Multi-Traffic Scene Perception

Thireesha Suryadevara[1], Mudassir Rafi*[1,2], Nandini Mokhamatam[1], Sai Keerthi Aluri[1], Vishnu Priya[1], and Harika Kommu[1]

[1] Department of Computer Science and Engineering, SRM University AP, Amaravati 522240, India
[2] Department of Computer Science, King Khalid University, Abha, 62529, Kingdom of Saudi Arabia
*Corresponding author: mudassir.r@srmap.edu.in

Abstract. Multi-traffic scene perception is critical for the safe routing and tracking of autonomous vehicles in complex urban environments. This paper proposes an enhanced YOLOv8-based vehicle detection framework for top-view traffic scenes by integrating Channel Attention Modules (CAM) and Multi-Scale Feature Fusion Modules (MSFFM). The proposed model achieves superior performance compared to YOLOv8n (mAP50 = 0.8822), YOLOv7 (mAP50 = 0.8822), and YOLOv9 (mAP50 = 0.8980), attaining an mAP50 of 0.9102 and mAP50–95 of 0.6968. The model is trained and validated on a dataset of 5,766 top-view images containing 2,254 annotated vehicle instances across eight classes. The MSFFM improves multi-scale object detection capability, while CAM enhances feature discriminability. The system demonstrates real-time suitability with an inference time of 4.5 ms per image and a compact model size of 11.47 MB. Although performance limitations are observed for densely packed vehicle classes, the overall results confirm the robustness and effectiveness of the proposed framework for intelligent transportation systems within IoT and AI-driven applications.

Keywords: Channel Attention Modules, Multi-Scale Feature Fusion Modules, YOLOv8

1 Introduction

The demand for correct and timely perception of dynamic traffic patterns has significantly accelerated with the advent of autonomous cars, smart cities, and smart transport systems. The key to the feasibility and success of such systems is multi-scene perception, particularly car perception, which enables applications such as real-time traffic analysis, crash prevention, and efficient transport systems in cities and roads. The distortion caused by perspective and the need for comprehensive analysis and understanding of car distribution can be alleviated with top view images, which can be captured using aerial drones or high-standing cameras, providing a distinct approach to analyzing traffic flow patterns. However, clarity, size variation, and high-density environments can hinder the identification of several cars in an image, which can be highly complex using simple computer vision approaches.

S. Sharma et al. (eds.), *Proceedings of the International Conference on Recent Advances in Intelligent and Sustainable Technologies (RAIST 2026)*, Atlantis Highlights in Intelligent Systems 17,
https://doi.org/10.2991/978-94-6239-707-1_14

Object recognition has greatly benefited from deep learning, particularly with convolutional neural networks, where models like YOLO (You Only Look Once) showed exceptional performance in real-time applications. Newer versions, such as YOLOv8, balance processing economy and precision, making them suitable for an embedded system such as self-driving car. However, the complex traffic situation where cars vary greatly in size and are partially occluded is usually difficult for a standard model to handle. Multi-scale feature processing and attention techniques have emerged as potential enhancements to overcome these limitations, improving contextual awareness and feature extraction. Given the base of the YOLOv8 architecture, this work proposes a novel vehicle identification framework tailored for top-view multi-traffic scene perception.

The main contributions of this work are summarized as follows:

- We developed an enhanced YOLOv8-based vehicle detection framework by integrating Multi-Scale Feature Fusion Modules (MSFFM) and Channel Attention Modules (CAM) to improve multi-scale representation and channel-wise feature discrimination in complex top-view traffic scenes.
- We evaluate the proposed model on a publicly available dataset comprising 5,126 training and 640 validation images across eight vehicle classes, and perform a comprehensive comparison with state-of-the-art models including YOLOv8n, YOLOv7, and YOLOv9 using mAP50, mAP50-95, precision, and recall.
- The proposed approach achieves superior detection performance (mAP50 = 0.9102 and mAP50-95 = 0.6968) while maintaining real-time efficiency, demonstrating its suitability for intelligent transportation and autonomous driving applications.

2 Literature Review

The research focuses on improving detection accuracy for autonomous vehicles through a dynamic neural network approach for real-time 2D object recognition on the roadways. Feature Pyramid Network(FPN) [6] used to perform visual object detection and embeds a CSPNet for feature extraction. For the model adaptation to various datasets, an auto-anchor generation technique is introduced. The model outperforms other detection networks with a high recall rate of 99.3% for bikers and pedestrians, according to experimental results on the KITTI dataset. Thomas *et.al* [8] examine the impact of scene complexity on the identification of vulnerable road users (VRUs) in traffic situations, including cyclists and pedestrians. They find that greater scene complexity dramatically decrease VRU detection rates using images with different degrees of visual crowding. When flanked by other cars, participants often missed VRUs; for walkers, miss rates might be as high as 65%.

In the study [11], roadside camera-based full-stack traffic scene perception system propose for infrastructure-assisted autonomous driving. Accurate real-time perception is made possible by the framework's multi-camera data fusion,

object detection, localization, and tracking features. A landmark-based 3D localization method is presented, which uses just 2D labels for training and does not require 3D annotations. To increase the accuracy level of complex traffic environments, this study proposes a novel vehicle detection method. Based on global context awareness, the proposed model uses a CAM to improve detection accuracy for occlusion and truncation cases. Moreover, a MSFFM is proposed to quickly detect cars of varying sizes in dynamic traffic environments. This will enable the model to adapt to challenging traffic scenarios like overlapping cars and small objects. The proposed model has outstanding accuracy and robustness performance when validated on the KITTI and Cityscapes Datasets. The applicability of this study to real-world autonomous driving projects has been verified through experimental results [4].

Wei Li *et.al*, use the approaches of image captioning to propose a new method for analyzing traffic scenes and improve situational awareness in autonomous driving by fusing NLP with visual perception. The authors develop a model that produces descriptive textual summaries of traffic scenes to enrich the understanding and prediction of dynamic traffic environments [2]. A CNN-based multi-task learning architecture for efficient traffic scene understanding is proposed in [7]. It is focused on object identification and road segregation. Task-specific decoders and a shared encoder are utilized in the architecture to prevent redundant computations and achieve fast inference suitable for low-cost embedded platforms. In addition to that, object orientation is predicted using Analytic Geometry for precise calculation of 3D bounding boxes. The proposed architecture integrates dynamic object localization and free space estimation to achieve a simplified environment map.

A vision-based system is proposed in [9] for the identification and recognition of traffic objects. The four main traffic items it focuses on are traffic signals, road vehicles, pedestrians, and traffic signs. The suggested approach uses multi-scale HOG-SVM and Faster R-CNN models for detection and combines a 3D gaze tracker and stereo imaging system to estimate the driver's attentional visual field. A ResNet-101 network is used for recognition. In urban driving situations, the framework showed a high detection accuracy of 91%. To improve local path planning, Xueqin *et.al* [1] suggest a unique deep reinforcement learning-based navigation system for unmanned ground vehicles (UGVs). By using multi-modal perception, the system decouples perception from control and allows for dependable real-time interaction with the surroundings. This system can handle dynamic impediments efficiently, such as cars and pedestrians, both in simulated and real-world scenarios by using semantic segmentation maps rather than raw RGB images. Modal separation learning and a multi-modal fusion strategy are employed to improve the performance and speed up the training process.

For the aim of enhancing the visualization of the intelligent marine vehicles during harsh weather conditions like haze, darkness, rain, and snow, the authors have designed an all-in-one low-visibility enhancement network referred to as AiOENet. The approach suggested in [3] proposes the classification of four types of low-visibility images. These include images characterized by haze, darkness,

rain, and snow. The approach utilizes the VGG16-powered scene discriminator. The images are transformed using respective encoders, transformers, and decoders. In the maritime environments, the approach introduced by the designed AiOENet enhances image visualization effectively and boosts navigation and detection accuracy. In regard to the intelligent vehicle driving system and the description of traffic situations, this paper proposes a two-stage merging network. The proposed system employs the adaptive two-stage merging network on the encoder-decoder framework model with the capability to change the ratio of the data automatically according to the data being processed. The proposed system has been tested on the datasets COCO2014 and Flickr30K to observe its effectiveness[10].

Through the simulation of the complex interactions between pedestrians and vehicles in traffic scenarios, the researchers propose a new solution for the risk estimation of pedestrian collisions within autonomous driving systems. For accurate risk estimation, the researchers establish traffic scene graphs with enhanced vehicle-pedestrian interactions, proposing a deep learning model utilizing the fusion of the Transformer architecture and the Graph Convolution Networks (GCN). Using the CAP-DATA and JAAD datasets, they generate two types of traffic scene graph datasets: Interaction-Enhanced Scene Graphs (IESG) and None Interaction-Enhanced Scene Graphs (Non-IESG). Experiments on the IESG dataset confirm the efficiency of the proposed model compared to baseline models, showing enhanced accuracy rate (94% against 84%), AUC (98% vs. 89%), and F1 score (93% against 84%) [5].

3 Methodology

3.1 Data Collection

The dataset used for this project was obtained from Kaggle. It consists of a set of top-view images that have been carefully chosen for vehicle detection applications. These images show a range of vehicle kinds in various traffic situations. The images are split into a training set and a validation set. The total number of vehicle occurrences for eight classes (ranging from 0 to 7) representing different vehicle categories used in the detection experiments with 5,126 images in the training set (comprising 5,118 images of vehicles and 8 images of backdrops) and 640 images in validation set. Although class names are not mentioned in the class of the images in the set, these classes consist of several categories of vehicles. The images have been preprocessed to a consistent 640x640 pixel resolution, and YOLO format annotations with class labels and bounding box coordinates are included. During training, data augmentation methods like blur, grayscale conversion, and CLAHE were used to increase the model's resilience to changes in the real environment.

3.2 Proposed Model Architecture and Implementation

Through the introduction of two novel modules—the CAM and the MSFFM —our effort aims to advance the YOLOv8 model for vehicle identification. By fo-

cusing on horizontal and vertical spatial contexts through adaptive average pooling and 1x1 convolutions, CAM improves feature extraction. Attention weights are then generated by a sigmoid activation. By applying these weights to the input characteristics and maintaining the original information through a residual link, the model is better able to concentrate on important areas. However, MSFFM uses an adaptive average pooling branch in addition to max pooling operations at kernel sizes of 1x1, 3x3, 7x7, and 11x11 to capture multi-scale features. In order to preserve feature integrity, these features undergo dimensionality reduction using 1x1 convolutions, concatenation, and processing with a final convolution and ReLU activation, once more utilizing a residual connection. By extending YOLOv8's C2f blocks, this dual-module method was incorporated, resulting in a unique C2f_CAMMSFFM module that includes the advantages of both improvements.

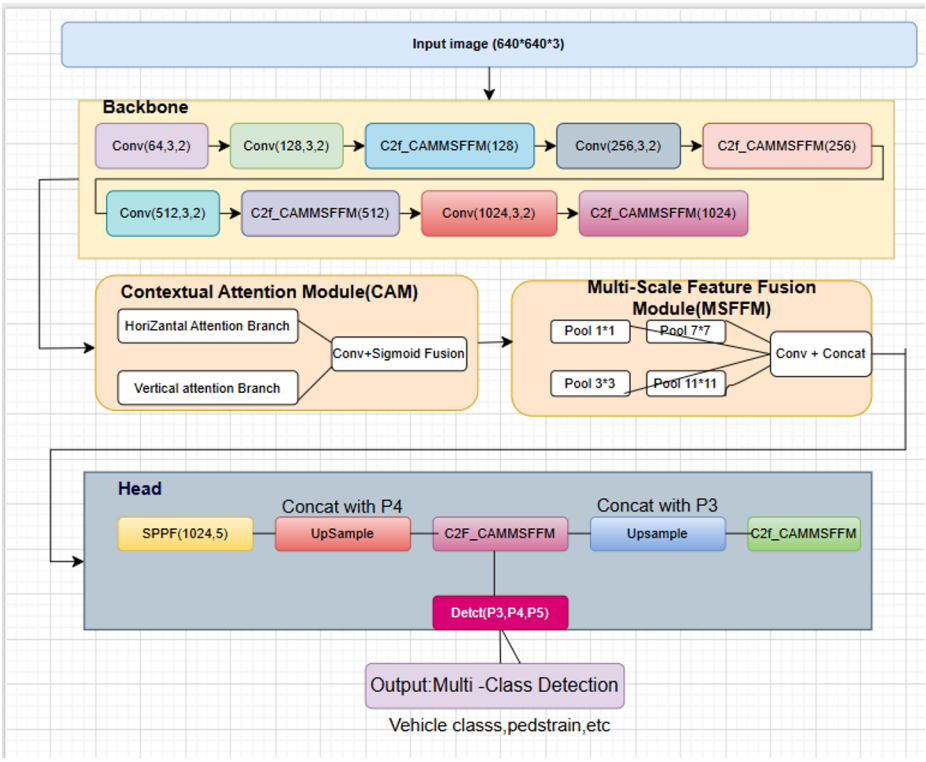


Fig. 1: Architecture of the proposed CAM-MSFFM-enhanced YOLOv8 model for multi-traffic scene perception.

To ensure robustness, our improved YOLOv8 model was implemented using two different methods. First, we used a dummy input (1x3x640x640) to dy-

namically infer the output shapes of the C2f layers of an existing pretrained Yolov8n.pt model. In accordance with the inferred shapes, hooks were then registered to add CAM and MSFFM modules to each C2f block, customizing their channel dimensions (e.g., 32, 64, 128, 256). If this strategy didn't work, a backup plan used a YAML file to create a custom YOLOv8 architecture with three classes (car, bus, and bicycle) and a backbone and head defined by C2f_CAMMSFFM modules. This custom model tried loading pretrained weights from yolov8n.pt, but if it didn't work, it started training from scratch. The main method coordinated the training process, which stored the final weights as cam_msffm_final.pt, set up folders, loaded the dataset from a data.yaml file, trained the model for 10 epochs with a batch size of 4, and image size of 640x640. Predictions and performance indicators (AP50, mAP) were also visualized in order to evaluate the effectiveness of the model.

Fig. 1 builds upon YOLOv8's efficient design by incorporating two novel modules: the CAM and MSFFM. These additions enhance the model's ability to focus on relevant spatial features and integrate multi-scale information for improved traffic object detection.The attention mechanisms help focus on relevant features, while the multi-scale fusion improves the model's capability to handle objects at different distances from the camera.

The proposed architecture takes RGB images of size 640×640×3 as input and employs a YOLOv8-based backbone with progressive channel expansion (64→128→256→512→1024) using 3×3 stride-2 convolutions for hierarchical feature extraction. Standard convolutional layers are interleaved with the custom C2f_CAMMSFFM blocks that integrate a Contextual Attention Module (CAM) and a Multi-Scale Feature Fusion Module (MSFFM). The CAM enhances feature discrimination through dual-branch spatial attention (horizontal and vertical pooling followed by 1×1 convolutions and sigmoid activation with residual connections), while the MSFFM captures multi-scale contextual information using parallel pooling operations (1×1, 3×3, 7×7, 11×11) along with global average pooling, followed by channel reduction, concatenation, and residual learning. The head network incorporates SPPF and FPN-like structure for multi-scale feature aggregation, enabling detection at three scales (P3, P4, P5). The final output layer performs multi-class vehicle detection for diverse traffic categories. Fig. 2 illustrates the implementation workflow for our custom YOLOv8 model enhanced with CAM and MSFFM modules. Table 1 represents the algorithm of proposed framework.

4 Results

For the multi-traffic environment, the use of the customized YOLOv8 model with CAM, MSFFM, and MVSFP technique delivered outstanding results to detect vehicles. With a total of 10 epochs, the model was able to train with a mAP50 of 0.9102 and a mAP50-95 of 0.6968 on the top-view vehicle detection dataset. Upon validation, using a total of 640 images with 2,254 vehicle annotations of eight classes, the model presented a precision of 0.8667 and a recall of 0.8424.

Table 1: Algorithmic workflow of the proposed CAM-MSFFM-based YOLO framework

Stage	**Description**
Start	Initial entry point of the implementation process.
Load Dependencies	Installation of required libraries and frameworks such as: – PyTorch, YOLO (Ultralytics) – NumPy and other essential packages for deep learning and visualization.
Load Dataset	Import traffic object detection dataset using `data.yaml`. Dataset includes eight vehicle classes like **bicycle, bus, car**.
Implementing CAM and MSFFM	Definition and implementation of two custom modules: – **CAM:** CAM with horizontal and vertical attention branches. – **MSFFM:** MSFFM with multi-kernel pooling operations.
Branching Implementation Approaches	Two parallel strategies were used in the implementation: **Modifying Existing YOLO:** – Load a pretrained YOLOv8 model. – Perform shape inference using a dummy input. – Identify all `C2f` blocks in the model. – Insert CAM and MSFFM modules using forward hooks. – **Primary method** used in training (connected directly to Training Loop). **Create YOLO Model with YAML:** – Define a custom YAML configuration. – Use `C2f_CAMMSFFM` blocks to construct a new architecture. – Ased as a **fallback** option, not as part of main training data.
Training Loop	Train the modified model with: – `epochs = 10`, `batch size = 4`, `image size = 640`.
Evaluating Model	Assess performance using validation dataset. Metrics used: – **mAP50**, **mAP50-95** – **Precision** and **Recall**
Validation Results	Analyze validation metrics to determine model effectiveness.
Comparative Study	Compare the enhanced YOLOv8 model with CAM + MSFFM: – Baseline YOLOv8n – Other variants (e.g., YOLOv7, YOLOv9)
Plotting Comparison	Visualize metrics to highlight performance improvements due to CAM and MSFFM.
Saving Results	Save: – Trained model weights – Evaluation metrics – Comparison plots
Stop	Completion of the implementation process.

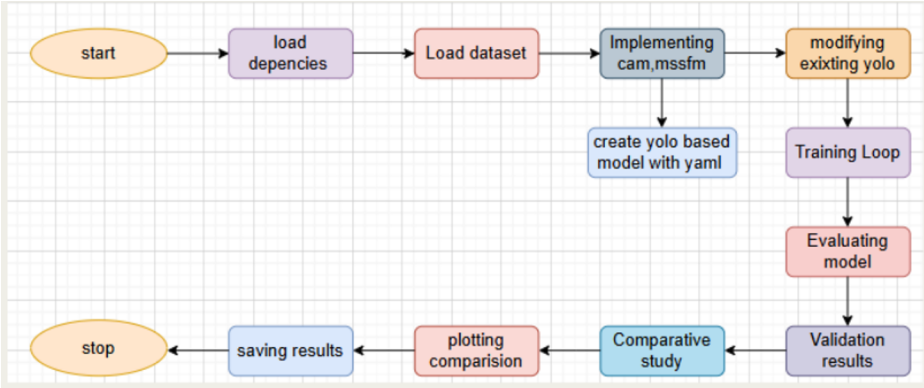


Fig. 2: Workflow of the proposed method from data loading to model evaluation.

In terms of performance, the vehicle detection was not uniform among classes, having the largest mAP50-95 of 0.844 on Class 2, assumed to belong to the usual vehicle category. Finishing the training session, which took place on a Tesla T4 GPU, with a final box loss of 0.9349, classification loss of 0.7677, and a distribution focal loss of 1.136, it consumed a total of 0.693 hours, signifying a successful convergence. Real-time potential was demonstrated by the average inference speed of 4.5 ms per image, with a total inference time of 18.02 seconds over the validation set.These outcomes demonstrate the accuracy and resilience of the model.

4.1 Training Result

The Fig. 3 shows the dataset from Google Drive includes 5,126 training and 640 validation images across 8 vehicle classes. Defined in data.yaml, it was trained on a Tesla T4 GPU for 10 epochs in 41 minutes. Batch size was 4, image size 640x640, with augmentations like blur and mosaic applied. Achieved mAP50 of 0.91 and mAP of 0.695, logged to runs/detect/cam_msffm_model. Weights saved as cam_msffm_final.pt

4.2 Comparative Study

In order to achieve a fair comparison, the performance of the suggested CAM-MSFFM-enhanced YOLOv8 model was thoroughly assessed and contrasted with three well-known baseline models: YOLOv8n, YOLOv7, and YOLOv9 as shown in Fig. 4. All three models were trained and tested using the same top-view vehicle detection dataset. The custom model obtained the highest mean Average Precision (mAP50) of 0.9102 and mAP50-95 of 0.6968, indicating the efficacy of the suggested improvements. The proposed model achieves superior performance compared to baseline YOLO variants as shown in Table 2. Additionally, the custom model surpassed YOLOv8n (0.8683) and YOLOv9 (0.8687) in recall,

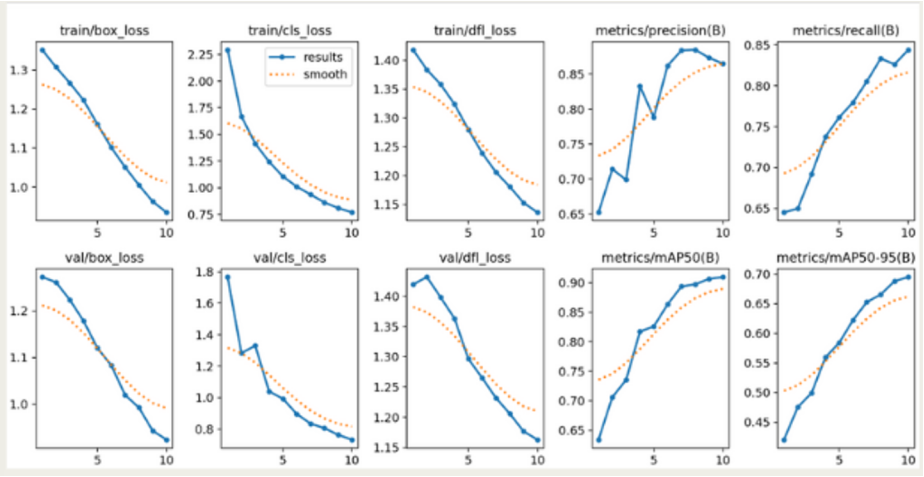


Fig. 3: Training and validation performance curves of the proposed model.

registering a value of 0.8424 as opposed to 0.7927 for YOLOv8n and 0.8202 for YOLOv9, despite achieving a precision of 0.8667, which was closely competitive with these models. This demonstrates how well the custom model can detect more true positives, particularly in settings that are difficult or congested. The main reasons for the improved performance are the MSFFM, which increases detection robustness across a range of object sizes and scales, and the CAM, which enhances the network's capacity to concentrate on pertinent features. When combined, these elements greatly improve the model's capacity to accurately and consistently manage intricate, real-world traffic situations. The following table and graphs illustrate these results:

Table 2: Comparison of performance metrics across different YOLO models

#	**Model**	**mAP50**	**mAP50-95**	**Precision**	**Recall**
0	CAM-MSFFM-YOLOv8	0.9102	0.6968	0.8667	0.8424
1	YOLOv8n	0.8822	0.6468	0.8683	0.7927
2	YOLOv7	0.8822	0.6468	0.8683	0.7927
3	YOLOv9	0.8980	0.6718	0.8687	0.8202

4.3 Per-Class Comparision

Fig. 5 shows that the custom CAM-MSFFM model outperforms YOLOv8n, YOLOv7, and YOLOv9, with peak AP50 (0.93 for Class 2) and AP (0.84 for Class 2), indicating superior per-class accuracy, especially for frequent classes.

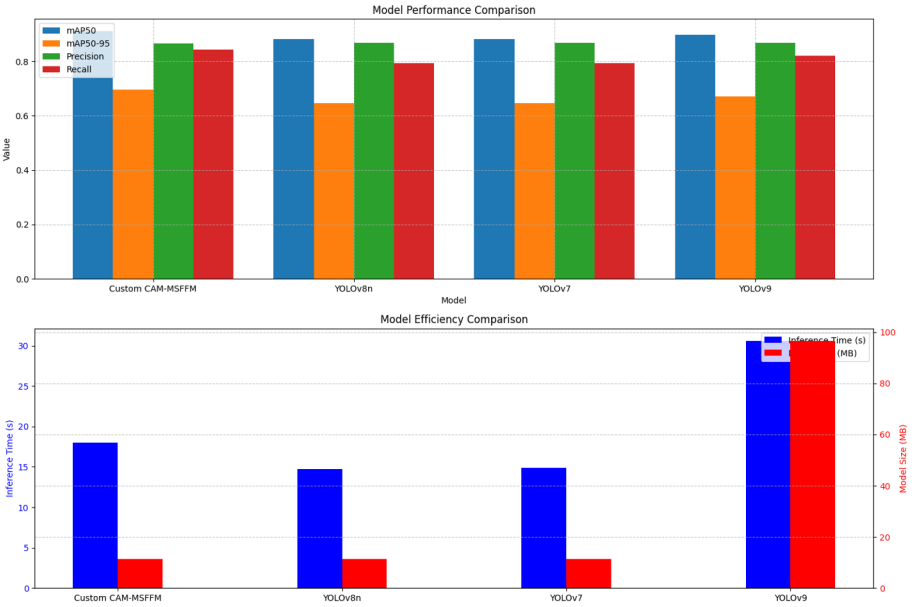


Fig. 4: Performance comparison of the proposed model with baseline YOLO variants.

4.4 Results of detection

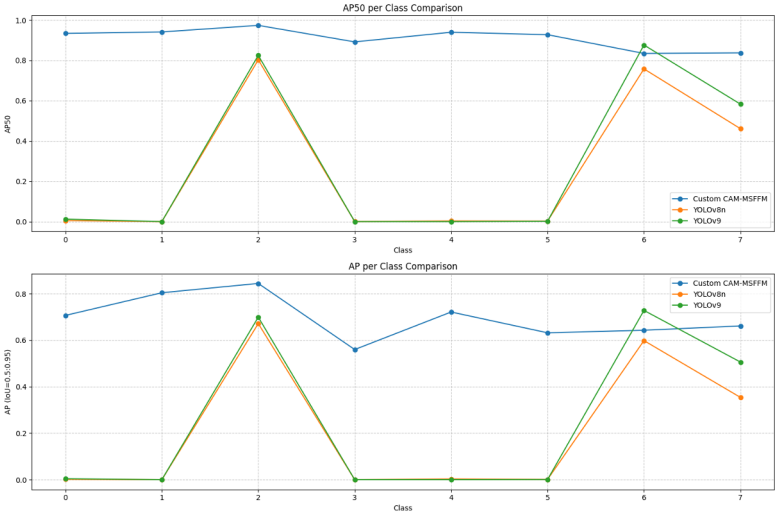
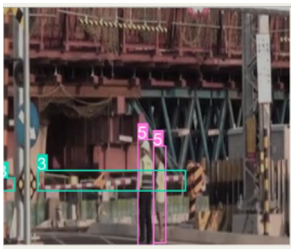


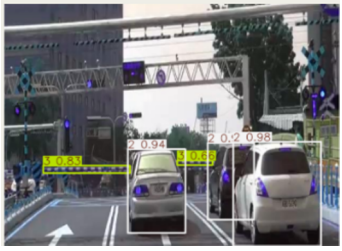
Fig. 5: Per-class AP50 and AP comparison across different YOLO models.



(a) Industrial traffic scene

(b) Occlusion and dense pedestrians

(c) Indoor transit detection



(d) Varying illumination

Fig. 6: Qualitative results of the proposed CAM-MSFFM-based YOLO model.

Fig. 6 demonstrates how well the suggested vehicle detection model performs in a variety of intricate traffic situations, such as scenes with cars and buses inside indoor transit stations, numerous cars and motorcycles negotiating city streets, and railroad crossings with a mix of pedestrian and vehicular traffic. The success of the proposed model is demonstrated by these visual findings, which reliably identify and localize different traffic participants with precise bounding boxes. The outcomes confirm the model's resilience and applicability for practical applications in intelligent traffic systems by demonstrating that it can successfully recognize a variety of object kinds in a range of environmental and scene conditions.

5 Conclusion

This study proposes an enhanced YOLOv8-based vehicle detection framework by integrating Channel Attention Modules (CAM) and Multi-Scale Feature Fusion Modules (MSFFM) for top-view traffic scene perception. The proposed model improved feature discrimination and multi-scale representation, achieves an mAP50 of 0.9102 and mAP50-95 of 0.6968, with a precision of 0.8667 and recall of 0.8424 on 640 validation images containing 2,254 annotated vehicle instances across eight classes. The highest class-wise performance was observed for Class 2 with an mAP50-95 of 0.844. Comparative analysis demonstrated that the proposed model outperformed baseline detectors such as YOLOv8n (mAP50

= 0.8822, mAP50-95 = 0.6468), YOLOv7, and YOLOv9, validating the effectiveness of combining attention mechanisms with multi-scale feature fusion. The model also showed robustness in handling occlusion, scale variations, and diverse vehicle orientations. Future work will focus on improving performance in densely populated traffic scenarios and enhancing generalization using larger and more diverse datasets.

References

1. Huang, X., Deng, H., Zhang, W., Song, R., Li, Y.: Towards multimodal perception-based navigation: A deep reinforcement learning method. IEEE Robotics and Automation Letters **6**(3), 4986–4993 (2021). https://doi.org/10.1109/LRA.2021.3064461
2. Li, W., Qu, Z., Song, H., Wang, P., Xue, B.: The traffic scene understanding and prediction based on image captioning. IEEE Access **9**, 1420–1427 (2021). https://doi.org/10.1109/ACCESS.2020.3047091
3. Liu, R.W., Lu, Y., Guo, Y., Ren, W., Zhu, F., Lv, Y.: Aioenet: All-in-one low-visibility enhancement to improve visual perception for intelligent marine vehicles under severe weather conditions. IEEE Transactions on Intelligent Vehicles **9**(2), 3811–3826 (2024). https://doi.org/10.1109/TIV.2023.3347952
4. Liu, W., Zhao, B., Zhu, Y., Deng, T., Yan, F.: Improving vehicle detection accuracy in complex traffic scenes through context attention and multi-scale feature fusion module. Applied Intelligence **55**(6), 389 (2025)
5. Liu, X., Zhou, Y., Gou, C.: Learning from interaction-enhanced scene graph for pedestrian collision risk assessment. IEEE Transactions on Intelligent Vehicles **8**(9), 4237–4248 (2023). https://doi.org/10.1109/TIV.2023.3309274
6. Mehtab, S., Yan, W.Q.: Flexible neural network for fast and accurate road scene perception. Multimedia Tools and Applications **81**(5), 7169–7181 (2022)
7. Oeljeklaus, M., Hoffmann, F., Bertram, T.: A fast multi-task cnn for spatial understanding of traffic scenes. In: 2018 21st International Conference on Intelligent Transportation Systems (ITSC). pp. 2825–2830 (2018). https://doi.org/10.1109/ITSC.2018.8569822
8. Sanocki, T., Islam, M., Doyon, J.K., Lee, C.: Rapid scene perception with tragic consequences: observers miss perceiving vulnerable road users, especially in crowded traffic scenes. Attention, Perception, & Psychophysics **77**, 1252–1262 (2015)
9. Shirpour, M., Khairdoost, N., Bauer, M.A., Beauchemin, S.S.: Traffic object detection and recognition based on the attentional visual field of drivers. IEEE Transactions on Intelligent Vehicles **8**(1), 594–604 (2023). https://doi.org/10.1109/TIV.2021.3133849
10. Song, H., Zhu, J., Jiang, Y.: Two-stage merging network for describing traffic scenes in intelligent vehicle driving system. IEEE Transactions on Intelligent Transportation Systems **23**(12), 25509–25520 (2022). https://doi.org/10.1109/TITS.2021.3083656
11. Zou, Z., Zhang, R., Shen, S., Pandey, G., Chakravarty, P., Parchami, A., Liu, H.X.: Real-time full-stack traffic scene perception for autonomous driving with roadside cameras. In: 2022 International Conference on Robotics and Automation (ICRA). pp. 890–896 (2022). https://doi.org/10.1109/ICRA46639.2022.9812137