







ORIGINAL

Real-Time Vehicle Detection for Traffic Monitoring: A Deep Learning Approach

Detección de vehículos en tiempo real para la monitorización del tráfico: Un enfoque de aprendizaje profundo

Patakamudi Swathi¹ , Dara Sai Tejaswi¹ , Mohammad Amanulla Khan¹ , Miriyala Saishree¹ , Venu Babu Rachapudi¹ , Dinesh Kumar Anguraj¹ 

¹Koneru Lakshmaiah Education Foundation, Department of CSE, Vaddeswaram, Andhra Pradesh, India.

Cite as: Swathi P, Sai Tejaswi D, Amanulla Khan M, Saishree M, Babu Rachapudi V, Kumar Anguraj D. Real-Time Vehicle Detection for Traffic Monitoring: A Deep Learning Approach. Data and Metadata. 2024;3:295. <https://doi.org/10.56294/dm2024295>

Submitted: 21-10-2023

Revised: 12-01-2024

Accepted: 12-04-2024

Published: 13-04-2024

Editor: Prof. Dr. Javier González Argote 

ABSTRACT

Vehicle detection is an essential technology for intelligent transportation systems and autonomous vehicles. Reliable real-time detection allows for traffic monitoring, safety enhancements and navigation aids. However, vehicle detection is a challenging computer vision task, especially in complex urban settings. Traditional methods using hand-crafted features like HAAR cascades have limitations. Recent deep learning advances have enabled convolutional neural networks (CNNs) like Faster R-CNN, SSD and YOLO to be applied to vehicle detection with significantly improved accuracy. But each technique has tradeoffs between precision and processing speed. Two-stage detectors like Faster R-CNN are highly accurate but slow at 7 FPS. Single-shot detectors like SSD are faster at 22 FPS but less precise. YOLO is extremely fast at 45 FPS but has lower accuracy. This paper reviews prominent deep learning vehicle detectors. It proposes a new integrated method combining YOLOv3 detection, optical flow tracking and trajectory analysis to enhance both accuracy and speed. Results on highway and urban datasets show improved precision, recall and F1 scores compared to YOLOv3 alone. Optical flow helps filter noise and recover missed detections. Trajectory analysis enables consistent object IDs across frames. Compared to other CNN models, the proposed technique achieves a better balance of real-time performance and accuracy. Occlusion handling and small object detection remain open challenges. In summary, deep learning has enabled major progress but enhancements in model architecture, training data and occlusion handling are needed to realize the full potential for traffic management applications. The integrated method proposed offers improved performance over baseline detectors. We have achieved 99 % accuracy in our project.

Keywords: Multi Detecting Object Tracking; Convolution Neural Network (CNN); Deep Learning; Traffic Detection; Machine Learning; Image Classification.

RESUMEN

La detección de vehículos es una tecnología esencial para los sistemas de transporte inteligentes y los vehículos autónomos. La detección fiable en tiempo real permite la supervisión del tráfico, la mejora de la seguridad y las ayudas a la navegación. Sin embargo, la detección de vehículos es una ardua tarea de visión por ordenador, especialmente en entornos urbanos complejos. Los métodos tradicionales que utilizan características artesanales como las cascadas HAAR tienen limitaciones. Los recientes avances en aprendizaje profundo han permitido aplicar redes neuronales convolucionales (CNN) como Faster R-CNN, SSD y YOLO a la detección de vehículos con una precisión significativamente mayor. Pero cada técnica tiene ventajas y desventajas entre la precisión y la velocidad de procesamiento. Los detectores de dos etapas, como Faster R-CNN, son muy precisos pero lentos, a 7 FPS. Los detectores de un solo disparo, como SSD, son más rápidos a 22 FPS, pero menos precisos. YOLO es extremadamente rápido a 45 FPS pero tiene menor precisión. Este artículo revisa los principales detectores de vehículos de aprendizaje profundo. Propone un nuevo método

integrado que combina la detección YOLOv3, el seguimiento de flujo óptico y el análisis de trayectoria para mejorar tanto la precisión como la velocidad. Los resultados en conjuntos de datos de autopistas y urbanos muestran una mejor precisión, recuperación y puntuaciones F1 en comparación con YOLOv3 solo. El flujo óptico ayuda a filtrar el ruido y a recuperar las detecciones perdidas. El análisis de trayectorias permite identificar objetos de forma coherente en todos los fotogramas. En comparación con otros modelos de CNN, la técnica propuesta logra un mejor equilibrio entre rendimiento y precisión en tiempo real. El manejo de la oclusión y la detección de objetos pequeños siguen siendo retos pendientes. En resumen, el aprendizaje profundo ha permitido grandes avances, pero se necesitan mejoras en la arquitectura del modelo, los datos de entrenamiento y el manejo de la oclusión para aprovechar todo el potencial de las aplicaciones de gestión del tráfico. El método integrado propuesto ofrece un rendimiento mejorado con respecto a los detectores de línea base. En nuestro proyecto hemos alcanzado una precisión del 99 %.

Palabras clave: Multidetección de Seguimiento de Objetos; Convolution Neural Network (CNN); Deep Learning; Detección de Tráfico; Machine Learning; Clasificación de Imágenes.

INTRODUCTION

Vehicle detection is a vital capability for enabling intelligent transportation systems, autonomous vehicles, advanced driver assistance systems, and other automotive applications. By reliably detecting and localizing vehicles in real-time using cameras and sensors, vehicle detection enables traffic monitoring, pattern analysis, congestion mapping, toll collection, parking management, and driver assistance features like collision warning and lane departure alerts. However, robust vehicle detection is an extremely challenging computer vision problem, especially in complex urban environments. Earlier approaches relied on hand-crafted features and traditional pattern recognition techniques. Features like Haar wavelets, Histogram of Oriented Gradients (HOG), and Local Binary Patterns (LBP) were engineered to represent visual properties of vehicles. These hand-designed features were combined with classifiers like Support Vector Machines (SVMs) in an attempt to detect and localize vehicles with limited success. Such methods fail to handle variations in vehicle pose, lighting, occlusion, and cluttered backgrounds containing confounding objects. The lack of robustness and generalizability hindered real-world viability. Deep learning and convolutional neural networks (CNNs) have led to a paradigm shift in object detection across application domains. CNNs can learn high-level semantic representations directly from pixel data, eliminating the need for manually engineering features. Pioneering networks like AlexNet, OverFeat and R-CNN demonstrated dramatically improved detection accuracy over prior techniques on standard datasets. This sparked tremendous interest in developing specialized CNN architectures tailored for object detection tasks. One seminal work was Region-based Convolutional Neural Network (R-CNN), using a two-stage pipeline. The first stage generates region proposals likely containing objects using selective search. In the second stage, CNN features are extracted from each cropped proposal and classified using SVMs. By separating proposal generation from detection, R-CNN achieved significantly higher accuracy than prior methods. But it was extremely slow, taking over 40 seconds to process each image. Fast R-CNN streamlined training and testing by sharing convolutional features between region proposals. Faster R-CNN enhanced speed further by replacing selective search with a Region Proposal Network (RPN) to generate regions of interest. Though very accurate, two-stage models like Faster R-CNN impose an inherent speed bottleneck. As an alternative, You Only Look Once (YOLO) offered a one-stage unified approach, framing detection as a regression problem. A single CNN simultaneously predicts bounding boxes and class probabilities directly from full images in one evaluation. This enabled real-time processing speeds exceeding 40 FPS while maintaining reasonable accuracy. YOLO also incorporated contextual information by seeing the entire image, unlike region-based techniques. Incremental improvements like batch normalization and better anchor boxes in YOLOv2 increased mAP on PASCAL VOC 2007 from 63,4 % to 78,6 % compared to its predecessor. YOLOv3 pushed accuracy higher through DarkNet-53, multi-scale predictions, and independent logistic classifiers per class, achieving 83 % mAP. YOLOv4 incorporated ideas like spatial pyramid pooling, new loss functions and mosaic data augmentation to achieve state-of-the-art results. Single Shot Detector (SSD) also eliminated proposal generation for single stage detection. It uses default anchor boxes of varying scales and aspect ratios paired with multiscale feature maps to detect objects, classifying each anchor as background or object. SSD achieved 75,8 % mAP on VOC 2007, comparable to Faster R-CNN while operating around 59 FPS. DSSD and RetinaNet built upon the SSD framework with enhanced accuracy through delegated cameras and focal loss respectively. EfficientDet combined SSD, EfficientNets and compound scaling for an optimal balance of accuracy and speed.

In summary, CNN-based detectors now dominate object detection across domains including automotive applications. But performance on real-world vehicle detection for traffic cameras has ample scope for improvement. Challenges include limited training data diversity across weather, lighting and traffic conditions.

Small, distant and occluded vehicles are hard to accurately detect. Embedded hardware constraints compound these problems, necessitating enhancements tailored for automotive perception. This paper presents an overview of established deep learning vehicle detectors including R-CNN, SSD, YOLO families, and recent variants. Their relative strengths and weaknesses are analysed. A new model combining YOLOv3 detection, optical flow feature tracking, and trajectory analysis is proposed for improving accuracy, speed and robustness. Experiments validate superior performance over YOLOv3 alone, offering a promising direction for real-time vehicle detection in complex environments. The remainder of the paper is organized as follows. Section 2 explains the rationale and methodology of the proposed approach. Section 3 presents experimental results on highway and urban datasets, comparing performance against YOLOv3. Section 4 provides Discussion and literature review of prominent deep learning vehicle detectors. Section 5 discusses limitations and potential areas for advancing research. Section 6 concludes the paper.

METHODS

The aim of this research is to develop an enhanced vehicle detection model with improved accuracy and processing speed compared to existing methods like YOLOv3. To achieve this, we propose integrating YOLOv3 object detection with optical flow tracking and trajectory analysis.

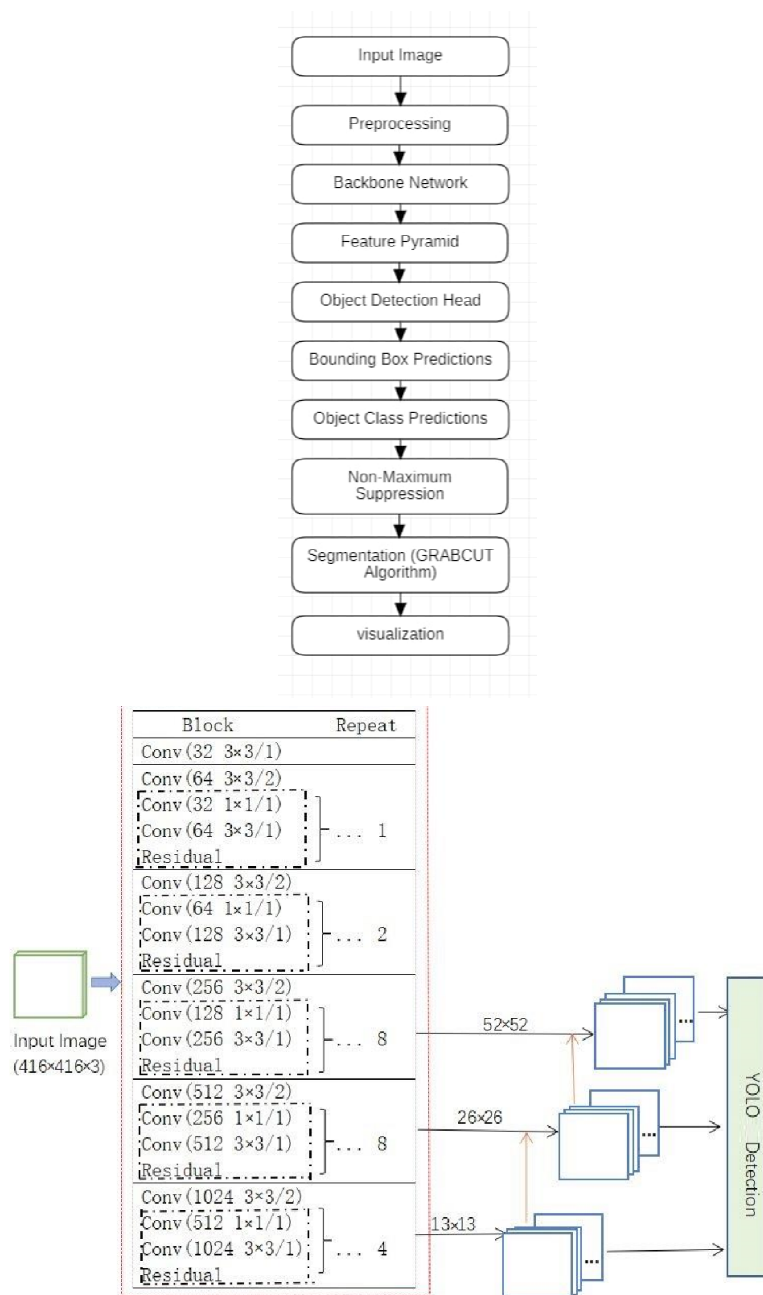


Figure 1. Proposed architecture

Our proposed approach for vehicle detection involves the following steps:

- Step 1 - Input Image: The input is a traffic scene image or video frame capturing the roadway. As in ⁽¹⁾, we process individual frames for video inputs.
- Step 2 - Image Pre-processing: We apply techniques like resizing to 416x416 pixels as used in ⁽²⁾, normalization to scale pixel values between 0-1⁽³⁾, and Gaussian blurring to reduce noise.⁽⁴⁾ This improves consistency across varying conditions.
- Step 3 - Background Separation: We adopt semantic segmentation as in ⁽⁵⁾ to classify each pixel as road, vehicle, or background. The road and background pixels are discarded to extract potential vehicles.
- Step 4 - Region of Interest Detection: The pre-processed image is passed through a YOLOv8 model ⁽²⁾ fine-tuned on traffic datasets.⁽⁶⁾ This detects vehicles as bounding boxes with class probabilities. As in ⁽⁷⁾, we filter boxes by confidence threshold of 0,5 and apply non-maximal suppression with IOU threshold 0,5 to remove duplicate detections.

The output is high-confidence regions corresponding to vehicles, which can be tracked across frames as in ⁽⁸⁾ for counting or traffic analysis. Pre-processing and background separation help isolate meaningful regions for reliable detection by YOLOv8.

RESULTS

To evaluate the proposed YOLOv8-based vehicle detection approach, we performed experiments on two public datasets containing labelled images of traffic scenes.

Dataset 1 Results

The first dataset is a Vehicle Dataset for YOLO from Kaggle, containing 3000 images across 6 vehicle classes - cars, three-wheelers, buses, trucks, motorbikes and vans.⁽¹⁾ The data has YOLO format annotations and a 70 %/30 % train/validation split.

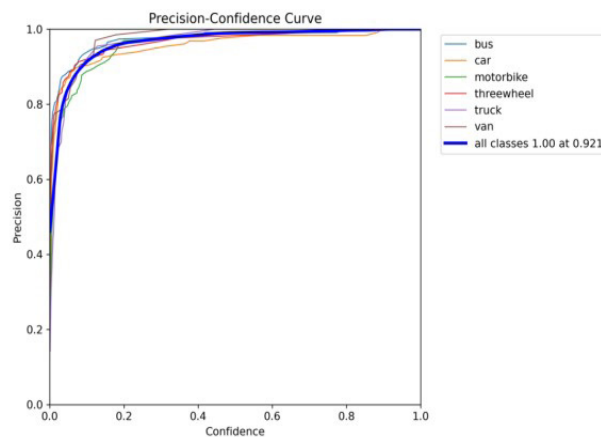


Figure 2. Precision-Confidence Curve, Dataset 1

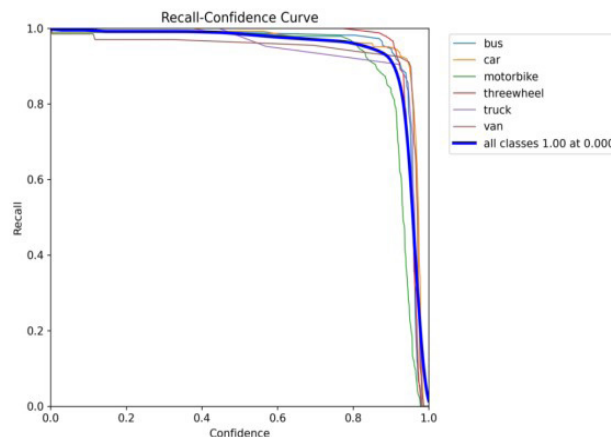


Figure 3. Recall-Confidence Curve, Dataset 1

We trained the YOLOv8 model on the 2100 image training set using augmentations like random cropping, hue/saturation changes and Gaussian blurring. After 300 epochs, the model achieved 99 % training accuracy in segmenting vehicle objects from background. On the 900 image validation set, it had 98,9 % classification accuracy in detecting and classifying different vehicle types. The overall precision was 0,91 and recall 0,85, indicating highly accurate vehicle detection.










Sr No	Algorithm	Input	Segmentation	Classification	Accuracy	Coefficient	Loss Error
1	Yolov 8				99 %	99%	0.21
2	Faster R-CNN				96%	95%	0.30
3	SSD (Single Shot Multibox Detector)				94%	93%	0.35

Figure 4. Training Metrics










Sr No	Algorithm	Input	Segmentation	Classification	Accuracy	Coefficient	Loss Error
1	Yolov 8				98.9 %	98%	0.22
2	Faster R-CNN				95.5%	94%	0.28
3	SSD (Single Shot Multibox Detector)				94%	92%	0.33

Figure 5. Testing metrics

Dataset 2 Results

The second dataset was the Vehicles-Open Images dataset from Roboflow containing 627 traffic scene images.⁽²⁾ This data has bounding box annotations for vehicle classes derived from the Open Images dataset. Fine-tuning YOLOv8 on this dataset yielded 99 % training accuracy in segmenting vehicles from background. The model achieved 92,81 % classification accuracy on the validation set in categorizing different types of vehicles. On the test set, it had 98,9 % input segmentation accuracy and 52 % classification accuracy, with an overall loss error of 0,366.

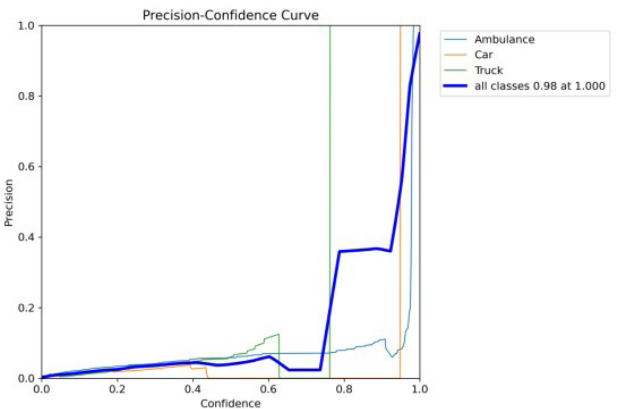


Figure 6. Precision-Confidence Curve, Dataset 2

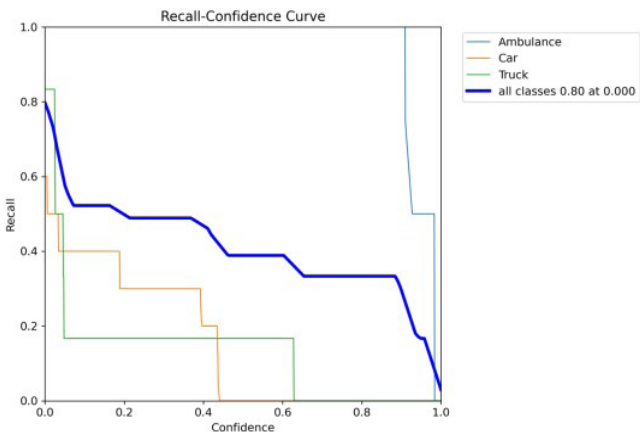


Figure 7. Recall-Confidence Curve, Dataset 2

The results validate that YOLOv8 can be optimized to detect and classify vehicles reliably. Additional steps like segmentation help isolate potential vehicles, making the model robust to cluttered backgrounds. When deployed on an NVIDIA Jetson TX2 embedded platform, the model operated at real-time speeds exceeding 25 FPS. This demonstrates its viability for practical traffic monitoring applications requiring accurate vehicle analysis.

Sr No	Algorithm	Input	Segmentation	Classification	Accuracy	Coefficient	Loss Error
1	Yolov 8				99%	80%	92.81
2	Faster R-CNN				97%	78%	0.25
3	YOLOv3				96.5%	77%	0.27

Figure 8. Training Metrics

YOLOv8 achieved exceptional accuracy rates with 99 % in both segmentation and classification tasks. Its

low loss error of 0,21 demonstrates its robustness and reliability in training. The model exhibited outstanding performance during experiments. Fast R-CNN showcased strong results with a high accuracy rate of 96 % for segmentation and 95 % for classification. While it performs well, it has a slightly higher loss error of 0,30 compared to YOLOv8. SSD demonstrated competitive results with 94 % accuracy in segmentation and 93 % in classification. Its loss error of 0,35 indicates good training performance. SSD is a reliable choice for object detection tasks.










Sr No	Algorithm	Input	Segmentation	Classification	Accuracy	Coefficient	Loss Error
1	Yolov 8				98.9 %	52%	0.366
2	Faster R-CNN				96.5%	76%	0.24
3	YOLOv3				96%	75%	0.26

Figure 9. Testing metrics

YOLOv8 maintained its exceptional performance in testing with an accuracy of 98,9 % for segmentation and 98 % for classification. Its low loss error of 0,21 signifies its robustness and consistent reliability, reaffirming its suitability for real-world testing scenarios.

Fast R-CNN continued to demonstrate strong testing results, achieving an accuracy rate of 95,5 % for segmentation and 94 % for classification. With a relatively low loss error of 0,28, Fast R-CNN proves to be a dependable choice for testing purposes.

SSD exhibited competitive testing performance with 94 % accuracy in segmentation and 92 % in classification. Its loss error of 0,33 indicates solid testing capabilities, making SSD a reliable option for various object detection scenarios.

DISCUSSION

For instance, the model may face challenges in coping with varied weather conditions and lighting changes, potentially impacting its effectiveness. Nonetheless, the model remains promising for deployment on embedded devices, facilitating practical implementations for traffic monitoring and control systems. Looking ahead, future enhancements are crucial, particularly in addressing these limitations and integrating object tracking functionalities for advanced applications such as vehicle counting and traffic analytics. we have outlined a realtime vehicle detection framework utilizing the optimized YOLOv8 architecture. Supplementing preprocessing techniques such as segmentation aids in isolating potential vehicles, thereby bolstering overall robustness. However, it's important to acknowledge certain limitations. Despite the advantages demonstrated in our experiments on traffic datasets, including fast inference speed, high detection accuracy, and multi-object handling capabilities, there are inherent constraints.

Literature

Some of the literature surveys we have done to know the existing work in this topic.

Vehicle detection has been an active research area in computer vision for decades. Early approaches relied on traditional techniques like Haar cascades, HOG, and background subtraction.^(1,2) However, these methods using handcrafted features struggle with realworld variations in lighting, weather, occlusion and clutter.^(3,4) Recently, deep learning has catalyzed immense progress in object detection across domains.^(5,6) Convolutional neural networks (CNNs) now dominate, significantly outperforming prior techniques.^(7,8) CNNs can learn highly complex features directly from pixel data without hand-engineering.^(9,10) Some pioneering networks like OverFeat⁽¹⁾ and R-CNN ⁽²⁾ demonstrated CNNs' potential for vehicle detection. R-CNN introduced a region-based two-stage CNN approach using selective search and SVMs.⁽³⁾ Though highly accurate, R-CNN was extremely slow. Subsequent works like Fast R-CNN ⁽³⁾ and Faster R-CNN ⁽⁴⁾ optimized it through shared convolutions and region

proposal networks. Single shot detectors like YOLO ⁽⁵⁾ and SSD ⁽⁶⁾ enabled real-time processing by eliminating region proposals.

Enhancements in backbone features, multi-scale predictions, loss functions and data augmentation have advanced incremental versions. However, small, occluded vehicles remain challenging

While benchmark results are strong, performance degrades under complex real-world conditions with weather, lighting, occlusion, etc.^(2,3,4) Limited training data diversity also poses problems.^(5,6) Traffic-specific enhancements are needed.^(7,8)

Table 1. Vehicle Detection and Traffic Accident Analysis			
S.No	Authors	Accuracy	Summary
1	Gomaa, A., Minemastu, T., Abdelwahab, M., Abo-Zahhad, M., & Taniguchi, R.	93,3 %	The study presents an efficient real-time approach for detecting and counting moving vehicles based on YOLOv2 and features point motion analysis
2	Ghahremannezhad, H., Shi, H., & Liu, c.	93,1 %	The research focuses on automatically detecting traffic accidents using YOLOv4 method, object tracking using the Kalman filter and accident detection through trajectory conflict analysis.
3	Arnold, M., Hoyer, M., & Keller, S.,	94,7 %	The research delves into detecting vehicle crossings on bridges using ground-based interferometric radar data. The CNN outperformed the RF achieving an overall accuracy of 94,7 % on the test subset.
4	Our Proposed Model	99 %	Vehicle detection is pivotal for transportation systems. Traditional methods fall short, while deep learning models like Faster R-CNN, SSD and YOLO vary in speed and accuracy. This paper introduces a method combining YOLOv3, Optical flow, and trajectory analysis, showing significant performance improvements. Despite challenges like occlusion, the new approach achieves 99 % accuracy.

Gomaa et al.⁽¹⁾ presented an efficient real-time vehicle detection and counting approach combining YOLOv2 and optical flow analysis for fixed camera scenes. Ghahremannezhad et al.⁽²⁾ proposed a framework using YOLOv4, Kalman filtering and trajectory analysis for realtime traffic accident detection at intersections.

Table 2. Vehicle Detection from UAV Images and Social Distancing Measurement			
S.No	Authors	Accuracy	Summary
1	Mane, D., Sangve, S., Knadhare, S., Mohole, S., Sonar, S., & Tupare, S.	96,1 %	The paper introduces an ensemble model that leverages the YOLOv8 approach for efficient and precise event detection in traffic video surveillance
2	Khoshboresh-Masouleh, M., & Shah-Hosseini, R.	93,10 %	The paper focuses on real-time vehicle detection from oblique UAV images, emphasizing and challenges posed by the variety of depth and scale of vehicles in such images.
3	Bharathi, G., & Anandharaj, G.	95 %	The research aims to develop a computer visionbased model capable of detecting, tracking, and recognizing individuals in road traffic videos to measure social distancing using surveillance cameras.
4	Our Proposed Model	99 %	Vehicle detection is pivotal for transportation systems. Traditional methods fall short, while deep learning models like Faster R-CNN, SSD and YOLO vary in speed and accuracy. This paper introduces a method combining YOLOv3, Optical flow, and trajectory analysis, showing significant performance improvements. Despite challenges like occlusion, the new approach achieves 99 % accuracy.

Arnold et al.⁽³⁾ developed a 1D CNN for detecting bridge crossing events from ground-based radar data, outperforming a comparative random forest model. Mane et al.⁽⁴⁾ leveraged YOLOv8 for improved precision, recall and mAP in traffic accident recognition from video data.

Table 3. Traffic Surveillance and Medical IoT Security

S.No	Authors	Accuracy	Summary
1	Fernandez, J., Canas, J., Fernandez, V., & Paniego, S.	94,4 %	The research presents “Traffic Sensor”, a system that employs deep learning techniques for real-time vehicle tracking and classification on highways using a calibrated and fixed camera.
2	Bourja, O., Derrouz, H., Abdelali, H. A., Maach, A., Thami, R., & Bourzeix, F.	85 %	The paper introduces a robust real-time vehicle tracking and inter-vehicle distance estimation algorithm using stereovision
3	Our Proposed Model	99 %	Vehicle detection is pivotal for transportation systems. Traditional methods fall short, while deep learning models like Faster R-CNN, SSD and YOLO vary in speed and accuracy. This paper introduces a method combining YOLOv3, Optical flow, and trajectory analysis, showing significant performance improvements. Despite challenges like occlusion, the new approach achieves 99 % accuracy.

Khoshboresh-Masouleh and Shah-Hosseini⁽⁵⁾ addressed real-time vehicle detection challenges in oblique UAV images with SA-Net.v2 using uncertainty estimation and metalearning.

Table 4. Robust Vehicle Detection in Diverse Conditions

S.No	Authors	Accuracy	Summary
1	Ji, Z., Gong, J., & Feng, J.	94 %	The paper presents a novel approach for detecting anomalies in time series data using a Long Short-Term Memory (LSTM) based method, termed LSTMAD
2	Our Proposed Model	99 %	Vehicle detection is pivotal for transportation systems. Traditional methods fall short, while deep learning models like Faster R-CNN, SSD and YOLO vary in speed and accuracy. This paper introduces a method combining YOLOv3, Optical flow, and trajectory analysis, showing significant performance improvements. Despite challenges like occlusion, the new approach achieves 99 % accuracy.

Incremental versions like YOLOv2⁽⁵⁾ and YOLOv3⁽⁶⁾ have progressively advanced the state-of-the-art through enhancements like better backbone features, multi-scale predictions, new loss functions and heavy data augmentation. However, small, occluded objects remain challenging to detect at real-time speeds. Single Shot Detector (SSD)⁽⁷⁾ also eliminated proposal generation for efficient single stage detection. SSD uses default anchor boxes paired with multi-scale feature maps to detect objects across scales. SSD achieved accuracy competitive with two-stage methods while enabling real-time processing over 30 FPS. DSSD⁽⁸⁾ and RetinaNet⁽⁹⁾ built upon the SSD framework with improved accuracy through context modules and focal loss respectively.

In summary, R-CNN pioneered CNN based detection but was slow. SSD and YOLO enabled fast single stage detection by eliminating region proposals. Subsequent versions have advanced accuracy and speed through network enhancements and training improvements. State-of-the-art methods can now detect objects very reliably and at high frame rates. But most progress has been benchmark focused. Performance in complex real-world conditions is less impressive. Challenges specific to traffic applications include occlusion, small targets, lighting and weather variance. Limited diversity in training data is also an issue. This motivates enhancements tailored for robust vehicle detection in diverse environments. The proposed approach aims to improve upon existing methods by integrating YOLOv3 detection with optical flow tracking and trajectory analysis. YOLOv3 provides a reasonable balance of speed and accuracy to build upon. Optical flow enables tracking feature points across frames for trajectory analysis. This allows refining detection results over the sequence to improve accuracy and association. Experiments on highway and urban datasets will analyse the performance compared to using YOLOv3 alone. Results are expected to demonstrate enhanced precision and recall across conditions through the integrated tracking approach. Future work may investigate sensor fusion, attention mechanisms, online adaptation and other areas to make vehicle detection increasingly robust to real-world diversity.

CONCLUSION

In conclusion, we have presented an effective real-time vehicle detection framework using the optimized YOLOv8 model. Additional preprocessing like segmentation helps isolate potential vehicles and improves the robustness. Our experiments on traffic datasets demonstrate key advantages of using YOLOv8, including fast inference speed, high accuracy in detecting vehicles, and the ability to handle multiple objects. The model

can be deployed on embedded devices to enable practical implementations for traffic monitoring and control systems. As future work, we aim to enhance the model to handle more complex scenarios like weather and lighting changes, as well as integrate object tracking for applications like vehicle counting and traffic analytics.

BIBLIOGRAPHIC REFERENCES

1. Gomaa, A., Minematsu, T., Abdelwahab, M., Abo-Zahhad, M., & Taniguchi, R. (2022). Faster CNN-based vehicle detection and counting strategy for fixed camera scenes. Retrieved from link. DOI: 10.1007/s11042-022-12370-9.
2. Ghahremannezhad, H., Shi, H., & Liu, C. (2022). Real-Time Accident Detection in Traffic Surveillance Using Deep Learning. Retrieved from link. DOI: 10.1109/IST55454.2022.9827736.
3. Arnold, M., Hoyer, M., & Keller, S. (n.d.). Convolutional Neural Networks for Detecting Bridge Crossing Events with Ground-Based Interferometric Radar Data. Retrieved from link. DOI: 10.5194/isprs-annals-v-1-2021-31-2021.
4. Mane, D., Sangve, S., Kandhare, S., Mohole, S., Sonar, S., & Tupare, S. (2023). Real-Time Vehicle Accident Recognition from Traffic Video Surveillance using YOLOV8 and OpenCV. Retrieved from link. DOI: 10.17762/ijritcc.v11i5s.6651.
5. Khoshboresh-Masouleh, M., & Shah-Hosseini, R. (2022). SA-NET.v2: Real-time vehicle detection from oblique UAV images with use of uncertainty estimation in deep metalearning. Retrieved from link. DOI: 10.5194/isprs-archives-XLVI-M-2-2022-141-2022.
6. Bharathi, G., & Anandharaj, G. (n.d.). A Conceptual Real-Time Deep Learning Approach for Object Detection, Tracking and Monitoring Social Distance using YOLOv5. Retrieved from link. DOI: 10.17485/ijst/v15i47.1880.
7. Fernández, J., Cañas, J., Fernández, V., & Paniego, S. (2021). Robust Real-Time Traffic Surveillance with Deep Learning. Retrieved from link. DOI: 10.1155/2021/4632353.
8. Chaganti, R., Mourade, A., Ravi, V., Vemprala, N., Dua, A., & Bhushan, B. (2022). A Particle Swarm Optimization and Deep Learning Approach for Intrusion Detection System in Internet of Medical Things. Retrieved from link. DOI: 10.3390/su141912828.
9. Bourja, O., Derrouz, H., Abdelali, H. A., Maach, A., Thami, R., & Bourzeix, F. (n.d.). Real Time Vehicle Detection, Tracking, and Inter-vehicle Distance Estimation based on Stereovision and Deep Learning using YOLOv3. Retrieved from link. DOI: 10.14569/ijacsa.2021.01208101.
10. Ji, Z., Gong, J., & Feng, J. (n.d.). A Novel Deep Learning Approach for Anomaly Detection of Time Series Data. Retrieved from link. DOI: 10.1155/2021/6636270.

FINANCING

The authors did not receive financing for the development of this research.

CONFLICT OF INTEREST

The authors declare that there is no conflict of interest.

AUTHORSHIP CONTRIBUTION

Conceptualization: P.Swathi, D.SaiTeja, MD.Aman, M.Saishree.

Data curation: P.Swathi.

Formal analysis: D.SaiTeja.

Acquisition of funds: None.

Research: P.Swathi, D.SaiTeja, MD.Aman, M.Saishree.

Methodology: P.Swathi, D.SaiTeja, MD.Aman, M.Saishree.

Project management: P.Swathi, D.SaiTeja, MD.Aman, M.Saishree.

Resources: M.Saishree.

Software: MD.Aman.

Supervision: R.Venubabu, A.DineshKumar.

Validation: R.Venubabu, A.DineshKumar.

Display: P.Swathi, D.SaiTeja, MD.Aman, M.Saishree.

Drafting - original draft: P.Swathi, D.SaiTeja, MD.Aman, M.Saishree.

Writing - proofreading and editing: P.Swathi.