



ARTICLE

Intelligent Urban Transportation over Complex Vehicle Networks with YOLOv8 for Traffic Flow Monitoring

Mohammed Alonazi¹, Muhammad Adeel Ahmed Tahir², Adnan Ahmed Rafique², Maha Abdelhaq³, Raed Alsaqour⁴, Ahmad Jalal^{5,6} and Jeongmin Park^{7,*}

¹Department of Information Systems, College of Computer Engineering and Sciences, Prince Sattam bin Abdulaziz University, Al-Kharj, Saudi Arabia

²Department of Computer Science, University of Poonch, Rawalakot, AJK, Pakistan

³Department of Information Technology, College of Computer and Information Sciences, Princess Nourah bint Abdulrahman University, Riyadh, Saudi Arabia

⁴Department of Information Technology, College of Computing and Informatics, Saudi Electronic University, Riyadh, Saudi Arabia

⁵Department of Computer Science, Air University, Islamabad, Pakistan

⁶Department of Computer Science and Engineering, College of Informatics, Korea University, Seoul, Republic of Korea

⁷Department of Computer Engineering, Tech University of Korea, 237 Sangidaehak-ro, Siheung-si, Gyeonggi-do, Republic of Korea

*Corresponding Author: Jeongmin Park. Email: jmpark@tukorea.ac.kr

Received: 05 June 2026; Accepted: 14 July 2026; Published: 15 September 2026

ABSTRACT: Accurate vehicle detection, tracking, and counting are fundamental components of Intelligent Transportation Systems (ITS) and urban traffic surveillance. However, real-world deployment remains challenging due to domain shifts, illumination variations, occlusions, dense traffic conditions, and heterogeneous data distributions. Existing studies often address detection, tracking, and counting as independent tasks, resulting in limited cross-domain generalization and inconsistent performance in complex traffic environments. To overcome these limitations, this paper proposes a unified cross-domain framework that jointly integrates vehicle detection, tracking, and lane-aware counting within a single intelligent traffic analytics pipeline. The proposed framework begins with image enhancement using Contrast Limited Adaptive Histogram Equalization (CLAHE) and Gamma Correction to improve visual quality under varying illumination conditions. A Domain-Adversarial Neural Network (DANN) equipped with a Gradient Reversal Layer is employed to learn domain-invariant feature representations across heterogeneous datasets. Vehicle localization is performed using a dual-stage detection strategy in which YOLOv8 generates high-recall object proposals and RetinaNet refines candidate detections to reduce false positives and improve localization accuracy. For multi-object tracking, ByteTrack is enhanced with appearance-based Re-Identification embeddings and Kalman filtering to improve identity preservation and trajectory consistency under occlusions and crowded traffic conditions. Furthermore, a lane-aware multi-zone counting mechanism is introduced to replace conventional line-crossing approaches by enforcing trajectory-consistent spatial transitions across lane-aligned regions. Experimental evaluation on the Cityscapes and KITTI datasets demonstrates significant improvements in detection accuracy, tracking robustness, identity preservation, cross-domain generalization, and vehicle counting performance. Ablation studies further validate the contribution of each module to the overall framework. The proposed system provides an effective solution for real-time urban traffic monitoring and intelligent transportation analytics across heterogeneous environments.

KEYWORDS: Unmanned vehicles; multi-object tracking; traffic dynamics; autonomous systems; complex networks; urban transportation systems; network resilience; intelligent transportation systems; congestion analysis

1 Introduction

Intelligent Transportation Systems (ITS) and unmanned vehicle perception have become essential for modern smart cities, enabling real-time traffic monitoring, congestion analysis, and safety management. Accurate vehicle detection, multi-object tracking, and counting are critical for reliable decision-making in dynamic urban environments. However, performance is often degraded due to domain shift, occlusion, illumination variation, and dataset imbalance. In addition, urban transportation can be represented as complex networks, where intersections act as nodes and road segments as edges, and traffic flow dynamics directly affect congestion propagation and system resilience.

Recent deep learning-based methods have significantly improved vehicle perception. One-stage detectors such as YOLO-based models achieve high-speed inference, while loss-optimized architectures improve robustness in dense scenes. Similarly, multi-object tracking has evolved from motion-only approaches to hybrid methods combining motion and appearance features. Despite these advances, most existing approaches treat detection, tracking, and counting as separate tasks, limiting their generalization across domains and real-world deployment scenarios.

To address these limitations, this work proposes a unified cross-domain vehicle perception framework integrating detection, tracking, and counting. The proposed pipeline begins with image enhancement using CLAHE and gamma correction to reduce illumination effects. A Domain-Adversarial Neural Network (DANN) with Gradient Reversal Layer [1] is employed to learn domain-invariant features across Cityscapes and KITTI. For detection, a YOLOv8-based high-recall model is refined using RetinaNet to improve precision [2,3]. For tracking, ByteTrack is enhanced with Re-ID embeddings and Kalman filtering to reduce identity switches and ensure temporal consistency [4]. Finally, a lane-aware multi-zone counting strategy is introduced, modeling traffic flow as structured transitions within spatial zones aligned with urban network topology. Experimental evaluation on Cityscapes [5] and KITTI [6] demonstrates improved detection accuracy, tracking stability, and counting performance, with ablation studies validating the contribution of each module.

Research Gap and Novelty

Although significant progress has been achieved in vehicle detection, tracking, and traffic monitoring, most existing studies focus on these tasks independently. Detection-oriented methods primarily emphasize localization accuracy, while tracking frameworks focus on identity preservation without explicitly considering cross-domain robustness and counting consistency. Furthermore, many traffic counting approaches rely on simple line-crossing mechanisms that are highly sensitive to occlusion, identity switches, and trajectory fragmentation.

Unlike existing approaches, the proposed framework integrates cross-domain adaptation, dual-stage vehicle detection, identity-aware tracking, trajectory refinement, and lane-aware counting within a unified architecture. The novelty of this work lies not in introducing a new detector or tracker, but in establishing a coherent cross-domain traffic analytics framework that improves consistency across all stages of vehicle perception and traffic analysis. The main contributions of this work are:

- A unified cross-domain vehicle analytics framework is proposed that jointly performs detection, tracking, and lane-aware counting within a single intelligent transportation pipeline.
- A feature-level domain adaptation mechanism based on Domain-Adversarial Neural Networks (DANN) is integrated to improve cross-dataset generalization between heterogeneous traffic environments.
- A dual-stage vehicle detection strategy combining YOLOv8 and RetinaNet is introduced to balance detection recall and localization precision under dense traffic conditions.

- An enhanced ByteTrack framework incorporating Re-Identification embeddings and Kalman filtering is developed to improve identity consistency and reduce tracking fragmentation.
- A novel lane-aware multi-zone counting strategy is proposed that leverages trajectory-consistent spatial transitions for robust vehicle counting in multi-lane urban traffic scenarios.
- Extensive experiments, cross-domain evaluations, and ablation studies demonstrate the effectiveness of the proposed framework for intelligent urban traffic surveillance applications.

The remainder of this paper is organized as follows. [Section 2](#) reviews related work. [Section 3](#) describes the proposed methodology. [Section 4](#) presents the experimental setup. [Section 5](#) discusses the results, and [Section 6](#) concludes the paper.

2 Literature Review

Vehicle detection, tracking, and counting are core components of Intelligent Transportation Systems (ITS) and have been extensively studied for applications in traffic monitoring and autonomous driving. Despite significant progress, existing methods still face challenges in handling occlusion, scale variation, and cross-domain generalization in complex urban environments.

2.1 Vehicle Detection Methods

Deep learning-based object detectors have significantly improved vehicle detection performance in recent years. Two-stage detectors such as Faster R-CNN [7] achieve strong localization accuracy but incur high inference cost, whereas one-stage detectors such as SSD [8] and YOLOv8 [9] enable real-time inference by directly predicting bounding boxes and class probabilities in a single forward pass, making them suitable for traffic surveillance applications. More recent variants, including YOLOv9 [10] and YOLOv10 [11], further improve the accuracy-latency trade-off, while transformer-based real-time detectors such as RT-DETR [12] eliminate the need for non-maximum suppression. However, their performance degrades in dense scenes due to occlusion and class imbalance. Focal-loss-based methods such as RetinaNet address this issue by prioritizing hard examples, improving robustness in challenging conditions [3]. Nevertheless, most existing detection approaches still struggle to maintain a balance between accuracy and real-time performance in highly congested traffic scenarios. In addition, evaluations are often limited to single datasets such as Cityscapes [5] and KITTI [6], while larger and more diverse benchmarks such as Mapillary Vistas [13] and nuScenes [14] remain comparatively underexplored in integrated detection-tracking pipelines.

2.2 Multi-Object Tracking Methods

Multi-object tracking (MOT) focuses on maintaining consistent object identities across video frames. Classical methods such as SORT [15] utilize Kalman filtering and Hungarian assignment, offering computational efficiency but suffering from poor performance under occlusion. To improve robustness, DeepSORT [16] incorporates Re-Identification (Re-ID) features, enhancing identity preservation in crowded scenes. More recent methods such as ByteTrack [4] improve tracking performance by leveraging both high- and low-confidence detections without requiring complex appearance modeling, while observation-centric and appearance-augmented variants such as OC-SORT [17], BoT-SORT [18], and StrongSORT [19] further improve robustness under occlusion. Despite these advancements, identity switches and fragmented trajectories remain significant challenges in dense traffic environments, particularly under occlusion and domain shift conditions.

2.3 Domain Adaptation in Visual Detection

Domain adaptation techniques aim to reduce performance degradation when models are transferred across different environments. The Domain-Adversarial Neural Network (DANN) proposed by Ganin et al. [1] introduces a Gradient Reversal Layer (GRL) to learn domain-invariant representations. Related approaches instead align feature distributions using maximum mean discrepancy [20], adversarial discriminative training [21], or cycle-consistent image translation [22]. While effective, most existing approaches focus on generic object detection tasks and do not fully address traffic-specific cross-dataset challenges. Generalization between Cityscapes and KITTI remains insufficiently explored in integrated detection–tracking pipelines. Moreover, only limited work integrates domain adaptation directly into end-to-end traffic analysis systems.

2.4 Trajectory Modeling and Motion Estimation

Trajectory modeling is essential for maintaining temporal consistency in multi-object tracking systems. The Kalman filter is widely used for motion prediction due to its efficiency and ability to handle uncertainty in object movement [23]. However, many modern tracking pipelines focus primarily on detection and association, often neglecting trajectory smoothing. This results in unstable bounding boxes and jitter, which negatively impacts downstream tasks such as vehicle counting and traffic flow estimation. Therefore, incorporating motion modeling remains critical for reliable real-world deployment.

2.5 Vehicle Counting Methods

Vehicle counting approaches are generally categorized into line-based, region-based, and trajectory-based methods. Line-crossing techniques detect vehicles when they cross predefined virtual lines but are highly sensitive to occlusion and tracking errors [24]. Region-based methods improve robustness by dividing the scene into spatial zones; however, they often ignore lane-level structure, leading to ambiguity in multi-lane traffic [25]. Trajectory-based methods improve counting accuracy but still depend heavily on stable tracking performance [26]. Overall, existing approaches rarely address detection uncertainty, tracking instability, and domain shift simultaneously, limiting their effectiveness in real-world traffic systems.

2.6 Complex Network Models in Transportation

Complex network theory provides a mathematical framework for analyzing urban transportation systems, where intersections are modeled as nodes and roads as edges [27]. Metrics such as centrality and clustering coefficients help identify critical nodes and congestion points. Although these models are useful for macroscopic traffic analysis, most existing studies rely on static or coarse-grained data. In contrast, this work integrates real-time detection, tracking, and lane-aware counting to enable fine-grained traffic flow analysis, bridging the gap between theoretical network models and practical ITS applications.

3 Materials and Methods

This study proposes a unified domain-adaptive framework for robust vehicle detection, tracking, and counting in complex traffic environments. The system integrates multiple stages including preprocessing, domain adaptation, multi-scale object detection, identity-aware tracking, trajectory refinement, and lane-based counting to handle challenges such as illumination variation, domain shift, occlusion, and dynamic traffic flow. Fig. 1 shows the architecture workflow of the proposed system.

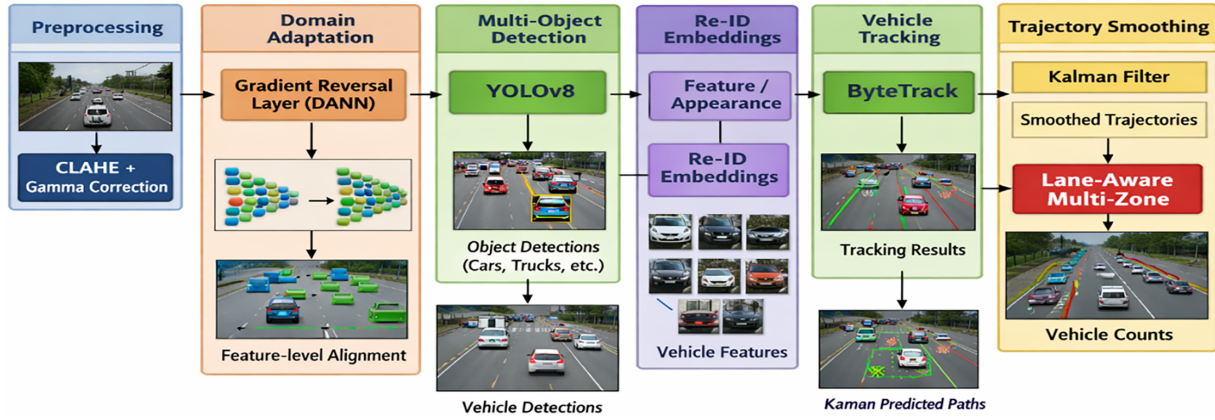


Figure 1: Architecture of the proposed intelligent traffic surveillance system.

3.1 Image Preprocessing

The preprocessing stage aims to improve input frame quality and reduce the effects of illumination variations, shadows, and noise commonly observed in real-world traffic scenes. To achieve this, a two-step enhancement process is applied, consisting of contrast enhancement and intensity normalization, ensuring more stable and discriminative visual features for subsequent detection and tracking modules. Initially, each input frame $I(x, y)$ is enhanced using Contrast Limited Adaptive Histogram Equalization (CLAHE), which improves local contrast by operating on small image regions while limiting noise amplification through histogram clipping [28]. The transformation can be expressed as in Eq. (1):

$$I_{CLAHE}(x, y) = \frac{CDF(I(x, y)) - CDF_{\min}}{(M \times N) - CDF_{\min}} \times (L - 1) \quad (1)$$

where CDF is the cumulative distribution function of the clipped histogram, $CDF_{\min}$ is the minimum non-zero value, $M \times N$ is the number of pixels in a tile, and L represents the intensity levels. This operation enhances local structural details such as vehicle edges and textures, particularly in low-light regions. After contrast enhancement, gamma correction is applied to normalize global illumination variations [29]. This non-linear transformation is defined in Eq. (2):

$$I_{\text{out}}(x, y) = c \cdot I_{CLAHE}(x, y)^\gamma \quad (2)$$

where c is a scaling constant and γ controls brightness adjustment. Values of $\gamma < 1$ brighten dark scenes, while $\gamma > 1$ suppress overly bright regions, enabling balanced intensity distribution across varying lighting conditions. Finally, the complete preprocessing pipeline is represented as:

$$I_{\text{final}}(x, y) = c \cdot (CLAHE(I(x, y)))^\gamma \quad (3)$$

This enhanced output preserves essential visual structures while reducing environmental distortions. Consequently, downstream modules such as domain adaptation, detection (YOLOv8/RetinaNet), and tracking receive more consistent and robust feature representations, leading to improved overall system stability and accuracy under diverse traffic conditions. The output of the CLAHE can be depicted in Fig. 2.



Figure 2: Enhanced preprocessing results using CLAHE and gamma correction on Cityscapes and KITTI.

3.2 Feature-Level Domain Adaptation

In cross-dataset traffic analysis, models often suffer from performance degradation due to domain shift caused by variations in imaging conditions, viewpoints, and scene dynamics. To address this issue, a Domain-Adversarial Neural Network (DANN) is incorporated to learn domain-invariant feature representations at the feature level, improving generalization between datasets such as Cityscapes and KITTI. The overall architecture of the proposed domain adaptation module is illustrated in Fig. 3.

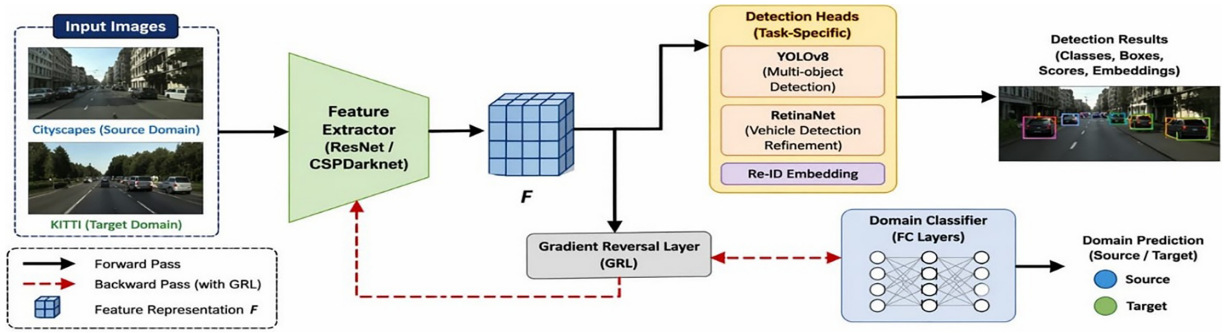


Figure 3: Feature-level domain adaption framework (DANN).

Given an input image I , a deep feature extractor G_f , implemented using a CSPDarknet and ResNet backbone, maps it to a latent representation:

$$F = G_f(I) \quad (4)$$

where F encodes high-level spatial and semantic information and is shared across detection and domain classification tasks. The structure of the feature extraction process is shown in Fig. 4. To enforce domain invariance, a domain classifier G_d predicts the domain label, as shown in Eq. (5).

$$d = G_d(F) \quad (5)$$

The corresponding domain classification loss is defined as

$$L_d = - \sum_{i=1}^N [y_i \log(d_i) + (1 - y_i) \log(1 - d_i)] \quad (6)$$

where $y_i = 1$ if the feature comes from the source domain and $y_i = 0$ for the target domain. This loss trains the domain classifier to distinguish source and target feature. A Gradient Reversal Layer (GRL) is introduced to achieve adversarial learning by reversing gradients during backpropagation:

$$\frac{\partial L_d}{\partial F} \leftarrow -\lambda \frac{\partial L_d}{\partial F} \quad (7)$$

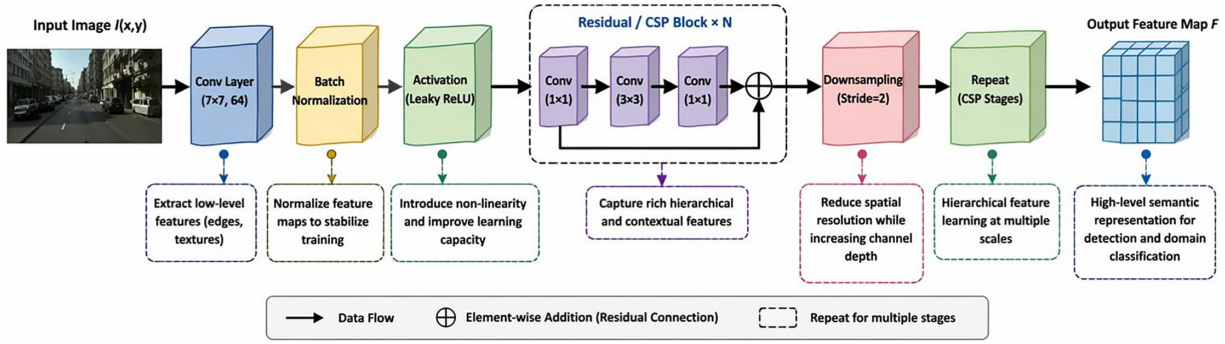


Figure 4: Feature extractor architecture (backbone network).

This mechanism forces the feature extractor to generate representations that are both discriminative for detection and invariant across domains [30]. The overall training objective is defined as:

$$L_{\text{total}} = L_{\text{task}} + \alpha L_d \quad (8)$$

where L_{task} represents detection and Re-ID losses, and α controls the contribution of domain alignment. As shown in the experimental results section, the proposed domain adaptation strategy improves cross-domain detection performance and reduces generalization error compared to non-adaptive baselines. Unlike conventional approaches that apply adaptation at the image level or within single-stage detectors, the proposed method integrates feature-level adversarial learning within a multi-stage detection and tracking pipeline. Furthermore, the learned domain-invariant features are propagated to downstream modules, including Re-ID embedding and ByteTrack tracking, ensuring consistency in both detection and identity preservation.

3.3 Multi-Object Detection Using YOLOv8

Following domain adaptation, the proposed framework performs multi-object detection using domain-invariant feature representations F , enabling robust localization across heterogeneous datasets. By directly utilizing features learned through adversarial training, the detection module mitigates performance degradation caused by domain shift and environmental variability [31]. To achieve efficient and real-time detection, YOLOv8 is employed as a one-stage detector due to its capability to jointly perform localization and classification in a single forward pass [9]. Given the adapted feature maps F , the detector G_y produces a set of object detections as defined in Eq. (9).

$$D = G_y(F) \quad (9)$$

Each detection D_i is defined as:

$$D_i = (x_i, y_i, w_i, h_i, c_i, s_i) \quad (10)$$

where (x_i, y_i) denote the bounding box center, w_i and h_i represent width and height, c_i is the class label, and s_i is the confidence score. To handle scale variations in traffic scenes, YOLOv8 employs multi-scale feature learning through Feature Pyramid Network (FPN) and Path Aggregation Network (PAN) structures, which effectively combine semantic and spatial information across different resolutions [32,33]. Let F_s , F_m and F_l denote feature maps at different scales. The detection process is formulated as:

$$D = \bigcup_{k \in \{s, m, l\}} G_y^k(F_k) \quad (11)$$

This strategy enables accurate detection of small distant vehicles as well as large nearby objects. The detection network is optimized using a composite loss function:

$$L_{det} = L_{box} + L_{cls} + L_{obj} \quad (12)$$

where L_{box} is the bounding box regression loss (IoU-based), L_{cls} is the classification loss, L_{obj} is the objectness confidence loss. Advanced IoU variants such as CIoU further improve localization accuracy. A key strength of the proposed approach lies in the tight integration between domain adaptation and detection. Specifically, the detector operates on features defined as $F = G_f(I)$, where G_f is trained adversarial via GRL. This enables the model to generalize effectively across domains without requiring extensive retraining. As demonstrated in the experimental section, this integration improves detection robustness and reduces cross-domain performance degradation. The background architecture of YOLOv8 can be seen in Fig. 5.

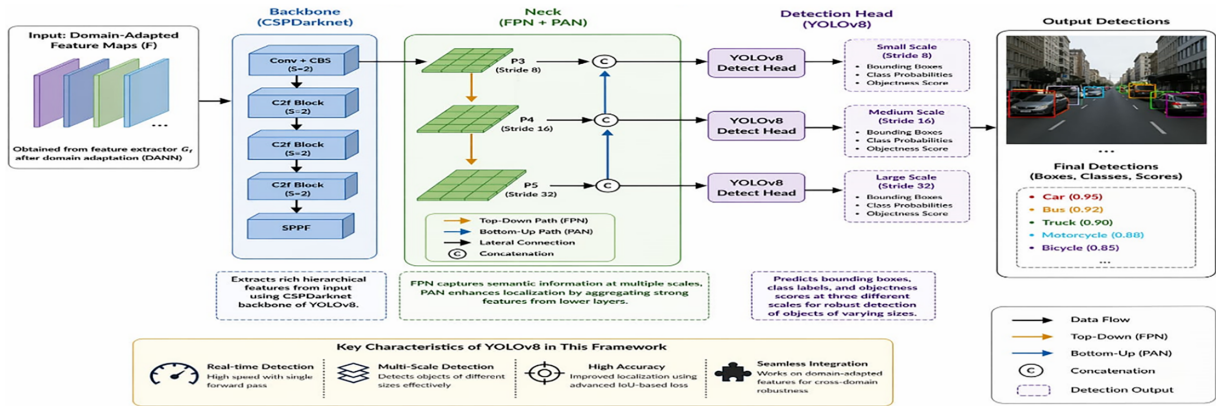


Figure 5: Multi-object detection using the YOLOv8 algorithm.

Furthermore, the proposed detection module provides high-recall object localization, which serves as a foundation for subsequent stages including vehicle refinement (RetinaNet), identity-aware tracking, and trajectory estimation. Compared to conventional YOLO-based frameworks, the use of domain-adaptive features significantly enhances detection stability under varying environmental and dataset conditions. The output of the multi-objects can be seen in Fig. 6.

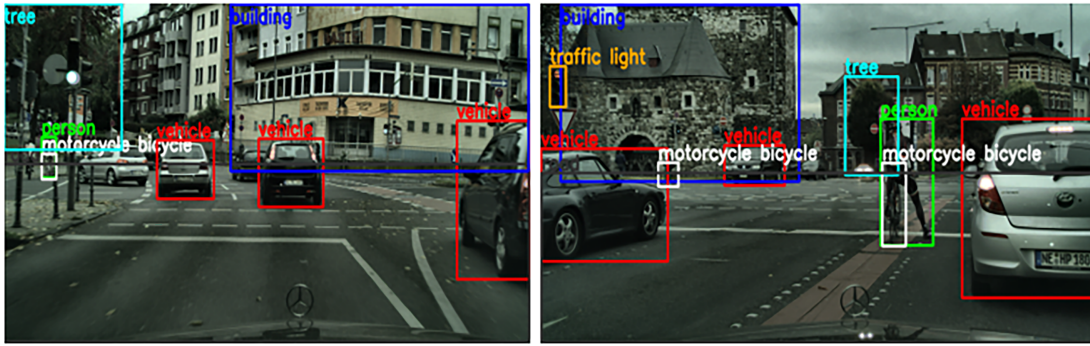


Figure 6: Multi-object detection using the YOLOv8.

3.4 Vehicle-Specific Detection Refinement Using RetinaNet

To improve vehicle localization accuracy in complex traffic environments, a RetinaNet-based refinement module is introduced after the multi-object detection stage. Although YOLOv8 provides high-recall detection by capturing all object instances, its predictions for vehicles may exhibit localization inaccuracies and false positives under occlusion, scale variation, and dense traffic conditions. To mitigate these limitations, a vehicle-specific second-stage detector is employed to refine candidate regions, thereby improving precision while preserving recall. The refined vehicle detections are formulated as:

$$V = G_r(I), \quad V_i = (x_i, y_i, w_i, h_i, c_i, s_i) \quad (13)$$

where G_r denotes the RetinaNet detector and V_i represents the bounding box coordinates, class label, and confidence score of the i -th vehicle instance. To maintain computational efficiency, RetinaNet operates either on the full image or on regions of interest derived from YOLOv8 outputs, effectively reducing redundant computations while focusing on relevant areas. RetinaNet leverages a Feature Pyramid Network (FPN) to handle scale variations effectively [32]. Let F_k denote feature maps at pyramid levels $k \in \{s, m, l\}$; the detection process is expressed as:

$$V = \bigcup_{k \in \{s, m, l\}} G_r^{(k)}(F_k) \quad (14)$$

This multi-scale aggregation is particularly important in traffic scenes, where vehicles appear at significantly different scales due to perspective variations. To further address the severe foreground–background imbalance inherent in dense detection tasks, RetinaNet utilizes Focal Loss:

$$FL(p_t) = -\alpha(1 - p_t)^\gamma \log(p_t) \quad (15)$$

where $\alpha = 0.25$ and $\gamma = 2$. This formulation suppresses easy negative samples and emphasizes hard examples, leading to improved discrimination of challenging vehicle instances and a reduction in false detections. The background for the RetinaNet can be seen in Fig. 7.

Unlike conventional two-stage detectors (e.g., Faster R-CNN) that rely on region proposal networks, the proposed framework integrates a high-recall one-stage detector (YOLOv8) with a precision-oriented dense detector (RetinaNet). This design introduces a *cross-detector refinement strategy*, where candidate regions generated by YOLOv8 are selectively re-evaluated by RetinaNet. Compared to single-detector pipelines, this approach reduces error propagation and improves localization stability, particularly in cluttered and high-density traffic scenes. Moreover, restricting refinement to ROIs significantly lowers computational

overhead compared to full-frame re-detection, making the framework suitable for real-time intelligent transportation applications.

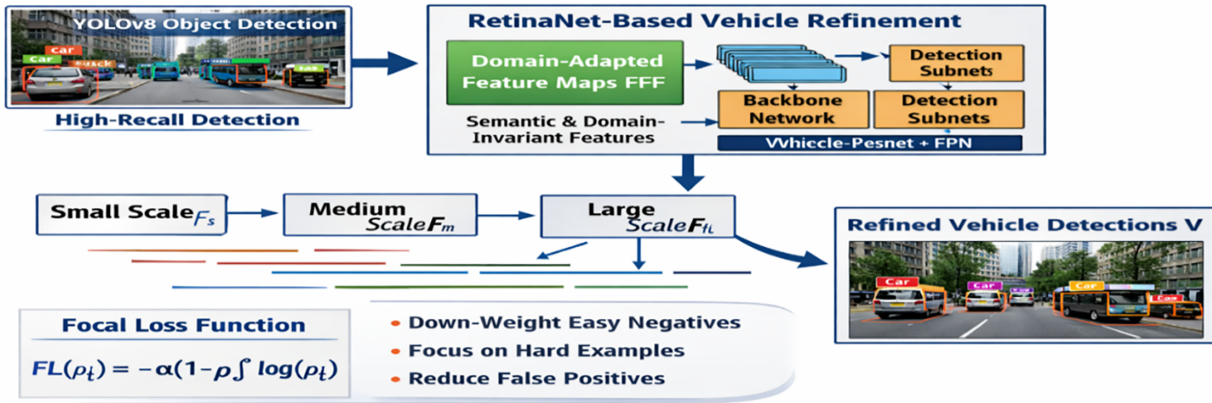


Figure 7: Refined vehicle detection using RetinaNet.

Overall, this hierarchical detection paradigm effectively balances recall and precision, resulting in more accurate and reliable vehicle localization, which directly benefits downstream tasks such as multi-object tracking and vehicle counting. The result for the vehicle detection can be seen in Fig. 8.

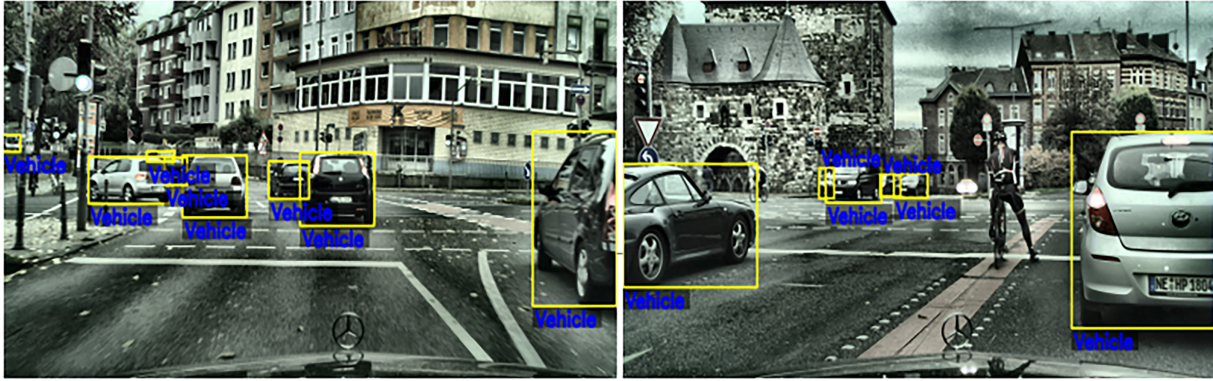


Figure 8: Vehicle detection via RetinaNet.

3.5 Feature Embedding and Multi-Object Tracking

To address the limitations of conventional tracking-by-detection approaches in complex urban environments, a hybrid multi-object tracking framework is proposed that integrates domain-adaptive appearance embedding, feature-aware association, and motion-guided prediction. By jointly exploiting semantic feature consistency and temporal dynamics, the framework achieves robust identity preservation under occlusion, illumination variation, and dense traffic conditions. Each detected vehicle V_i is encoded into a discriminative embedding vector, as expressed in Eq. (16).

$$e_i = G_e(F_i) \quad (16)$$

where G_e denotes the embedding function and F_i represents the refined feature representation. The use of domain-aligned features ensures robustness to cross-dataset variations, improving association

reliability [34]. Inter-frame association is performed using a dual-consistency similarity measure that combines appearance and spatial cues. The cosine similarity between embeddings is defined as:

$$S(e_i, e_j) = \frac{e_i \cdot e_j}{\|e_i\| \|e_j\|} \quad (17)$$

This enables reliable matching under occlusion and motion ambiguity, complementing geometric overlap [35]. The ByteTrack framework is extended by incorporating feature-aware association. Unlike the original formulation, which relies primarily on detection confidence, the proposed method integrates appearance similarity into both high- and low-confidence matching stages, reducing identity switches and improving trajectory continuity. For each pair of detection i and existing track j , the association cost $C(i, j)$ is defined as:

$$C(i, j) = \lambda_1 (1 - IoU_{ij}) + \lambda_2 (1 - S(e_i, e_j)) \quad (18)$$

where IoU_{ij} denotes the Intersection over Union between detection i and track j ; $S(e_i, e_j)$ is the cosine similarity between their respective embedding vectors e_i and e_j defined in Eq. (17); and λ_1, λ_2 are scalar weighting coefficients controlling the relative contribution of spatial and appearance cues. Optimal assignment across all detection-track pairs is obtained by minimizing the total association cost over all feasible one-to-one assignments using the Hungarian algorithm. To maintain motion continuity during association, a Kalman filter [23] models the vehicle state and measurement as:

$$x_k = Ax_{k-1} + w_k, z_k = Hx_k + v_k \quad (19)$$

This enables stable tracking during short-term detection failures and occlusions. A lightweight track management strategy is further applied to control initialization and termination, preventing noisy trajectories while preserving identity consistency [36]. The proposed framework extends ByteTrack with domain-adaptive embeddings and unified appearance–motion association, enabling robust cross-domain tracking and improved identity stability. The effectiveness of each component is validated through ablation studies in Section 4. After association, Kalman filtering, and track management, the tracker outputs a set of n confirmed vehicle trajectories across the video sequence:

$$T = \{T_1, T_2, \dots, T_n\} \quad (20)$$

where each trajectory T_i represents a temporally consistent sequence of bounding-box states for a single vehicle across frames, produced by the combined appearance–motion association defined in Eq. (18) and the motion model in Eq. (19). The index $i \in \{1, \dots, n\}$ is consistent with the detection index i used in Eqs. (16)–(19) above. The output can be seen in Fig. 9.

3.6 Lane-Aware Multi-Zone Vehicle Counting

Following tracking, a lane-aware multi-zone vehicle counting strategy is introduced that reformulates counting as a trajectory-based zone transition problem rather than a traditional line-crossing event. The scene is partitioned into structured spatial zones aligned with traffic lanes, enabling direction-aware and occlusion-robust counting in dense urban traffic. The scene is defined as:

$$Z = \{Z_1, Z_2, \dots, Z_K\}, z_i^t = Z(x_i^t, y_i^t) \quad (21)$$

A vehicle is counted only when its trajectory exhibits a valid transition between adjacent zones.

$$C = \sum_i \mathbb{I}(Z_a \rightarrow Z_b) \quad (22)$$

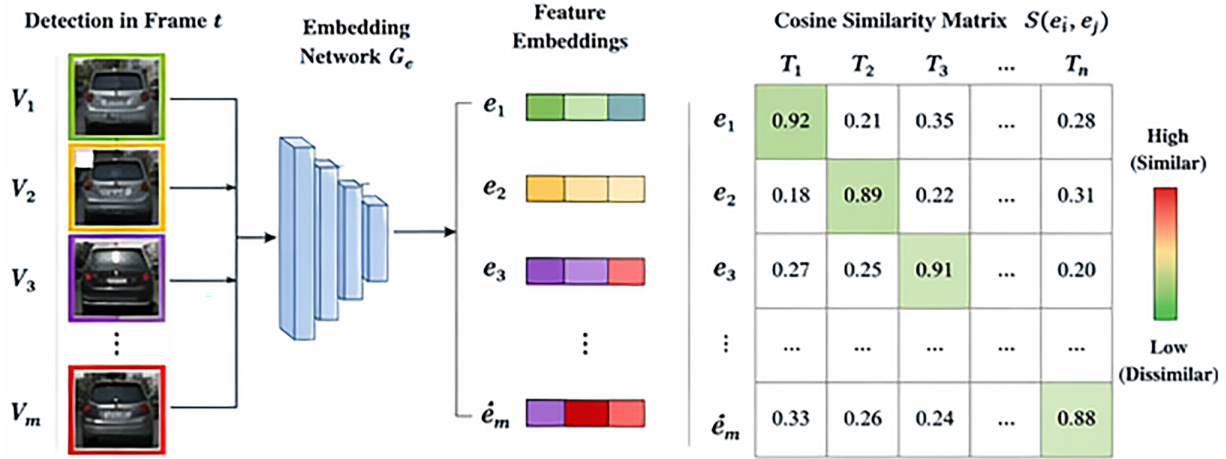


Figure 9: Re-ID embeddings and similarity computations.

In Eq. (22), the indicator function $I(\cdot)$ returns 1 if and only if trajectory T_i (as defined in Eq. (20)) passes through zone Z_a before entering the adjacent zone Z_b where Z_a and Z_b are distinct elements of the zone partition Z defined in Eq. (21). The subscript i in the summation is the same vehicle index used throughout Eqs. (16)–(20):

$$v_i \cdot d_{ab} > 0 \quad (23)$$

where d_{ab} represents the direction vector between zones. This ensures that only motion aligned with traffic flow contributes to counting, eliminating spurious and reverse transitions. To suppress noisy trajectories, a persistence constraint is applied:

$$\text{valid}(T_i) = \begin{cases} 1, & |T_i| > \tau \\ 0, & \text{otherwise} \end{cases} \quad (24)$$

This filtering mechanism removes short-lived or fragmented trajectories, thereby minimizing counting errors caused by identity switches or detection noise. The total lane-wise count is computed as:

$$C_{\text{total}} = \sum_{k=1}^{K-1} C(Z_k \rightarrow Z_{k+1}) \quad (25)$$

This formulation enables fine-grained lane-level traffic analysis within a unified framework. The proposed method reformulates vehicle counting as a trajectory-consistent, direction-constrained zone transition problem. Unlike conventional line-crossing methods, it leverages lane-aware spatial decomposition and motion-guided validation, ensuring robustness against occlusion, identity switches, and detection noise. The integration of tracking, ReID embeddings, and Kalman filtering enables reliable spatiotemporal consistency, while supporting scalable multi-lane traffic analytics in a unified formulation. The final output is shown

in Eq. (26) where each element corresponds to a specific lane or zone. The result for the Lane aware can be determined in Fig. 10.

$$C = \{C_1, C_2, \dots, C_K\} \quad (26)$$

4 Experimentation and Results

The proposed method was implemented and validated in the Python 3.8 environment, utilizing widely adopted deep learning and computer vision libraries. The framework integrates preprocessing, domain adaptation, detection, tracking, and counting modules into a unified pipeline. Key dependencies include:

- PyTorch 1.10 (for YOLOv8, RetinaNet, and DANN-based training)
- OpenCV 4.5 (for image preprocessing and visualization)
- NumPy (for matrix operations)
- scikit-learn 0.24 (for evaluation metrics)

Experiments were performed on an Intel Core i5-12500H 2.50 GHz processor with 24 GB RAM and an NVIDIA RTX-3050 GPU with 4 GB memory. The model demonstrated superior performance across both Cityscapes and KITTI datasets, validating its robustness in diverse traffic environments.

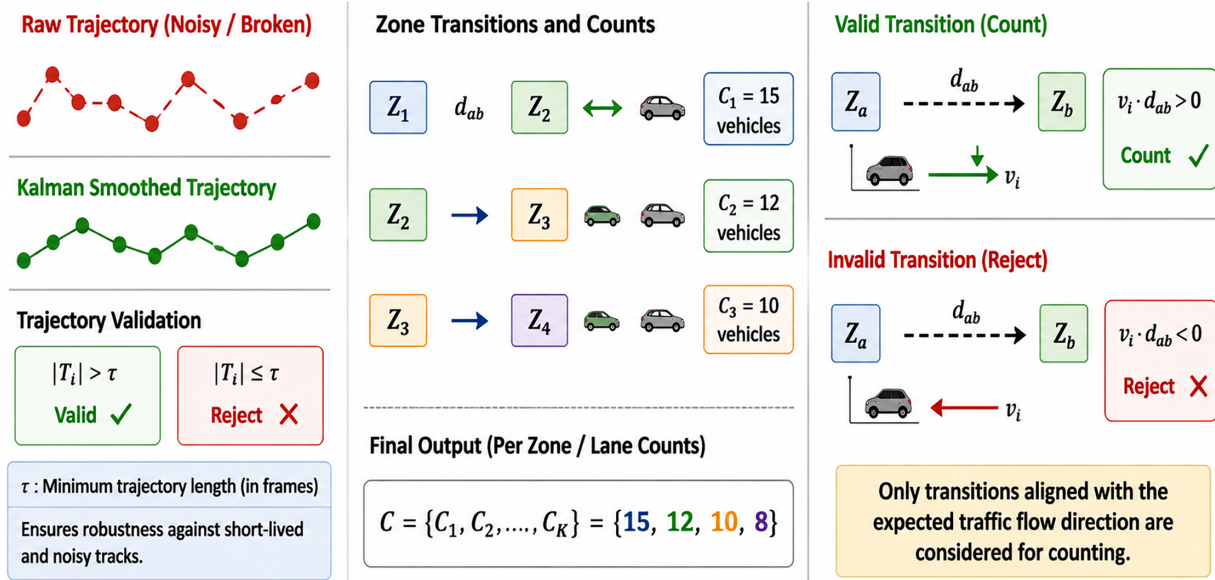


Figure 10: Lane-aware zone counting.

4.1 Dataset Description

4.1.1 Cityscapes Dataset

The Cityscapes Dataset comprises high-resolution urban street images captured under varying environmental conditions. It contains 5000 finely annotated images with pixel-level labels. The dataset includes vehicle categories such as cars, trucks, and buses, with dense traffic scenes and complex occlusions. Its high resolution (2048×1024 pixels) and diverse lighting conditions make it suitable for evaluating detection, tracking, and counting performance in real-world urban scenarios.

4.1.2 KITTI Dataset

The KITTI Dataset is widely used for autonomous driving research and provides challenging real-world driving sequences. It contains RGB images with a resolution of approximately 1242×375 pixels and includes multiple traffic scenarios such as urban roads and highways. The dataset introduces challenges such as motion blur, scale variation, and camera movement, making it ideal for evaluating cross-domain generalization.

4.2 Model Evaluation and Experimental Results

Detection performance is evaluated using Precision, Recall, F1-score, and mean Average Precision (mAP) at IoU thresholds of 0.5 and 0.5:0.95, following the standard evaluation protocols of the Cityscapes [5] and KITTI [6] benchmarks. Tracking performance is assessed using Multiple Object Tracking Accuracy (MOTA), Identity F1-score (IDF1), and Higher Order Tracking Accuracy (HOTA), which collectively measure detection quality, identity consistency, and overall tracking performance [35,37]. Vehicle counting is evaluated using Mean Absolute Error (MAE), Root Mean Square Error (RMSE), and counting accuracy. Vehicle counting results (Table 1) are averaged over three independent runs and reported as mean $\pm$ standard deviation. Detection, tracking, cross-domain, efficiency, and ablation results (Tables 2–6) are obtained from a single representative run under a fixed random seed and are therefore reported as point estimates. The results confirm that the integration of multi-scale detection and feature refinement improves overall accuracy. Table 2 shows the comparison across detection model with SOTA techniques.

Table 1: Vehicle counting accuracy.

Method	MAE $\downarrow$	RMSE $\downarrow$	Accuracy $\uparrow$
Line-Based Counting	12.5 ± 1.2	15.2 ± 1.5	85%
Region-Based Counting	9.8 ± 1.0	12.1 ± 1.2	88%
Tracking-Based Counting	7.2 ± 0.8	9.5 ± 1.0	91%
Lane-Aware Multi-Zone (Proposed)	4.3 ± 0.5	6.1 ± 0.6	95%

Table 2: Comparison of model detection rate with other state-of-art methods.

Dataset	Method	Precision	Recall	F1-Score	mAP@0.5	mAP@0.5:0.95
Cityscapes	Faster R-CNN	0.82	0.80	0.81	0.85	0.62
	RetinaNet	0.86	0.84	0.85	0.88	0.66
	YOLOv8	0.91	0.89	0.90	0.92	0.71
	YOLOv9	0.92	0.90	0.91	0.93	0.73
	YOLOv10	0.93	0.91	0.92	0.94	0.74
	RT-DETR	0.94	0.92	0.93	0.95	0.76
	Proposed	0.95	0.94	0.945	0.96	0.78
KITTI	Faster R-CNN	0.84	0.82	0.83	0.86	0.64
	RetinaNet	0.85	0.83	0.84	0.87	0.65
	YOLOv8	0.90	0.88	0.89	0.91	0.70
	YOLOv9	0.91	0.89	0.90	0.92	0.72
	YOLOv10	0.92	0.90	0.91	0.93	0.74
	RT-DETR	0.93	0.91	0.92	0.94	0.75
	Proposed	0.94	0.93	0.935	0.95	0.76

Table 3: Multi-object tracking performance comparison.

Method	MOTA ↑	IDF1 ↑	HOTA ↑	ID Switches ↓	FPS ↑
SORT	0.78	0.75	0.70	120	35
DeepSORT	0.82	0.80	0.76	95	28
ByteTrack	0.87	0.85	0.81	70	30
OC-SORT	0.88	0.86	0.83	60	32
BoT-SORT	0.89	0.87	0.84	55	29
Proposed Method	0.90	0.88	0.85	45	27

Table 4: Cross domain generalization evaluation.

Training → Testing	Method	Precision	Recall	F1-Score
Cityscapes → KITTI	YOLOv8	0.82	0.80	0.81
	Proposed (DANN)	0.90	0.88	0.89
KITTI → Cityscapes	YOLOv8	0.80	0.78	0.79
	Proposed (DANN)	0.89	0.87	0.88

Table 5: Computational efficiency analysis.

Method	Params (M)	FLOPs (G)	FPS ↑	Latency (ms)
YOLOv8	25.3	78.2	32	31
YOLOv10	28.1	82.5	30	33
RT-DETR	42.6	120.4	24	41
ByteTrack	18.2	45.3	30	28
BoT-SORT	22.5	60.1	28	30
Proposed	45.8	140.2	27	37

Table 6: Ablation study of proposed framework.

Configuration	Detection	Tracking	Domain Adaptation	Counting	mAP@0.5	MOTA
Baseline (YOLOv8 only)	✓	✗	✗	✗	0.92	0.72
+Tracking Module	✓	✓	✗	✗	0.92	0.80
+DANN Domain Adaptation	✓	✓	✓	✗	0.94	0.83
+Lane-Aware Counting	✓	✓	✓	✓	0.96	0.86

The proposed method consistently outperforms baseline detectors across both datasets. The improvement is primarily attributed to enhanced feature representation and vehicle-specific refinement, which improves localization accuracy and reduces false positives.

Fig. 11 presents the confusion matrix of the proposed model. The strong diagonal dominance indicates high classification accuracy across vehicle categories, while minimal off-diagonal values demonstrate reduced misclassification.

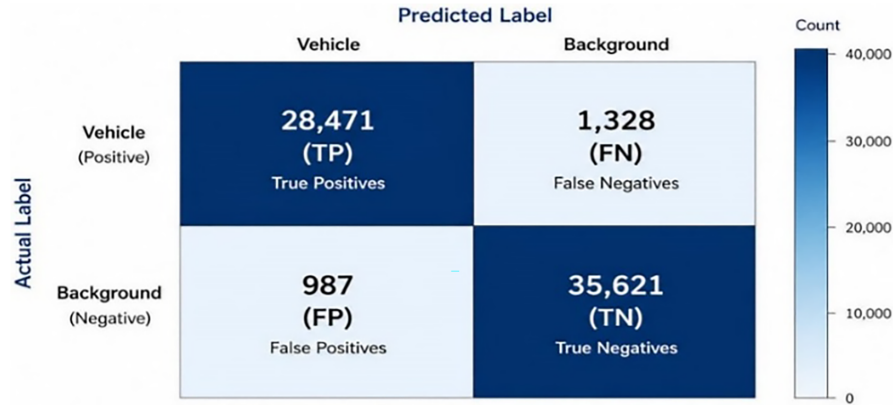


Figure 11: Confusion matrix for vehicle detection (binary classification).

Table 3 shows the proposed method achieves superior tracking performance, significantly reducing identity switches while maintaining high accuracy. The use of Re-ID embeddings improves identity preservation, while Kalman filtering stabilizes trajectories. Table 1 shows the vehicle counting accuracy.

The proposed lane-aware multi-zone counting method significantly reduces counting errors by incorporating spatial lane constraints and trajectory consistency, thereby avoiding double counting and improving robustness under occlusion.

Domain adaptation significantly improves generalization across datasets by aligning feature distributions between source and target domains, reducing domain discrepancy. Table 4 shows the Cross domain generalization.

Table 5 compares the computational efficiency of the proposed framework with state-of-the-art detection and tracking methods in terms of parameters, FLOPs, FPS, and latency. The results show that lightweight models such as YOLOv8 and ByteTrack achieve higher speed but with limited representational capacity, while transformer-based RT-DETR incurs higher computational cost with reduced real-time performance. In contrast, the proposed method maintains a balanced trade-off, achieving competitive real-time speed (27 FPS) despite higher complexity, while delivering improved overall detection and tracking accuracy, making it suitable for practical intelligent transportation system deployment.

Table 6 demonstrates that each component contributes incrementally to overall performance. Domain adaptation provides the most significant improvement due to reduced domain shift, while RetinaNet enhances detection precision. Fig. 12 shows the precision comparison.

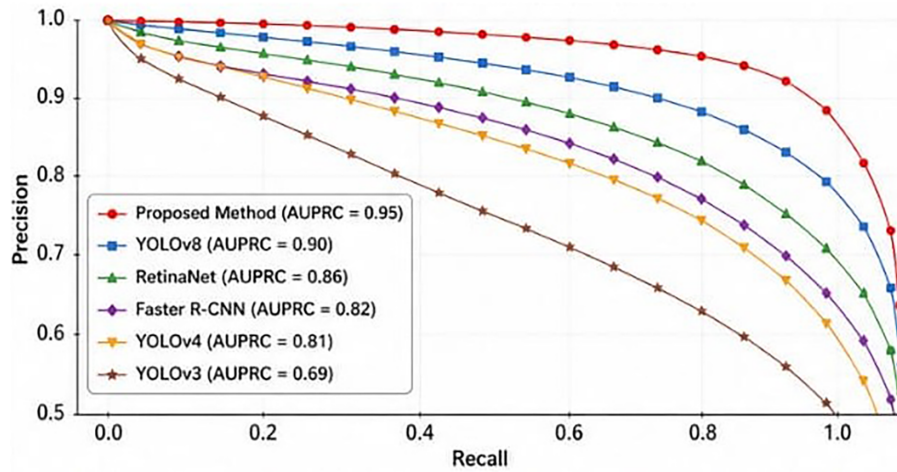


Figure 12: Precision-recall curve comparison.

5 Discussion/Limitations

5.1 Overall Performance Analysis

The proposed framework demonstrates strong performance for vehicle detection, tracking, and counting in complex urban environments by integrating CLAHE-based enhancement, domain adaptation (DANN), dual-stage detection (YOLOv8 with RetinaNet refinement), Re-Identification embeddings, ByteTrack association, and Kalman filtering. Experimental evaluation on Cityscapes Dataset and KITTI Vision Benchmark Suite confirms consistent improvements in detection accuracy, tracking stability, and counting reliability compared to baseline approaches.

5.2 Generalization and Dataset Limitations

The evaluation is limited to two benchmark datasets, which restricts exposure to broader environmental variability such as night-time scenes, adverse weather, and geographically diverse traffic patterns; extending the evaluation to larger benchmarks such as BDD100K [38] and the Waymo Open Dataset [39] remains an important direction for future work. Although domain adaptation partially reduces dataset bias, residual domain shift still affects cross-scene generalization. Additionally, class imbalance in aerial-style distributions leads to biased learning behavior, where dominant classes (e.g., cars and buses) achieve higher recall compared to underrepresented categories such as trucks and bicycles.

5.3 Robustness under Challenging Conditions

Performance degradation is observed under low-contrast, motion-blurred, and heavily occluded conditions. While deep detection backbones mitigate a significant portion of these issues, residual errors persist in densely populated traffic scenes. Moreover, any significant variation in UAV altitude, viewing angle, or sensor configuration introduces distribution shifts that are not fully compensated by the current adaptation mechanism.

5.4 Computational and Deployment Constraints

The multi-stage architecture improves accuracy but introduces additional computational overhead due to sequential processing across enhancement, adaptation, detection, feature extraction, and tracking stages. This increases inference latency and limits direct deployment on low-power edge devices.

It should be noted that model compression techniques such as pruning and INT8 quantization were not applied in the current experimental setup; they are proposed here as directions for future work to reduce the computational cost of YOLOv8 with minimal accuracy loss. Similarly, replacing heavy feature extraction modules with lightweight architecture and applying dimensionality reduction techniques are additional future directions that can further improve efficiency. For current deployment scenarios requiring real-time processing on resource-constrained devices, a hybrid UAV-edge or ground-station architecture is recommended.

5.5 Accuracy-Latency Trade-Off Analysis

The integration of RetinaNet refinement and appearance-based Re-Identification embeddings introduces additional computational overhead compared with the original ByteTrack framework. Experimental results indicate that the proposed framework achieves superior detection accuracy and tracking stability while maintaining near real-time performance. Specifically, the FPS decreases from 30 FPS (ByteTrack) to 27 FPS in the proposed framework. However, this moderate reduction in processing speed is accompanied by substantial improvements in MOTA, IDF1, HOTA, and identity-switch reduction. As confirmed by the empirical data in Table 5, the proposed framework operates at 27 FPS with a latency of 37 ms compared to 30 FPS (28 ms) for ByteTrack alone while achieving higher MOTA (0.90 vs. 0.87), IDF1 (0.88 vs. 0.85), HOTA (0.85 vs. 0.81), and a reduction in identity switches from 70 to 45 (Table 3). Therefore, the proposed framework achieves a favorable balance between computational efficiency and analytical accuracy, making it suitable for intelligent transportation applications where reliability is prioritized over marginal speed improvements.

In summary, while the proposed framework significantly improves multi-stage vehicle detection, tracking, and counting performance, its limitations stem from dataset diversity, computational complexity, and sensitivity to extreme environmental conditions. Future work will focus on large-scale cross-domain training, fully lightweight end-to-end architectures, and improved robustness under dynamic UAV configurations and real-world deployment constraints.

6 Conclusion

The proposed framework provides a unified approach for vehicle detection, tracking, and lane-aware counting in complex urban traffic scenes by integrating image enhancement, domain adaptation, dual-stage detection, Re-ID embeddings, ByteTrack association, and Kalman filtering. Within urban transportation networks, it models traffic zones as network components and vehicle trajectories as flow measurements for fine-grained congestion and traffic analysis. Experimental results on the Cityscapes and KITTI datasets show improved detection, tracking, and counting performance under occlusion and cross-domain variations, while also identifying bottleneck zones effectively.

Limitations and Future Research Directions

Despite the promising performance achieved by the proposed framework, several limitations remain. First, the detection accuracy may decrease when vehicles occupy only a small number of pixels in distant traffic scenes, making small-object localization challenging. Second, although the integration of Re-Identification embeddings improves tracking consistency, severe occlusions and dense traffic conditions can still introduce identity switches and fragmented trajectories. Third, the framework has been evaluated primarily on standard urban driving datasets and may experience performance degradation under adverse environmental conditions such as heavy rain, fog, low illumination, or extreme weather scenarios. Fourth, while Domain-Adversarial Neural Networks improve cross-domain generalization, residual domain discrepancies may still exist when transferring to unseen traffic environments with significantly different visual

characteristics. Finally, the combination of image enhancement, dual-stage detection, domain adaptation, and tracking modules introduces additional computational overhead, which may limit deployment on resource-constrained edge devices. Future work will focus on lightweight model optimization, weather-robust perception, advanced occlusion handling, and graph-based traffic network analysis for large-scale intelligent transportation systems.

Acknowledgement: Princess Nourah bint Abdulrahman University Researchers Supporting Project number (PNURSP2026R97), Princess Nourah bint Abdulrahman University, Riyadh, Saudi Arabia.

Funding Statement: This work was supported by the IITP (Institute of Information & Communications Technology Planning & Evaluation)-ICAN (ICT Challenge and Advanced Network of HRD) (IITP-2026-RS-2022-00156326,33) grant funded by the Korea government (Ministry of Science and ICT). This research is supported and funded by Princess Nourah bint Abdulrahman University Researchers Supporting Project number (PNURSP2026R97), Princess Nourah bint Abdulrahman University, Riyadh, Saudi Arabia. This study is supported via funding from Prince sattam bin Abdulaziz University project number (PSAU/2026/R/1447).

Author Contributions: Study conception and design: Muhammad Adeel Ahmed Tahir, Adnan Ahmed Rafique and Maha Abdelhaq; data collection: Muhammad Adeel Ahmed Tahir, Raed Alsaqour and Mohammed Alonazi; analysis and interpretation of results: Maha Abdelhaq, Raed Alsaqour and Ahmad Jalal; draft manuscript preparation: Mohammed Alonazi, Maha Abdelhaq, Jeongmin Park and Ahmad Jalal. All authors reviewed and approved the final version of the manuscript.

Availability of Data and Materials: All publicly available datasets are used in the study.

Ethics Approval: Not applicable.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Ganin Y, Ustinova E, Ajakan H, Germain P, Larochelle H, Laviolette F, et al. Domain-adversarial training of neural networks. *J Mach Learn Res.* 2016;17(59):1–35.
2. Redmon J, Divvala S, Girshick R, Farhadi A. You only look once: unified, real-time object detection. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*; 2016 Jun 27–30; Las Vegas, NV, USA. p. 779–88. doi:10.1109/CVPR.2016.91.
3. Lin TY, Goyal P, Girshick R, He K, Dollár P. Focal loss for dense object detection. *IEEE Trans Pattern Anal Mach Intell.* 2020;42(2):318–27. doi:10.1109/TPAMI.2018.2858826.
4. Zhang Y, Sun P, Jiang Y, Yu D, Weng F, Yuan Z, et al. ByteTrack: multi-object tracking by associating every detection box. In: *Proceedings of the European Conference on Computer Vision (ECCV)*; 2022 Oct 23–27; Tel Aviv, Israel. p. 1–21. doi:10.1007/978-3-031-20047-2_1.
5. Cordts M, Omran M, Ramos S, Rehfeld T, Enzweiler M, Benenson R, et al. The cityscapes dataset for semantic urban scene understanding. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*; 2016 Jun 27–30; Las Vegas, NV, USA. p. 3213–23. doi:10.1109/CVPR.2016.350.
6. Geiger A, Lenz P, Urtasun R. Are we ready for autonomous driving? The KITTI vision benchmark suite. In: *Proceedings of the 2012 IEEE Conference on Computer Vision and Pattern Recognition*; 2012 Jun 16–21; Providence, RI, USA. p. 3354–61. doi:10.1109/CVPR.2012.6248074.
7. Ren S, He K, Girshick R, Sun J. Faster R-CNN: towards real-time object detection with region proposal networks. *IEEE Trans Pattern Anal Mach Intell.* 2017;39(6):1137–49. doi:10.1109/TPAMI.2016.2577031.
8. Liu W, Anguelov D, Erhan D, Szegedy C, Reed S, Fu CY, et al. SSD: single shot MultiBox detector. In: *Proceedings of the European Conference on Computer Vision (ECCV)*; 2016 Oct 11–14; Amsterdam, The Netherlands. p. 21–37. doi:10.1007/978-3-319-46448-0_2.

9. Jocher G, Chaurasia A, Qiu J. Ultralytics YOLOv8. Computer software. 2023 [cited 2026 Jul 14]. Available from: <https://github.com/ultralytics/ultralytics>.
10. Wang CY, Yeh IH, Mark Liao HY. YOLOv9: learning what you want to learn using programmable gradient information. In: Proceedings of the European Conference on Computer Vision (ECCV); 2024 Sep 29–Oct 4; Milan, Italy. p. 1–21. doi:10.1007/978-3-031-72751-1_1.
11. Wang A, Chen H, Liu L, Chen K, Lin Z, Han J, et al. YOLOv10: real-time end-to-end object detection. *Adv Neural Inf Process Syst*. 2024;37:107984–8011. doi:10.52202/079017-3429.
12. Zhao Y, Lv W, Xu S, Wei J, Wang G, Dang Q, et al. DETRs beat YOLOs on real-time object detection. In: Proceedings of the 2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR); 2024 Jun 16–22; Seattle, WA, USA. p. 16965–74. doi:10.1109/CVPR52733.2024.01605.
13. Neuhold G, Ollmann T, Bulò SR, Kotschieder P. The Mapillary Vistas dataset for semantic understanding of street scenes. In: Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV); 2017 Oct 22–29; Venice, Italy. p. 5000–9. doi:10.1109/ICCV.2017.534.
14. Caesar H, Bankiti V, Lang AH, Vora S, Liong VE, Xu Q, et al. nuScenes: a multimodal dataset for autonomous driving. In: Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR); 2020 Jun 13–19; Seattle, WA, USA. p. 11618–28. doi:10.1109/cvpr42600.2020.01164.
15. Bewley A, Ge Z, Ott L, Ramos F, Upcroft B. Simple online and realtime tracking. In: Proceedings of the 2016 IEEE International Conference on Image Processing (ICIP); 2016 Sep 25–28; Phoenix, AZ, USA. p. 3464–8. doi:10.1109/ICIP.2016.7533003.
16. Wojke N, Bewley A, Paulus D. Simple online and realtime tracking with a deep association metric. In: Proceedings of the 2017 IEEE International Conference on Image Processing (ICIP); 2017 Sep 17–20; Beijing, China. p. 3645–9. doi:10.1109/ICIP.2017.8296962.
17. Cao J, Pang J, Weng X, Khirodkar R, Kitani K. Observation-centric SORT: rethinking SORT for robust multi-object tracking. In: Proceedings of the 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR); 2023 Jun 17–24; Vancouver, BC, Canada. p. 9686–96. doi:10.1109/CVPR52729.2023.00934.
18. Aharon N, Orfaig R, Bobrovsky BZ. BoT-SORT: robust associations multi-pedestrian tracking. *arXiv:2206.14651*. 2022. doi:10.48550/arXiv.2206.14651.
19. Du Y, Zhao Z, Song Y, Zhao Y, Su F, Gong T, et al. StrongSORT: make DeepSORT great again. *IEEE Trans Multimed*. 2023;25:8725–37. doi:10.1109/TMM.2023.3240881.
20. Long M, Cao Y, Wang J, Jordan MI. Learning transferable features with deep adaptation networks. In: Proceedings of the 32nd International Conference on Machine Learning (ICML); 2015 Jul 6–11; Lille, France. p. 97–105.
21. Tzeng E, Hoffman J, Saenko K, Darrell T. Adversarial discriminative domain adaptation. In: Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR); 2017 Jul 21–26; Honolulu, HI, USA. p. 2962–71. doi:10.1109/CVPR.2017.316.
22. Hoffman J, Tzeng E, Park T, Zhu JY, Isola P, Saenko K, et al. CyCADA: cycle-consistent adversarial domain adaptation. In: Proceedings of the 35th International Conference on Machine Learning (ICML); 2018 Jul 10–15; Stockholm, Sweden. p. 1989–98.
23. Kalman RE. A new approach to linear filtering and prediction problems. *J Basic Eng*. 1960;82(1):35–45. doi:10.1115/1.3662552.
24. Barcellos P, Bouvié C, Escouto FL, Scharcanski J. A novel video based system for detecting and counting vehicles at user-defined virtual loops. *Expert Syst Appl*. 2015;42(4):1845–56. doi:10.1016/j.eswa.2014.09.045.
25. Bui KHN, Yi H, Cho J. A multi-class multi-movement vehicle counting framework for traffic analysis in complex areas using CCTV systems. *Energies*. 2020;13(8):2036. doi:10.3390/en13082036.
26. Song H, Liang H, Li H, Dai Z, Yun X. Vision-based vehicle detection and counting system using deep learning in highway scenes. *Eur Transp Res Rev*. 2019;11(1):51. doi:10.1186/s12544-019-0390-4.
27. Du J, Cui J, Ren G, Thompson RG. Recognizing the traffic state of urban road networks: a resilience-based data-driven approach. *Transp Res Rec*. 2025;2679(6):183–202. doi:10.1177/03611981241312914.
28. Pizer SM, Amburn EP, Austin JD, Cromartie R, Geselowitz A, Greer T, et al. Adaptive histogram equalization and its variations. *Comput Vis Graph Image Process*. 1987;39(3):355–68. doi:10.1016/S0734-189X(87)80186-X.

29. Gonzalez RC, Woods RE. Digital image processing. 4th ed. London, UK: Pearson; 2017.
30. Ganin Y, Lempitsky V. Unsupervised domain adaptation by backpropagation. In: Proceedings of the 32nd International Conference on Machine Learning; 2015 Jul 6–11; Lille, France. p. 1180–9. doi:10.5555/3045118.3045244.
31. Wang CY, Bochkovskiy A, Liao HM. YOLOv7: trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. In: Proceedings of the 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR); 2023 Jun 17–24; Vancouver, BC, Canada. p. 7464–75. doi:10.1109/CVPR52729.2023.00721.
32. Lin TY, Dollár P, Girshick R, He K, Hariharan B, Belongie S. Feature pyramid networks for object detection. In: Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR); 2017 Jul 21–26; Honolulu, HI, USA. p. 936–44. doi:10.1109/CVPR.2017.106.
33. Liu S, Qi L, Qin H, Shi J, Jia J. Path aggregation network for instance segmentation. In: Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition; 2018 Jun 18–23; Salt Lake City, UT, USA. p. 8759–68. doi:10.1109/CVPR.2018.00913.
34. Zheng L, Yang Y, Hauptmann AG. Person re-identification: past, present and future. arXiv:1610.02984. 2016.
35. Ristani E, Solera F, Zou R, Cucchiara R, Tomasi C. Performance measures and a data set for multi-target, multi-camera tracking. In: Proceedings of the European Conference on Computer Vision; 2016 Oct 8–16; Amsterdam, The Netherlands. p. 17–35. doi:10.1007/978-3-319-48881-3_2.
36. Bergmann P, Meinhardt T, Leal-Taixe L. Tracking without bells and whistles. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV); 2019 Oct 27–Nov 2; Seoul, Republic of Korea. p. 941–51. doi:10.1109/iccv.2019.00103.
37. Zhang Y, Wang C, Wang X, Zeng W, Liu W. FairMOT: on the fairness of detection and re-identification in multiple object tracking. *Int J Comput Vis.* 2021;129(11):3069–87. doi:10.1007/s11263-021-01513-4.
38. Yu F, Chen H, Wang X, Xian W, Chen Y, Liu F, et al. BDD100K: a diverse driving dataset for heterogeneous multitask learning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR); 2020 Jun 14–19; Seattle, WA, USA. p. 2636–45. doi:10.1109/CVPR42600.2020.00271.
39. Sun P, Kretschmar H, Dotiwalla X, Chouard A, Patnaik V, Tsui P, et al. Scalability in perception for autonomous driving: Waymo open dataset. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR); 2020 Jun 14–19; Seattle, WA, USA. p. 2446–54. doi:10.1109/CVPR42600.2020.00252.