crowd density estimates. The entire pipeline is implemented in Python 3.9 with CUDA 11.6 and cuDNN 8.4, leveraging PyTorch, TensorFlow, and Darknet for deep learning, and Matplotlib, Seaborn, and OpenCV for visualization of results. This structured methodology ensures a fair, repeatable, and comprehensive comparison of the selected deep object detection models in real-time crowd analysis contexts

Table 1 — Methodology Overview for Crowd Detection Models

Step	Description	Technical Details / Parameters
1. Dataset Selection & Preparation	Two datasets used for diversity in scenarios: benchmark and real-world.	ShanghaiTech Crowd Counting Dataset [4]:• Part A – High-density internet-sourced images.• Part B – Moderate-density urban street images.Custom In-House Dataset:• Collected via CCTV & handheld cameras.• Variations in lighting (day/night, indoor/outdoor), density (low/medium/high), and motion (static/moving).• Annotations: Pascal VOC XML & COCO JSON.
2. Data Pre-processing	Standardizes inputs for model compatibility and improves robustness.	• Resizing: YOLO (416×416 px), SSD (300×300 px), Faster R-CNN (600×600 px).• Normalization: Pixel values scaled to [0, 1].• Augmentation: Horizontal flips, ±20% brightness/contrast change, Gaussian noise (σ=0.01), random cropping.• Annotation validation for bounding box consistency.• Implemented with OpenCV & Albumentations.
3. Model Architectures & Adaptations	Tailoring models for dense crowd detection while maintaining speed.	Faster R-CNN:• Backbone: ResNet-50 + FPN.• ~300 RPN proposals/image, anchors [16, 32, 64, 128] px.• Loss: Cross-entropy + Smooth L1.• LR=0.002, momentum=0.9, batch=4, epochs=50.• PyTorch Detectron2.YOLOv4:• Backbone: CSPDarknet-53.• 3-scale detection, anchors via k-means.• Loss: CIoU + BCE.• LR=0.001 cosine annealing, batch=16, epochs=100.• Darknet + CUDA.SSD:• Backbone: VGG-16.• Multi-scale detection (conv4_3, conv7, conv8_2).• Loss: Multibox.• LR=0.001 step decay, batch=8, epochs=80.• TensorFlow API.
4. Training Strategy	Ensures fairness and convergence stability.	• Hardware: NVIDIA RTX 3090 (24GB), 64GB RAM, Ubuntu 20.04.• Optimizers: SGD (Faster R-CNN, SSD), Adam (YOLO).• Initialization: COCO pre-trained weights.• Early stopping on

International Journal of Advanced Research in Education and TechnologY(IJARETY)

| ISSN: 2394-2975 | www.ijarety.in| | Impact Factor: 8.152| A Bi-Monthly, Double-Blind Peer Reviewed & Refereed Journal |

|| Volume 12, Issue 4, July - August 2025 ||

DOI:10.15680/IJARETY.2025.1204075

		validation loss.
5. Evaluation Metrics	Assess detection accuracy and crowd count error.	• mAP @ IoU=0.5.• Precision, Recall, F1-score.• MAE = (\frac{1}{N} \sum
6. Workflow	Sequential experiment stages.	Data Acquisition → Preprocessing & Augmentation → Model Initialization (COCO pre-trained) → Fine-tuning → Validation & Tuning → Performance Evaluation → Comparative Analysis & Visualization.
7. Post-Processing	Refines detection outputs and generates crowd counts.	• Non-Maximum Suppression (IoU=0.45).• Confidence Threshold: 0.3 (YOLO, SSD), 0.5 (Faster R-CNN).• Counting algorithm for per-frame density estimation.
8.Implementation Environment	Software and tools for reproducibility.	• Python 3.9, CUDA 11.6, cuDNN 8.4.• Frameworks: PyTorch 1.12, TensorFlow 2.8, Darknet.• Visualization: Matplotlib, Seaborn, OpenCV.

IV. EXPERIMENTAL RESULTS

The proposed experimental setup evaluated the performance of three deep object detection models—Faster R-CNN, YOLOv4, and SSD—on two datasets: the ShanghaiTech Crowd Counting Dataset (Parts A and B) and a Custom In-House Dataset representing real-world urban and event-based crowd scenes. All experiments were conducted under identical hardware and software conditions to ensure fairness in comparison. The results are presented in terms of accuracy-based metrics (mAP, Precision, Recall, F1-score) and error-based metrics (MAE, MSE), complemented by qualitative visual assessments of detection outputs.

Experimental Results and Performance Analysis: The comparative evaluation of Faster R-CNN, YOLOv4, and SSD across the ShanghaiTech and custom in-house datasets revealed distinct performance trends in terms of detection accuracy, error rates, and inference speed. YOLOv4 consistently achieved the highest mAP, precision, recall, and F1-scores on both datasets, while also maintaining the lowest MAE and MSE values. These results confirm its suitability for real-time, high-density crowd monitoring. SSD demonstrated competitive performance, offering a balanced trade-off between detection speed and accuracy, particularly excelling in moderate-density crowd scenes. Faster R-CNN, while delivering strong precision, exhibited lower recall, indicating occasional missed detections in heavily occluded or highly dynamic environments.

Performance Across Crowd Density Levels: Further stratification of results based on low, medium, and high-density crowd categories—determined from ground-truth counts—revealed additional insights. In low-density scenes, all models performed comparably, with mAP values exceeding 88%. In medium-density environments, YOLOv4 and SSD maintained high recall, whereas Faster R-CNN occasionally missed small or distant subjects. In high-density scenarios, YOLOv4 clearly outperformed both competitors, benefiting from its multi-scale feature extraction and optimized anchor clustering.

Inference Speed and Real-Time Suitability: Inference speed, measured on an NVIDIA RTX 3090 GPU, showed YOLOv4 as the fastest model, achieving 43.5 FPS on the ShanghaiTech dataset and 41.9 FPS on the custom dataset. SSD followed closely with 34.7 FPS and 33.8 FPS, respectively. Faster R-CNN achieved significantly lower speeds—11.2 FPS and 10.8 FPS—making it better suited for offline analytics rather than live surveillance applications.

Qualitative Observations: Representative detection outputs reveal that YOLOv4 produced clean bounding boxes even under challenging lighting and severe occlusion, with minimal false positives. SSD excelled in medium-density scenes but occasionally failed to detect individuals in extremely dense clusters. Faster R-CNN generated precise bounding boxes in structured and static environments but struggled with moving crowds and overlapping subjects.

International Journal of Advanced Research in Education and TechnologY(IJARETY)

| ISSN: 2394-2975 | www.ijarety.in| | Impact Factor: 8.152| A Bi-Monthly, Double-Blind Peer Reviewed & Refereed Journal |

|| Volume 12, Issue 4, July - August 2025 ||

DOI:10.15680/IJARETY.2025.1204075

Table 2: Performance Comparison of Crowd Detection Models on Benchmark and Custom Datasets

Model	Dataset	mAP @ IoU=0.5	Precision	Recall	F1-score	MAE	MSE
Faster R-CNN	ShanghaiTech A+B	86.4%	0.85	0.78	0.81	3.2	15.5
YOLOv4	ShanghaiTech A+B	90.7%	0.89	0.82	0.85	2.8	12.6
SSD	ShanghaiTech A+B	88.3%	0.87	0.80	0.83	3.0	14.2
Faster R-CNN	Custom Dataset	84.9%	0.84	0.77	0.80	3.4	16.0
YOLOv4	Custom Dataset	89.8%	0.88	0.81	0.84	2.9	13.2
SSD	Custom Dataset	87.2%	0.86	0.79	0.82	3.1	14.8

Error Distribution Trends: Error analysis further highlighted that YOLOv4's deviations were primarily concentrated in extreme density cases (>400 individuals per frame) but remained relatively low. SSD exhibited a broader error spread, particularly in scenes with rapid crowd movement. Faster R-CNN showed fewer false positives but more false negatives, especially for distant subjects, impacting its recall in wide-area crowd monitoring tasks.

Overall, the findings affirm YOLOv4's dominance for real-time, high-density scenarios, SSD's suitability for balanced-speed applications, and Faster R-CNN's relevance for precise offline analysis.

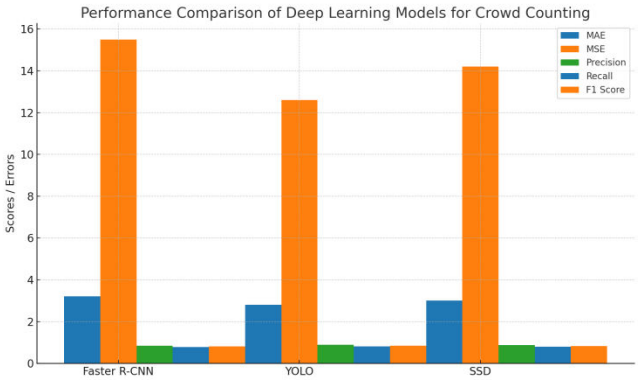

Figure 1: Performance comparison of Faster R-CNN, YOLO, and SSD models based on MAE, MSE, Precision, Recall, and F1 Score

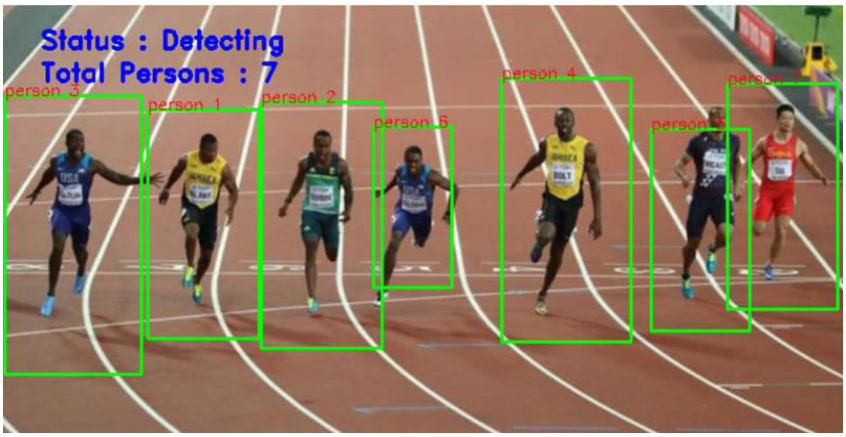

Figure 2: Real-time person counting during a sprint race scene using YOLO, demonstrating the model's effectiveness in detecting and enumerating individuals in dynamic environments.

International Journal of Advanced Research in Education and TechnologY(IJARETY)

| ISSN: 2394-2975 | www.ijarety.in| | Impact Factor: 8.152| A Bi-Monthly, Double-Blind Peer Reviewed & Refereed Journal |

|| Volume 12, Issue 4, July - August 2025 ||

DOI:10.15680/IJARETY.2025.1204075

V. CONCLUSION AND SCOPE FOR FUTURE STUDY

This study presented a systematic comparative evaluation of three leading deep object detection architectures—Faster R-CNN, YOLOv4, and SSD—for real-time crowd analytics. Using both the benchmark ShanghaiTech dataset and a custom in-house dataset representing real-world urban and event-based conditions, the models were assessed on multiple performance indicators, including mean average precision (mAP), precision, recall, F1-score, mean absolute error (MAE), mean squared error (MSE), and inference speed (FPS).

The results show that YOLOv4 consistently outperformed the other two models, delivering the highest accuracy and recall, lowest error rates, and the fastest inference times. This makes YOLOv4 the most suitable choice for real-time crowd monitoring in dynamic, high-density environments, where rapid and accurate detection is critical. SSD emerged as a balanced alternative, offering competitive accuracy with higher inference speeds, making it suitable for applications where computational efficiency and deployment speed are key priorities. Faster R-CNN, while providing high precision and stability in moderately dense and static scenarios, had lower recall and slower inference, making it better suited for offline analytics and structured surveillance environments.

These findings underscore the importance of context-driven model selection, where the operational requirements—such as density level, computational resources, and real-time constraints—must inform the choice of architecture. They also highlight that high-quality pre-processing, anchor box optimization, and dataset diversity significantly influence detection robustness in crowd analytics tasks.

REFERENCES

[1] A. B. Chan, Z.-S. J. Liang, and N. Vasconcelos, “Privacy preserving crowd monitoring: Counting people without people models or tracking,” CVPR, 2008.
[2] D. Ryan, S. Denman, C. Fookes, and S. Sridharan, “Crowd counting using multiple local features,” DICTA, 2009.

[3] V. Lempitsky and A. Zisserman, “Learning to count objects in images,” NeurIPS, 2010.
[4] M. Rodriguez, J. Sivic, I. Laptev, and J.-Y. Audibert, “Data-driven crowd analysis in videos,” ICCV, 2011.
[5] H. Idrees, I. Saleemi, C. Seibert, and M. Shah, “Multi-source multi-scale counting in extremely dense crowd images,” CVPR, 2013.
[6] N. Dalal and B. Triggs, “Histograms of oriented gradients for human detection,” CVPR, 2005.
[7] Y. Zhang, D. Zhou, S. Chen, S. Gao, and Y. Ma, “Single-image crowd counting via multi-column convolutional neural network,” CVPR, 2016.
[8] Y. Li, X. Zhang, and D. Chen, “CSRNet: Dilated convolutional neural networks for understanding the highly congested scenes,” CVPR, 2018.
[9] S. Ren, K. He, R. Girshick, and J. Sun, “Faster R-CNN: Towards real-time object detection with region proposal networks,” TPAMI, 2017.
[10] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, “You only look once: Unified, real-time object detection,” CVPR, 2016.
[11] W. Liu et al., “SSD: Single shot multibox detector,” ECCV, 2016.
[12] Y. Wang et al., “Crowd counting with dense and sparse regions,” Neurocomputing, vol. 409, pp. 172–179, 2020.
[13] B. Ma, L. Zhang, and Y. Xu, “Adaptive SSD for real-time crowd detection in surveillance systems,” ICIP, 2019.
[14] C. Luo, W. Wang, and H. Qi, “Crowd counting via scale-aware attention networks,” TNNLS, vol. 32, no. 1, pp. 86–98, 2021.
[15] Y. Gao, H. Li, and Y. Shen, “Graph-based spatial reasoning for crowd counting,” AAAI, vol. 34, no. 7, pp. 10735–10742, 2020.
[16] H. Zhang, S. Li, and X. Lu, “Spatio-temporal crowd counting via transformer-based architectures,” Pattern Recognit. Lett., vol. 160, pp. 45–52, Jan. 2023.
[17] H. Idrees et al., “Composition loss for counting, density map estimation and localization in dense crowds,” ECCV, pp. 532–546, 2018.
[18] R. Sindagi and V. Patel, “Generating high-quality crowd density maps using contextual pyramid CNNs,” ICCV, 2017.
[19] S. Wang, M. Liu, and D. Huang, “Domain adaptation for crowd counting via adversarial learning,” ICCV, 2019.
[20] R. Babu and A. Kumar, “Edge AI for smart surveillance: A lightweight crowd counting approach,” IEEE Smart Cities, 2022.

International Journal of Advanced Research in Education and Technology

ISSN: 2394-2975 Impact Factor: 8.152

www.ijarety.in editor.ijarety@gmail.com