Improving the Vehicle Small Object Detection Algorithm of Yolov5

Yuanyuan Liu[1,*], Jianlin Zhu[1], Haili Ma[1]

[1]Engineering Innovation College (Engineering Training Center), Shanghai Institute of Technology, Shanghai, 201418, China

Received 14 May 2024; received in revised form 18 June 2024; accepted 19 June 2024

DOI: https://doi.org/10.46604/ijeti.2024.13717

Abstract

To address the problems of low accuracy and poor robustness in vehicle small object detection for autonomous driving tasks, this study aims to propose an improved vehicle small object detection algorithm model based on YOLOv5. Firstly, some convolutions in the backbone network are replaced with receptive field attention convolutions, and the weights of the convolution kernels are dynamically assigned based on the importance of image features to ensure the extraction of important features. Secondly, adding a channel attention mechanism to the backbone network enhances the attention to small target features. Finally, the Focal-EIoU loss function is introduced to increase the attention on high-quality samples in the regression stage of object detection boxes. When the model is applied to the small object test set of the KITTI dataset, the precision rate, recall rate and mean average precision are 88.5%, 82.8%, and 84.9%, respectively, and the frame processing rate reaches 87.83FPS.

Keywords: object detection, yolov5, autonomous driving, deep learning, attention mechanism

1. Introduction

Vehicle detection technology in autonomous driving environment perception is a research hotspot in computer vision for intelligent driving. In the task of vehicle detection in autonomous driving, even a tiny error may have catastrophic consequences. In vehicle driving, it is necessary to use complex methods to achieve high-precision real-time detection of the surrounding environment and transmit environmental information to the underlying of the vehicle for data fusion, analysis, and processing with the self-vehicle information, thereby realizing autonomous driving [1]. When cameras are used in the process of autonomous driving to detect distant vehicles, vehicles can make decisions in advance, making driving safer. However, in the acquired image data, the targets of distant vehicles are small and have few pixel values, making the detection of small vehicle targets a major challenge in the practical applications of autonomous driving. It can be seen that improving the accuracy and robustness of small vehicle target detection is of great significance for the safety and reliability of autonomous driving.

In recent years, deep learning methods are commonly used in vision-based vehicle detection tasks to analyze and process image data from on-board cameras. Deep learning object detection algorithm models can be divided into one-stage (One-stage) detection and two-stage (Two-stage) detection. One-stage detection models, represented by SSD [2] and the YOLO (You Only Look Once, YOLO) series [3-5], locate target boxes according to the regression logic, and predict them all at once after extracting image features through a convolutional neural network, with fast speed but slightly lower accuracy. Two-stage detection models, such as Fast R-CNN [6-7] and Faster R-CNN [8], generate detection proposal boxes by calculating multiple regions of interest (Regions of Interest, ROI) from the feature map. These models classify and locate the features within the

[*] Corresponding author. E-mail address: yuanyuanliu012@163.com

English language proofreader: Yen-Chun Hsieh

regions with high accuracy but at a slow speed, which cannot meet the requirements of environmental perception tasks in autonomous driving.

Ma et al. [9] improved detection accuracy by integrating the YOLOv3 and Deep-SORT algorithms and using object tracking. Dong et al. [10] improved the data augmentation method of the original YOLOv5s and optimized the non-maximum suppression priori box, effectively improving the missed detection of the original vehicle target. Liang et al. [11] introduced the S-RFB module into the YOLOv3 algorithm model to expand the model’s receptive field and better utilize contextual information, thereby improving the ability to detect small target vehicles. Ahmed et al. [12] proposed a technique for vehicle detection and classification in remotely sensed images based on transfer learning, using Mask Region-based Convolutional Neural Network (RCNN) technology to detect vehicles, and once detected, using Fuzzy Wavelet Neural Network (FWNN) models to classify them. Kong et al. [13] proposed a two-stage vehicle detection framework, first using convolutional layers with different receptive fields to alleviate the problem of scale variation. Additionally, a scale-based non-maximum suppression (NMS) to hierarchically filter redundant proposals from different levels of the feature pyramid was proposed. Dong et al. [14] use lightweight methods to reduce the YOLOv5 model parameters to ensure the effectiveness of the model, which is suitable for devices with smaller computing power.

Although the above-mentioned algorithms have achieved good results in the field of object detection, they still do not achieve ideal results in detecting small targets in autonomous driving scenarios. Therefore, this study proposes several improvements to the YOLOv5 algorithm. During the data input stage, image enhancement techniques such as skewing and stretching are implemented to increase the robustness of the trained detection model. Additionally, some convolutions in the backbone network are replaced with receptive field attention convolutions, which dynamically allocate convolutional kernel weights based on the importance of image features within the receptive field, thereby improving the network's feature extraction capability. Furthermore, an attention mechanism is introduced into the backbone structure of the model to enhance the feature extraction functionality of feature maps. Finally, the Focal EIoU loss function is incorporated at the output end to increase the focus on high-quality samples during the object detection bounding box regression phase, thereby improving detection accuracy.

2. YOLOv5 Algorithm

The network structure model of YOLOv5 is shown in Fig. 1. The Backbone layer consists of the Focus structure and the cross-stage partial (Cross Stage Partial, CSP) [15] Network structure composition: Focus is a special convolution operation that divides the feature image into four sub-images and stitches them together, allowing the neural network to fully parse image features. CSPNet serves as the backbone network to extract image features. The network structure adopts the CSPDarkNet-53 module, whose main idea is reflected in the C3 module, which realizes gradient shunt and effectively improves the accuracy of the model while having fewer model parameters and computing resources.

The Neck layer includes Spatial Pyramid Pooling (Spatial Pyramid Pooling, SPP) [16] Structure and Path Aggregation Network (Path Aggregation Network, PAN) [17] Structure. SPP pools and stitches feature maps of different sizes to enhance the model’s perception ability for targets of different scales. PAN performs upsampling and downsampling separately to fuse the semantic information and location information of the acquired images.

The Intersection over Union (IoU) loss at the output end of the head is a loss function for calculating the Bounding Box, which is used to calculate the difference between the predicted box and the real bounding box, and eliminates ineffective predicted bounding boxes through Non-Maximum Suppression. YOLOv5 is divided into four versions according to the number of network layers and the overall capacity, namely s, m, l, and x, with more network layers and capacity having higher accuracy, but resulting in larger model volume and slower inference time. Therefore, in considering the trade-off between accuracy and speed, this paper selects YOLOv5l as the benchmark model for the experiment.

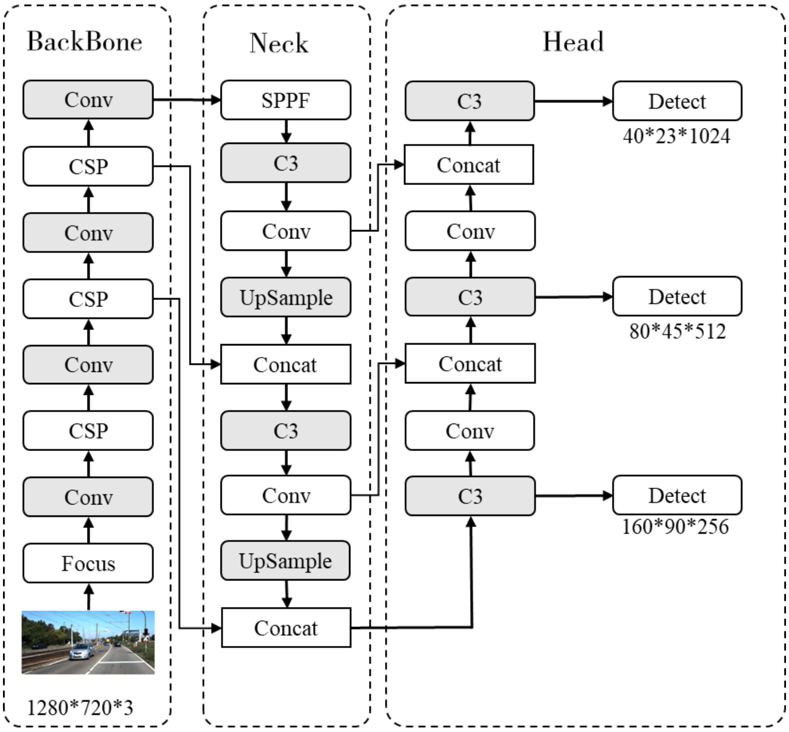


Fig. 1 YOLOv5 network structure

The YOLOv5 algorithm combines various neural network structures to fully extract image feature information and optimize the network structure to reduce computation and accelerate detection speed. However, the backbone network still lacks accuracy in feature extraction, and the IoU of the original bounding box loss function is greatly affected by the localization gap. For autonomous driving, which requires high accuracy in small target detection, the original YOLOv5 algorithm is difficult to apply to the actual detection of small vehicle targets. Therefore, the algorithm is improved to address these shortcomings.

3. Algorithm Improvement

Ordinary convolution uses the same convolution kernel in the feature extraction process, without considering the different information of features from different positions, nor the importance of each feature. Therefore, it limits the ability of feature extraction, thereby affecting the performance of the model. Based on this, this paper introduces the RFAConv (Receptive-Field Attention Convolution) [18].

3.1 Receptive-Field Attention Convolution

Ordinary convolution uses the same convolution kernel in the feature extraction process, without considering the different information of features from different positions, nor the importance of each feature. Therefore, it limits the ability of feature extraction, thereby affecting the performance of the model. Based on this, this paper introduces the RFAConv (Receptive-Field Attention Convolution) [18].

First, the feature map uses average pooling to aggregate the global information of each receptive field feature, uses 1x1 convolution to interact information, normalizes to emphasize the importance of each feature in the receptive field feature, grades the importance of different features in the receptive field slider, and prioritizes the features in the receptive field space, thereby ensuring that the obtained convolution kernel can extract important features. The attention map is used to allocate weights to the subsequent convolution kernel. Then, the original feature map is convolved to obtain the receptive field space feature with the same dimensions as the attention map. Both the attention map and the receptive field space feature are obtained through grouped convolution, reducing the number of parameters and the amount of computation in the network. Finally, the features of the receptive field space are extracted according to the weight of the attention map, and adjusted to the appropriate size to obtain the output result of the RFAConv. The calculation process of the RFAConv is shown in Equation (1):

$$\mathrm{F} = Soft\max(g^{1\times1}(\mathrm{AvgPool(X)})) \times \mathrm{ReLU}(\mathrm{Norm}(\mathrm{g}^{k\times k}(X))) = Arf \times Frf \tag{1}$$

Among them, the grouped convolution represents a size of i×i, k represents the size of the convolution kernel, Norm represents normalization, X represents the input feature map, and the output feature F is obtained by multiplying the attention map with the transformed receptive field space feature. The process of the RFAConv is shown in Fig. 2.

RFAConv considers the importance of features within each receptive field in convolutional neural networks, addressing the performance limitations of ordinary convolutions due to shared parameters and insensitivity to positional changes. By combining the spatial features of the receptive field with convolution operations, non-parametric shared convolution operations were achieved.

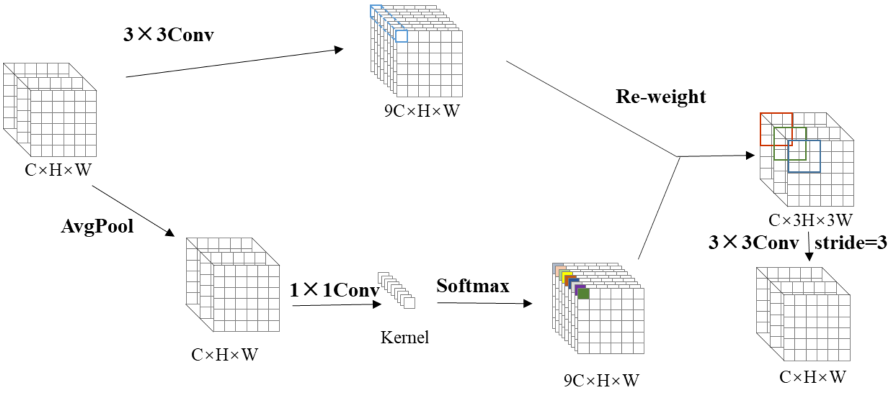


Fig. 2 Process diagram of RFAConv

3.2 Network structure

Attention mechanism is a technique used to enhance the performance of object detection, helping the network better focus on important features and target areas. The Squeeze-and-Excitation (SE) attention mechanism module [19] differs from other works by enhancing the network's feature perception ability from the perspective of enhancing spatial dimension coding. Unlike others, it focuses on the network channel dimension, modeling the dependency relationships between channels to adaptively adjust the feature response values of each channel. Adding the SE module to the neural network structure greatly improves network performance while consuming very little computation. The basic structure of the SE module is shown in Fig. 3, where is a convolutional structure, X (C'×H'×W') and U (C×H×W) are the input and output of respectively, which exist in other network structures. The specific process of the SE template is: first perform a global average pooling (GAP) on U, called the Squeeze process. The output 1×1×C data is subjected to a two-level fully connected, called the Excitation process. Finally, a sigmoid activation function is used to multiply the scale calculated by the computation module onto U as the input data for the next level.

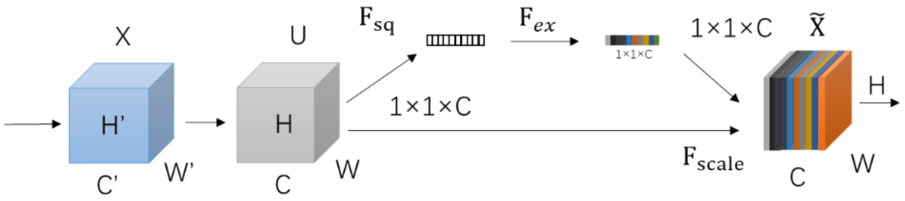


Fig. 3 Squeeze-and-Excitation block

The formula for the squeeze operation is:

$$F_{sq}\left(u_c\right)=\frac{1}{H\times W}\sum_{i=1}^{H}\sum_{j=1}^{W}u_c\left(i,j\right) \tag{2}$$

Among them, the pixel value of the channel represents c at position (i, j) in the feature image. H and W represent the height and width of the two-dimensional convolution spatial dimension features, respectively. To utilize the correlation between channels, the average value of information for all points in space is obtained, and finally, the calculated scale is applied to the entire channel. The formula for the excitation operation is as follows:

$$F_{ex}\left(z,w\right)=Sigmoid\left(W_2\times ReLU\left(W_1\times z\right)\right) \tag{3}$$

Among them, W_1 and W_2 are the weight matrices of the FC layer, and z represents the value obtained by the squeezing operation. The excitation part is implemented through two levels of fully connected: the first level compresses the C-layer channels through the ReLU function, and the second level restores to C channels through the Sigmoid function, so that the correlation of the channels is utilized to calculate the effective scale and add attention to the network. The SE module explicitly models the interdependence between channels, enabling the model to better understand and distinguish the features of different channels, which helps to improve the accuracy of vehicle detection.

3.3 Loss function

In YOLOv5, the loss of bounding boxes is calculated to judge the quality of the model in the training stage. Based on the performance of the model in target category classification and bounding box confidence, training and optimization are performed using backward propagation and gradient descent methods. The calculation formula for IoU Loss is:

$$\text{IoU}=\frac{\left|B\cap B^{gt}\right|}{\left|B\cup B^{gt}\right|} \tag{4}$$

$$L_{IoU}=1-\text{IoU} \tag{5}$$

Here, B represents the predicted bounding box, and G represents the ground truth bounding box. A larger IoU value indicates greater overlap between the two boxes, reflecting a better detection effect. However, when the predicted bounding box does not intersect with the ground truth bounding box, both the IoU value and the gradient are 0, which fails to reflect the proximity of the two boxes, making it impossible to perform backpropagation and optimization. Moreover, it ignores the imbalance issue in bounding box regression, that is, a large number of bounding boxes with small overlaps with the target bounding box contribute the most to the optimization of bounding box regression. Therefore, this paper introduces the Focal-EIoU loss function. Among them, the EIoU Loss consists of three components: the IoU loss, the distance loss, and the width-height loss (overlap area, center point distance, width-height ratio), as shown in the following formula:

$$L_{EIoU}=1-IoU+\frac{\rho^2\left(B,B^{gt}\right)}{\left(w^c\right)^2+\left(h^c\right)^2}+\frac{\rho^2\left(w,w^{gt}\right)}{\left(w^c\right)^2}+\frac{\rho^2\left(h,h^{gt}\right)}{\left(h^c\right)^2} \tag{4}$$

Among them, w^c and h^c represent the width and height of the minimum bounding rectangle of the predicted bounding box and the ground truth bounding box, respectively. The ρ represents the Euclidean distance between two points. The width-height loss directly minimizes the difference in height and width between the predicted bounding box and the ground truth bounding box, resulting in faster convergence speed and better localization results during the training process. However, due to the imbalance in the number of high-quality and low-quality predicted bounding boxes in an image, there will be a large gradient during the regression, affecting the training results. Therefore, directly using EIoU Loss may not yield the desired results. To address this issue, the author combined Focal Loss to propose Focal-EIoU Loss, which separates high-quality and low-quality anchor boxes from a gradient perspective. The formula is as follows:

$$L_{Fcoal-EIoU} = IoU^{\gamma} L_{EIoU} \tag{5}$$

The weight for IoU, the higher the IoU, the greater the loss of the sample, which helps improve the regression accuracy. Zhang et al. [20] found through ablation experiments that when γ= 0.5, the loss function achieves the best effect, as shown in Fig. 4. The Focal-EIoU loss function can reflect the true difference between the width and height of the bounding box and its confidence during bounding box regression, accelerating the training convergence.

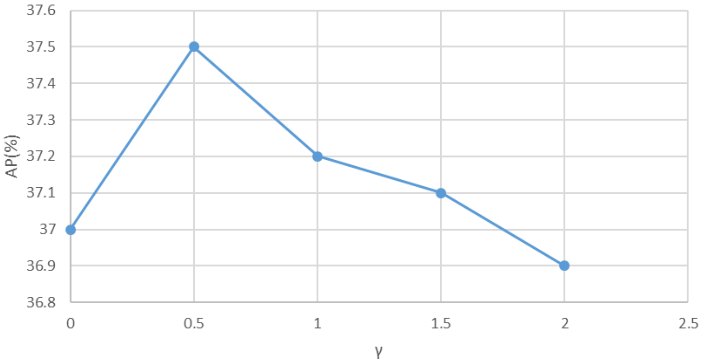


Fig. 4 Mean average precision of γ with different weights

4. Experiment and Analysis

KITTI Dataset is a publicly available dataset widely used in autonomous driving and computer vision research. It was created in collaboration between the Karlsruhe Institute of Technology (KIT) in Germany and the Technical University of Munich (TUM). It primarily comprises a large amount of sensor data collected by onboard sensors in urban environments.

4.1 KITTI Dataset

KITTI Dataset is a publicly available dataset widely used in autonomous driving and computer vision research. It was created in collaboration between the Karlsruhe Institute of Technology (KIT) in Germany and the Technical University of Munich (TUM). It primarily comprises a large amount of sensor data collected by onboard sensors in urban environments. Using this dataset as training data can better simulate the real driving environment of autonomous driving, enabling the trained model to apply to environmental perception tasks in autonomous driving. In this experiment, 5281 images were selected as the training set, 1100 images as the validation set, and 1100 images as the test set. The detection targets include common vehicles, such as cars, bicycles, trucks, vans, and trams (car, cycle, truck, van, and tram).

4.2 Experimental environment and model evaluation metrics

All experiments were conducted under the Ubuntu 22.04 system, with Python 3.10 as the programming environment, Pytorch 2.0 as the deep learning framework, an Intel® Xeon(R) W-1370P @ 3.60GHz CPU, 64GB memory, and a GeForce RTX 3090 graphics card. For experiments with different improvement strategies, the model parameters were kept consistent, the batch size was 16, the number of threads was 12, the image input size was 640×640, the number of training iterations was 100, the initial learning rate was 0.01, momentum is 0.937, attenuation coefficient is 0.0005, and no pre-trained model was used during training.

The experiment mainly uses the precision rate, recall rate, mean average precision (mAP) with an IoU threshold of 0.5, and frames per second (FPS) as the model evaluation metrics. The specific calculation formula for precision rate, recall rate, and mAP is as follows:

$$\Pr\text{ecision} = \frac{TP+TN}{N} \tag{6}$$

$$\text{Re call} = \frac{\text{TP}}{TP+FN} \tag{7}$$

$$\text{mAP} = \frac{1}{n}\sum_{i=1}^{n} P_i^{IoU=0.5}\left(R_i^{IoU=0.5}\right) \tag{8}$$

Among them, TP and TN represent the number of positive and negative samples that were correctly predicted, respectively, and FN represents the number of positive samples that were incorrectly predicted as negative; n represents the number of detection target categories, P represents the precision rate, which refers to the ratio of the number of correct samples detected to the total number of positive samples predicted; R represents the recall rate, which refers to the ratio of the number of correct samples detected to the total number of actual targets. A high precision rate means that the model has a lower error rate in its detection results, reducing false positives, while a high recall rate means that the model can detect most of the real targets well, reducing missed detections. The mean average precision calculated based on precision rate and recall rate is an indicator that can comprehensively evaluate the performance of the target detection model, reflecting the accuracy and recall rate performance of the model in multiple categories.

4.3 Automatic driving scene detection experiment

Firstly, the effectiveness of the model in object detection in common autonomous driving scenarios is verified. This paper proposes three improvement methods for the YOLOv5 model, namely RFAConv, SEAttention, and Focal-EIoU loss (hereinafter referred to as R, SE, and FE). To verify the effectiveness of the improvement method, this experiment approaches the verification from two directions: 1) adding a module separately based on the benchmark model YOLOv5l; 2) removing a module separately from the benchmark model YOLOv5l. During the training of the model, random stretching, tilting, and other processing of the image are performed in the data input model stage to simulate the image information obtained by the camera from different angles of the vehicle, thereby enhancing the robustness of the model. In the experiment, except for the change in the module combination, other model parameters remain unchanged. The experimental results are shown in Table 1.

Table 1 Comparison of Training Results of YOLOv5l Ablation Experiment

Model	Precision/%	Recall/%	mAP/%	FPS
YOLOv5l	94.7	84.8	91.2	92.72
YOLOv5l+R	94.5	87.0	92.7	90.63
YOLOv5l+SE	95.2	86.4	93.6	87.10
YOLOv5l+FE	96.0	87.3	93.4	90.88
YOLOv5l+ R +SE	94.7	87.7	93.1	86.30
YOLOv5l+ R +FE	95.9	86.0	93.2	91.35
YOLOv5l+SE+FE	94.5	87.0	93.2	85.10
YOLOv5l+ R +SE+FE (Improved model)	95.8	87.6	93.8	88.06

Analyzing Table 1, it is observed that the improved YOLOv5l model shows increases in the precision rate, recall rate, and mAP by 1.1, 2.8, and 2.6 percentage points, respectively, compared with the original model. This indicates that the improved model is effective and more accurate than the original model. The speed only loses 4.66 FPS, and the result remains well above the real-time image processing capability of 30 FPS, which is the minimum requirement in autonomous driving. A slight decrease in speed for improved detection accuracy is acceptable. This ablation experiment also shows that the RFAConv, SEAttention, and Focal-EIoU loss three modules will slightly increase the computation compared to the original model, but the improvement in the detection effect is considerable.

To verify the effectiveness of the improved model, a comparative experiment was designed to compare it with other models. This experiment selected the YOLOv3, YOLOv5 series algorithms, and YOLOv8l. Except for the different network structures of each model, other experimental parameters remain consistent, and the experimental results are shown in Table 2.

Table 2 Comparison of training results of different detection algorithms

Model	Precision/%	Recall/%	mAP/%	FPS
YOLOv3	93.5	82.7	90.3	113.32
YOLOv5s	92.2	82.5	89.3	130.27
YOLOv5m	93.8	84.5	90.4	102.97
YOLOv5l	94.7	84.8	91.2	92.72
YOLOv5x	95.2	85.6	91.2	79.22
YOLOv8l	94.8	85.0	90.5	90.09
Improved model	95.8	87.6	93.8	88.06

By analyzing Table 2, the vehicle detection performance of the model in this paper is higher than that of other YOLOv5 models, and even higher than that of its heavyweight model YOLOv5x, with an increase of 0.6%, 2.0%, and 2.6% in precision rate, recall rate, and mAP, respectively. The detection speed is 11.8% higher than that of the YOLOv5x model. The improvement effect of this paper on the algorithm model is significant and effective. Improve the model performs better than state-of-the-art YOLOv8l, with a 1% higher Precision, 2.6% higher Recall, and 3.3% higher mAP. At the same time, the detection performance of the improved YOLOv5l algorithm model in this paper is better than that of the YOLO series models YOLOv3 and YOLOv8l. The model volumes trained by YOLOv3, YOLOv5l, this model, and YOLOv8l on this dataset are 117MB, 88.4MB, 88.7MB, and 83.5MB respectively. Combined with the data in the table, it can be seen that YOLOv5 and YOLOv8 are lighter and more efficient than YOLOv3 in detection effect, which meets the actual demand of edge devices for resource limitations. Improve the proposed model has the highest mAP, and the detection speed meets the minimum requirement of 30 FPS for autonomous driving, indicating its best detection performance and suitability for practical traffic scenarios.

4.4 Vehicle small target detection experiment

To verify the detection effect of the improved model on small targets in autonomous driving scenarios, images containing small targets were selected by screening target labels in the test set of the KITTI dataset. Fig. 5 shows the scatter plot of the length and width of the targets in the test set of the KITTI dataset, where the distribution of small targets is relatively concentrated. A total of 1,907 target objects were selected, with a length of no more than 20% and a width of no more than 10%.

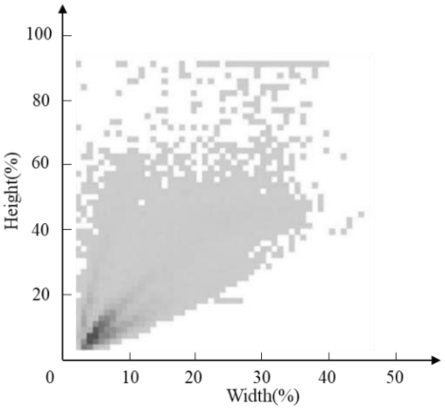


Fig. 5 The scatter plot of target widths and heights in the test dataset

To verify more possibilities, this paper designed experiments by adding other attention mechanisms to the same network structure as designed in this paper and conducted multiple comparative experiments on the small-target test set. The experimental results are shown in Table 3.

Table 3 Comparison of Small Target Detection Effects with Different Attention Mechanisms

Model	Precision /%	Recall/%	mAP/%	FPS
YOLOv5l	85.6	81.2	82.6	88.36
YOLOv5l+SimAM	85.9	81.0	83.1	87.32
YOLOv5l+GAM	86.2	80.7	82.9	83.89
YOLOv5l+ECA	85.8	82.2	82.6	86.38.
YOLOv5l+SE	86.8	82.3	83.8	85.27

Based on the comparison results, it is observed that introducing SimAM and ECA into the baseline model does not significantly improve the small object detection effect. However, GAM shows a significant improvement in accuracy but results in a significant decrease in recall rate. In contrast, the SE attention mechanism proposed in this paper achieves significant performance improvement in small object detection tasks, including precision rate, recall rate, and mAP. The uniqueness of the SE attention mechanism lies in its ability to dynamically adjust the weight of the feature maps of different channels in the convolutional neural network, highlighting the key feature information. This dynamics enables the SE attention mechanism to better capture the feature representation of small targets, thereby improving the performance of small object detection. Therefore, this paper chooses to integrate the SE attention mechanism into the network structure to enhance the small object detection performance. Experiments were conducted using the previously trained YOLOv3, YOLOv5 series algorithms, and YOLOv8l on this small-object test set. The experimental results are shown in Table 3.

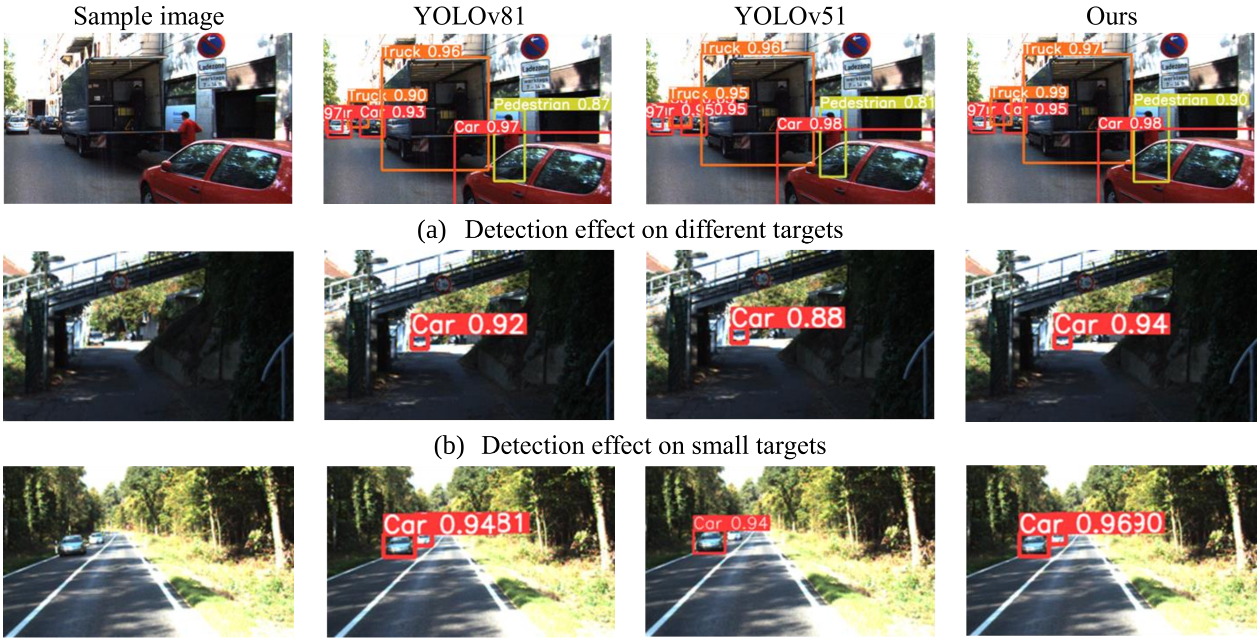
Table 4 Comparison of Different Algorithms for Small Target Detection

Model	Precision/%	Recall/%	mAP/%	FPS
YOLOv3	84.6	79.3	82.2	110.45
YOLOv5s	83.9	78.6	81.7	132.61
YOLOv5m	85.2	80.7	82.4	100.23
YOLOv5l	85.6	81.2	82.6	88.36
YOLOv5x	87.3	81.9	83.2	77.28
YOLOv8l	86.1	80.9	82.7	88.7
Improved model	88.5	82.8	84.9	87.83

Based on the results in Table 4, the model proposed in this study demonstrates excellent performance in detecting small target vehicles, achieving precision, recall, and mAP rates of 88.5%, 82.8%, and 84.9%, respectively. Compared to the original YOLOv5l model, these performance metrics have improved by 2.9%, 1.6%, and 2.3%, respectively. It is worth noting that the proposed model in this paper exhibits significantly higher accuracy in detecting small target vehicles compared to other models of the same kind. Meanwhile, it should be pointed out that the detection speed of these models on the small target dataset is slightly lower compared to that in ordinary autonomous driving scenarios. This is because images containing small targets typically have a wider field of view, with a larger number of targets, resulting in an increased computational burden. The experimental results demonstrate that the proposed model successfully achieves excellent small target vehicle detection performance in autonomous driving scenarios.

The actual detection effects of each target detection algorithm are presented in Fig. 6. In Fig. 6 (a), the detection effects of the model on various targets can be observed. It is worth noting that the model proposed in this paper demonstrates higher detection accuracy compared to the original model and exhibits more excellent performance in the detection tasks of different target categories. Furthermore, in Fig. 6(b), the performance of the model in small target detection is investigated in detail. The results show that the model proposed in this paper has higher confidence compared to other models, particularly demonstrating more excellent performance in small target detection. Additionally, in Fig. 6(c), the detection effects of the model on targets affected by lighting are explored. Given the challenges of lighting conditions, the YOLOv5l model experiences missed detections. However, the model proposed in this paper still demonstrates good detection performance in this scenario affected by lighting, showcasing higher robustness.

Through comprehensive comparative experiments on different models, it can be concluded that the improved YOLOv5 algorithm model proposed in this paper significantly outperforms the original model and other models of the same kind in terms of detection accuracy and robustness. These results fully validate the effectiveness of the proposed improved algorithm model. The images in the visualization experiment are all from the actual traffic environment, and the experiment shows that the proposed model is suitable for real road environments.



(a) Detection effect on different targets

(b) Detection effect on small targets

(c) Detection effect on targets affected by lighting

Fig. 6 Comparison of detection effects of different models

5. Conclusion

To solve the problems of low detection accuracy and poor robustness of small target vehicles in autonomous driving scenarios, this paper proposes an improved YOLOv5 algorithm for vehicle detection. In the data input stage, image enhancement is performed to improve the robustness of the trained detection model. In the backbone network, ordinary convolutions are replaced with sensory field attention convolutions, where the weights of convolutional kernels are dynamically adjusted according to the importance of features, thereby obtaining richer feature information. Additionally, a channel attention mechanism is added to the backbone of the model network to focus on the channel dimension of the network, strengthening the feature extraction of the feature map. In the output end, the Focal-EIoU loss function is introduced to increase the attention on high-quality samples in the regression stage of the target detection box, thereby improving the accuracy of detection.

The experimental results show that the proposed model not only has a better detection effect compared to other models in ordinary autonomous driving scenarios but also exhibits higher accuracy and better robustness in detecting small target vehicles. This model can make accurate predictions and avoid obstacles in autonomous driving scenarios.

Conflicts of Interest

The authors declare no conflict of interest.

References

[1] J. F. Yang, X. Q. Wang, H. Lin, L. X. Li, Y. Y. Yang, K. C. Li, et al., “Review of One-Stage Vehicle Detection Algorithms Based on Deep Learning,” Computer Engineering and Applications, vol. 58, no. 7, pp. 55-67, 2022.

[2] S. Zhai, D. Shang, S. Wang, and S. Dong, “DF-SSD: An Improved SSD Object Detection Algorithm Based on DenseNet and Feature Fusion,” IEEE Access, vol. 8, pp. 24344-24357, 2020.

[3] U. Sirisha, S. P. Praveen, P. N. Srinivasu, P. Barsocchi, and A. K. Bhoi, “Statistical Analysis of Design Aspects of Various YOLO-Based Deep Learning Models for Object Detection,” International Journal of Computational Intelligence Systems, vol. 16, article no. 126, 2023.

[4] G. Oreski, “YOLO* C—Adding Context Improves YOLO Performance,” Neurocomputing, vol. 555, article no. 126655, August 2023.

[5] C. H. Kang and S.Y. Kim, “Real-time Object Detection and Segmentation Technology: An Analysis of the YOLO Algorithm,” JMST Advances, vol. 5, no. 2, pp. 69-76, 2023.

[6] A. Bochkovskiy, C. Y. Wang, and H. Y. M. Liao, “YOLOv4:Optimal Speed and Accuracy of Object Detection,” arXiv preprint arXiv:2004.10934, 2004.

[7] N. Arora, Y. Kumar, R. Karkra, and M. Kumar, “Automatic Vehicle Detection System in Different Environment Conditions Using Fast R-CNN,” Multimedia Tools and Applications, vol. 81, no. 13, pp. 18715-18735, 2022.

[8] R. X. Li, J. Y. Yu, F. Li, R. T. Yang, Y. D. Wang, and Z. H. Peng, “Automatic Bridge Crack Detection Using Unmanned Aerial Vehicle and Faster R-CNN,” Construction and Building Materials, vol. 362, article no. 129659, January 2023.

[9] Y. J. Ma, Y. T. Ma, S. S. Cheng, and Y. D. Ma, “Road Vehicle Detection Method Based on Improved YOLO v3 Model and Deep-SORT Algorithm,” Journal of Traffic and Transportation Engineering, vol. 21, no. 2, pp. 222-231, August 2021.

[10] X. D. Dong, S. Yan, and C. Q. Duan, “A Lightweight Vehicles Detection Network Model Based on YOLOv5,” Engineering Applications of Artificial Intelligence, vol. 113, article no. 104914, August 2022.

[11] J. R. Liang, Z. Chen, G. J. Dong, Q. Chen, and Y. L. Xu, “Vehicle Detection Based on Ghost Convolution and Channel Attention Mechanism Cascade Structure,” Journal of Tianjin University (Science and Technology), vol. 56, no. 02, pp. 193-199, Feburary 2023.

[12] M. A. Ahmed, S. A. Althubiti, V. H. C. de Albuquerque, M. C. dos Reis, C. Shashidhar, T. S. Murthy, et al., “Fuzzy Wavelet Neural Network Driven Vehicle Detection on Remote Sensing Imagery,” Computers and Electrical Engineering, vol. 109, Part A, article no. 108765, July 2023.

[13] X. H. Kong, Y, Zhang, S. T. Tu, C. Xu and W. Yang, “Vehicle Detection in High-Resolution Aerial Images with Parallel RPN and Density-Assigner,” Remote Sensing, vol. 15, no. 6, article no. 1659, March 2023.

[14] X. D. Dong, S. Yan, and C. Q. Duan, “A Lightweight Vehicles Detection Network Model Based on YOLOv5,” Engineering Applications of Artificial Intelligence, vol. 113, article no. 104914, August 2022.

[15] C. Y. Wang, H. Y. M. Liao, Y. H. Wu, P. Y. Chen, J. W. Hsieh, and I. H. Yeh, “CSPNet:A New Backbone That Can Enhance Learning Capability of CNN,” Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, IEEE Press, pp. 390-391, 2020.

[16] Y. S. Tan, K. M. Lim, C. Tee, C. P. Lee, and C. Y. Low, “Convolutional Neural Network with Spatial Pyramid Pooling for Hand Gesture Recognition,” Neural Computing and Applications, vol. 33, pp. 5339-5351, 2021.

[17] Y. Zhang, H. F. Zhang, Q. Q. Huang, Y. Han, and M. H. Zhao, “DsP-YOLO: An Anchor-Free Network with DsPAN for Small Object Detection of Multiscale Defects,” Expert Systems with Applications, vol. 241, article no. 122669, May 2024.

[18] X. Zhang, C. Liu, D. Yang, T. Song, Y. Ye, K. Li, and Y. Song, “RFAConv: Innovating spatial attention and standard convolutional operation,” arXiv preprint arXiv:2304.03198, 2023.

[19] J. Hu, L. Shen, S. Albanie, G. Sun, and E.H. Wu, “Squeeze-and-Excitation Networks,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 42, no. 8, pp. 2011-2023, August 2020.

[20] Y. F. Zhang, W. Ren, Z. Zhang, Z. Jia, L.Wang, and T. Tan, “Focal and Efficient IoU Loss for Accurate Bounding Box Regression,” Neurocomputing, vol. 506, pp. 146-157, September 2022.