A Unified Framework for Camera-based Automated Traffic Analysis in Unstructured Environments

Vivek Kumar Paswan
School of Computer and Systems Sciences
Jawaharlal Nehru Univeristy
New Delhi, India

Ayesha Choudhary
School of Computer and Systems Sciences
Jawaharlal Nehru Univeristy
New Delhi, India

ABSTRACT

In this paper, we propose a camera-based unified framework for automated traffic analysis in unstructured environments. The proposed framework utilizes computer vision and deep learning-based algorithms on the camera feed of an infrastructure camera observing a traffic scene. Detection and tracking of vehicles infer information regarding their spatial position over time. This information is further utilized to perform crucial tasks such as class-wise and direction-wise vehicle detection and counting, traffic density and volume estimation, and the detection of wrongly parked vehicles as well as wrong-side driving. The experimentation leverages an real-world traffic dataset of an unstructured driving environment to show the efficacy of the proposed system. The insights derived from this framework are vital for understanding complex traffic patterns, enabling informed optimization of traffic management strategies, and ultimately enhancing road safety in challenging, real-world conditions.

Keywords

Traffic Analysis; Traffic Volume; Wrongly Parked Vehicle; Wrong Direction Driving; Road Safety; Intelligent Transportation Systems.

1. INTRODUCTION

The acceleration of urbanization and the rapid growth of urban population places immense pressure on urban traffic management systems worldwide, especially in low-to-middle income countries(LMIC). This rapid urbanization has sharply increased the number of vehicles on the road, resulting in a highly heterogeneous mix including two-wheelers, cars, trucks, bus and non-standard vehicles such as vendor carts and auto-rickshaws. In LMICs, this vehicle diversity has led to unpredictable and erratic traffic behavior, increasing the risk of road accidents. Adding to this complexity, unstructured traffic environments present additional challenges due to the absence of lane discipline and the ubiquitous presence of anomalies, pedestrians, animals, etc. on the road. Parking remains a major concern, as roadside parking is a widespread practice that obstructs the drivable area. Static vehicles parked along the road force moving vehicles to brake suddenly or change lanes, increasing the risk of accidents and hindering overall traffic flow. Road users like pedestrians and cyclists become highly vulnerable when they are compelled to walk on the roadway because of encroached sidewalks or obstructed road edges, as shown in Figure 1.


(a) (b)

Fig. 1: Traffic congestion caused by static vehicles wrongly parked on both sides of the road [29].

To effectively address these multifaceted challenges, Intelligent Transportation System (ITS) has become an indispensable part of smart city initiatives. It aims to enhance transportation safety, efficiency, and sustainability through real-time data and actionable insights for drivers, operators, and planners, thereby fostering adaptive and responsive urban mobility. A foundational component of any effective ITS is traffic analysis, which involves the systematic collection, processing, and interpretation of data related to vehicular traffic movement. However, the traditional reliance on manual observation for traffic analysis is increasingly unsustainable. It is difficult to provide objective, uninterrupted data across vast and complex traffic scenes through manual observations, making it impossible to capture the dynamic, real-time nuances of vehicle behavior and flow comprehensively. Consequently, camera-based automated traffic analysis has emerged as a critical requirement for effective traffic management. These advanced systems overcome manual shortcomings by offering continuous, objective data collection, processing and analysis at large scale. The accurate, real-time insights generated by these automated frameworks are indispensable: they can dynamically guide infrastructure planning, enable highly responsive adaptive traffic control systems to optimize flow and reduce congestion, and robustly support enforcement mechanisms by precisely identifying violations, all of which are essential for creating safer, more efficient, and more sustainable urban mobility.

In this paper, we present a unified traffic analysis framework that takes as input a continuous video stream of the traffic from an infrastructure camera, detects and tracks the vehicles does the automated analyses for multiple sub-tasks and finally, outputs these for a holistic analysis of the area under observation. The insights derived from this analysis are crucial for optimizing traffic signal timings, designing safer road infrastructure and enhancing overall transportation system efficiency and reliability. Traffic analysis is foundational to numerous real-world applications; for instance, it enhances road safety by discovering accident hotspots, optimizes infrastructure use through dynamic lane control, and aids urban planning by predicting growth. This analysis also significantly improves emergency response and commercial logistics via efficient routing.

The unique characteristics of unstructured traffic pose significant challenges that generic traffic analysis solutions often fail to address. These environments, with their diverse mix of variety of vehicles and unpredictable behaviors, demand locality-specific solutions that are adaptable and cost-effective. The inherent ubiquity and relatively low cost of camera-based surveillance infrastructure make it an ideal foundation for developing specialized traffic analysis frameworks that can provide the necessary granular insights without requiring prohibitive investments in new, expensive sensing technologies. This further motivates the need for robust, vision-based solutions capable of accurately interpreting complex, unstructured traffic scenes. However, camera based methods face various challenges based on camera placement, occlusion, perspective distortion, diverse weather conditions such as fog, rain, snow, illumination variation, time of day, as well as lack of depth information in uncalibrated cameras. Unstructured traffic brings its own challenges for automated analysis since lane based driving is absent, and the driving rules are not strictly followed leading to wrong direction driving and parking of vehicles on roadsides.

Traffic analysis in such unstructured environments, involves several key components that help in understanding the flow and behavior of vehicles. It is therefore, important that the traffic management system not only measures the traffic volume and density to understand the possible congestion, but also it is important to find which vehicles are static, which are dynamic and the number of vehicles moving in the wrong direction. Traffic volume is the total number of vehicles passing a specific point or section of a road within a defined time interval and reflects overall vehicular flow. On the other hand, traffic density is the number of vehicles occupying a unit length of road at a given time and therefore, it helps assess how closely spaced vehicles are on the road. Therefore, both volume and density influence the traffic situation on a road and congestion arises when traffic demand exceeds roadway capacity, leading to slower speeds, increased delays, and unstable flow. It is therefore, important to estimate both the density and volume of traffic to analyze the causes of congestion and take appropriate measures to avoid congestion.

The proposed framework leverages static infrastructure-mounted cameras to continuously capture real-world traffic scenes. Vehicle detection and tracking serve as the foundational steps for extracting precise spatial and temporal movement patterns of vehicles. Based on these fundamental observations, the proposed framework performs the following crucial tasks: (a) class-wise and direction-wise detection and counting of vehicles, (b) estimation of traffic density and volume, (c) detection of static vehicles that are wrongly parked on the roadside, and (d) etection of vehicles moving in the wrong direction

2. RELATED WORK

The continuous improvements in camera sensors, hardware acceleration, and artificial intelligence techniques have significantly transformed traffic related task such as vehicle detection, tracking, speed estimation, flow forecasting, and traffic violation detection. In this section, related works addressing these key tasks for traffic analysis and monitoring have been presented.

A method for spatio-temporal congestion detection was introduced in [11], where Multi-Channel Singular Spectrum Analysis (MSSA) is used to analyze real-time traffic sensor data. The technique captures spatial and temporal patterns to identify contextual anomalies. Another example is presented in [5], where traffic flow is modeled as a 3D tensor based on time slots, vehicle types, and spatial zones. Homography-based methods [20] and classical computer vision techniques such as background subtraction, and feature extraction [19, 10] have been widely adopted for vehicle localization and classification. A DBSCAN-based Adaptive Traffic Management (ATM) framework is demonstrated in [17], utilizing IoT devices to dynamically adjust signal timings based on traffic volume and intersections.

Deep learning based techniques for object detection [8, 28, 4] and multi-object tracking such as SORT [3], DeepSORT [30], and attention-based methods [27] have enhanced real-time performance in traffic monitoring. A YOLOv3-based system is proposed in [1], which tracks vehicle centroids to monitor intersection traffic. Although effective, the method performs best in controlled settings. Domain adaptation for nighttime vehicle detection is addressed in [16]. The authors use Faster R-CNN trained on daytime-labeled images, improving detection across lighting conditions using two large datasets with over 57,000 labeled vehicles and 7,200 frames.

A UAV-based system, MultEYE [2], integrates object detection and segmentation for real-time speed estimation from aerial views. Designed for embedded platforms, it achieves high accuracy and efficiency, although field validation is limited. Legacy surveillance cameras are used in [18] to extract vehicle metadata. The system uses transfer learning and monocular homography for real-world measurement, achieving 60 FPS. Calibration remains a challenge when camera positions shift. A real-time IoT-optimized vehicle counting system is introduced in [25], utilizing high-recall detection and cosine similarity-based trajectory analysis to improve counting accuracy. Wrong-way driving detection is discussed in [24], where the RLB-CCTV and MBCDD algorithms utilize lane boundary detection and motion direction analysis. Wrongly parked vehicle detection is further explored in [7, 12, 6, 23], combining deep learning with handcrafted features and crowd sourced video data. Multi-camera tracking is tackled in [31] through an integrated system combining single-camera tracking, re-identification (Re-ID), and spatial-temporal cues. The model ranks highly in the AI City Challenge but shows limitations under occlusion and ambiguity. A two-stream convolutional architecture is presented in [13], combining spatial appearance and temporal motion features. The model performs real-time detection, tracking, and near-accident identification across various camera angles, including fisheye and overhead views.

A substantial body of research has addressed individual tasks such as vehicle detection, tracking, speed estimation, and traffic rule violation detection. However, these efforts are largely fragmented, with most studies focusing on isolated objectives rather than a holistic approach. This highlights a clear research gap and the need for a unified traffic analysis framework capable of performing multiple tasks simultaneously to enable comprehensive and real time traffic monitoring.

3. PROPOSED WORK

In this work, a camera based unified framework has been presented which takes the visual inputs of an unstructured traffic scene and performs the vehicle detection and tracking tasks in real time. The results obtained from the detection and tracking module are further processed for automated traffic analysis task including class-wise and direction-wise vehicle counts, traffic volume and density estimation. It also detects wrongly parked (static) vehicles as well as vehicles driving in wrong directions. The architecture of the proposed framework is shown in Figure 2 encompassing several deep learning based modules which are being discussed below:

3.1 Input Data Acquisition

The framework relies on input from a traffic scene obtained from an infrastructure camera strategically positioned on a particular place of the road at a pole. It assumes that the camera captures the unstructured traffic scene, providing continuous video streams that serve as the basis for automated real-time traffic analysis.

3.2 Vehicle Detection

The proposed framework utilizes YOLOv11 [15] model for vehicle detection task, which is a recently released version of the You Only Look Once(YOLO) family. YOLOv11 model detect these vehicles and provides information as the class of vehicle, bounding box and confidence score. Its architecture can be divided into three main categories: The backbone network, tasked with feature extraction from input images, employs CSPDarknet53, a variant of the Darknet architecture. It integrates Cross-Stage Partial (CSP) connections to facilitate better information flow across layers. This architectural choice enhances model accuracy while reducing computational demands, thus ensuring efficient operation.
The neck component is responsible for integrating features extracted from various levels of the backbone network. Rather than adopting conventional Feature Pyramid Network (FPN), it utilizes C3k2 module for feature fusion. It efficiently fuses high-level semantic features with low-level spatial details while keeping the parameter count low. Such a design is particularly advantageous for accurately detecting smaller objects.
The head of the network is where final predictions are generated. It employs multiple detection branches to estimate bounding boxes, objectness scores, and class probabilities corresponding to different regions of the input image. The detection head incorporates advancements like dynamic anchor assignment and a new Intersection over Union (IoU) loss function, which together enhance bounding box prediction accuracy and improve the model's capability to manage overlapping objects. There are some other features to improve the performance such as spatial attention, a mechanism designed to guide the network's attention toward the most informative regions of the image, enabling more accurate object localization. Moreover, the backbone incorporates bottleneck layers that decrease computational complexity while maintaining accuracy. The Spatial Pyramid Pooling Fast (SPPF) layer further improves detection performance by extracting features across multiple scales, particularly benefiting the recognition of objects with diverse sizes.

3.3 Vehicle Tracking

The bounding box details of detected vehicles in the traffic scene are subsequently passed to ByteTrack [33] for the tracking task. ByteTrack assigns a unique tracking ID to each detected vehicle across consecutive frames. Conventional methods generally assign object identities by linking detection boxes that have confidence scores exceeding a predefined threshold. This often results in the exclusion of low-confidence detections, such as occluded objects, causing missed true detections and broken tracking paths. ByteTrack addresses this issue through a simple yet effective association strategy that considers almost all detection boxes rather than relying solely on high-confidence ones. Low-confidence detections are compared with existing tracklets in ByteTrack to recover true objects while suppressing background noise. This approach efficiently utilizes detection outputs to improve multi-object tracking accuracy. ByteTrack has demonstrated state-of-the-art performance on MOT2017 [21] and BDD100K [32], a large-scale dataset designed for multi-category vehicle tracking. Due to its high accuracy, computational efficiency, and simplicity, ByteTrack has been integrated into the framework for vehicle tracking.

3.4 Vehicle Motion Direction Detection

To determine the direction of motion of each detected vehicle, the framework first calculate the center point $P(x, y)$ of the bounding box of the detected vehicle, where:

$$x = \frac{x_1 + x_2}{2}, \quad y = \frac{y_1 + y_2}{2} \tag{1}$$

In this context, (x_1, y_1) and (x_2, y_2) denote the top-left and bottom-right coordinates of the bounding of the detected vehicle. The computed point $P(x, y)$ serves to evaluate whether the vehicle is positioned within any of the predefined zones based on traffic direction rules given as Equation 2.

$$P(x, y) \in \begin{cases} Z_R, & \text{Right side} \\ Z_L, & \text{Left side} \end{cases} \tag{2}$$

Here, Z_R and Z_L denote polygonal zones corresponding to the right and left sides of the road, which are based on predefined traffic rules.
It consider Left-Hand Side (LHS) driving in according to the given data and therefore, the directional interpretation with respect to the camera capturing the traffic scene is as given by Equation 3.

$$M_D = \begin{cases} \text{Outgoing}, & \text{if } P(x, y) \in Z_L \\ \text{Incoming}, & \text{if } P(x, y) \in Z_R \end{cases} \tag{3}$$

Thus, a vehicle detected in the left zone (Z_L) is considered to be moving away from the camera (Outgoing), while one in the right zone (Z_R) is considered to be moving towards the camera (Incoming).

3.5 Wrongly Parked Vehicle Detection

Continuous tracking of vehicle ensures precise detection of the static and moving vehicles in traffic scene. The system identifies and counts static vehicles based on a predefined threshold for vehicle displacement, given in Equation 5, over a specified number of frames. Vehicles with displacement exceeding this threshold are classified as moving, whereas those vehicle with insignificant displacement below this defined threshold are considered static vehicle, i.e., wrongly parked on road. This approach facilitates proactive interventions, improving overall road safety and mobility.
Let S and E represent the start and end positions of the vehicle trajectory, and D is the displacement given as:

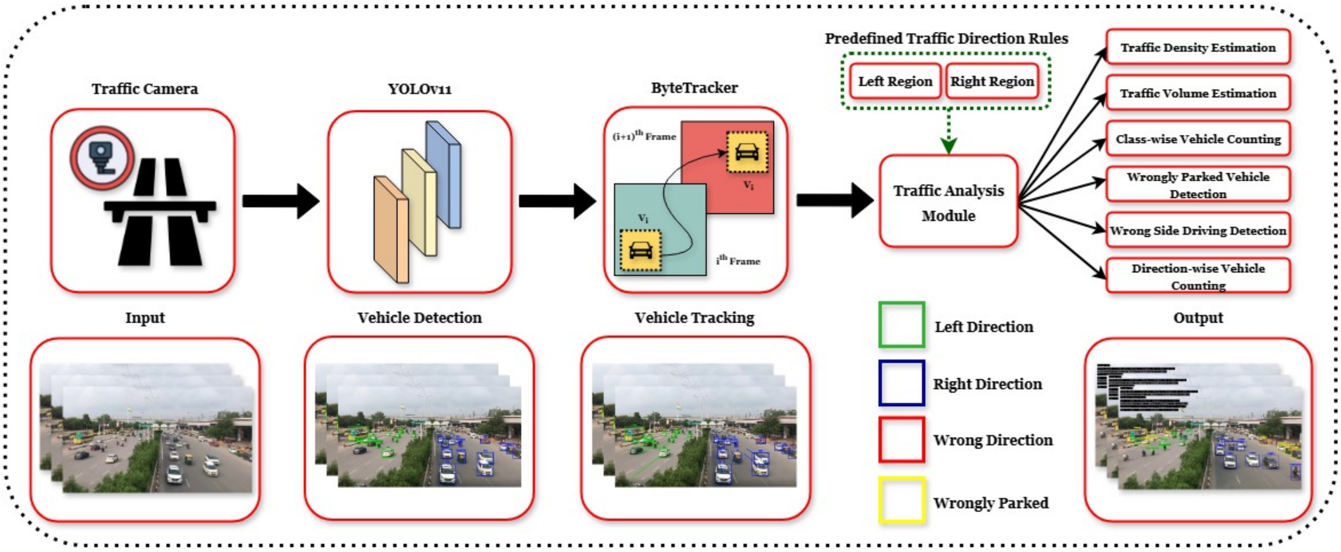

Fig. 2: Proposed Framework of Automated Traffic Analysis

$$D = |S - E| \quad (4)$$

A vehicle is considered Static if:

$$D \leq \delta \quad (5)$$

where δ is the displacement threshold to differentiate static and moving vehicles.

3.6 Wrong Direction Driving Detection

There are certain rules governing vehicle movement to ensure smooth and safe traffic flow where traffic regulations mandate that vehicles must drive on either left or right side of the road. To identify traffic violations in this regard, the proposed framework tracks vehicle movement and detects instances of wrong-side driving. It takes input of the pre-define the left side traffic flow region and right side traffic flow region in the scene, and vehicle trajectories are analyzed to determine their direction of motion. If a vehicle is detected moving against the designated traffic flow, it is classified as a wrong-direction driving and counted accordingly, the methods is defined below. The implementation of this methodology facilitates proactive traffic management, helping authorities mitigate traffic safety risks.

Let V_i denote the i-th vehicle, with S_i and E_i representing its starting and ending central coordinate positions in the scene, respectively. The traffic scene is divided into two regions: the right-side region Z_R and the left-side region Z_L. The current position of vehicle V_i is denoted by P_i, such that $P_i \in Z_R$ or $P_i \in Z_L$.

If $P_i \in Z_R$:

$$S_i > E_i \Rightarrow V_i \text{ is in Wrong Direction}$$
$$S_i < E_i \Rightarrow V_i \text{ is in Correct Direction}$$

If $P_i \in Z_L$:

$$S_i < E_i \Rightarrow V_i \text{ is in Wrong Direction}$$
$$S_i > E_i \Rightarrow V_i \text{ is in Correct Direction}$$

3.7 Vehicle Counting

The proposed framework first identifies the moving direction of the vehicle based on the tracking details of vehicles as well as predefined directional zones in the scene and then counts the number of vehicles running in both directions, wrongly parked vehicles, as well as numbers of vehicles running in wrong direction. The proposed framework also provides this information for each vehicle class separately. The method used for performing this task is given in Algorithm 1

3.8 Traffic Volume Estimation

The information regarding the vehicle class type provides the facility to analyze the traffic volume using the Passenger Car Unit (PCU) definition [14]. Under mixed traffic conditions, accurately determining the traffic volume is challenging due to the varied sizes of different vehicle on the road. For uniformity in traffic data analysis, different vehicle types are converted into a standardized measure known as the PCU. The PCU indicates the number of passenger cars that would exert an equivalent operational effect as one heavy vehicle type, considering specific road, traffic, and control scenarios. This enables a consistent comparison of traffic flow among various vehicle categories. Table 1 lists the PCU conversion values.

The proposed framework calculates the traffic volume (Q) in terms of PCUs when the vehicles cross a given section in Region of Interest (RoI) of the traffic scene based on the detected vehicle class and the PCU conversion table 1 using the formulas given in equations 6 and 7.

Traffic volume in right side:

$$Q_{in} = \sum_{i=1}^{N} (R_{in,i} \times \text{PCU}_i) \quad (6)$$

Algorithm 1 Vehicle Counting Method

Require: For each vehicle V_i: position P_i, start coordinate S_i, end coordinate E_i, distance D_i; Threshold δ, regions Z_R (right), Z_L (left)
Ensure: Total vehicles in left region N_{Left}, total vehicles in right region N_{Right}

```
Initialize counters: R_stop ← 0, R_in ← 0, R_wrong ← 0
Initialize counters: L_stop ← 0, L_out ← 0, L_wrong ← 0
for each vehicle V_i do
    if P_i ∈ Z_R then
        if D_i ≤ δ then
            R_stop ← R_stop + 1
        else if S_i > E_i then
            R_wrong ← R_wrong + 1
        else if S_i < E_i then
            R_in ← R_in + 1
        end if
    else if P_i ∈ Z_L then
        if D_i ≤ δ then
            L_stop ← L_stop + 1
        else if S_i < E_i then
            L_wrong ← L_wrong + 1
        else if S_i > E_i then
            L_out ← L_out + 1
        end if
    end if
end for
N_Right ← R_stop + R_in + R_wrong
N_Left ← L_stop + L_out + L_wrong
```

Vehicle Class	Passenger Car Unit Value
Car or Autorikshaw	1
Bus or Truck	3
Motorcycle	0.5
Cycle Rickshaw	1.5

Table 1. : Passenger Car Units for Different Vehicle Classes [14].

Similarly, traffic volume in left side:

$$Q_{out} = \sum_{i=1}^{N} (L_{out,i} \times \text{PCU}_i) \tag{7}$$

Here, N denotes the total number of vehicle categories detected; $R_{in,i}$ and $L_{out,i}$ represent the number of vehicles of type i correctly moving in the right and left directions, respectively. The term PCU_i corresponds to the PCU value assigned to vehicle type i, as specified in Table 1.
The above equation allows the traffic volume to be expressed as the cumulative PCU-weighted count of each vehicle type, giving a standardized measure of traffic volume based on diverse vehicle classes.

3.9 Traffic Density Estimation

The proposed framework also calculates the traffic density (K) for both sides of given directional regions in terms of PCU values using the table 1 for each detected vehicle in the respective region at a specific time. The method for calculating the traffic density is given below:

traffic density in right side:

$$K_{in} = \sum_{i=1}^{N} (N_{Right,i} \times \text{PCU}_i) \tag{8}$$

Similarly, traffic density in left side:

$$K_{out} = \sum_{i=1}^{N} (N_{Left,i} \times \text{PCU}_i) \tag{9}$$

Here, N denotes the total number of vehicle categories detected; $N_{Right,i}$ and $N_{Left,i}$ represent the number of vehicles of type i in the right and left regions, respectively. The term PCU_i corresponds to the PCU value assigned to vehicle type i, as specified in Table 1.
The information regarding the traffic volume and traffic density is essential for evaluating road networks and enhancing traffic management. It helps prioritize road improvements and expansions, aiding in traffic operations planning, facility control, and infrastructure development.
For better visualization, detected vehicles are annotated with bounding boxes displaying their tracker ID and class name. Further the color of the bounding box displays their status as:

—**Red**: Vehicles driving in wrong direction.
—**Yellow**: Wrongly parked static vehicles.
—**Blue**: Vehicles driving in right direction (incoming).
—**Green**: Vehicles driving in left direction (outgoing).

This categorization helps identify vehicles violating traffic rules, enabling proactive traffic management and road safety measures. The overall insights derived from the traffic analysis tasks support in decision making to facilitate safe, smooth and efficient traffic flow.

4. RESULT AND DISCUSSION

4.1 Experimental Environment

All experiments were conducted using Google Colaboratory [9], a virtual and interactive Jupyter Notebook environment that offers free GPU resources. In particular, we utilized the NVIDIA Tesla T4 GPU with 16GB of memory available on Google Colab to implement and train the proposed framework. This platform facilitates efficient and cost-effective data storage, dataset uploading, and result retrieval within its virtual environment.

4.2 Dataset Description

In this research, the Indian Driving Dataset (IDD) [26] has been utilized which is a comprehensive dataset specifically designed to support Intelligent Transportaion Systems (ITS) realted research in unstructured traffic environments. The dataset comprises of finely annotated 34 distinct classes collected from 182 drive sequences from urban and semi-urban locations around Hyderabad, Bangalore, and their surrounding areas.
The proposed work is focused on six vehicle-related categories from the total 34 classes: motorcycle, car, truck, bus, and autorickshaw. The dataset includes a total of 12,535 labeled images along with corresponding annotations divided into training, validation, and test subsets in the ratio of 70%, 20%, and 10%, respectively. This split ensures a balanced and representative distribution across the entire dataset that are resilient to the variations and complexities inherent in unstructured traffic environments., supporting robust model training and performance evaluation. The AI City Challenge

dataset [22], along with another publicly available dataset [29], has been further utilized to validate the robustness of the proposed framework under unseen traffic scenarios.

4.3 Model Training

The YOLOv11 model is trained on the IDD dataset for vehicle detection over 300 epochs. The model demonstrated steady improvement, ultimately achieving a mean Average Precision (mAP50) of 60% at a confidence threshold of 50%. Figure 3 illustrates the training and validation loss trends over the course of training, highlighting the model's learning progression. The gradual reduction in both training and validation loss indicates effective optimization and improved prediction accuracy.
In addition, Figure 4 shows the evolution of the mAP50 accuracy of YOLOv11, compared alongside other YOLO model variants. The figure clearly reflects the progressive enhancement in detection performance across epochs. A comparative summary of performance metrics: Precision, Recall, and Accuracy (mAP50) of YOLOv11 and other variants is provided in Table 2, emphasizing YOLOv11's superior detection capability.

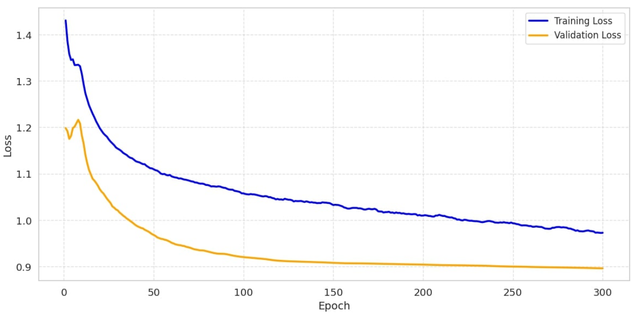

Fig. 3: YOLOv11 model training and validation loss over the epochs

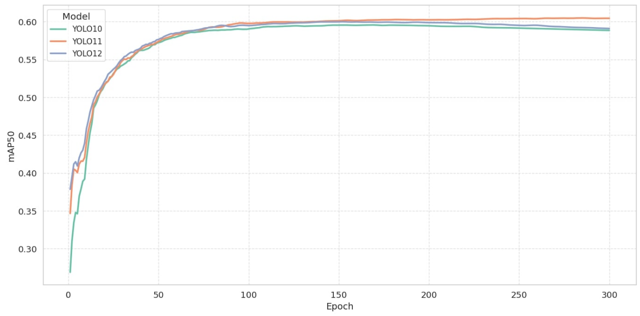

Fig. 4: Training accuracy comparison of YOLOv11 model with its other versions for vehicle detection on IDD dataset [26]

Model	Precision	Recall	Accuracy (mAP50)
YOLOv10	0.68	0.53	0.59
YOLOv11	**0.69**	**0.55**	**0.60**
YOLOv12	0.68	0.54	0.59

Table 2. : Performance Comparison of YOLO Models

4.4 Experimental Results

The framework assumes a scenario where a surveillance camera is strategically mounted at a fixed location along a road to continuously monitor the unstructured traffic environment. The video feed captured by this camera is used as input to the proposed automated traffic analysis framework. This framework is designed to process the input in real-time and automatically extract meaningful traffic insights. Specifically, it computes and reports key metrics such as class-wise and direction-wise vehicle counts, traffic volume, traffic density, detection of vehicles driving in the wrong direction as well as static (wrongly parked) vehicles.
The performance and applicability of the proposed system in real-world traffic scenarios are demonstrated through the traffic analysis results presented in Figure 5 and Figure 6. These visual results demonstrate how the framework accurately detects and categorizes various vehicle behaviors across both directions of traffic flow. The framework provides the statistics regarding the traffic density and traffic volume in both the directions. It also provides information regarding the tracking ID and class of each detected vehicle.

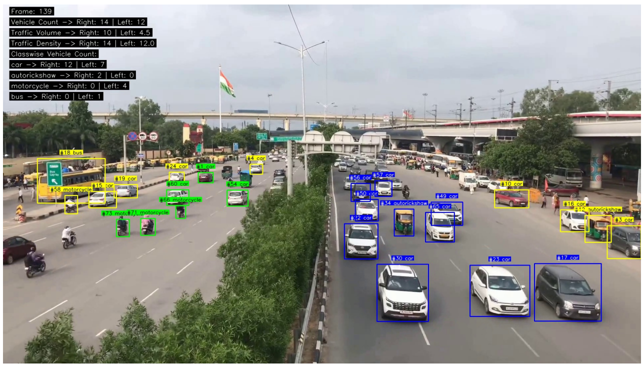

Fig. 5: Experimental results of the proposed automated traffic analysis framework. The outputs include direction-wise and class-wise vehicle counts, traffic density, and traffic volume. The framework employs color-coded bounding box annotations to highlight key traffic conditions: red for wrong-direction vehicles, yellow for wrongly parked(static) vehicles, blue for incoming vehicles (right side), and green for outgoing vehicles (left side). Each vehicle is labeled with its class and corresponding tracker ID. These visualizations assist in detecting traffic anomalies and contribute to improved road safety analysis.

Furthermore, the system is also capable of identifying traffic rule violations, such as vehicles that are wrongly parked on the road. These static vehicles often obstruct pedestrian movement, forcing people to walk in the middle of the road and thereby increasing the risk of accidents. The detection of such safety-critical scenarios is also integrated into the proposed system, as illustrated in Figure 7. This highlights the framework's potential for enhancing road safety and enabling proactive traffic enforcement in unstructured and congested urban environments.

4.5 Discussion

The proposed automated traffic analysis framework offers a robust solution for enhancing real-time surveillance and traffic management in complex, unstructured road environments providing granular insights into class-wise and direction-wise vehicle detection, overall traffic density, and traffic volume. A key advantage

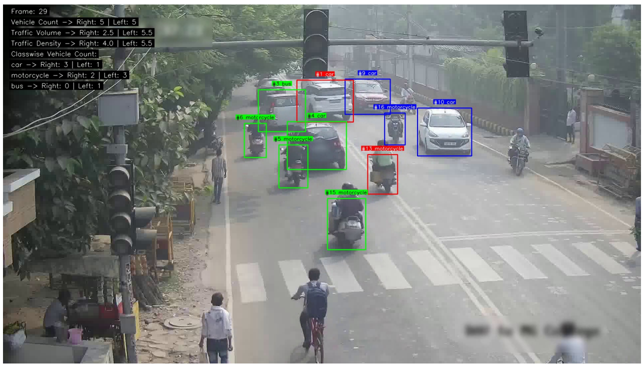

Fig. 6: In this experimental output, alongside other traffic analysis results, two vehicles moving in the wrong direction are successfully detected. They are highlighted with red-colored bounding boxes with tracker IDs 1 and 13, respectively.

Fig. 7: In this experimental output, static vehicles wrongly parked on the road are accurately detected and shown in yellow-coded bounding boxes. Their presence forces pedestrians to walk in the middle of the road, thereby increasing the risk of accidents and compromising pedestrian safety.

of the framework is its ability to generate interpretable outputs through visual annotations, where vehicles are consistently marked with color-coded bounding boxes, tracker IDs, and vehicle classes. These visual cues assist human operators in quickly assessing traffic situations and open avenues for integrating the system with alert mechanisms. Beyond standard traffic analytics, the framework places a strong emphasis on the detection of critical traffic violations allowing authorities to intervene promptly. The consequences of wrong parking and wrong direction driving are not limited to congestion; they often force pedestrians to walk on the road, increasing the risk of accidents.

The ability of the proposed framework to operate in real time is especially critical for unstructured traffic environments, where rapid changes in traffic volume and unpredictable driver behavior demand continuous monitoring. A detailed time analysis conducted on multiple video samples from the AI City Challenge [22] shows an average processing time of approximately 36 milliseconds per frame, corresponding to about 28 FPS. The vehicle detection and tracking module achieves an average of 29 FPS, while the traffic analysis module operates at over 500 FPS. These results validate the robustness of the proposed framework under complex traffic conditions, ensuring real-time performance suitable for intelligent transportation and traffic management applications.

5. CONCLUSION

In this work, a robust and efficient framework has been proposed for automated traffic analysis aimed at improving road safety and optimizing traffic management in unstructured environments. The proposed system integrates the YOLOv11 model for vehicle detection and ByteTrack for tracking, using camera-based inputs to monitor real-time traffic scenarios. It effectively performs direction-wise and class-wise vehicle counting and estimates traffic density and volume in PCU. Moreover, the system accurately identifies wrongly parked vehicles on the road, as well as vehicles moving in the wrong direction - both critical indicators of traffic violations and safety hazards.

A key advantage of the proposed approach is its reliance on a single fixed infrastructure camera, eliminating the need for expensive, sensor-heavy setups which makes it cost-effective, highly scalable and suitable for deployment in both urban and rural areas. Experimental results validate the reliability of the framework in analyzing traffic flow and detecting risky traffic behavior, thereby contributing significantly to safer and more efficient roads. In the future, multi-camera setups and predictive models utilizing historical data can be developed to forecast traffic congestion and potential accident hotspots.

6. REFERENCES

[1] J. Azimjonov, A. Özmen, and M. Varan. A Vision-Based Real-Time Traffic Flow Monitoring System for Road Intersections. *Multimedia Tools and Applications*, 82:25155–25174, 2023.

[2] Navaneeth Balamuralidhar, Sofia Tilon, and Francesco Nex. MultEYE: Monitoring System for Real-Time Vehicle Detection, Tracking and Speed Estimation from UAV Imagery on Edge-Computing Platforms. *Remote Sensing*, 13(4), 2021.

[3] A. Bewley, Z. Ge, L. Ott, F. Ramos, and B. Upcroft. Simple Online and Realtime Tracking. In *IEEE International Conference on Image Processing*, pages 3464–3468, 2016.

[4] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-End Object Detection with Transformers. In *Computer Vision – ECCV 2020*, pages 213–229, Cham, 2020. Springer International Publishing.

[5] Hui Gao, Zhi Wang, Zhen Yan, Zhi Yu, Wen Luo, and Lin Yuan. Synchronized Entry-Traffic Flow Prediction for Regional Expressway System Based on Multidimensional Tensor. *Transportation Research Record*, 2675(10):291–302, 2021. Original work published 2021.

[6] X. Gao, P. M. Birch, R. C. D. Young, and C. R. Chatwin. Illegally Parked Vehicle Detection Using Deep Learning and Key-point Tracking. In *9th International Conference on Imaging for Crime Detection and Prevention (ICDP)*, pages 7–12, 2019.

[7] Xing Gao, Phil Birch, Rupert Young, and Chris Chatwin. Illegally Parked Vehicle Detection Using Deep Learning and Key-Point Tracking. In *9th International Conference on Imaging for Crime Detection and Prevention (ICDP)*, pages 7–12. IET, 2019.

[8] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation. In *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pages 580–587, 2014.

[9] Google. Google Colaboratory.

[10] J. Guo, H. Lou, H. Chen, H. Liu, J. Gu, L. Bi, and X. Duan. A New Detection Algorithm for Alien Intrusion on Highway. *Scientific Reports*, 13(1):10667, 2023.

[11] Mai H. Hassan, Ali Tizghadam, and Alberto Leon-Garcia. Spatio-temporal Anomaly Detection in Intelligent Transportation Systems. *Procedia Computer Science*, 151:852–857, 2019. The 10th International Conference on Ambient Systems, Networks and Technologies (ANT 2019) / The 2nd International Conference on Emerging Data and Industry 4.0 (EDI40 2019) / Affiliated Workshops.

[12] Ruixue Huang, Lianghua Cheng, Zhenghan Li, Chaocan Xiang, and Yulan Guo. iPatrol: Illegal Roadside Parking Detection Leveraging On-road Vehicle Crowdsensing. In *2024 IEEE/ACM 32nd International Symposium on Quality of Service (IWQoS)*, pages 1–10, 2024.

[13] Xiaohui Huang, Pan He, Anand Rangarajan, and Sanjay Ranka. Intelligent Intersection: Two-stream Convolutional Networks for Real-time Near-accident Detection in Traffic Video. *ACM Trans. Spatial Algorithms Syst.*, 6(2), January 2020.

[14] Indian Roads Congress. *Guidelines for Capacity of Roads in Rural Areas*. The Indian Roads Congress, Jamnagar House, Shahjahan Road, New Delhi-110011, first revision edition, 1990. IRC: 64-1990. First Published: March 1976; First Revision: November 1990; Reprinted: August 2005, April 2007, July 2011.

[15] G. Jocher, A. Chaurasia, and J. Qiu. YOLOv11 by Ultralytics. `https://github.com/ultralytics/ultralytics`, 2024.

[16] Jinlong Li, Zhigang Xu, Lan Fu, Xuesong Zhou, and Hongkai Yu. Domain Adaptation from Daytime to Nighttime: A Situation-Sensitive Framework for Vehicle Detection and Traffic Flow Parameter Estimation. *Transportation Research Part C: Emerging Technologies*, 124:102946, 2021.

[17] Umesh Kumar Lilhore, Agbotiname Lucky Imoize, Chun-Ta Li, Sarita Simaiya, Subhendu Kumar Pani, Nitin Goyal, Arun Kumar, and Cheng-Chi Lee. Design and Implementation of an ML and IoT Based Adaptive Traffic-Management System for Smart Cities. *Sensors*, 22(8), 2022.

[18] Chenghuan Liu, Du Q. Huynh, Yuchao Sun, Mark Reynolds, and Steve Atkinson. A Vision-Based Pipeline for Vehicle Counting, Speed Estimation, and Classification. *IEEE Transactions on Intelligent Transportation Systems*, 22(12):7547–7560, 2021.

[19] D. C. Luvizon, B. T. Nassu, and R. Minetto. A Video-Based System for Vehicle Speed Measurement in Urban Roadways. *IEEE Transactions on Intelligent Transportation Systems*, 18(6):1393–1404, 2016.

[20] H. Mejia, E. J. Palomo, E. López-Rubio, I. Pineda, and R. Fonseca. Vehicle Speed Estimation Using Computer Vision and Evolutionary Camera Calibration. In *LatinX in AI at Neural Information Processing Systems Conference*, 2021.

[21] Anton Milan, Laura Leal-Taixé, Ian Reid, Stefan Roth, and Konrad Schindler. MOT16: A Benchmark for Multi-Object Tracking. *arXiv preprint arXiv:1603.00831*, 2016.

[22] Milind Naphade, Shuo Wang, David C. Anastasiu, Zheng Tang, Ming-Ching Chang, Yue Yao, Liang Zheng, Mohammed Shaiqur Rahman, Meenakshi S. Arya, Anuj Sharma, Qi Feng, Vitaly Ablavsky, Stan Sclaroff, Pranamesh Chakraborty, Sanjita Prajapati, Alice Li, Shangru Li, Krishna Kunadharaju, Shenxin Jiang, and Rama Chellappa. The 7th AI City Challenge. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2023.

[23] Xinggan Peng, Rongzihan Song, Qi Cao, Yue Li, Dongshun Cui, Xiaofan Jia, Zhiping Lin, and Guang-Bin Huang. Real-Time Illegal Parking Detection Algorithm in Urban Environments. *IEEE Transactions on Intelligent Transportation Systems*, 23(11):20572–20587, 2022.

[24] Pintusorn Suttiponpisarn, Chalermpol Charnsripinyo, Sasiporn Usanavasin, and Hiro Nakahara. An Enhanced System for Wrong-Way Driving Vehicle Detection with Road Boundary Detection Algorithm. *Procedia Computer Science*, 204:164–171, 2022. International Conference on Industry Sciences and Computer Science Innovation.

[25] Vu-Hoang Tran, Le-Hoai-Hieu Dang, Chinh-Nghiep Nguyen, Ngoc-Hoang-Lam Le, Khanh-Phong Bui, Lam-Truong Dam, Quang-Thang Le, and Dinh-Hiep Huynh. Real-time and Robust System for Counting Movement-Specific Vehicle at Crowded Intersections. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 4223–4230, 2021.

[26] Girish Varma, Anbumani Subramanian, Anoop Namboodiri, Manmohan Chandraker, and C.V. Jawahar. IDD: A Dataset for Exploring Problems of Autonomous Navigation in Unconstrained Environments. In *IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1743–1751, 2019.

[27] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is All You Need. In *31st International Conference on Neural Information Processing Systems*, NIPS'17, page 6000–6010, Red Hook, NY, USA, 2017. Curran Associates Inc.

[28] Ao Wang, Hui Chen, Lihao Liu, Kai Chen, Zijia Lin, Jungong Han, and Guiguang Ding. YOLOv10: Real-Time End-to-End Object Detection. *arXiv preprint arXiv:2405.14458*, 2024.

[29] WildFilmsIndia. New Delhi rush hour traffic at Yusuf Sarai. `https://www.youtube.com/watch?v=pdSDHlayBk4`, 2019. Video posted on August 17, 2019.

[30] N. Wojke, A. Bewley, and D. Paulus. Simple Online and Realtime Tracking with a Deep Association Metric. In *IEEE conference on computer vision and pattern recognition workshops*, pages 459–466, 2017.

[31] Minghu Wu, Yeqiang Qian, Chunxiang Wang, and Ming Yang. A Multi-Camera Vehicle Tracking System based on City-Scale Vehicle Re-ID and Spatial-Temporal Information. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 4072–4081, 2021.

[32] Fisher Yu, Haofeng Chen, Xin Wang, Wenqi Xian, Yingying Chen, Fangchen Liu, Vashisht Madhavan, and Trevor Darrell. BDD100K: A Diverse Driving Dataset for Heterogeneous Multitask Learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[33] Yifu Zhang, Peize Sun, Yi Jiang, Dongdong Yu, Fucheng Weng, Zehuan Yuan, Ping Luo, Wenyu Liu, and Xinggang Wang. ByteTrack: Multi-object Tracking by Associating Every Detection Box. In *Computer Vision – ECCV 2022*, pages 1–21, Cham, 2022. Springer Nature Switzerland.