Academic Journal of Computing & Information Science
ISSN 2616-5775 Vol. 9, Issue 4: 49-55, DOI: 10.25236/AJCIS.2026.090406

Road Object Detection Algorithm Based on Improved YOLOv8n

Changchun Wang[1], **Yibo Yin**[1], **Dongmei Liu**[1,*]

[1]Changchun University, Jilin, Changchun, China
**Corresponding Author*

Abstract: *The accuracy of road object detection is crucial for ensuring the safe operation of autonomous vehicles. However, existing road object detection models suffer from problems such as missed detection of small objects, excessive parameters, and low accuracy. To address these issues, the present study proposes a road object detection algorithm based on YOLOv8n. First, to reduce model parameters and computational complexity, a lightweight C2f-FB module is constructed by combining C2f and Faster Block to replace the original C2f in the backbone network. Second, lightweight depthwise separable convolution is introduced into the neck network for downsampling, further reducing the number of parameters and computations. Finally, to improve the detection performance of small road objects, a high-resolution branch and processing detection head are added for small object feature extraction. Comparative experiments on the KITTI dataset evaluated the improved algorithm compared with mainstream methods, the object detection accuracy reached 0.901 and 0.634 (in terms of mAP@50 and mAP@50-95), which is a 2.7% and 3.3% improvement compared to the baseline model, while the number of parameters was reduced by 40.95%. The results demonstrate that the model achieves high detection accuracy and a lightweight design, highlighting its reliability and effectiveness in complex traffic scenarios.*

Keywords: *road object detection, lightweight network, small object detection*

1. Introduction

With the continuous advancement of autonomous driving technology [1], road object detection plays an increasingly critical role in intelligent transportation systems, particularly in key components such as environmental perception [2] and traffic state monitoring. As a fundamental part of the perception module in autonomous driving systems, the primary task of road object detection is to accurately identify and localize key objects in the surrounding environment, thereby providing reliable and high-quality information for subsequent path planning, decision-making, and vehicle control, ensuring that the system can respond in a reasonable, timely, and safe manner. In complex and dynamic traffic scenarios, such as dense traffic flow, severe object occlusion, and low-visibility conditions including nighttime, rainy weather, and haze, environmental uncertainty is significantly increased, posing higher demands on the robustness and real-time performance of detection algorithms. Under such conditions, autonomous vehicles must accurately detect and locate various critical objects, including vehicles [3], pedestrians [4], lane markings, and traffic signs, to achieve comprehensive environmental perception. This capability not only helps the system identify potential hazards in advance and issue effective warnings, but also significantly reduces the likelihood of traffic accidents, improves driving safety and stability, and further enhances the operational smoothness of autonomous driving systems.

In recent years, deep learning-based road object detection methods have attracted significant attention because of their high detection accuracy and computational efficiency. Existing research has primarily focused on lightweight architectures and small object detection techniques to further enhance road object detection performance. Zhu et al. [5] replaced the original backbone network with a lightweight MobileNetV3 feature extraction network based on YOLOv8n, reducing the model's computational complexity and parameter count; Huang et al. [6] replaced the standard convolution in the network's Neck with GSConv convolution, dynamically selecting and adjusting features to maintain model accuracy while reducing the number of parameters; Sang et al. [7] added a detection layer specifically for small objects to the network based on YOLOv8s, effectively improving the accuracy of small object detection, and replaced the original network's CIoU with Focal EIoU, accelerating model convergence and improving regression accuracy; Zhu et al. [8] introduced EIoU as the network's loss function to better adapt to objects of different sizes, improving the model's localization accuracy. Gao et al. [9] constructed a C2fDBB module based on C2f using diverse branch blocks (DBB) to strengthen multi-scale feature extraction, thereby enhancing detection performance. Wang [10] integrated a feature-focused diffusion pyramid neck into the YOLOv8n network to enhance multi-scale contextual representation. In addition, a dynamic task-aligned detection head was introduced to effectively exploit joint features while reducing computational overhead. Wang [11] enhanced the YOLOv5 algorithm by introducing a Bidirectional Feature Pyramid Network (BiFPN) to improve detection performance for objects of different scales. In addition, Convolutional Block Attention Module (CBAM) was incorporated to strengthen feature representation capability.

2. Improving the YOLOv8n algorithm

YOLOv8 [12] achieves high-precision, real-time object detection through optimized network architecture and multi-scale feature fusion, making it well suited for performance-critical scenarios such as autonomous driving. It is divided into five versions based on parameter count and performance model, among which YOLOv8n is the lightest and fastest, and consists of three components: Backbone, Neck, and Head. Although YOLOv8n performs excellently in object detection tasks, it still suffers from issues such as missed detection of small objects, a large number of parameters, and insufficient detection accuracy in complex scenarios in autonomous driving environments. Therefore, the present study proposes an improved YOLOv8n architecture, whose overall structure is illustrated in Figure 1. The specific improvements are described as follows.

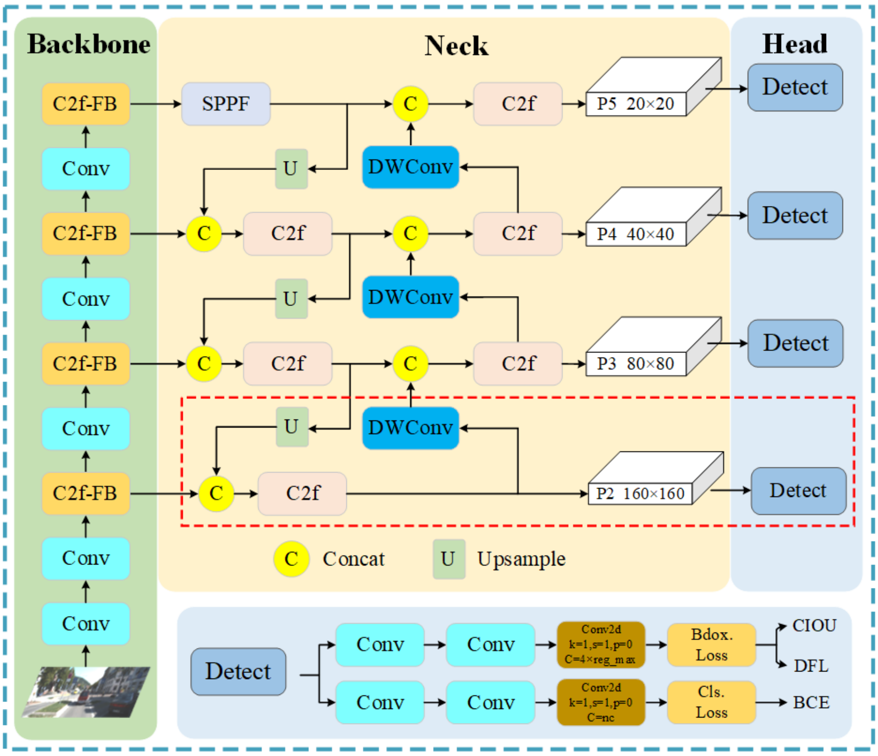


Figure 1. Structure of the improved model

2.1 Improving the C2f module

The C2f module in YOLOv8n significantly improves feature extraction capabilities by increasing the number of Bottleneck layers. However, this design increase in network parameters and high similarity between feature maps across different channels, resulting in a large amount of redundant information and increased memory access and computational overhead. Therefore, the present study introduces the lightweight Faster Block proposed in FasterNet [13], which consists of partial convolution (PConv) and pointwise convolution (PWConv), as shown in Figure 2. PConv reduces redundant feature computation and significantly lowers memory access by performing convolution only on a portion of the input feature map channels, while the remaining channels remain unchanged, thus improving the model's inference speed.

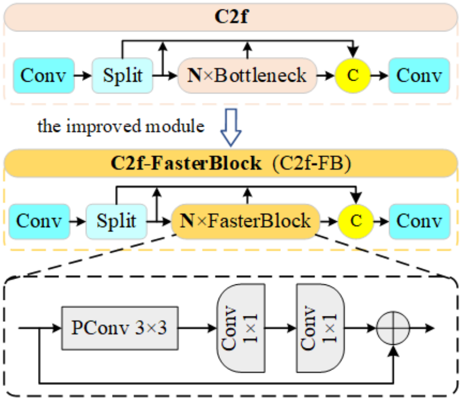
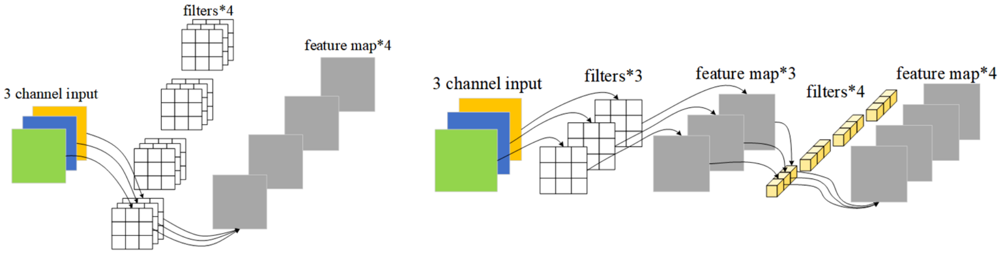


Figure 2. Architecture of the C2f-FB module

Compared with the original C2f module, C2f-FB incorporates a lightweight Faster Block design, effectively reducing model complexity and parameter count, thereby lowering computational cost and improving both training and inference efficiency. While maintaining low resource consumption, the proposed structure enhances feature extraction capability, achieving a favorable balance between model size and detection performance. Moreover, the combination of C2f and Faster Block improves adaptability and robustness in complex traffic scenarios, such as severe occlusion, dense object distribution, and significant illumination variations. Therefore, the improved structure is more suitable for real-time autonomous driving applications that require both high efficiency and accuracy.

2.2 DWConv depthwise separable convolution

Depthwise separable convolution[14] consists of two components: depthwise convolution and pointwise convolution. Compared to the standard convolution in C2f, the main difference is that the convolution operation is decoupled in the spatial and channel dimensions. In standard convolution, each convolution kernel operates on all channels of the input feature map simultaneously, performing spatial feature extraction and channel feature fusion through cross-channel weighted summation. While this method has strong feature representation capabilities, it introduces a large number of parameters and substantial computational overhead. In contrast, depthwise separable convolution first performs depthwise convolution, applying channel-wise convolution to each input channel so that feature extraction is conducted independently in the spatial dimension, producing an output feature map with the same number of channels as the input. Then, 1×1 pointwise convolution is applied to extract channel features and achieve information fusion between different channels, thereby compensating for the shortcomings of depthwise convolution in terms of channel information. Two convolutional methods are compared in Figure 3.



(a) Standard convolution (b)Depthwise convolution

Figure 3. Comparison between standard convolution and depthwise separable convolution.[15]

Depthwise separable convolutions significantly reduce the number of model parameters and computational complexity while maintaining feature representation capabilities, making them particularly suitable for mobile devices with limited computing power and storage resources, as well as real-time object detection and other scenarios.

2.3 Improving the feature extraction network for small objects

When detecting road objects of different scales, YOLOv8n still exhibits certain limitations in handling small objects. Due to their small size, road small objects are more susceptible to occlusion, background interference, and inter-class feature similarity, which leads to the loss of critical information and consequently results in missed and false detections. This issue becomes particularly pronounced in complex traffic scenarios.To effectively enhance the detection capability for small objects, this study introduces targeted improvements to the feature extraction and detection architecture based on YOLOv8n. On the one hand, the PANet structure in the Neck is extended by adding a high-resolution feature branch (160×160) to preserve more shallow-layer details and improve the perception of tiny objects. On the other hand, a dedicated small-object detection head is introduced in the Head to process high-resolution features from shallow layers, thereby strengthening the learning of fine-grained features, as illustrated by the red dashed box in Fig. 1.

Through these improvements, the small-object feature extraction network is able to more effectively fuse multi-scale feature information, significantly reducing information loss during feature propagation and enhancing the completeness and accuracy of detecting objects of different sizes. Meanwhile, the proposed method improves the adaptability and robustness of the model under complex environmental conditions, leading to a comprehensive enhancement in detection performance in real-world road scenarios.

3. Experimental results and analysis

3.1 Dataset

This experiment uses the KITTI autonomous driving dataset, jointly developed by the Karlsruhe Institute of Technology in Germany and the Toyota Technical Institute in America. The dataset contains numerous overlapping and small objects and covers diverse road scenarios, including rural areas, urban centers, and highways. It defines nine categories: Tram, Truck, Car, Person sitting, Pedestrian, Van, Cyclist, Misc, and DontCare. To better reflect real-world autonomous driving applications, Car, Tram, Truck, and Van were merged into a single Car; Person sitting and Pedestrian were merged into Pedestrian; Cyclist was retained; and Misc and DontCare were removed, resulting in three final categories: Car, Pedestrian, and Cyclist. Because the KITTI test set does not provide labels, only the labeled training images were used, totaling 7,481 samples. The dataset was randomly split into training, validation, and test sets at a ratio of 8:1:1.

3.2 Experimental environment and parameter configuration

All experiments in this study were conducted under the platform configuration shown in Table 1. All models were trained for 200 epochs with a batch size of 16. The input image size was set to 640 × 640, with an initial learning rate of 0.01, momentum of 0.937, and weight decay of 0.0005.

Table 1. Experimental platform configuration

Configuration	Parameter
CPU	12th Gen Intel(R) Core(TM) i5-12600KF
GPU	NVIDIA GeForce RTX 3060
Operating system	Windows 10
Python	3.8.0
Accelerated environment	CUDA 11.8, CUDNN 8.6.0
Development environment	Pycharm 2023.2.6
Deep learning framework	Pytorch 2.0.0

3.3 Evaluation metrics

This study employs four key evaluation metrics in object detection to comprehensively evaluate the performance of different models, including precision (P), recall (R), mean average precision (mAP), and the number of parameters.

The relevant formulas are as follows:

$$P = \frac{TP}{TP + FP}$$

$$R = \frac{TP}{TP + FN}$$

$$mAP = \frac{1}{N}\sum_{i=1}^{N} AP_i$$

Here, TP represents the number of instances correctly classified as positive, FP represents the number of instances incorrectly classified as positive, FN represents the number of instances incorrectly classified as negative, APi is the average precision for the i-th class, N is the total number of classes, and parameters are used to evaluate the complexity of the model; the more complex the model structure, the greater the number of parameters.

3.4 Experimental results

3.4.1 Ablation study

To investigate the impact of the C2f-FB module, DWConv module, and the improved small object feature extraction network on the model's detection performance in this study, the effectiveness of each improvement was validated based on the baseline model. Eight sets of ablation experiments were designed using the KITTI dataset, and the test results are shown in Table 2.

Table 2. Ablation study results

Model	Improved method			P	R	mAP@50	mAP@50-95	Parameters/10^6
	A	B	C					
1				0.916	0.774	0.874	0.601	3.006
2	√			0.880	0.794	0.871	0.604	2.646
3		√		0.871	0.778	0.86	0.585	1.896
4	√	√		0.864	0.800	0.865	0.571	**1.536**
5	√		√	0.886	0.843	0.907	0.639	1.853
6		√	√	0.891	0.826	0.899	0.641	2.117
7	√	√	√	**0.882**	**0.821**	**0.901**	**0.634**	1.775

Note: A represents the lightweight and efficient C2f-FB module, B is the DWConv module, C is the improved small object feature extraction network, and Model 1 is the YOLOv8n baseline model.

The experimental results of Models 2, 3, and 4 show that the incorporation of C2f-FB and DWConv significantly reduces the number of parameters, alleviates frequent memory access and channel redundancy, and improves training efficiency. From the comparative analysis of Models 5 and 6, it can be observed that optimizing the small object feature extraction network leads to consistent improvements across all evaluation metrics, effectively reducing missed detections and false positives of small objects in road scenarios, thereby enhancing overall detection performance. Furthermore, Model 7 integrates the three proposed improvements. Compared with the baseline Model 1, although the precision (P) slightly decreases, the recall (R), mAP@50, and mAP@50–95 increase by 4.7%, 2.7%, and 3.3%, respectively, while the number of parameters is reduced by 40.95%. These results demonstrate that the proposed method significantly improves road object detection performance and achieves an effective balance between detection accuracy and model complexity.

3.4.2 Comparative experiments of different mainstream algorithms

To evaluate the effectiveness of the proposed algorithm for road object detection in autonomous driving, comparative experiments were conducted on the KITTI dataset, comparing it with different mainstream algorithms. The experimental results are shown in Table 3.

Table 3. Comparison results of different mainstream algorithms

Algorithm	P	R	mAP@50	mAP@50-95	Parameters/10^6
YOLOv3-tiny	0.889	0.731	0.834	0.508	8.671
YOLOv5n	0.892	0.799	0.877	0.566	**1.763**
YOLOv7-tiny	0.862	0.812	0.87	0.557	6.013
YOLOv8n	**0.916**	0.774	0.874	0.601	3.006
YOLOv10n	0.86	0.791	0.855	0.592	2.696
YOLOv11n	0.895	0.785	0.867	0.592	2.583
Improved algorithm	0.882	**0.821**	**0.901**	**0.634**	1.775

Experimental results demonstrate that the proposed method achieves a favorable balance between model lightweighting and detection performance. The number of parameters is only 1.775M, which is significantly lower than that of YOLOv3-tiny (8.671M) and YOLOv7-tiny (6.013M), while still attaining competitive detection performance. Furthermore, compared with lightweight models such as YOLOv5n, YOLOv10n, and YOLOv11n, the proposed method exhibits superior detection accuracy while maintaining a compact model size. Relative to the baseline model, the R is improved by 4.7%, while mAP@50 and mAP@50-95 increase by 2.7% and 3.3%, respectively, with a 40.95% reduction in parameter count. In summary, the proposed method demonstrates clear advantages in achieving both lightweight design and high detection accuracy for road object detection tasks.

4. Conclusion

To address the issues of missed small object detection, excessive parameters, and low detection accuracy in existing models in complex road scenarios, this study proposes an improved road object detection algorithm based on YOLOv8n. First, a lightweight C2f-FB module is constructed by combining Faster Block and C2f. By performing feature extraction on only a portion of the input channels, this module effectively reduces redundant calculations and model parameters while maintaining feature representation capabilities. Second, lightweight depthwise separable convolutions are introduced into the neck network for downsampling, further reducing parameters and computation while preserving feature representation. Finally, a high-resolution feature extraction network is added to capture fine details of small objects in road scenes, reducing interference from complex road environments and improving detection accuracy. Training and testing results show that, compared to the baseline model on the KITTI dataset, the improved algorithm achieves a 4.7% and 2.7% increase in accuracy (in terms of R and mAP@50), while reducing the number of parameters by 40.95%. Compared to other mainstream algorithms, the improved algorithm achieves lightweight design while also demonstrating superior detection accuracy, effectively addressing the problems of missed small object detection, excessive parameters, and low detection accuracy in complex road scenarios, providing new methods and ideas for autonomous driving environment perception. Despite these improvements, this study still has some limitations, and future research will focus on the following two aspects:

(1) Due to hardware limitations, the improved algorithm proposed in this study was only implemented on a GPU platform. Therefore, future work will focus on the deployment and optimization of the improved algorithm on embedded platforms to further verify its feasibility in actual autonomous driving systems.

(2) Future work will further introduce datasets containing complex scenarios such as curves, rainy weather, and nighttime conditions, expanding the training samples to enhance the model's ability to learn complex environmental features and its robustness to different road conditions.

References

[1] Wang H, Liu C, Cai Y, et al. YOLOv8-QSD: An improved small object detection algorithm for autonomous vehicles based on YOLOv8[J]. IEEE Transactions on Instrumentation and Measurement, 2024, 73: 1-16.

[2] Natan O, Miura J. DeepIPC: Deeply integrated perception and control for an autonomous vehicle in real environments[J]. IEEE Access, 2024, 12: 49590-49601.

[3] Kang L, Lu Z, Meng L, et al. YOLO-FA: Type-1 fuzzy attention based YOLO detector for vehicle detection[J]. Expert Systems with Applications, 2024, 237: 121209.

[4] Chen F, Gu X, Gao L, et al. Pedestrian detection method based on fcos-defpn model[J]. IEEE Access, 2024, 12: 144337-144349.

[5] Zhu J, Liu Y, Zhang D. A Lightweight Road Crack Detection Model Based on Improved YOLOv8n[J]. Journal of Data Acquisition and Processing, 2025, 40(05): 1333-1347.

[6] Huang C, Xu H, Zhang X, et al. BGR-YOLO: An Improved Object Detection AlgorithmUnder Traffic Scenarios Based On YOLOv8[J/OL]. Computer Engineering & Science, 2025: 1-13.

[7] Sang J, Zhang Z, Xiao C, et al. An improved YOLOv8s method and its applicationin road traffic target detection[J]. Infrared and Laser Engineering, 2024, 53(11): 303-316.

[8] Zhu Y, Zhang Y. SEP-YOLO: ROAD OBJECT DETECTION ALGORITHM IMPROVED BASEDON YOLOV8[J/OL]. Computer Applications and Software, 2024: 1-8.

[9] Gao D, Chen T, Miao L. Improved Road Object Detection Algorithm for YOLOv8n[J]. Computer Engineering and Applications, 2024, 60(16): 186-197.

[10] Liu Z, Li L, Hu Y, et al. Enhanced road object detection with DFPD-YOLO: focusing on small and occluded targets [J]. The Journal of Supercomputing, 2025, 81(14): 1329.

[11] Chen H, Chen Z, Yu H. Enhanced YOLOv5: an efficient road object detection method[J]. Sensors, 2023, 23(20): 8355.

[12] ULTRALYTICS.YOLOv8. https://github.com/ultralytics/ultralytics.

[13] Chen J, Kao S, He H, et al. Run, don't walk: chasing higher FLOPS for faster neural networks[C]//Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2023: 12021-12031.

[14] Chollet F. Xception: Deep learning with depthwise separable convolutions[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2017: 1251-1258.

[15] JIANG Jian, YUAN Zhi qun, GAO Xiu jing, HE Hong zheng, GU Zi-shuo. Object detection algorithm for smart port based on improved YOLOX-S[J]. Object detection algorithm for smart port based on improved YOLOX-S, 2025,46(07):2045-2053.