Based on Improved Lightweight YOLOv8 for Vehicle Detection

Jie Zhou [1, a], Hongwei Xu [2, b], Runjing Zhou [1, c], Xin Du [1, d]

[1] School of Electronic and Information Engineering, University of Inner Mongolia, Hohhot, MA 010000, China;

[2] Inner Mongolia Hetao Irrigation Area Water Conservancy Development Center, Bayannur.

[a] 13679158194@163.com, [b] 36403704@qq.com, [c] auzhourj@163.com, [d] du_xin_4503@163.com

Abstract. In urban road traffic, vehicle detection in intelligent transportation system can effectively improve road traffic operation efficiency and road safety. Based on this goal, this paper is based on the YOLOv8 framework, using the improved lightweight YOLOv8 model to realize vehicle detection, the method replaces the Backbone network in the YOLOv8n model with a more efficient and lightweight EfficientViT network, and further adds the CBAM attention mechanism in Neck to enhance the detection precision, and later, Conv is replaced with the GhostConv model in Head to reduce the number of parameters and increase the speed of detection. This model greatly improves the detection precision and efficiency. We validate our approach on the UA-DETRAC vehicle detection dataset, and the experimental results show that the proposed detection model Precision reaches 91.17%, mAP@0.5 reaches 75.45%, and Recall reaches about 70.01%. Compared with the original YOLOv8n model, Precision improves by 4.6%, mAP@0.5 improves by about 4.3%, and Recall improves by about 9.1%. Through experimental validation, the model performs well in detecting various complex road traffic scenarios, and this innovative approach can provide strong support for the development of intelligent transportation systems, such as deployed on devices with limited mobile hardware resources, and is expected to make further breakthroughs in future applications.

Keywords: Target Detection; Vehicle Detection; Improvement of YOLOv8; Deep Learning.

1. Introduction

In recent years, with the continuous updating of deep learning has become an ideal choice for solving complex road traffic management. Currently, deep learning-based target detection algorithms can be mainly categorized into two main groups: regression-based single-stage target detection algorithms represented by SSD [1], YOLO [2], EfficientDet [3], etc., and candidate region-based two-stage target detection represented by R-CNN [4], SPP-Net [5], Detectron2 [6], etc. For the vehicle detection problem applied to road traffic, YOLOv8 [7], as a representative of mainstream single-stage target detection algorithms, is the first choice for solving the vehicle detection in road traffic, as it guarantees high detection precision while also having good real-time performance. Therefore, this paper adopts the improved YOLOv8 algorithm to realize vehicle detection, selects the YOLOv8n model as the base model, and proposes an improved lightweight vehicle detection model method for the vehicle detection problem under road traffic, and the contributions of this paper are summarized as follows:

(1) Replace the Backbone network in the YOLOv8n model with an EfficientViT network. This network structure effectively improves the overall detection precision and speed of the model.

(2) Neck introduces the CBAM attention mechanism, which effectively improves the model's ability to extract and integrate feature information in complex traffic scenarios and increases the model's detection performance without adding too much computational cost.

(3) Improve the lightweight Head by replacing the Conv in the Head with the GhostConv module to reduce the number of model parameters, which greatly improves the detection speed of the lightweight model, and has good detection performance when deployed in mobile hardware resource-constrained devices.

2. YOLOv8 Algorithm

YOLO (You Only Look Once) series is a popular real-time target detection algorithm, which is one of the mainstream detection algorithms nowadays for its high speed and accuracy. Among them, the overall network structure of YOLOv8 is divided into four parts: Input, Backbone, Neck and Head layers [8], as shown in Figure 1. And in this paper, improvements will be made in the latter three parts.

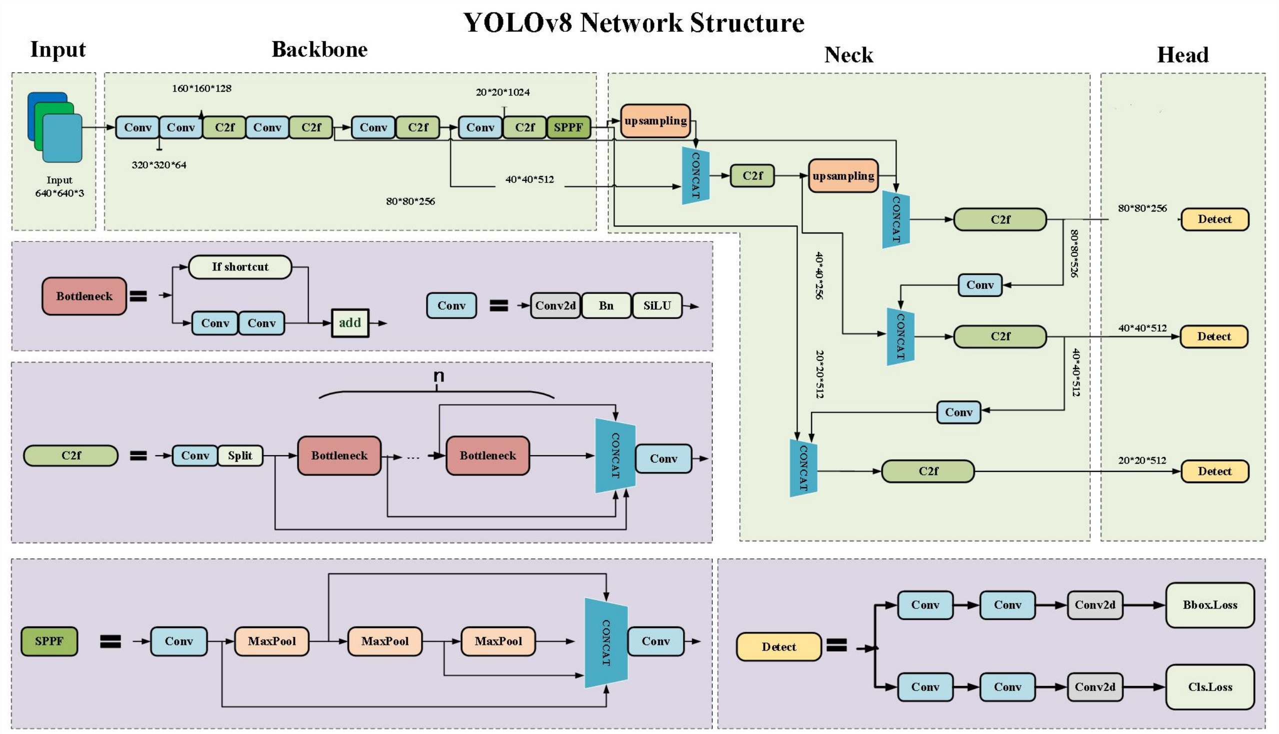


Fig. 1 YOLOv8 network structure

3. Improved YOLOv8 vehicle detection model

3.1 EfficientViT Model

In YOLOv8, due to the complex network structure and more parameters, the detection speed is relatively slow, which makes it difficult to balance the detection speed and detection precision in application scenarios with limited hardware resources. In order to improve the detection precision and speed of the target detection algorithm YOLOv8, this paper adopts the lighter EfficientViT [9] network as the backbone network for replacement, which not only improves the accuracy of target detection, but also greatly improves the computational speed compared with the original model.

The EfficientViT network model structure is shown in Fig.2. Different from the traditional converter model, its structure adopts a novel multi-scale linear attention mechanism to achieve global perceptual field and multi-scale learning, which can achieve significant speedup on various hardware platforms by optimizing memory efficiency and computational redundancy. The EfficientViT network architecture consists of a multiscale linear attention module and an FFN with Dependency convolution (FFN+DWConv). Multiscale linear attention is used to obtain contextual information, while FFN+DWConv is used to capture local information. After obtaining the Q/K/V tokens through the linear projection layer, the method generates the multiscale tokens by DWConv and 1*1GConv convolution, and finally connects the ReLU linear attention outputs to be sent to the linear projection layer for final feature fusion.

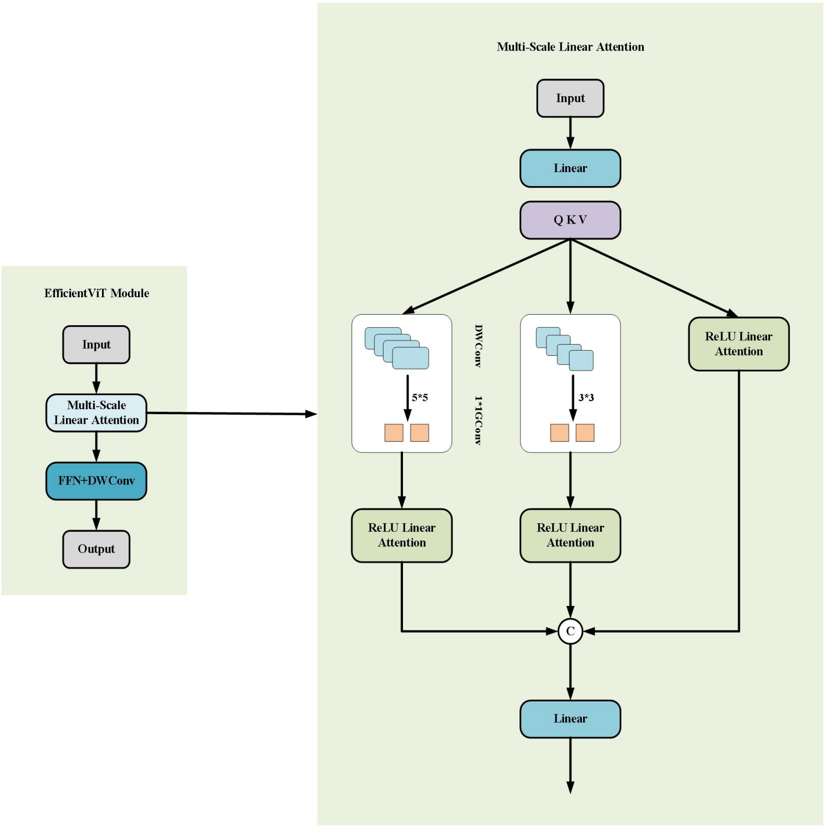


Fig. 2 EfficientViT model structure

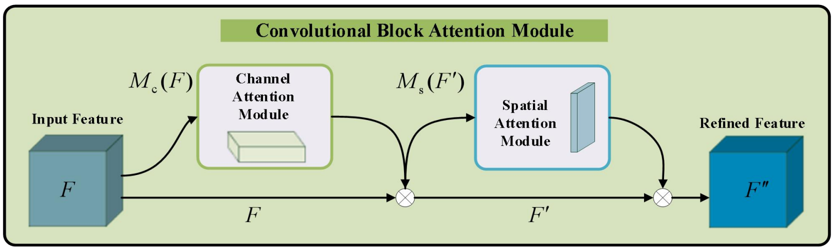


Fig. 3 CBAM module

3.2 CBAM Module

The attention mechanism can help the model to focus on the feature information of interest when processing image data, thus improving the performance and efficiency of the model. In deep learning, CBAM is a module that combines spatial attention (SAM) and channel attention (CAM). It not only considers the spatial information of each location in the feature map, but also pays attention to the importance of each channel, so as to improve the model's ability to understand the input data at multiple scales. Compared with other complex attention models, the CBAM module is relatively simple in design and can effectively improve the model performance without adding much computational cost [10].

The structure of the CBAM module is shown in Figure 3, and we assume that the input feature diagram is：

$$F = R^{C*H*W} \tag{1}$$

Where C, H, W represent: number of channels, width, height, respectively.

The input feature map first passes through the CAM module, and is convolved with this module in one dimension to enhance the model's ability to perceive different feature channels, and the result after convolution is shown in Eq. 2:

$$F' = M_c(F) \otimes F \tag{2}$$

After that, the spatial 2D convolution with the SAM module is performed to enhance the localization accuracy and perception range of the model, and the output is shown in Eq. 3:

$$F'' = M_s(F') \otimes F' \tag{3}$$

The method incorporates the channel attention mechanism and the spatial attention mechanism, CBAM module enables the model to re-sense and utilize the important information in the input feature maps through these operations, which improves the target detection capability.

3.3 Improved Head

In YOLOv8, the number of parameters in the header accounts for the majority of the total number of parameters because it involves a large number of convolution, feature map processing, and prediction operations. Therefore, in order to improve the inference speed and not too much affect the detection precision, the GhostConv[11] convolution module is used to replace the standard convolutional Conv in the header.

The GhostConv convolution module is shown in Figure 4 above, which is a series of linear transformations to uncover the required information from the original features with a small amount of computation. The improved head structure is shown in Figure 5 above. Taking YOLOv8n as an example, the number of head parameters is about 750,000, and about 393,000 after the replacement, which reduces the number of parameters by about half, and greatly accelerates the speed of detection and reasoning.

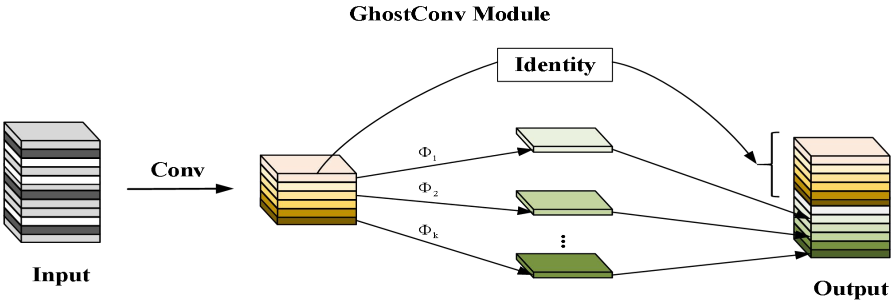


Fig. 4 GhostConv module

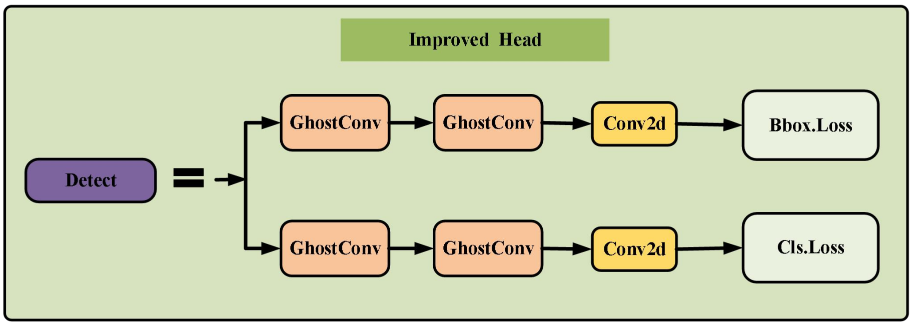


Fig. 5 Improved head

4. Experimental Results and Analysis

4.1 Datasets

In this paper, we use the open-source UA-DETRAC[12] vehicle detection dataset as the experimental dataset, which is based on images taken in Beijing and Tianjin, China. A total of 4000 images are selected from this dataset, and the training set is extended to 8000 images by data enhancement methods to ensure that the model can learn the vehicle information features in different environments and conditions during the training process, so as to improve the model's adaptability and generalization ability in various scenes.

4.2 Experimental Environment

The experiments in this paper use a cloud server for training, and the configuration is shown in Table 1 below. After many experiments, it is concluded that the convergence state can be reached when the number of iteration rounds of model training is 200, the initial learning rate is set to 0.01, and the optimizer is chosen to be Adam, which contributes to the stability of model training.

Table 1. Experimental environment configuration

Configuration	Version
GPU	RTX 3090(24GB) * 1
CPU	14 vCPU Intel(R) @2.80GHz
Operating system	Windows11
Internal memory	45GB
Python	v3.8
Cuda	v11.3

4.3 Evaluate Metrics

In order to intuitively and accurately represent the effect of the improved lightweight YOLOv8 model in detecting traffic flow in road traffic, the experiments in this paper adopt Precision, mAP@0.5, Recall and Parameters as the metrics of the improved model in this paper [13].

(1) Precision, an assessment metric that serves as a visual representation of the model's detection precision, is composed of Eq. 4:

$$P = \frac{TP}{TP + FP} \tag{4}$$

where TP indicates that vehicles in the test set are correctly classified as vehicles, FP indicates that non-vehicles in the test set are incorrectly classified as vehicles, TN indicates that vehicles in the test set are incorrectly classified as non-vehicles, and FN indicates that non-vehicles in the test set are classified as non-vehicles. The higher the value calculated by this method, the higher the accuracy of vehicle detection.

(2) mAP@0.5, which measures how good the model is at detecting on all categories, is an assessment metric that indicates the precision and comprehensiveness of the model's detection.

$$\mathrm{m}AP = \frac{\sum_{i=1}^{k} AP_i}{K} \tag{5}$$

where K is the number of classes and AP is the average precision of the i class.

(3) Recall, which is the ratio of the number of positive examples successfully identified by the model to the total number of actual positive examples, the closer the recall is to 1, the better the model performs in identifying positive examples.

$$R = \frac{TP}{TP + FN} \tag{6}$$

(4) Parameters, it directly affects the capacity and complexity of the model, usually more parameters means that the model can fit the complex data better, but at the same time the model file is larger, for mobile hardware resource constrained devices instead the smaller the smoother the operation.

4.4 Analysis of Experimental Results

4.4.1 Comparative analysis of algorithms

In this paper, YOLOv8n, a lightweight variant of YOLOv8, is selected as the basic network and improved as the above-mentioned network. In order to verify the feasibility and superiority of the improvements, a total of 8 sets of comparative experiments were set up on the experimental dataset, and the comparison range included the classic algorithm in the field of target detection

(Faster-RCNN, YOLOv3-tiny, YOLOv5s) and four model variants of YOLOv8 with different sizes (YOLOV8n, YOLOV8s, YOLOV8m, YOLOV8x), as well as the improved model in this paper. When conducting comparison experiments, the input image resolution is uniformly set to be uniformly adjusted to 640×640, the number of training Epoch is set to 200 rounds, the batchsize is set to 32, the learning rate is 0.01, and the rest of the hyper-parameter settings are consistent by default. The experimental results are shown in Table 2.

Table 2. Algorithm comparison test results

Datasets	Models	Precision/%	mAP@0.5/%	Recall/%	Parameters/M
UA-DET RAC	Faster-RCNN	80.31	65.21	58.45	25.6
	YOLOv3-tiny	83.88	67.55	60.69	12.1
	YOLOv5s	86.26	70.76	62.67	7.2
	YOLOV8n	87.13	72.35	64.18	3.2
	YOLOV8s	89.76	72.94	64.55	11.1
	YOLOV8m	90.45	74.18	68.49	25.8
	YOLOV8x	91.01	75.13	69.71	68.2
	Ours	91.17	75.45	70.01	3.1

From Table 2, it can be seen that the Ours model achieves 91.17% in Precision, 75.45% in mAP@0.5, and 70.01% in Recall. Compared with Faster-RCNN Precision, the method in this paper is improved by 13.5%, mAP@0.5 by 15.7%, and Recall by 18.8%. It is also 1.6% higher than YOLOv8s Precision, 3.4% more mAP@0.5, and 8.5% more Recall. Although the YOLOV8m and YOLOV8x variant models are similar to the proposed method in terms of detection precision, the number of parameters of the proposed model is much higher than that of the proposed method, which is not friendly to mobile hardware resource-constrained devices. The proposed method maintains high detection precision while maintaining a small number of parameters, which can well meet the requirements of lightweight model and real-time detection.

Figure 6 shows the data results obtained from the training of this paper's method, and it can be seen that the loss curve trained by this paper's method is very smooth, indicating that the model gradually converges to the local optimal solution during the training process. The Precision curves, mAP@0.5 curves, and Recall curves are consistent with the information in Table 2.

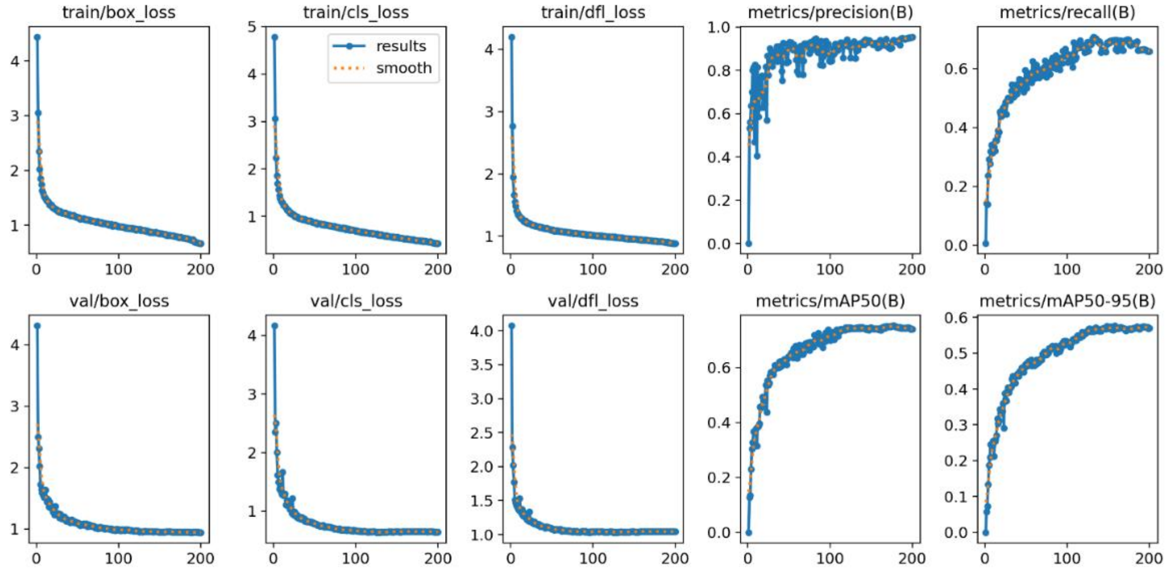
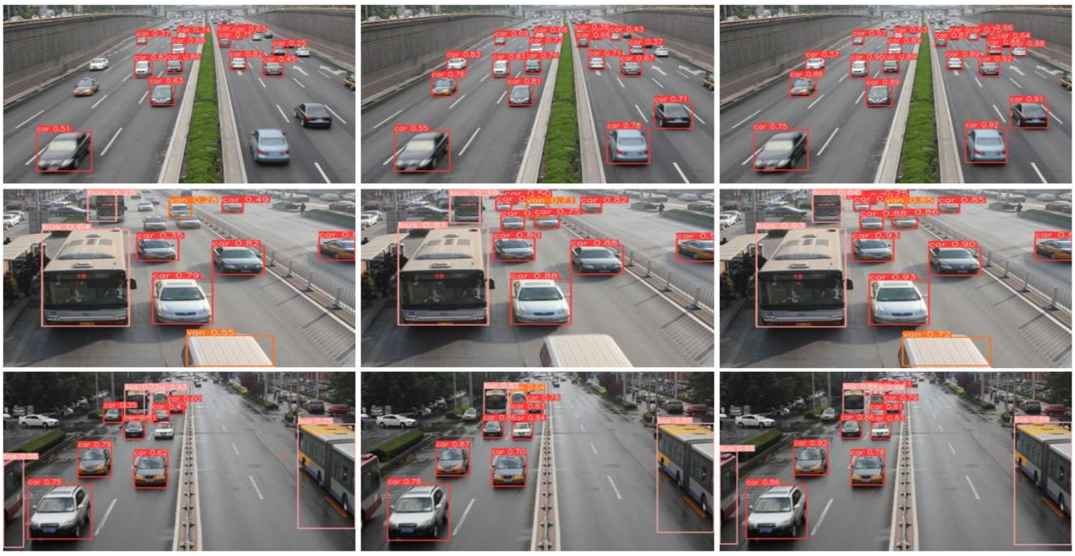


Fig. 6 Training results graph

Fig. 7 shows the comparison chart of vehicle detection effect, through comparative analysis, it is found that the method of this paper has better recognition effect compared to other methods and meets the demand of traffic flow detection.



(YOLOv3-tiny) (YOLOv8n) (Ours)

Fig. 7 Vehicle detection comparison results

4.4.2 Ablation experimental analysis

In order to verify the degree of influence of different improvement strategies on the initial model YOLOV8n, 1-3 improvement modules are added to the initial algorithm to compare the experimental data with the initial model respectively. The experimental comparison results are shown in Table 3.

Table 3. Comparative results of ablation experiments

Datasets	Models	Precision/%	mAP@0.5/%	Recall/%	Parameters/M
UA-DET RAC	——	87.13	72.35	64.18	3.2
	1：EfficientViT	89.74	74.24	66.68	3.4
	2：CBAM	88.24	73.17	65.71	3.3
	3：GhostConv	87.15	72.31	64.02	2.8
	4：CBAM + GhostConv	88.89	73.67	66.12	2.9
	5:EfficientViT + GhostConv	89.17	73.91	66.15	3.0
	6：EfficientViT + CBAM	90.56	75.01	69.22	3.5
	7：All	91.17	75.45	70.01	3.1

As can be seen from Table 3, Method 1 replaces the backbone network EfficientViT on the basis of the YOLOv8n model, and compared with the base YOLOV8n model, Precision is increased by 3%, and mAP@0.5 is increased by about 2.6%. Methods 2, 3 use CBAM and GhostConv, with little effect. Models such as Methods 4, 5, 6 use two of the three improved methods with significantly improved results. Method 7 is to add all three enhancement modules at the same time, the method is better than the rest of the methods in terms of detection precision, detection speed and the number of parameters, compared with the original model Precision is improved by 4.6%, mAP@0.5 is improved by 4.3%, and Recall is improved by about 9.1%.

5. Conclusion

The improvement method proposed in this paper can effectively improve the problems of YOLOv8 algorithm in vehicle detection, by replacing Backbone network with EfficientViT network, Neck introduces the CBAM attention mechanism, and Head replaces the GhostConv module, which effectively improves the overall detection precision and detection speed of the model. Through comparative experiments, the Precision, mAP@0.5 and Recall of the improved algorithm in this paper reach 91.17%, 75.45% and 70.01%, respectively, and the number of parameters is only 3.1M. While guaranteeing the detection precision, the detection speed is also faster than other methods, which meets the requirements of lightweight traffic flow detection model, and can perform real-time detection tasks well, meanwhile, this improved algorithm can be adapted to different datasets and achieve good results, and has good generalization ability and robustness.

References

[1] Zhai S, Shang D, Wang S, et al. DF-SSD: An improved SSD object detection algorithm based on DenseNet and feature fusion[J]. IEEE access, 2020, 8: 24344-24357.

[2] Jiang P, Ergu D, Liu F, et al. A Review of Yolo algorithm developments[J]. Procedia computer science, 2022, 199: 1066-1073.

[3] Tan M, Pang R, Le Q V. Efficientdet: Scalable and efficient object detection[C]//Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2020: 10781-10790.

[4] Bharati P, Pramanik A. Deep learning techniques—R-CNN to mask R-CNN: a survey[J]. Computational Intelligence in Pattern Recognition: Proceedings of CIPR 2019, 2020: 657-668.

[5] Purkait P, Zhao C, Zach C. SPP-Net: Deep absolute pose regression with synthetic views[J]. arXiv preprint arXiv:1712.03452, 2017.

[6] Pham V, Pham C, Dang T. Road damage detection and classification with detectron2 and faster r-cnn[C]//2020 IEEE International Conference on Big Data (Big Data). IEEE, 2020: 5592-5601.

[7] Liu Q, Liu Y, Lin D. Revolutionizing target detection in intelligent traffic systems: Yolov8-snakevision[J]. Electronics, 2023, 12(24): 4970.

[8] Bai R, Shen F, Wang M, et al. Improving detection capabilities of YOLOv8-n for small objects in remote sensing imagery: towards better precision with simplified model complexity[J]. 2023.

[9] Cai H, Li J, Hu M, et al. Efficientvit: Lightweight multi-scale attention for high-resolution dense prediction[C]//Proceedings of the IEEE/CVF International Conference on Computer Vision. 2023: 17302-17313.

[10] Woo S, Park J, Lee J Y, et al. Cbam: Convolutional block attention module[C]//Proceedings of the European conference on computer vision (ECCV). 2018: 3-19.

[11] Cao J, Bao W, Shang H, et al. GCL-YOLO: A GhostConv-based lightweight yolo network for UAV small object detection[J]. Remote Sensing, 2023, 15(20): 4932.

[12] Wen L, Du D, Cai Z, et al. UA-DETRAC: A new benchmark and protocol for multi-object detection and tracking[J]. Computer Vision and Image Understanding, 2020, 193: 102907.

[13] Oksuz K, Cam B C, Akbas E, et al. Localization recall precision (LRP): A new performance metric for object detection[C]//Proceedings of the European conference on computer vision (ECCV). 2018: 504-519.