# Vehicle Detection Algorithm in Complex Scenes Based on Improved YOLOv8

Qing Yu, Xinyu Ouyang, Bochao Su, Nannan Zhao and Hongman You

***Abstract*—An algorithm based on improved YOLOv8 for detecting densely small-scale vehicles in complex scenes is designed. Using YOLOv8s as a baseline model, the Global Attention Module (GAM) is first introduced into the backbone network of YOLOv8 to improve the ability to extract detailed features in complex scenes. Secondly, the Efficient RepGFPN-ASFF network structure is used instead of the original neck network and detection heads to enhance the fusion of shallow detail features and deeper high-level features, and weighting parameters are introduced to improve the network's interest in dense, small-scale vehicles based on multilevel functionality. Then, an inner-SIoU loss function is adopted, and the size of the auxiliary border is controlled by adjusting the scale factor ratio, which improves the model's detection performance for overlapping occluded vehicle targets. Finally, the robustness of the model in complex scenes is verified in different datasets, and trained and tested in a self-built Complex Vehicle Dataset (CVD) using transfer learning. The experimental results show that compared with the original model, the precision, recall and mAP@0.5 of the optimised model are improved by 4.3%, 6.2% and 6.0%, respectively, and its detection effect is significantly improved, which proves the effectiveness and reliability of the algorithm.**

***Index Terms*—Improved YOLOv8, vehicle detection, complex scenes, Global Attention Mechanism (GAM), loss function.**

## I. Introduction

With the continuous development of the world economy, the number of cars in the world is also increasing rapidly. So, although cars have greatly facilitated people's daily lives, they have also caused a series of serious social problems, such as frequent traffic accidents, traffic congestion, and safety hazards caused by illegal parking. In order to solve these complex traffic problems, the use of intelligent traffic management systems has become a trend, and vehicle object detection is one of the key technologies of this system. Effectively solving these problems is of great significance for improving the digitalization and intelligence level of road traffic management [1].

Currently, object detection algorithms based on deep learning mainly include two types: one is single-stage object detection algorithms, such as YOLO [2], SSD [3], RetinaNet [4]. The other is a two-stage object detection algorithm based on candidate regions, such as R-CNN [5], SPP-Net [6], Faster R-CNN [7]. In recent years, scholars have conducted many studies on vehicle detection based on this. For example, references [8], [9], and [10] proposed an improved YOLO algorithm, which improved the detection capability of the model in dim scenes, by replacing the backbone network and adopting the image dark-light enhancement method. References [11] and [12] added new modules to the backbone network to improve the robustness of the model, and to solve the problem of small targets. In [13], the improved Inception module was added to the SSD network to improve the detection ability of small target vehicles. In the literature [14], the S-YOLOv3 based on the YOLOv3 [15] was presented, and the extraction capability of features was improved by using ResNet [16]. In the literature [17], a feature fusion module was designed by introducing attention mechanism, which not only improved the detection accuracy of small target vehicles, but also ensured real-time performance. In the literature [18], the feature extraction effect of the model was further enhanced, by replacing the conventional convolution in YOLOv4 with a deformable convolution [19]. while in reference [20], a coordinate attention module was inserted , which reduces the loss of feature information and improves the accuracy of vehicle detection. Literature [21] combined YOLOv7 [22], [23] and GhostNet [24], [25] to reduce the number of parameters. Finally, reference [26] improved the overall accuracy of the model by improving the loss function of the YOLOv3 algorithm.

However, there are still some issues that need to be addressed. First, due to the large number of pedestrians, intersections, buildings, and other interference factors on the roads, it is necessary to make the dataset more rich and diverse. Secondly, due to the different angles and distances of the firing vehicles, the target vehicles can be blocked to varying degrees, increasing the difficulty of model detection. Finally, when there is a lack of training samples for different scenes in the training data, the generalizability of the model performance will be reduced. Therefore, how to improve the accuracy of vehicle detection in complex scenes and reduce the missed detection rate has become a major challenge in this research field. The main contributions include:

(1) Introduce attention module. The GAM [27] is introduced into the backbone network. When processing vehicle images, it enables the model to automatically pay attention to the important parts of the image, so as to extract richer feature information.

(2) Improve feature pyramid structure. Replace the original structure with the Efficient RepGFPN-ASFF network structure, combining the deep and shallow feature information of the network to increase the interest in various vehicle targets.

Manuscript received September 5, 2024; revised January 24, 2025.

Qing Yu is a postgraduate student of School of Electronic and Information Engineering, University of Science and Technology Liaoning, Anshan, Liaoning, 114051, China (e-mail: 1725274801@qq.com).

Xinyu Ouyang is a professor of the School of Electronic and Information Engineering, University of Science and Technology Liaoning, Anshan, Liaoning, 114051, China. (Corresponding author, e-mail: 13392862@qq.com.)

Bochao Su is a professor of the Institute of Future Technology, Shenzhen Polytechnic University, Nanshan District, Shenzhen, 518000, China (Corresponding author, e-mail: subochao@szpt.edu.cn).

Nan-Nan Zhao is a professor of the School of Electronic and Information Engineering, University of Science and Technology Liaoning, Anshan, Liaoning, 114051, China (e-mail: 723306003@qq.com).

Hongman You is a postgraduate student of School of Electronic and Information Engineering, University of Science and Technology Liaoning, Anshan, Liaoning, 114051, China (e-mail: 738525169@qq.com).

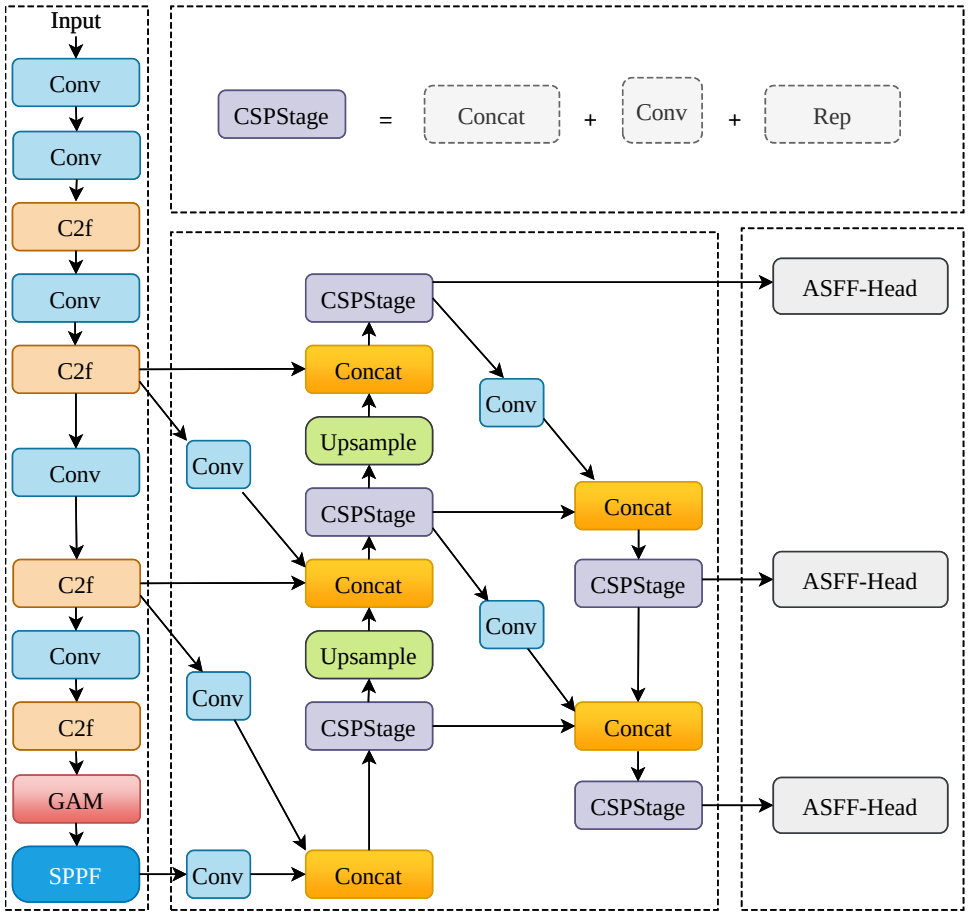



Fig. 1. Improved YOLOv8 structure diagram

(3) Replace the loss function. Improving the loss function to inner-SIoU improves the boundary regression accuracy by adjusting the scale factor ratio.

(4) Construct the dataset. The CVD provides a large amount of data support for the training and testing of vehicle detection algorithms, enhancing the richness and diversity of the dataset.

(5) Adopt the idea of transfer learning [28]. First, use the COCO vehicle dataset as the source domain to train the YOLOv8 model, and then use the CVD as the target domain to train and test the improved YOLOv8.

## II. Related Work

Due to its simplicity and efficiency, YOLOv8 has a good performance among many target detection algorithms. It can be divided into five models according to the network width and depth. The parameters are YOLOv8-n, s, m, l and x from small to large. After comprehensive consideration, YOLOv8s is selected as the baseline model for this improvement.

The YOLOv8 structure includes: Input, Backbone, Neck and Head. In the Input stage, the image to be detected will be pre-processed by random segmentation, splicing and normalization to enable the network to handle a variety of images, and enhance the generalization ability of the network. Backbone first uses the Conv module to implement downsampling operations without losing information, and then uses the C2f module to extract features from shallow to deep. Finally, the SPPF module uses convolution kernels of different sizes to max pool the feature maps and concatenate the results. The Neck network achieves the fusion of shallow detail semantics and deep advanced semantics, by using the top-down FPN structure and the bottom-up PAN structure. Head is a prediction network, which outputs prediction results on three feature maps of different sizes.

## III. Improved Model

The network structure of the improved YOLOv8 is shown in Fig. 1. The GAM is introduced in the deep layer of the backbone network, the feature pyramid network is replaced with the efficient RepGFPN, a new detection head ASFF-Head is constructed, and the loss function is improved to inner-SIoU.

### A. Introducing Attention Module

In order to improve the detection performance of YOLOv8 for complex backgrounds, GAM was introduced. The GAM can not only effectively reduce information loss, but also enhance the global interactivity of the network, thereby improving network performance. The whole process is shown in Fig. 2.

Where $M_c$ and $M_s$ are channel and spatial attention maps respectively. $\otimes$ represents element-wise multiplication. In the following formulas (1) and (2), $F_1 \in R^{C*H*W}$ is the input feature, the intermediate state $F_2$ and output $F_3$ are defined as:

$$F_2 = M_c(F_1) \otimes F_1 \tag{1}$$

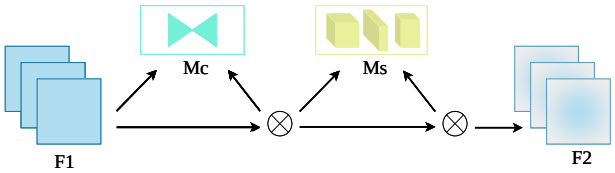



Fig. 2. GAM structure diagram

$$F_3 = M_s(F_2) \otimes F_2 \tag{2}$$

The whole structure mainly includes channel attention module and spatial attention module. The former is mainly used to store dimensional information. Its principle is based on 3D sequence, and enhances cross-dimensional channel-space dependency through a two-layer encoder-decoder structure (MLP). The channel attention submodule structure is shown in Fig. 3. The latter contains two convolutional layers, which helps to focus on more spatial information. To avoid information loss and performance impact caused by max pooling operations, pooling operations are omitted to better preserve mapping characteristics. The spatial attention submodule is shown in Figure 4.

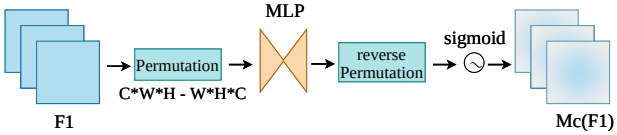



Fig. 3. Channel attention submodule

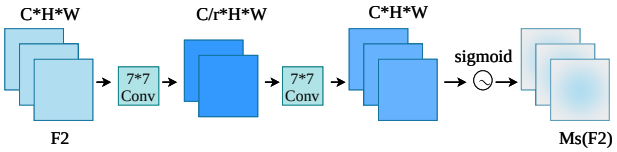



Fig. 4. Spatial attention submodule

### *B. Improve Feature Pyramid Structure*

The traditional feature pyramid network (FPN) [29] merges multi-scale features through a top-down approach, but this increases the amount of computation. The bidirectional feature pyramid network (BiFPN) [30] improves model performance, by removing nodes with only a single input and adding direct paths between the same levels. The global feature pyramid network (GFPN) [31] can achieve better performance, However,it shares the same channel dimension between features of different scales, and its Queen-Fusion mechanism involves a large number of upsampling and downsampling operations. The efficient representative global feature pyramid network (Efficient RepGFPN) [32] improves these problems. It controls the computational cost by setting different channel dimensions for feature maps of different scales, and effectively captures multi-scale features in vehicle images. At the same time, on the basis of ensuring real-time detection, it removes the redundant upsampling steps in Queen-Fusion, so as to perform feature fusion and processing more efficiently. The network structure of Efficient RepGFPN is shown in Fig. 5.

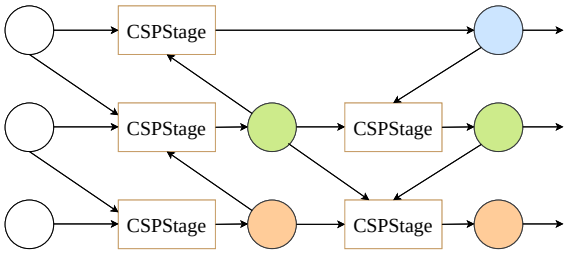



Fig. 5. Efficient RepGFPN network structure

CSPStage is the core fusion module of the Efficient RepGFPN network. First, the input feature maps are concatenated using the Concat operation. Then, the channel dimension is reduced through 1x1 convolution. Next, multiple Rep 3x3 convolutions and 3x3 convolutions are used to transform the features and generate multiple output layers. Finally, these output layers are concatenated again through Concat to obtain the final result. The structure of CSPStage is shown in Fig. 6.

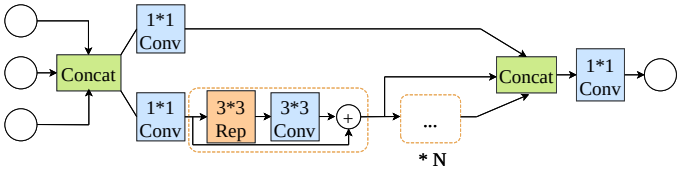



Fig. 6. CSPStage network structure

The structural feature of Rep is that it is heavily parameterized. It uses two branches during training and merges them together during reasoning, greatly saving inference time. Its structure is shown in Fig. 7.

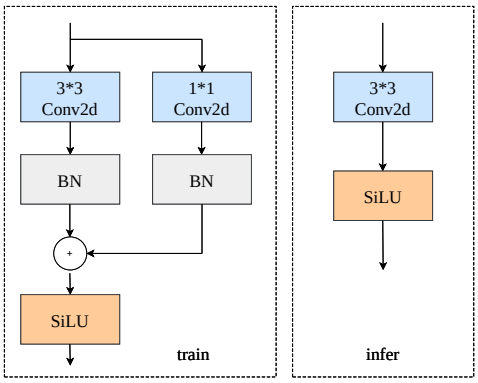



Fig. 7. Rep network structure

Combined with the above structure, the network structure of Efficient RepGFPN-ASFF is shown in Fig. 8.

The core feature of ASFF [33] is to perform feature filtering through learning parameters, so when applying this structure, if the scope of the information object is limited to the attention information at the current level, then based on the algorithmic logic of the structure, the information will be layered, which will help improve the learning efficiency of the model. For example, ASFF-1 contains feature layers of different scales, each of which is multiplied by the corresponding weight parameters $\alpha$, $\beta$, $\gamma$, and then these

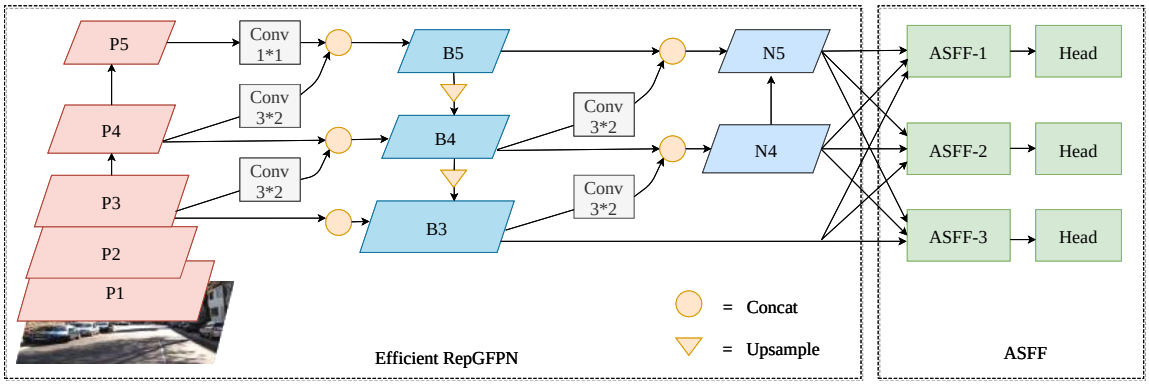



Fig. 8. Efficient RepGFPN-ASFF network structure

results are added together to obtain the following formula:

$$y_{ij}^{1} = \alpha_{ij}^{1} \cdot x_{ij}^{1\to 1} + \beta_{ij}^{1} \cdot x_{ij}^{2\to 1} + \gamma_{ij}^{1} \cdot x_{ij}^{3\to 1} \tag{3}$$

Where $y_{ij}^{1}$ is the new feature map obtained by ASFF-1, which is the output of different layer weights $\alpha_{ij}^{1}$, $\beta_{ij}^{1}$, $\gamma_{ij}^{1}$ and different features $x_{ij}^{1\to 1}$, $x_{ij}^{2\to 1}$, $x_{ij}^{3\to 1}$. Since addition is used, it is necessary to ensure that the ASFF layer obtains the same output feature dimension and number of channels from different levels. The size of the convolution kernel is $1 \times 1$, and it uses convolution layers with the same number of channels. The sum of the weight parameters $\alpha$, $\beta$, and $\gamma$ is 1, and the value of [0.1] is locked by the normalization function.

### C. Replace Loss Function

The loss of YOLOv8 include distribution focus loss, category classification loss and bounding box regression loss. Compared with CIoU, SIoU considers the vector angle problem between the required regressions, so it has been verified experimentally that replacing CIoU with SIoU can improve the performance of bounding box regression. The SIoU loss function includes angle cost, shape cost, IoU cost and distance cost. Fig. 9 shows the calculation process of angle cost.

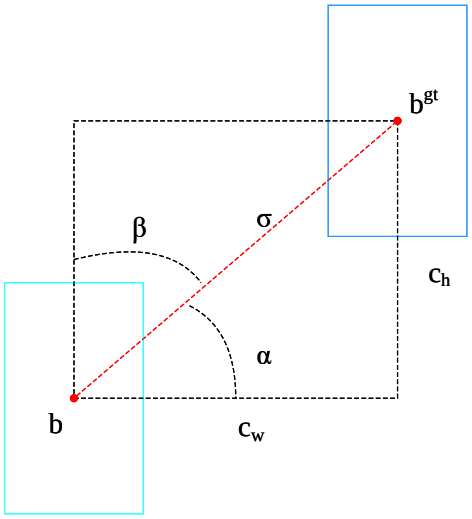



Fig. 9. Angle cost calculation process

In order to realize the above process, the following formula (4) is introduced for explanation.

$$\wedge = 1 - 2 \cdot \sin^2\left(\arcsin(x) - \frac{\pi}{4}\right)^{d} \tag{4}$$

Where:

$$x = \frac{c_h}{\sigma} = \sin(\alpha) \cdot e^{i\theta} \tag{5}$$

$$\sigma = \sqrt{(b_{c_x}^{gt} - b_{c_x})^2 - (b_{c_y}^{gt} - b_{c_y})^2} \tag{6}$$

$$c_h = \max(b_{c_y}^{gt}, b_{c_y}) - \min(b_{c_y}^{gt}, b_{c_y})^{d} \tag{7}$$

With the above angle cost, the distance cost can be redefined as shown in formula (8).

$$\triangle = \sum_{t=x,y} (1 - e^{-\gamma \rho_t})^{d} \tag{8}$$

Where:

$$\rho_x = \left(\frac{b_{c_x}^{gt} - b_{c_x}}{c_w}\right)^2, \quad \rho_y = \left(\frac{b_{c_y}^{gt} - b_{c_y}}{c_h}\right)^2 \tag{9}$$

$$\gamma = 2 - \wedge \tag{10}$$

When $\alpha \to 0$, the impact of distance cost is significantly reduced. On the contrary, when $\alpha$ is close to $\frac{\pi}{4}$, $\triangle$ is larger. Therefore, a shape cost needs to be defined, as shown in formula (11).

$$\Omega = \sum_{t=w,h} (1 - e^{-\omega t})^{\theta} \tag{11}$$

Where:

$$\omega_w = \frac{|w - w^{gt}|}{\max(w, w^{gt})}, \quad \omega_h = \frac{|h - h^{gt}|}{\max(h, h^{gt})} \tag{12}$$

Finally, SIoU is defined as formula (13).

$$SIOU = IoU - \frac{\triangle + \Omega}{2} \tag{13}$$

Among them, IoU is the intersection over union ratio:

$$IoU = \frac{|B \cap B^{gt}|}{|B \cup B^{gt}|} \tag{14}$$

Although the introduction of new loss terms speeds up the convergence of the model, these methods do not fully

consider the limitations of IoU itself. Therefore, the inner-IoU [34] is proposed to calculate IoU, which uses an auxiliary bounding box for calculation to enhance the model's generalization ability. Specifically, the size of the auxiliary bounding box is determined by a scaling factor $ratio$, and its calculation formula is shown in (15)-(16) below.

$$b_l = x_c - \frac{w \cdot ratio}{2}, \quad b_r = x_c + \frac{w \cdot ratio}{2} \tag{15}$$

$$b_t = y_c - \frac{h \cdot ratio}{2}, \quad b_b = y_c + \frac{h \cdot ratio}{2} \tag{16}$$

By using formulas (17)-(19), the vertex position of the auxiliary detection frame can be determined by transforming the center of the detection frame. This transformation also applies to the model's predicted detection frame and actual detection frame, where $b^{gt}$ and $b$ note the calculation outputs of the real detection frame and the predicted detection frame, respectively.

$$inter = (\min(b_r^{gt}, b_r) - \max(b_l^{gt}, b_l)) \cdot (\min(b_b^{gt}, b_b) - \max(b_t^{gt}, b_t)) \tag{17}$$

$$union = (w^{gt} \cdot h^{gt}) \cdot (ratio)^2 + (w \cdot h) \cdot (ratio)^2 - inter \tag{18}$$

$$IoU^{inner} = \frac{inter}{union} \tag{19}$$

Therefore, the core of inner-IoU is to calculate the intersection over union ratio between auxiliary bounding boxes. The scaling factor $ratio$ generally falls in the range of [0.5, 1.5]. When $ratio$ is less than 1, the auxiliary bounding box will be smaller than the actual bounding box, and the effective regression range will also be reduced. Compared with traditional IoU, the absolute value of its gradient is greater, which accelerates the convergence speed of high IoU samples. When $ratio$ is greater 1, the opposite is true, which is not conducive to high IoU regression. Therefore, introducing inner-IoU to transform SIoU can significantly improve detection performance. As shown in the following formula (20).

$$SloU^{\text{inner}} = IoU^{\text{inner}} - \frac{\triangle + \Omega}{2} \tag{20}$$

## IV. Experiments

### A. Experimental Condition Setting

The number of training rounds is set to 200, the image size is 640×640, the batch size is 32. The initial value of the learning rate is set to 0.001, the momentum size is set to 0.98, and the weighted decay parameter is set to 0.001. The model is based on the pytorch framework using CUDA 11.7, with an operating system of Ubuntu 18.04. The a GPU processor of Quadro RTX5000×2, the RAM memory size of 128G, and a hard disk size of 4TB.

### B. Experiment Dataset

Two datasets are used in this paper, one of which is the COCO vehicle dataset. The image in this dataset has a single viewpoint, which is less accurate in real scenarios, and has more misdetections and omissions when encountering complex and changing scenarios. Therefore, this dataset is used to train the network in advance to form a pre-trained model, which saves time and resources for further training. The other dataset is a combination of the CVD, the VisDrone, the KITTI, and some selected COCO vehicle datasets.

During training, the YOLOv8 model is first trained using the COCO vehicle dataset as the source domain. Then the improved model is initialised using the weight parameters obtained from the training. Finally, the CVD is used as the target domain to train the model again. The CVD contains 9604 vehicle images with occluded and small scale targets, which are randomly divided into training, validation and test sets in the ratio of 8:1:1.

### C. Evaluation Index

In order to evaluate the performance improvement of the improved algorithm for vehicle detection in complex scenes, the calculation formula for the evaluation indicators used in this paper is as follows:

$$P = \frac{TP}{TP + FP} \tag{21}$$

$$R = \frac{TP}{TP + FN} \tag{22}$$

$$AP = \int_0^1 P(r)\, dr \tag{23}$$

$$mAP = \frac{1}{N} \sum_{i=1}^{N} AP_i \tag{24}$$

where P, R, AP, and mAP represent precision, recall, average precision and mean average accuracy, respectively.TP and TN denote the positive and negative samples with correct predictions, while FP and FN denote the positive and negative samples with incorrect predictions, respectively. Draw mAP@0.5 curve can more intuitively reflect the improvement of the algorithm.

### D. Ablation Studies

In order to verify the effectiveness of the improved module, the YOLOv8s algorithm is used as a benchmark, and these modules are introduced sequentially for the ablation experiments: the GAM, the Efficient RepGFPN-ASFF, and the inner-SIoU. The mAP@0.5 curve is shown in Fig. 10, and the comparative results of the ablation experiment are shown in Table I.

The experimental results are shown in Table I, and in combination with the mAP@0.5 curve, it can be seen that the improved YOLOv8s algorithm improves P, R, mAP@0.5 and mAP@0.5 to 0.95 by 4.3%, 6.2%, 6.0% and 5.6%, respectively, compared to the YOLOv8s. The first group of experiments represents the baseline performance without any improvement of the YOLOv8s algorithm. The second group of experiments only added the GAM. Its performance was

TABLE I
ABLATION EXPERIMENT

| Models | GAM | RepGFPN | ASFF | Inner-SIOU | P | R | mAP@0.5 | mAP@0.5:0.95 |
|---|---|---|---|---|---|---|---|---|
| 1 | | | | | 0.837 | 0.676 | 0.763 | 0.485 |
| 2 | ✓ | | | | 0.849 | 0.694 | 0.782 | 0.499 |
| 3 | ✓ | ✓ | | | 0.871 | 0.702 | 0.784 | 0.494 |
| 4 | ✓ | ✓ | ✓ | | **0.883** | 0.719 | 0.805 | **0.548** |
| 5 | ✓ | ✓ | ✓ | ✓ | 0.880 | **0.738** | **0.823** | 0.541 |

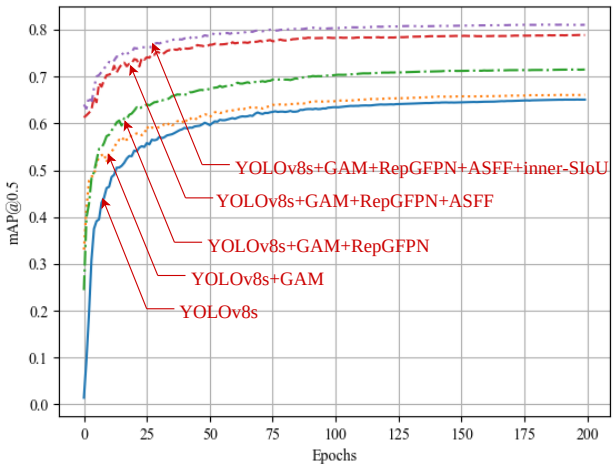



Fig. 10. mAP@0.5 curve

compared with the basic group. The four indicators were improved by 1.2%, 1.8%, 1.9% and 1.4% respectively. The third group of experiments used the GAM and the RepGFPN network structure at the same time. Compared to the previous group, except for the fourth index, the first three indexes were improved by 2.2%, 0.8%, and 0.2% respectively. The fourth group of experiments used the GAM and the RepGFPN-ASFF network structure at the same time. Compared with the third group of experiments, the four indexes were improved by 1.2%, 1.7%, 2.1%, and 5.4% respectively. In the fifth set of experiments, all four components were added. While there was a slight decrease in P and mAP@0.5 to 0.95, there was a significant improvement in R and mAP@0.5, which increased by 1.9% and 1.8%, respectively. Therefore, the improved YOLOv8 vehicle detection algorithm greatly improves the detection accuracy with a slight increase in the model size, and can more accurately detect occluded and small-scale vehicles in complex scenes.

For the ratio value in the inner-SIoU, the ablation experiment results of trying different values are shown in Table II. When $ratio$ = 0.5, the experimental effect is the best. Because vehicles in complex environments are small detection targets with low IoU, it is more conducive to rapid regression with low IoU when the annotation box is also small. When $ratio$ > 1, the regression speed will be slightly slowed down. Therefore, effectively adjusting the $ratio$ can achieve optimization of the entire model.

Under the same experimental conditions, for SSD, Faster RCNN, YOLOv5s, YOLOv7-tiny, YOLOv8s, and YOLOv9c algorithms were compared on the CVD to verify the performance of their algorithms. The results obtained from the

TABLE II
COMPARISON OF EFFECTS OF DIFFERENT RATIO VALUES

| ratio | P | R | mAP@0.5 | mAP@0.5:0.95 |
|---|---|---|---|---|
| 0.5 | **0.880** | **0.738** | **0.823** | **0.541** |
| 0.6 | 0.873 | 0.734 | 0.815 | 0.535 |
| 0.7 | 0.871 | 0.722 | 0.804 | 0.523 |
| 0.8 | 0.869 | 0.719 | 0.801 | 0.515 |
| 1.0 | 0.872 | 0.720 | 0.806 | 0.529 |
| 1.25 | 0.868 | 0.717 | 0.798 | 0.513 |
| 1.5 | 0.869 | 0.716 | 0.797 | 0.511 |

experiments are shown in Table III.

TABLE III
PERFORMANCE COMPARISON OF MAINSTREAM MODELS

| Models | P | R | mAP@0.5 |
|---|---|---|---|
| SSD | 0.612 | 0.509 | 0.574 |
| Faster-RCNN | 0.634 | 0.521 | 0.613 |
| YOLOv5s | 0.712 | 0.652 | 0.721 |
| YOLOv7-tiny | 0.627 | 0.531 | 0.615 |
| YOLOv8s | 0.837 | 0.676 | 0.763 |
| YOLOv9c | 0.876 | 0.750 | 0.834 |
| Ours | 0.880 | 0.738 | 0.823 |

According to Table III, the proposed algorithm has great advantages in P, R and mAP@0.5 indicators compared with most other algorithms. However, when compared with YOLOv9c, it is slightly lower in terms of R and mAP@0.5. Nevertheless, the proposed algorithm has fewer parameters and lower computational complexity, achieving a better balance between accuracy and real-time performance.

In order to further verify the robustness and generalisability of the improved algorithm, experiments were carried out on different datasets to observe its performance effect, and the results are shown in Table IV. For the COCO, KITTI and UA-DETRAC datasets, the improved model has increased by 2.6%, 4.1% and 4.7% on mAP@0.5 over the original model, respectively, but with varying increases in the Params and the GFLOPs.However, on the CVD, the mAP@0.5 increases by 6.0%. When applied to our proposed model, the number of parameters decreases when GFLOPs increases slightly. The experimental results show that the improved algorithm outperforms the original algorithm on different datasets, and the performance is more prominent on the self-constructed CVD, which further confirms the good robustness.

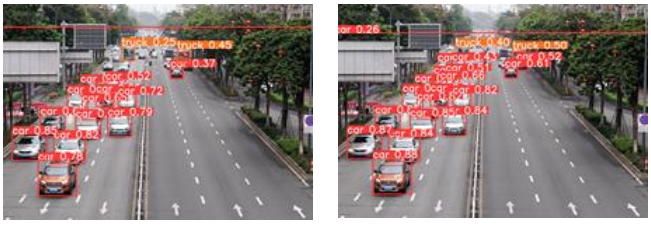
### *E. Visualization Analysis*

In order to visualise the improvement effect after each operation, a comparison picture of the detection results is given. As shown in Fig. 11(a) shows the COCO vehicle

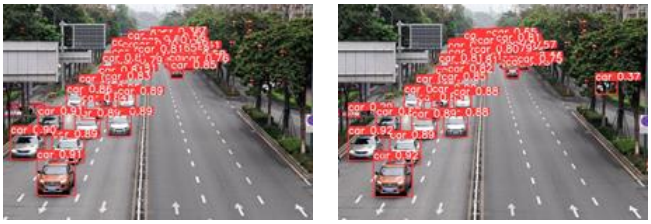
TABLE IV
COMPARISON OF EXPERIMENTS WITH DIFFERENT DATASETS

| Datasets | Models | mAP@0.5 | Params/M | GFLOPs/G |
|---|---|---|---|---|
| COCO | YOLOv8s | 0.715 | 11.1 | 28.4 |
| | Ours | 0.741 | 17.1 | 33.7 |
| KITTI | YOLOv8s | 0.728 | 11.1 | 28.4 |
| | Ours | 0.769 | 15.6 | 37.1 |
| UA-DETRAC | YOLOv8s | 0.734 | 11.1 | 28.4 |
| | Ours | 0.781 | 17.6 | 39.7 |
| CVD | YOLOv8s | 0.763 | 11.1 | 28.4 |
| | Ours | 0.823 | 10.2 | 36.5 |

dataset and Fig. 11(b) shows the Complex Vehicle Dataset, the left and right pictures are the training results of the original model and the improved model, respectively.



(a) COCO

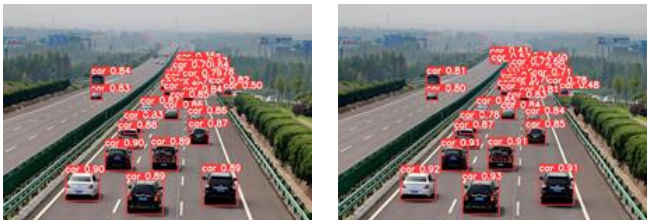


(b) CVD

Fig. 11. Comparison of COCO and CVD detection results

The results in Fig. 11 show that the models trained using the COCO vehicle dataset have many misdetections and miss-detections, both in the original model and the improved model, especially for the small-scale vehicles behind and those obscured by trees. Moreover, the model trained with the CVD is more generalizable, and the training effect is significantly better than the COCO vehicle dataset with higher detection accuracy. However, it is undeniable that the improved model outperforms the original model on both datasets.

Fig. 12 compares the detection results of the original YOLOv8s (left) and the improved YOLOv8s (right). It can be seen that the improved model has significant advantages in small scale and false detection in dense scenarios. From Fig. 12(a), it can be seen that the original algorithm can't detect the small-scale vehicles behind the left lane when the target size changes and there are more small target vehicles, while the improved algorithm can. From Fig. 12(b), it can be seen that the original algorithm misjudges the dotted line on the road as a vehicle, while the improved algorithm doesn't have the phenomenon of misjudgment, so the improved algorithm has a better recognition ability for the dense small-scale vehicle targets in complex scenes. In summary, the proposed algorithm has better performance in detecting small-scale vehicle targets in complex scenes, and can more effectively detect occlusions and small-scale vehicles, reducing missed detections and false detections.



(a) Small scale

(b) False detection

Fig. 12. Comparison of detection effects between YOLOv8s and improved YOLOv8s algorithms

## V. CONCLUSION

The paper proposes a dense vehicle detection method based on the improved YOLOv8 in complex scenes, which solves the problems such as low detection accuracy due to dense occlusion in this scene. To address the problem of a single sample dataset, data fusion is used to form a new Complex Vehicle Dataset from selected images from multiple datasets, which ensures the diversity of data samples.To address the problem of detecting occluded and dense small-scale vehicles, by means of transfer learning, this paper introduces the GAM in the deep layer of the backbone network, replaces the feature pyramid networks with the Efficient RepGFPN, constructs a new detection head ASFF-Head and improves the loss function to inner-SIoU. Then experiments are conducted on the CVD and the mAP@0.5 reaches 82.3% and the amount of parameters is reduced by 1.1 M. In addition, experiments were conducted on several datasets to verify the robustness of the model in complex scenes. It can be seen that the improved YOLOv8 algorithm provides more accurate detection of occluded dense small-scale vehicle targets in complex scenes, which verifies the effectiveness and generalisability of the algorithm in this paper.

## REFERENCES

[1] B. Jan, H. Farman, M. Khan, M. Talha, and I. U. Din, "Designing A Smart Transportation System: An Internet of Things and Big Data Approach," *IEEE Wireless Communications*, vol. 26, no. 4, pp. 73–79, 2019.

[2] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 779–788.

[3] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "SSD: Single Shot Multibox Detector," in *Computer Vision-ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part I 14*. Springer, 2016, pp. 21–37.

[4] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal Loss for Dense Object Detection," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 2980–2988.

[5] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 2961–2969.

[6] J. Jeon, B. Jeong, S. Baek, and Y.-S. Jeong, "Hybrid Malware Detection Based on Bi-LSTM and SPP-Net for Smart IoT," *IEEE Transactions on Industrial Informatics*, vol. 18, no. 7, pp. 4830–4837, 2021.

[7] Q. Fan, L. Brown, and J. Smith, "A Closer Look at Faster R-CNN for Vehicle Detection," in *2016 IEEE intelligent vehicles symposium (IV)*. IEEE, 2016, pp. 124–129.

[8] X. Liu and Y. Lin, "YOLO-GW: Quickly and Accurately Detecting Pedestrians in a Foggy Traffic Environment," *Sensors*, vol. 23, no. 12, p. 5539, 2023.

[9] X. Wang and C. Wang, "Vehicle multi-target detection in foggy scene based on foggy env-yolo algorithm," in *2022 IEEE 7th International Conference on Intelligent Transportation Engineering (ICITE)*. IEEE, 2022, pp. 451–456.

[10] X. Chu, B. Zhang, and R. Xu, "Moga: Searching Beyond Mobilenetv3," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 4042–4046.

[11] Q. Xu, R. Lin, H. Yue, H. Huang, Y. Yang, and Z. Yao, "Research on Small Target Detection in Driving Scenarios Based on Improved YOLO Network," *IEEE Access*, vol. 8, pp. 27 574–27 583, 2020.

[12] M.-T. Pham, L. Courtrai, C. Friguet, S. Lefèvre, and A. Baussard, "YOLO-Fine: One-Stage Detector of Small Objects Under Various Backgrounds in Remote Sensing Images," *Remote Sensing*, vol. 12, no. 15, p. 2501, 2020.

[13] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going Deeper with Convolutions," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 1–9.

[14] X. Liu, J. Cheng, Y. Gu, X. Lei, B. Wang, and Y. Cheng, "A Highway Distance Posts Detection Method Based on S-YOLOv3," in *2021 International Conference on Control, Automation and Information Sciences (ICCAIS)*. IEEE, 2021, pp. 974–979.

[15] L. Zhao and S. Li, "Object Detection Algorithm based on Improved YOLOv3," *Electronics*, vol. 9, no. 3, p. 537, 2020.

[16] A. K. Pandey, D. Jain, T. K. Gautam, J. S. Kushwah, S. Shrivastava, R. Sharma, and P. Vats, "Tomato leaf disease detection using generative adversarial network-based resnet50v2." *Engineering Letters*, vol. 32, no. 5, pp. 965–973, 2024.

[17] J. Lian, Y. Yin, L. Li, Z. Wang, and Y. Zhou, "Small Object Detection in Traffic Scenes Based on Attention Feature Fusion," *Sensors*, vol. 21, no. 9, p. 3031, 2021.

[18] Y. Cai, T. Luan, H. Gao, H. Wang, L. Chen, Y. Li, M. A. Sotelo, and Z. Li, "YOLOv4-5D: An Effective and Efficient Object Detector for Autonomous Driving," *IEEE Transactions on Instrumentation and Measurement*, vol. 70, pp. 1–13, 2021.

[19] Z. Hao, Z. Zhang, H. Li, B. Xu, X. Zhang, M. Xu, and W. Wang, "Multi-view 3D Reconstruction Based on Deformable Convolution and Laplace Pyramid Residuals," *IAENG International Journal of Computer Science*, vol. 51, no. 7, pp. 896–905, 2024.

[20] Y. Cao, C. Li, Y. Peng, and H. Ru, "MCS-YOLO: A multiscale Object Detection Method for Autonomous Driving Road Environment Recognition," *IEEE Access*, vol. 11, pp. 22 342–22 354, 2023.

[21] K. Zhao, L. Zhao, Y. Zhao, and H. Deng, "Study on Lightweight Model of Maize Seedling Object Detection Based on YOLOv7," *Applied Sciences*, vol. 13, no. 13, p. 7731, 2023.

[22] C.-Y. Wang, A. Bochkovskiy, and H.-Y. M. Liao, "YOLOv7: Trainable Bag-of-Freebies Sets New State-of-the-Art for Real-Time Object Detectors," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 7464–7475.

[23] K. Jiang, T. Xie, R. Yan, X. Wen, D. Li, H. Jiang, N. Jiang, L. Feng, X. Duan, and J. Wang, "An Attention Mechanism-Improved YOLOv7 Object Detection Algorithm for Hemp Duck Count Estimation," *Agriculture*, vol. 12, no. 10, p. 1659, 2022.

[24] K. Han, Y. Wang, Q. Tian, J. Guo, C. Xu, and C. Xu, "Ghostnet: More Features From Cheap Operations," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 1580–1589.

[25] W. Teng, H. Zhang, and Y. Zhang, "X-ray security inspection prohibited items detection model based on improved yolov7-tiny." *IAENG International Journal of Applied Mathematics*, vol. 54, no. 7, pp. 1279-1287, 2024.

[26] X. Sun, Q. Huang, Y. Li, and Y. Huang, "An Improved Vehicle Detection Algorithm Based on YOLOV3," in *2019 IEEE Intl Conf on Parallel & Distributed Processing with Applications, Big Data & Cloud Computing, Sustainable Computing & Communications, Social Computing & Networking (ISPA/BDCloud/SocialCom/SustainCom)*. IEEE, 2019, pp. 1445–1450.

[27] Z. Liu, L. Li, X. Fang, W. Qi, J. Shen, H. Zhou, and Y. Zhang, "Hard-rock tunnel lithology prediction with tbm construction big data using a global-attention-mechanism-based lstm network," *Automation in Construction*, vol. 125, p. 103647, 2021.

[28] A. Zhao, S. Xing, X. Wang, and J.-Q. Sun, "Radial Basis Function Neural Networks for Optimal Control with Model Reduction and Transfer Learning," *Engineering Applications of Artificial Intelligence*, vol. 136, p. 108899, 2024.

[29] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature Pyramid Networks for Object Detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 2117–2125.

[30] M. Tan, R. Pang, and Q. V. Le, "Efficientdet: Scalable and Efficient Object Detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 10 781–10 790.

[31] Z. Tan, J. Wang, X. Sun, M. Lin, H. Li *et al.*, "Giraffedet: A heavy-neck paradigm for object detection," in *International conference on learning representations*, 2021.

[32] G. Shi, J. Li, L. Shi, Y. Li, Y. Li, H. Ma, D. Sun, and C. Zhang, "Anomaly detection of transmission line large metal based on egfpn-yolo and uavs," in *International Conference on Intelligent Computing*. Springer, 2024, pp. 36–46.

[33] M. Qiu, L. Huang, and B.-H. Tang, "ASFF-YOLOv5: Multielement Detection Method for Road Traffic in UAV Images Based on Multiscale Feature Fusion," *Remote Sensing*, vol. 14, no. 14, p. 3498, 2022.

[34] P. Huang, S. Tian, Y. Su, W. Tan, Y. Dong, and W. Xu, "Ia-ciou: An improved iou bounding box loss function for sar ship target detection methods," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 14, pp. 10 569–10 582, 2024.