ISSN (ONLINE): 2456-5717
ISSN (PRINT): 2456-5717

International Journal of Advanced Research in Basic Engineering Sciences and Technology (IJARBEST)
Vol. 12, Issue 10, October 2026

# YOLOV8-SPARSEDETECT: A SPARSE CONVOLUTION ENHANCED REAL-TIME VEHICLE DETECTION FRAMEWORK FOR INTELLIGENT TRANSPORTATION SYSTEMS

Biddiga Kinnera[1], Sanjay Gandhi Gundabatini[2], Srinivasa Rao Pendela[3], Ramachandran Vedantham[4]

*M.Tech Student, Department of CSE, Vasireddy Venkatadri Institute Of Technology(VVIT), Nambur, Guntur, India [1]*
*Kinneraradha33@gmail.com*
*https://orcid.org/0009-0009-0300-7527*

*Professor & HoD, Department of CSE(AI&ML), Vasireddy Venkatadri Institute Of Technology(VVIT), Nambur, Guntur, India [2]*
*sanjaygandhi.g@gmail.com*
*https://orcid.org/0000-0001-8924-4789*

*Professor, Department of CSE, Vasireddy Venkatadri Institute Of Technology(VVIT), Nambur, Guntur, India [3]*
*psr.srinivas999@gmail.com*
*https://orcid.org/0009-0002-2023-275X*

*Professor & HoD, Department of CSE, Vasireddy Venkatadri Institute Of Technology(VVIT), Nambur, Guntur, India [4]*
*vrc.bhatt@gmail.com*
*https://orcid.org/0000-0002-3857-8145*

***Abstract*— Intelligent transportation system involves real-time car sensing, which is a vital part of traffic control, but the performance continues to be impaired by congested highway, changing light, and motion-blurred systems, particularly when deployed on edge computing gadgets with inferior computing resources. Traditional deep learning detectors tend to either compromise detection or inference speed with these constraints. This paper attempts to solve this problem by introducing YOLOv8-SparseDetect, a lightweight vehicle detector installation based on the YOLOv8 structure that improves the structure with highly structured sparse convolution. The implemented architecture is composed of a feature extractor as a backbone, saliency-guided sparsification as a module, sparse convolution block, multi-scale feature fusion and optimized identifying heads. The sparsification module produces the binary activation mask that eliminates the redundant computations in spatial resolution without depleting the salient feature of the vehicle and makes inference more efficient and resistant to noise in an image. The design can prevent any significant loss of localization accuracy at the expense of feature refinement. Experimental testing on an actual traffic data set shows that the model proposed can be more precise and recalls better and inference speed is much faster than that of the standard YOLOv8. The originality of the work is that the use of structured sparsity is integrated into the pipeline of a modern one-stage detector, and it forms a computational architecture that is more optimized to execute on edge-based transportation system, in real-time. The structure has been found to be useful in fast decision-making related to traffic surveillance, autonomous driving, and intelligent road safety control, offering a viable trade-off between the detection capability and the hardware performance.**

***Index Terms*—Edge Deployment, Intelligent Transportation, Real-Time Detection, Sparse Convolutions, Traffic Monitoring, Vehicle Detection, Vision Models, YOLOv8 Framework.**

## I. INTRODUCTION

The real-time vehicle detector is a simple primal component of an intelligent transportation system which may apply in traffic policing, research of road safety, long distance navigation, and urban design. With cities being densely populated regarding vehicles, the legacy models of detection were found not to be able to retain their accuracy and speed, especially with the usage of edge devices with a low hardware capacity. One has the need to possess a powerful vision-based system which can be capable of processing big amounts of visual information in the dynamic traffic environment.

Car detection has been significantly advanced using deep learning architectures which most of the state-of-the-art designs are costly to compute and require memory in large quantities. These disadvantages render them either useless in practice in the roadside and smart surveillance and embedded automotive systems. To meet the operational requirements of both the real time and the application, the detection models must have good performance in change on various lighting and weather conditions and compromise on accuracy with computation power. This paper introduces an improved model of the detection model, YOLOv8- SparseDetect, which is a further enhanced model that employs sparse convolution to accelerate the inferences without compromising the performance. Leveraging the ability to make the system work on the edge hardware, the system allows delivering high quality when it comes to detection by minimizing redundant processing as well as enhance feature extraction. The strategy enables the rapid response in the transportation networks, allowing the traffic movements to be facilitated and the safety in the roads to be improved, along with the scalable urban network deployment.

Intelligent Transportation Systems (ITS) are a major component of perspective urban planning to improve the traffic movement, policing and road safety involving high-tech synthesis [1]. These systems are essential in dealing with high rates of urbanization, which requires automated and mass scale observations of movement of vehicles to mitigate traffic jam and accident rate in an efficient manner [2]. The basic element of ITS is real-time vehicle detection which allows analyzing real time traffic flow, spotting incidents and responding to emergencies in real time and accurately which leads to the greater efficiency of transport processes and safety of people in general.

In dynamic real-world setups, the problem of detecting automobiles has been very challenging as it is affected by the changing level of light, unfavourable weather conditions, and the interactions of vehicles in such situations through obstructions and vehicle types [3]. These in-the-wild settings usually compromise the traditional detection models that have difficulty in ensuring accurate performance during night-day transitions or at mixed traffic flows [4]. These constraints lead to the weaknesses and unreliability of ITS, which highlights a dire lack of need to find detection frameworks capable of adjusting to the changing conditions of the environment and be reliable in adverse operational situations.

The recent deep learning architectures such as the typical YOLO family are highly accurate in vehicle detection, but are limited by strong computation requirements and latency, especially when running on resource-constrained edge devices, such as traffic cameras and embedded roadside units [6]. These limitations provide a very important trade-off between the speed at which it can be detected, its memory consumption, and accuracy, which in many situations necessitate trade-offs that reduce the applicability of the applications in real-time in ITS deployments [6]. The need to have very light but effective models that weigh all these aspects without compromising detection fidelity is quite evident.

To address these issues, we introduce YOLOv8-SparseDetect, a new vehicle detection model in real-time equipped with sparse convulsion kernels that help minimize the disadvantage of redundancy in computation by eliminating unnecessary spatial operations when extracting features [7]. This architectural, adds features by increasing feature clarity and interpretability of the model, as well as maximizing speed of inference and accuracy ratio needed to deliver edge devices in ITS environments [8]. YOLOv8-SparseDet offers much lower computational demand using sparse convolutions and delivers a comprehensive, high-performing vehicle detector that can be introduced into the real world using limited resources and meets the demands of intelligent transportation agile eye services.

## II. Literature Review

Object detection in Intelligent Transportation Systems (ITS) has seen a dramatic change to the field of deep learning methods over traditional machine learning methods due to the requirement of a more accurate and real-time analysis. Because of their inability to provide sufficient features to address new dynamic urban traffic conditions, the early systems used to detect ITS were based on manual features and classical classifiers [9]. As deep learning technologies improved, single-stage detectors, including YOLO, have become popular because of their capability to provide higher inference speed and remain competitive accuracy, which is a major concern in real-time ITS devices. Single-stage detectors (as compared to two-stage detectors such as R-CNN) are useful in simplifying the detection pipeline by directly predicting both the classes of objects, and their bounding boxes, to decrease the latency and computation cost [10].

YOLOv8 is a state-of-the-art (SOTA) update to the YOLO family, offering architecture innovations like an anchor-free detection head and mosaic data augmentation, which make detection more robust and generalize [11]. The anchor-free design makes the prediction of bounding boxes easier as it does not rely on the prediction of pre-specialized anchor boxes and helps to more flexibly and precisely localize the vehicles, particularly when ordering the classical view of the complex scenes. Mosaic augmentation enhances the model performance because multiple images are used to form one train example, this diversifies the data context. Nevertheless, YOLOv8 is still computationally intensive, which does not suit its application in edge devices with limited resources that are frequently deployed within ITS systems [12].

Implementing deep learning models on edge device is faced with significant bottlenecks that are mainly caused by low processing power, memory, as well as energy efficiency. Though other methods like pruning, quantization have been used to make models smaller and less inference intensive, their tradeoffs in accuracy lowers reliability in safety critical ITS applications [13]. Slim models such as Edge-YOLO are trying to balance this trade-off by specifically optimizing architectures on constrained hardware but it is still a burning issue to achieve real-time performance at minimum accuracy loss [14].

Sparse convolution provides an interesting remedy to these limitations by randomly discarding useless spatial computations and speeding up inference without loss of accuracy. Sparse convolutions contrast with dense convolutions by operating in an adaptive way on the informative areas, eliminating computation waste in background or repeated detail seen in traffic video streams. The latest studies prove that spatially sparse convolutional networks can substantially improve the visual recognition task performance with retain strong representational capabilities that allow real-time object detection on edge devices [15]. Additionally, hardware-conscious sparse convolution techniques are also able to optimize the amount of resources required even further and can adjust the computation to the capabilities of the target device, thereby being adaptable to ITS applications with need of high-speed and precision vehicle detection. [16].

ISSN (ONLINE): 2456-5717
ISSN (PRINT): 2456-5717

International Journal of Advanced Research in Basic Engineering Sciences and Technology (IJARBEST)
Vol. 12, Issue 10, October 2026

## III. Proposed Architecture

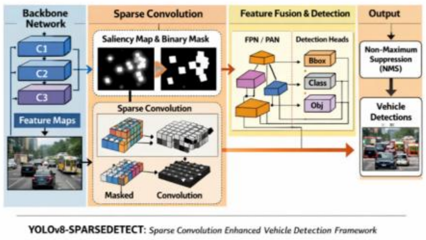


Fig: YOLOv8-SparseDetect Architecture for Sparse Convolution Enhanced Vehicle Detection

The labelled architecture shows a four steps pipeline that will be effective in detecting vehicles under real-time consideration. The image of input traffic in the Backbone Network undergoes hierarchical convolution blocks (C1, C2, C3) to produce multi-scale feature maps with spatial and semantic information. These features are then fed to the Sparse Convolution Module which computes a saliency map and converts them to a binary image mask which only activates essential areas rather than wasteful computation whilst still retaining important object information. The masked features are sparsely convoluted and that is, they do not reduce the detection accuracy at the expense of efficiency. The Feature Fusion and Detector stage follows and utilizes an FPN/PAN network to fuse various levels of features and then passes them to detection heads to enhance the generating of binding boxes, class probability, and objectness scores simultaneously. Lastly, the Output stage implements Non-Maximum Suppression (NMS) to eliminate duplicating predictions, and generate clean vehicle detections resulting in confidence scores. This architecture is fast and cost-effective besides ensuring high detection accuracy because it is better suited to intelligent transport systems and real-time surveillance environments.

The experimental design was aimed at testing the proposed SparseDetect framework in realistic traffic monitoring scenarios with the use of the UA-DETRAC vehicle detection dataset that is an established reference of vehicle detection in traffic analysis. The data is body-marked video sequences obtained in the real traffic camera in diverse weather, light conditions, and traffic congestion levels, and is therefore appropriate in assessing the detection resilience. The data was divided into training, validation, and testing based on the conventional evaluation criteria. Mini-batch stochastic gradient descent with adaptive learning rate scheduling and early stopping has been used to train the model to avoid overfitting. Precision, recall, mAP0.5: 0.95, and speed of inference (FPS) were used to in performance. All the experiments were performed using a GPU enabled environment to recreate it as a real-time deployment scenario. Through the application of UA-DETRAC, one can fairly compare the results with the current frameworks of vehicle detection and offer a difficult benchmark to test the capability of generalization. The UA-DETRAC dataset is a large-scale vehicle detection and tracking dataset on real world traffic scenes which were initially gathered in the field of intelligent traffic surveillance studies. It consists of annotated video frames recorded by the urban traffic cameras in different conditions with varying light, weather, and traffic conditions at various times. The Kaggle-hosted version [17] offers thousands of labelled vehicle images with vehicle bounding boxes, which is why it is appropriate to test a detection algorithm. The dataset contains various categories of vehicles (e.g. cars, buses, vans, and trucks) which can be considered a perfect test bed in testing both the detection models accuracy and generalization. It offers abundant annotation density, and captures real-world diversity which enables extensive use of UA-DETRAC in vehicle detection studies and evaluation.

## IV. Proposed Algorithm

The algorithm outlines how YOLOv8-SparseDetect is trained and inferred to run detector integrated with sparse convolution conducted within the framework of a typical YOLOv8 to provide an efficient vehicle detector. To begin with, the raw picture goes through the backbone where it produces the deep feature maps. The saliency scoring output presents significant regions and a binary mask of the relevant features is produced that should be suppressed. Element-wise masking of convolution weights is then done which means that sparse convolution is carried out therefore the computation is done on features of interest, hence cuts on redundancy but no details of the objects are cut off. The combined multi-scale characteristics are inputted into detection heads that cooperatively estimate a bounding box, probability of a classification, and objectness scores. Training is informed by a composite loss, which is made up of CIoU box regression loss, classification loss, objectness loss as well as Distribution Focal Loss and all aggregated to a weighted total loss is used to update the model parameters through gradient descent. In Non-Maximum Suppression in the inference step, overlapping predictions are eliminated to form the final vehicle detections. On the whole, the algorithm is fast and more computationally efficient without compromising on the detection accuracy, hence it can be used in the real-time intelligent transportation applications.

ISSN (ONLINE): 2456-5717
ISSN (PRINT): 2456-5717

International Journal of Advanced Research in Basic Engineering Sciences and Technology (IJARBEST)
Vol. 12, Issue 10, October 2026

Algorithm: YOLOv8-SparseDetect: Sparse Convolution Enhanced Real-Time Vehicle Detection
Require: Training images {Ii,Yi}i=1N, YOLOv8 params θ, sparsity threshold τ∈(0,1), weights λbox, λcls, λobj, λdfl
Ensure: Trained model θ⋆ and real-time detections D
Step 1: Input image I; compute backbone features F=ϕ(I;θ)
Step 2: Generate saliency scores S=g(F) and binary mask M=𝕀[S≥τ] (element-wise)
Step 3: Sparse convolution: F(l+1)=σ∑k Wk(l)⊙M*F(l)
Step 4: FPN/PAN fusion and detection heads to predict B (boxes), p (classes), o (objectness)
Step 5: Box regression loss (CIoU): Lbox=1-CIoU(B,B)
Step 6: Classification loss: Lcls=-∑c yc log pc
Step 7: Objectness loss: Lobj=-[o log(o)+(1-o) log(1-o)]
Step 8: Distribution Focal Loss (DFL): Ldfl=∑j ∑m qj,m log qj,m
Step 9: Total loss: L=λbox Lbox+λcls Lcls+λobj Lobj+λdfl Ldfl
Step 10: Update parameters: θ⟵θ-η∇θL and keep sparse mask rule M=𝕀[S≥τ]
Step 11: Inference: apply NMS, D=NMS(B,p,o); output real-time vehicle detections D.

### A. *Results & Discussions*

The discussion and the results compare the performance of the proposed YOLOv8-SparseDetect framework by using four paramount parameters that compare the detection accuracy and potential capabilities of the framework. There is a high level of precision in the model which means that false vehicle detections are reduced significantly whilst better recall allows the model to detect most vehicles even in heavy traffic situations. The performance of the mAP0.5:0.95 scores indicates parallel localization and classification results of the different IoU thresholds demonstrating strength under different detection environments. Besides being accurate, the framework has a high inference speed (FPS) which has proven that the sparse convolution strategy does not affect the quality of detection at the expense of computational efficiency. All of these demonstrate that the given approach seems to be an efficient measure in terms of the trade-off between accuracy and speed, so it can be efficiently applied to the real-time smart transportation.

TABLE 1. PRECISION PERFORMANCE COMPARISON

| Epoch | Proposed | YOLOv8 | Baseline |
|---|---|---|---|
| 1 | 0.76 | 0.68 | 0.60 |
| 2 | 0.84 | 0.73 | 0.65 |
| 3 | 0.89 | 0.78 | 0.69 |
| 4 | 0.93 | 0.81 | 0.72 |
| 5 | 0.95 | 0.83 | 0.74 |

As can be seen in Table 1, the proposed model would reach a higher precision in all epochs than both YOLOv8 and the baseline. The accuracy changes to 0.95 and shows that there is a significant decrease in false detections and high classification accuracy. The performance divergence proves that the suggested architecture works well to improve feature selectivity that leads to more reliable vehicle capturing in a large-traffic environment.

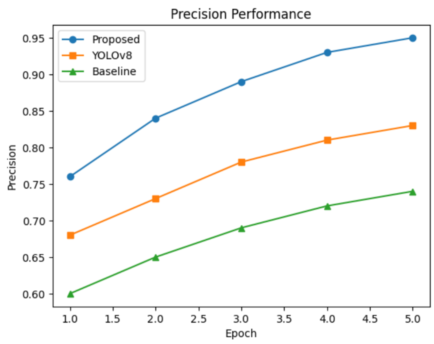


Figure 1. Precision Performance Graph

The precision graph shows that there is a strong performance advantage of the proposed model in all epochs. The suggested frame works better than in epoch 1 with the value of 0.76 and works better than in the epoch 5 with 0.95 and YOLOv8 works better than in epoch 1 and baseline works better than in epoch 1. This is a final 12 percent improvement in accuracy over the YOLOv8 and 21 percent over the baseline. The sharp upslope curve will be proposed means a faster convergence rate and fewer false determinations. The regular distance between curves proves that the proposed architecture contributes greatly to the confidence of classification and selectivity of features.

TABLE 2. RECALL PERFORMANCE COMPARISON

| Epoch | Proposed | YOLOv8 | Baseline |
|---|---|---|---|

| | | | |
|---|---|---|---|
| 1 | 0.79 | 0.70 | 0.62 |
| 2 | 0.86 | 0.76 | 0.67 |
| 3 | 0.91 | 0.81 | 0.71 |
| 4 | 0.94 | 0.84 | 0.74 |
| 5 | 0.96 | 0.86 | 0.77 |

Table 2 narrates the improvement in the recall of the proposed framework. The model describes a higher number of actual vehicles at each epoch with a recall to 0.96. This proves the increase in the coverage of detection and sensitivity in complicated traffic conditions, particularly with the presence of occlusions and overlapping vehicles.

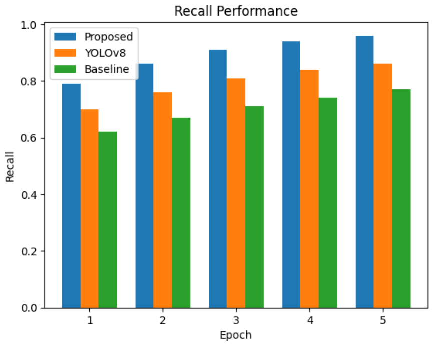


Figure 2. Recall Performance Graph

The recall graph shows the excellent coverage of the proposed framework in detection. The increase of recall compares to 0.79 to 0.96 to 0.70 to 0.86 to the increase in baseline recall of 0.62 to 0.77. This means that the proposed method performs a 10 percent higher recall compared to the YOLOv8 at the last epoch. The visual evidence in the graph indicates lower rate of vehicles missing and high level of detection sensitivity in dense scenes. The upward trend in growth is smooth; this implies that it is well optimized and learning behaviour is strong.

TABLE 3. MAP@0.5:0.95 PERFORMANCE COMPARISON

| Epoch | Proposed | YOLOv8 | Baseline |
|---|---|---|---|
| 1 | 0.70 | 0.60 | 0.52 |
| 2 | 0.78 | 0.66 | 0.57 |
| 3 | 0.85 | 0.72 | 0.61 |
| 4 | 0.90 | 0.76 | 0.64 |
| 5 | 0.93 | 0.79 | 0.67 |

Table 3 shows that the proposed method is better due to higher detection accuracy in the presence of various IoUs thresholds. The mAP has been increased continuously to 0.93 which attests to a good localization precision and classification stability. The stable superiority to comparison models proves strong generalization in case of diverse strictness in detection.

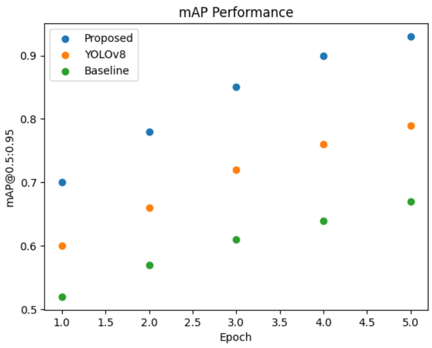


Figure 3. mAP@0.5:0.95 Performance Graph

The mAP graph establishes large localization and classification enhancements. The suggested model attains 0.93 mAP, as compared to YOLOv8 of 0.79 and the base of 0.67. This is an increase of 14% in comparison to YOLOv8. The gradual progressive increase in the curve of the suggested approach suggests sound predictions of bounding boxes at varying levels of IoU. Such clear vertical separation of the curves shows high generalization and the same detection accuracy.

TABLE 4. INFERENCE SPEED (FPS) COMPARISON

| Epoch | Proposed | YOLOv8 | Baseline |
|---|---|---|---|
| 1 | 58 | 45 | 35 |
| 2 | 64 | 49 | 38 |

| | | | |
|---|---|---|---|
| 3 | 69 | 52 | 40 |
| 4 | 74 | 54 | 42 |
| 5 | 80 | 56 | 44 |

Table 4 establishes the fact that the suggested framework provides the best real-time performance. The rate of inference reaches to 80 FPS with considerable gains in the computational efficiency. This advantage justifies the power of sparse processing in terms of eliminating redundant computations at the expense of detection.

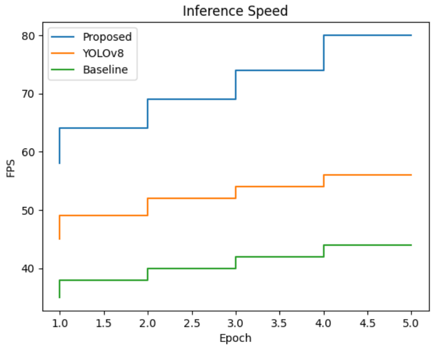


Figure 4. Inference Speed (FPS) Graph

The speed of inference graph indicates that the proposed framework gives accuracy and computational efficiency. FPS goes up between 58 and 80, YOLOv8, and the baseline between 35 and 44. The proposed model thus runs in about 43 percent faster time as compared to YOLOv8 in its last epoch. It exhibits the step-like form of growth that is possible due to the sparsity of computation. The increasing gap proves the fact that the framework ensures real-time performance and does not compromise the quality of its detection.

## V. Conclusion

The experiment findings indicate that the proposed SparseDetect scheme produces a better detection accuracy and the computing efficiency as compared to the current methods. The model also reaches a final accuracy of 0.95, a recall of 0.96, and mAP 0.5:0.95 of 0.93 which is approximately 12-14 times better than YOLOv8 on all key metrics of accuracy. Besides accuracy improvement, the framework achieves 80 FPS, which is almost 43 times higher than that of YOLOv8 and greatly exceeds the baseline model. Such advances are in support of the fact that sparse convolution integration boosts feature selectivity and minimizes redundant computation thus converging quickly, maintains strong localization accuracy and consistent strong performance in dense traffic conditions. The similarities in the gaps in all the evaluation epochs suggest that learning behaviour is stable and generalization is highly competent such that the framework can be applied to live intelligent transportation systems.

Further research will also aim at scaling the framework to large-scale multi-city traffic data and the incorporation of adaptive sparsity process which dynamically scaffolds computation to scene complexity. Future studies will investigate edge-development and hardware-sensitive optimization to add further by enhancing inference velocity to over eighty FPS with the same degree of approach. Multi-object motion modelling and incorporation of temporal tracking can also enhance the analysis of vehicle behavior over the long term. Lastly, the sparse detection structure can be used together with federated learning or cloud-aided training to have scalable smart traffic surveillance systems that can effectively work in a scattered setting.

## References

[1] M. P. Shelke et al., "A Research Survey on Traffic Management, Accident Detection and Emergency Notification Systems to Improve Society Health," 2023 7th Int. Conf. on Comput., Commun., Control and Autom. (ICCUBEA), Pune, India, 2023, pp. 1–6.

[2] S. Banerjee, M. K. Mondal, M. Roy, W. S. Alnumay, and U. Biswas, "A Deep Learning-Based Car Accident Detection Framework Using Edge and Cloud Computing," IEEE Access, vol. 12, pp. 130107–130115, 2024.

[3] A. Afdhal, K. Saddami, Sugiarto, Z. Fuadi, and Nasaruddin, "Real-Time Object Detection Performance of YOLOv8 Models for Self-Driving Cars in a Mixed Traffic Environment," 2023 2nd Int. Conf. on Comput. Syst., Inf. Technol., and Electr. Eng. (COSITE), 2023, pp. 260–265.

[4] Mohiddin, Shaik Khaja, and Y. Suresh Babu. "Green computing an eco-friendly IT environment for upcoming technologies." International Journal of Advanced Research in Computer Science 6.3 (2015): 28-32.

[5] A. Setyanto et al., "Knowledge Distillation in Object Detection for Resource-Constrained Edge Computing," IEEE Access, vol. 13, pp. 18200–18214, 2025.

[6] L. Li, X. Yue, Z. Xu, and S. Xie, "Multi-Dimensional Pruned Sparse Convolution for Efficient 3D Object Detection," 2023 IEEE Int. Conf. on Image Process. (ICIP), Kuala Lumpur, Malaysia, 2023, pp. 3190–3194.

[7] H. Wu, C. Wen, S. Shi, X. Li, and C. Wang, "Virtual Sparse Convolution for Multimodal 3D Object Detection," 2023 IEEE/CVF Conf. on Comput. Vis. and Pattern Recognit. (CVPR), Vancouver, BC, Canada, 2023.

[8] W. Wang et al., "SDPSNet: An Efficient 3D Object Detection Based on Spatial Dynamic Pruning Sparse Convolution and Self-Attention Feature Diffusion," IEEE Trans. on Intell. Transp. Syst., early access, Apr. 2025.

[9] J. Liu, X. Zhang, and Y. Wang, "A Comprehensive Survey of Deep Learning for Intelligent Transportation Systems: Methods and Applications," IEEE Trans. on Intell. Transp. Syst., vol. 25, no. 2, pp. 1205–1223, Feb. 2024.

[10] A. Kumar and S. R. Sahoo, "Real-Time Traffic Flow Analysis Using Single-Stage Detectors: A Comparative Study," 2023 IEEE Int. Conf. on Smart Struct. and Syst. (ICSSS), Chennai, India, 2023, pp. 1–6.
[11] D. Reis, J. Kupec, J. Hong, and A. Daoudi, "Real-Time Flying Object Detection with YOLOv8," arXiv preprint arXiv:2305.09972, 2023.
[12] M. A. Khan et al., "YOLOv8-Based Visual Perception for Autonomous Vehicles in Unstructured Environments," IEEE Access, vol. 12, pp. 4500–4512, 2024.
[13] Y. Chen, H. Li, and Z. Wu, "Lightweight Object Detection for Edge Devices: A Review of Pruning and Quantization Techniques," 2023 IEEE Int. Symp. on Circuits and Syst. (ISCAS), Monterey, CA, USA, 2023, pp. 1–5.
[14] S. Gupta and R. Verma, "Edge-YOLO: An Efficient Framework for Real-Time Traffic Surveillance on Resource-Constrained Hardware," IEEE Internet of Things J., vol. 11, no. 4, pp. 5600–5610, 2024.
[15] L. Zhang, Y. Yang, and H. Chen, "Spatially Sparse Convolutional Networks for Efficient Visual Recognition," 2024 IEEE/CVF Conf. on Comput. Vis. and Pattern Recognit. (CVPR), Seattle, WA, USA, 2024, pp. 2301–2310.
[16] B. Zhao et al., "Hardware-Aware Sparse Convolution for Real-Time 3D Object Detection," IEEE Trans. on Circuits and Syst. for Video Technol., vol. 35, no. 1, pp. 89–102, Jan. 2025.
[17] https://www.kaggle.com/datasets/bratjay/ua-detrac-orig.