Improved YOLOv5s Traffic Sign Detection

Xiaoming Zhang, Ying Tian

***Abstract*—Aiming at the small proportion of traffic signs in natural scenes, fuzzy and complex, and the problems of low detection accuracy, missed detection, and false detection in current traffic sign detection algorithms, a traffic sign detection algorithm based on YOLOv5s was proposed. Firstly, the Regional Feature Enhancement Module (RFEM) is presented, which uses dilated convolution with different dilated rates and 1×1 convolution to expand the receptive field and change the feature dimension. The feature fusion is carried out by adding a method to increase each dimension information of the image. Improve the final classification accuracy of the model. Secondly, a 160×160 size detection layer was added to the detection layer of the original algorithm, and the feature fusion was performed with the local small target information extracted from the backbone network to increase the detection accuracy of small targets. Finally, K-means++ was used to recluster the initial anchor box, which accelerated the convergence speed of the model, reduced the border loss, and improved the detection accuracy of the model. The experimental results show that the improved algorithm has achieved 90.10%Precision, 82.36%Recall, and 87.98%mAP, on the TT100K dataset. Compared with the original YOLOv5s algorithm, the improved YOLOv5s algorithm has improved the accuracy of the algorithm. It increased by 7.89%, 5.05%, and 4.36%, respectively. This method can be effectively applied for traffic sign detection.**

***Index Terms*—traffic sign detection, YOLOv5s, receptive field, feature fusion, dilated convolution**

I. Introduction

AS a crucial branch of object detection [1-4], traffic sign detection holds significant practical value for unmanned and assisted driving and has been extensively researched in recent years. However, detecting traffic signs in complex and dynamic real-world scenarios poses a significant challenge. Distant traffic signs appear smaller within the overall detection environment. In contrast, nearby traffic signs occupy a more substantial portion of the detection area, resulting in large-scale and small-scale transformation issues for the target to be detected. In essence, this paper aims to address the problems of multi-scale transformation and small target detection [5, 6], which are the primary focus of our research.

Research on traffic sign detection can be classified into two categories: traditional methods and deep learning-based methods. Traditional methods rely on shape and color features of traffic signs, including algorithms such as Histograms of Oriented Gradient (HOG) [7], and Scale Invariant Feature Transform (SIFT) [8]. These approaches involve manual feature extraction and employ machine learning algorithms for classification. However, traditional algorithms are susceptible to external factors such as weather, resulting in poor robustness. On the other hand, deep learning-based traffic sign detection algorithms can be broadly classified into one-stage and two-stage detection approaches. One-stage object detection algorithms, such as OverFeat [9], YOLOv3 [10], SSD [11], and RetinaNet [12], treat positioning and classification as regression problems, enabling end-to-end detection. While these approaches improve detection speed, they usually have lower detection accuracy. In contrast, two-stage object detection algorithms, including R-CNN [13], SPP-Net [14], Fast R-CNN [15], Faster R-CNN [16], and R-FCN [17], use a Region Proposal Network to identify regions of interest. These algorithms typically achieve high detection accuracy in classifying the region of interest, but their detection speed is slower than that of one-stage approaches.

II. Related Work

To address the challenges of small object detection and large-scale transformations in traffic sign detection, researchers often use Feature Pyramid Networks [18], fusion attention mechanisms [19-23], improved multi-scale detection heads [24], and data augmentation techniques to improve detection accuracy. The new Traffic Sign Detection Benchmark (Tsinghue-Tencent100K, TT100K) [25] offers a more extensive data scale than the widely used detection benchmark (CCTSDB, CSUST Chinese Traffic Sign Detection Benchmark) [26], with more target categories, better data quality, and higher image resolution. Researchers can select the appropriate benchmark dataset based on their specific needs. In China, traffic signs can be broadly categorized into three groups: instructions, warnings, and prohibitions, denoted by blue, yellow, and red, respectively [27]. They come in various shapes, such as circles, triangles, and rectangles. Currently, color and shape features are used to detect traffic signs, with the K-means method used for color grouping and convolutional neural networks employed for detection.

To address the challenges of small-scale, fuzzy, and complex traffic sign recognition in natural scenes, a traffic sign detection method based on RetinaNext-NeXt has been proposed. This approach utilizes a new backbone network, ResNeXt, to improve the detection accuracy and effectiveness of RetinaNet. Further, an improved Sparse R-CNN has been proposed to address the mismatch between existing detection algorithms and their practical application in natural traffic scenes. This approach combines the

Manuscript received June 09,2023; revised October 19,2023.

This work was funded by the foundation of Liaoning Educational committee under the Grant No. LJKZ0310.

Xiaoming Zhang is a postgraduate student of School of Computer Science and Software Engineering, University of Science and Technology Liaoning, Anshan, 114051, China. (e-mail: 2468988768@qq.com).

Ying Tian is a Professor of School of Computer Science and Software Engineering, University of Science and Technology Liaoning, Anshan, 114051, China. (corresponding author to provide phone: +8613898015263; e-mail: astianying@126.com).

coordinate attention block with ResNeSt to construct a feature pyramid that modifies the backbone, enabling the extracted features to focus on important information and improve detection accuracy. Despite the promising results achieved by existing traffic sign detection algorithms, several challenges still need to be solved, including low detection accuracy, missed detection, false detection, and other problems. Therefore, further research is needed to address these challenges and enhance the effectiveness of traffic sign detection.

Small target detection poses a critical challenge in traffic sign detection, given that the majority of traffic signs are characterized as small targets within real-world detection scenes, and their scale may dynamically vary with the movement of the vehicle camera. To improve the accuracy of small target detection and address the issue of multi-scale changes in the target, this study aims to enhance the application of traffic sign detection in real-life scenarios. To achieve this, the YOLOv5s [28] algorithm has been improved. The proposed improvements include the following: (a) A Regional Feature Enhancement Module (RFEM) based on Dilated Convolution and 1x1 convolution is proposed and applied to the Neck part of the basic network. This enriches the language information of the small target before the feature map enters the detection Head, thereby improving the detection accuracy of the small target. (b) The Multi-scale Detection Head (MDH) part of the basic network is improved by adding a 160x160 feature map for the fourth detection head. The receptive field of this feature map, corresponding to the original image, is smaller than that of the 80x80 feature map, making it more effective in detecting small objects on larger feature maps. (c) The anchor calculation method of the basic network is abandoned, and the K-means++ clustering algorithm is used to re-cluster the anchor boxes suitable for the dataset. This not only accelerates the convergence speed of the model but also improves the final detection accuracy of the model. By implementing these improvements, the proposed approach aims to address the challenges of small target detection and multi-scale changes in the target, thereby improving the accuracy and effectiveness of traffic sign detection in real-life scenarios.

As with the network structure of the YOLO series, YOLOv5s consists of four primary parts: Input, Backbone, Neck, and Head. This lightweight model of YOLOv5 has a smaller model volume and faster inference speed than other YOLO models. The structure of YOLOv5s is illustrated in Fig. 1.

The input of YOLOv5s is RGB images with three channels, and the feature map size is 640x640x3. Mosaic data augmentation is used to enrich the background of the detected target image and reduce the model's dependence on batch size. The backbone network is responsible for feature extraction, and the CSPDarkNet53 structure is used to divide the input feature map into two parts, with cross-stage partial connections introduced between the two parts to enable the network to learn feature information of different scales. This design significantly reduces the model's number of parameters and calculations while improving its accuracy and generalization ability. The neck employs the Path Aggregation Network (PANet) to effectively fuse the feature maps output by the backbone and achieve information fusion across different feature layers. PANet better integrates the feature information of shallow and deep layers, enabling the network to fully extract the features of each level in the network. This strengthens feature extraction and provides richer feature information, including strong semantic information, edges, textures, and other details. The output part, Head, has three YOLO head detectors that output feature maps of different scales for object prediction.

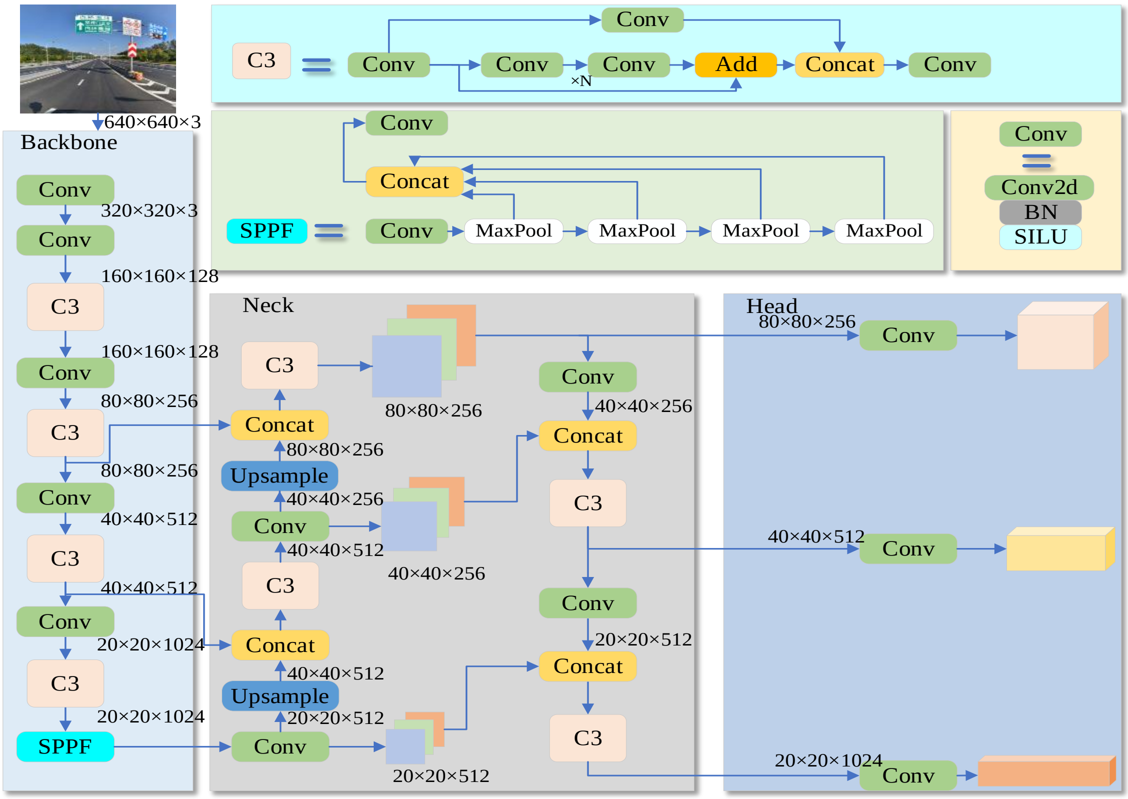


Fig. 1. YOLOv5s model

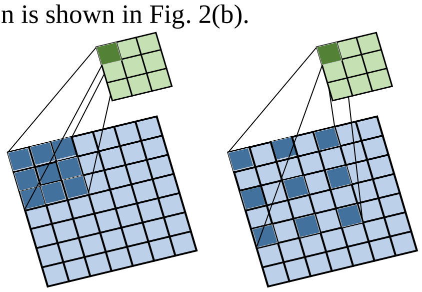
III. Improved Model

A. Region Feature Enhancement Module

In general, downsampling is a common method to increase the receptive field and reduce the amount of calculation in deep neural networks. However, downsampling can also sacrifice resolution and result in loss of input information. On the other hand, dilated convolution not only increases the receptive field but also enhances the resolution and accuracy of the target location compared to downsampling. To improve the accuracy of small object detection, we propose the Regional Feature Enhancement Module (RFEM). As the depth of the neural network increases, the semantic information contained in small targets can gradually be lost, which limits the model's ability to extract relevant information. We propose RFEM, implemented based on Dilated Convolution and 1×1 convolution, to address this issue. Dilated convolution is a standard operation in convolutional neural networks that increases the receptive field without adding parameters. This allows the feature map to capture more target information from the upper layer of the network, thereby improving the model's performance. The essence of dilated convolution involves inserting interval zero elements into the input tensor, which enables the convolution kernel to span a more considerable distance during convolution. For instance, a 3x3 kernel with dilation=1 can only move one pixel at a time. However, if dilation=2, the kernel can cross an interval of one pixel each time, effectively expanding the adequate size of the convolution kernel to 5x5. After dilated convolution, each pixel of the feature map corresponds to a 5x5 receptive field size of the feature map from the previous layer. The receptive field is calculated as follows:

$$N = K + (K-1)\times(d-1) \qquad (1)$$

K denotes the original convolution kernel size, d denotes the dilation rate parameter. The larger the dilation rate parameter is, the larger the receptive field of each pixel on the feature map after dilated convolution corresponds to the original image. $d-1$ means that for dilation rate d , the convolution spans $d-1$ pixel intervals each time; N represents the size of the corresponding receptive field after the dilation convolution with the dilation rate of d . The ordinary convolution is shown in Fig. 2(a), and the dilated convolution is shown in Fig. 2(b).



(a) dilation=1 (b) dilation=2

Fig. 2. Comparison of normal convolution and dilated convolution

In the RFEM module, the role of a 1×1 convolution is to modify the channel dimension and fuse features. The 1×1 convolution has a kernel size of 1×1, which not only reduces computational complexity but also enhances the network's expressive ability and classification performance. In this module, the input is the feature map from the feature enhancement module. Initially, the channel dimension is adjusted using a 1×1 convolution to control the number of channels within a suitable range for the layer. Subsequently, the subsequent operation is carried out on the modified feature map.

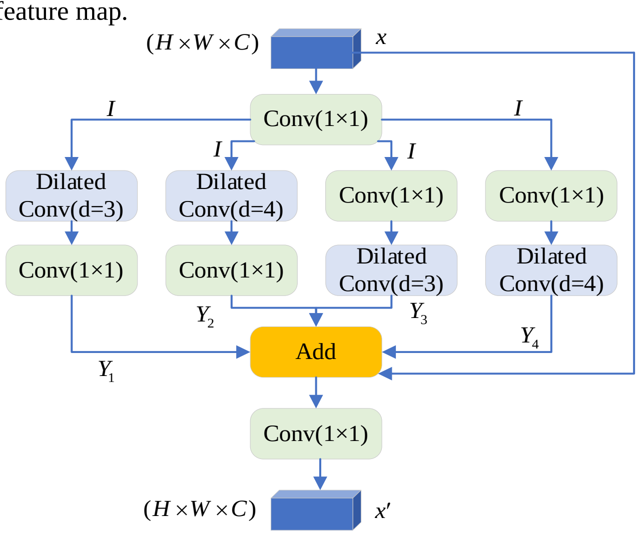


Fig. 3. RFEM structure diagram

Fig. 3 illustrates the implementation of the regional feature enhancement module based on dilated convolution and 1×1 convolution. The process follows: the input feature map is denoted as, and the channel dimension is modified using a 1×1 convolution, resulting in the output feature map. It is then simultaneously fed into four parallel branches. In the first branch, dilated convolution is applied with a dilation rate of 3, expanding the receptive field of the extracted features. A 1×1 convolution follows this. In the second branch, dilated convolution with a dilation rate of four is performed, followed by a 1×1 convolution. Branches three and four start with a 1×1 convolution, followed by dilated convolutions with dilation rates of 3 and 4, respectively. The feature maps obtained from these branches contain richer semantic information about small targets, as different dilation rates yield different receptive field sizes in the original images.

$$I = f_{conv}(x) \qquad (2)$$

$$Y_1 = f_{conv}(f'_{conv2d}(I)) \qquad (3)$$

$$Y_2 = f_{conv}(f''_{conv2d}(I)) \qquad (4)$$

$$Y_3 = f'_{conv2d}(f_{conv}(I)) \qquad (5)$$

$$Y_4 = f''_{conv2d}(f_{conv}(I)) \qquad (6)$$

x represents the input feature map of the RFEM module. f_{conv} denotes the 1×1 convolution. I represents the feature map obtained by applying f_{conv} to x . f'_{conv2d} represents the dilation convolution with a dilation rate of 3, and f''_{conv2d} represents the dilation convolution with a dilation rate of 4. Y_1 , Y_2 , Y_3 , Y_4 correspond to the outputs of the four branches, respectively. The outputs of the four branches are ultimately fused using the Add operation to preserve more valuable information.

$$x' = f_{conv}(Y1 \oplus Y2 \oplus Y3 \oplus Y4 \oplus x) \qquad (7)$$

$\oplus$ signifies that the output feature maps from the four branches are fused using the Add method. This feature fusion approach does not alter the channel number of the feature map; instead, it adds the corresponding pixel values from each feature map. This enables the retention of more semantic information about small targets and enhances the model's classification performance. x' represents the final output of the RFEM module.

B. Multi-scale Detection Head (MDH)

The YOLOv5 model extracts features through the backbone network and then fuses the features extracted at each stage using the neck feature fusion network. Finally, these fused features are passed to the multi-scale prediction head. In the original network, the input to the prediction head is divided into three feature maps of sizes 80×80×256, 40×40×512, and 20×20×1024. The prediction head performs object detection and classification on these feature maps at different levels. The larger feature map size allows for smaller receptive fields of the original image corresponding to each pixel on the feature map, enabling the detection of smaller targets. Conversely, smaller feature map sizes have larger receptive fields, making them more suitable for detecting larger targets. The variation in receptive field sizes corresponding to feature maps of different dimensions is illustrated in Fig. 4.

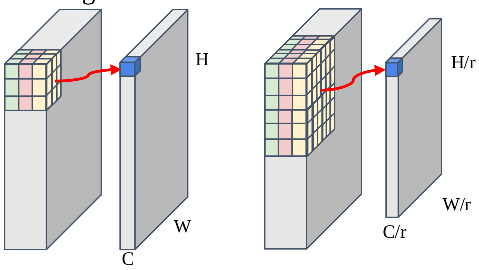


Fig. 4. Feature maps of different dimensions correspond to receptive field sizes

For the TT100K dataset, which predominantly contains small traffic signs, smaller feature maps are needed to detect these smaller signs. Therefore, a smaller feature map of size 160×160×256 is added to the multi-scale prediction head specifically for detecting smaller traffic signs. Additionally, the 160×160×128 feature maps extracted from the backbone network are fused to provide local feature information from the initial part of the network. This fusion plays a crucial role in improving the accuracy of detecting small targets. While the initial part of the backbone network may not extract high-dimensional target information, it contains valuable information about small targets. The enhanced multi-scale prediction head is illustrated in Fig. 5.

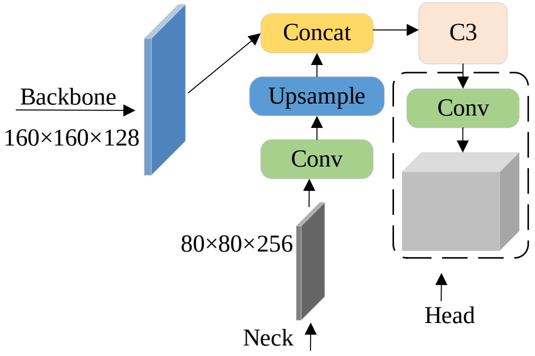


Fig. 5. This added multi-scale detection head aggregates low-level semantic information from the backbone shallow network

C. Re-cluster

YOLOv5 utilizes the K-means clustering algorithm to cluster the TT100K dataset and generate 9 initial anchors of fixed size. However, due to the small number of small targets in the TT100K dataset, the anchors generated by the original YOLOv5 clustering algorithm are large in size. This can have an adverse impact on the detection speed and accuracy of small target traffic signs. Therefore, this study employs the K-means++ clustering algorithm to re-cluster the training set of the TT100K dataset and identify initial anchors that are more suitable for the dataset used in this study. In the K-means++ algorithm, the initial cluster center is selected through a specific strategy rather than random selection as in K-means. It gives priority to points that are far away from the selected cluster center as the new cluster center. This approach helps to better avoid issues caused by random initialization. As a result, compared to K-means, K-means++ can avoid falling into local optima, thereby enhancing the accuracy and stability of the clustering process. Furthermore, three anchors with smaller sizes are generated compared to the original anchors. These smaller anchors are not only suitable for the new detection head but also contribute to the detection accuracy of small targets. The process of the K-means++ clustering algorithm is as follows:

(1) Randomly select a sample data as the first cluster center.

(2) Calculate the minimum distance between each sample x_i and the cluster center C_j.

$$D(x_i) = \arg\min \left\| x_i - C_j \right\|_2^2 \qquad (8)$$

x_i is the number of samples and C_j represents the C-th cluster center, where $j = 1, 2, ..., k$.

(3) The sample point with the maximum distance is selected as the cluster center.

(4) Repeat (2) and (3) until the number of clusters k is reached.

(5) The k cluster centers are used as the initial cluster centers to run the K-means++ algorithm.

When the cluster center is set to 12, the K-means++ clustering analysis of the TT100K dataset is depicted in Fig. 6. The analysis reveals that the target sizes in the TT100K dataset used in this paper fall within a small range. Therefore, it is appropriate to employ the K-means++ clustering algorithm to re-cluster the anchors in accordance with the initial anchors of the dataset. In Fig. 6, the red dots represent the 12 cluster center points. The horizontal coordinate represents the initial frame width (w), and the vertical coordinate represents the initial frame height (h). Both coordinates are normalized within the range of 0 to 1, relative to the image width and height.

As depicted in Fig. 6, the clustering process with 12 cluster centers yields 12 initial anchors, which are smaller than those generated by the baseline model and more suitable for the dataset used in this paper. The sizes of the anchors after re-clustering are presented in Table I. The smaller anchors are assigned to the larger feature map, which improves the model's detection accuracy for small targets by utilizing smaller anchors on the larger feature map.

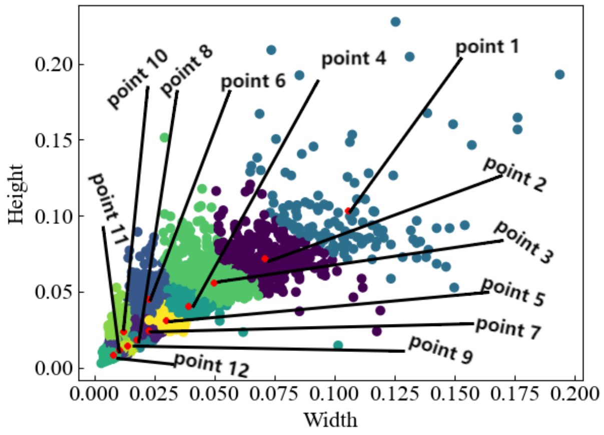


Fig. 6. Anchors clustering distribution

TABLE I
THE ANCHOR SIZE CORRESPONDING TO EACH DETECTION HEAD

Feature Map Size	Anchors Size
160×160	(4×5); (6×7); (7×15)
80×80	(8×9); (11×12); (14×15)
40×40	(14×28); (18×19); (24×26)
20×20	(37×35); (45×45); (67×66)

D. Improved YOLOv5s Model

The improved structure of YOLOv5s is illustrated in Fig. 7. The red module in the neck network represents the RFEM (Regional Feature Enhancement Module) proposed in this paper. The feature map entering each prediction head undergoes this module, resulting in more comprehensive and detailed feature information. The newly added scale prediction head, denoted by the red dashed box, combines local feature information extracted by the backbone network, which contains more details about small targets. It utilizes a detection head that is specifically designed for small target detection to detect smaller traffic signs in the dataset.

IV. Experiments and Analysis

A. Introduction to Dataset

We utilize the TT100K dataset, a collaborative effort between Tsinghua University and Tencent. This dataset provides a total of 100,000 high-resolution images, including 30,000 instances of traffic signs. It covers images captured under different lighting and weather conditions, offering a large-scale dataset with rich semantic information. Compared to other traffic sign datasets, TT100K offers a greater number of categories and smaller-sized targets, making the detection task more challenging. Due to the imbalance in instance numbers across categories in the dataset, and the presence of uncommon traffic signs in real-world scenes, this paper focuses on selecting categories with more than 100 instances for training and testing. The processed dataset used for network training and testing consists of 45 categories, with 7,198 instances in the training set and 1,850 instances in the testing set.

B. Experimental Environment and Parameter Configuration

Experimental environment: The experiment was conducted on a CentOS7 operating system. The CPU model used was Intel E5-2650. CUDA version 11.0 was employed for accelerated training, and the deep learning framework used was PyTorch 1.10.0. The programming language used

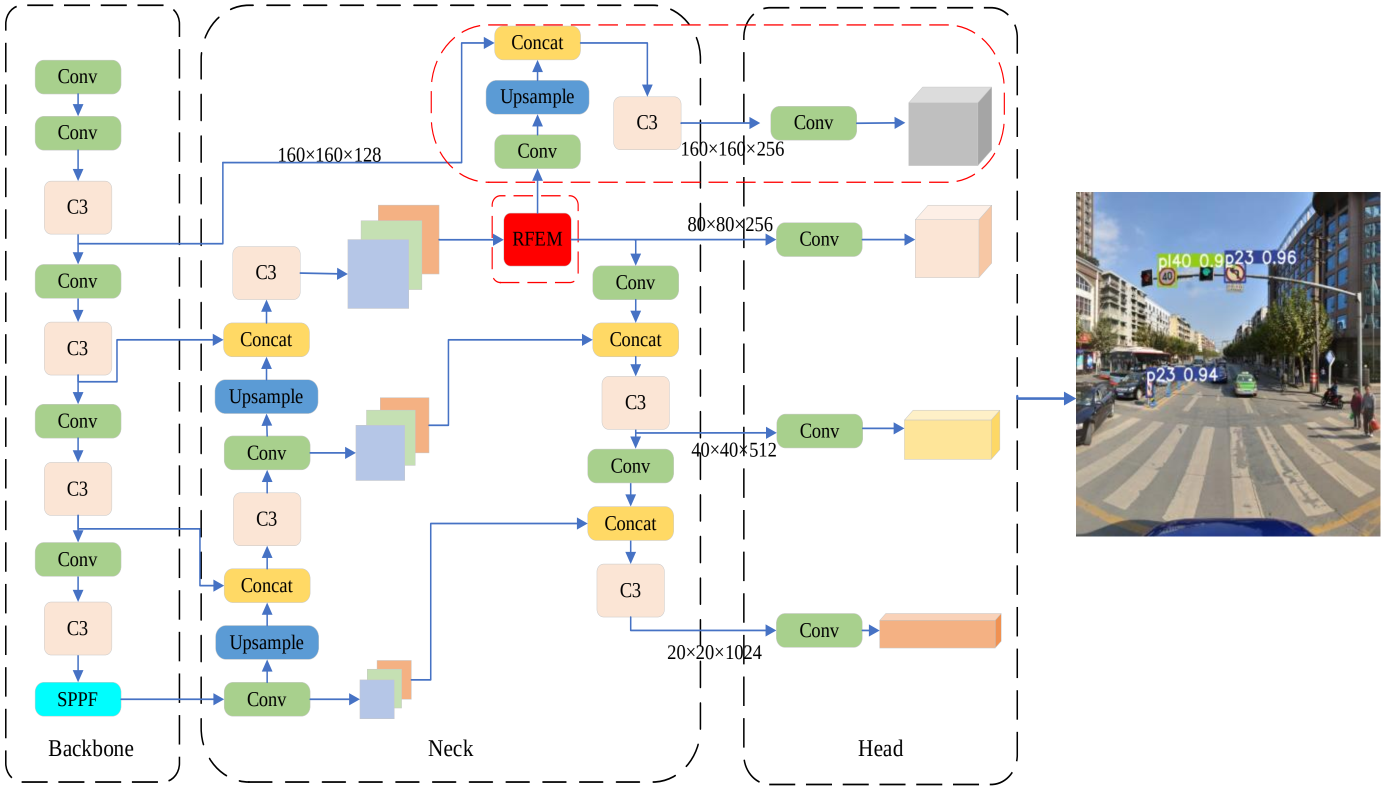


Fig. 7. Improved YOLOv5s structure diagram

was Python 3.7. The GPU model utilized was a single NVIDIA GTX1080ti with 11GB of video memory. The hyperparameters were set as follows: batch size of 32, initial learning rate of 0.01, cosine annealing strategy for learning rate adjustment, weight decay coefficient of 0.0005, momentum size of 0.937, SGD optimizer, and a total of 200 iterations.

C. Evaluation Index

Precision, Recall, and mAP (mean Average Precision) are selected as evaluation metrics to evaluate the proposed algorithm's performance. Precision refers to the probability of correct predictions among all positive samples, providing an intuitive measure of the model's false detection. Recall refers to the probability of predicting all positive samples, offering an intuitive measure of the model's missed detection. Model performance can be assessed using Precision and Recall, and typically, as Recall increases, Precision may decrease, and vice versa.

$$mAP = \frac{1}{n}\sum_{j=1}^{n} AP(j) \tag{9}$$

n =45, which represents the 45 classes in the dataset used in this paper, and AP represents the Average Precision of a class in the dataset.

$$p = \frac{TP}{TP+FP} \tag{10}$$

$$R = \frac{TP}{TP+FN} \tag{11}$$

$$AP = \int_0^1 P(R)dR \tag{12}$$

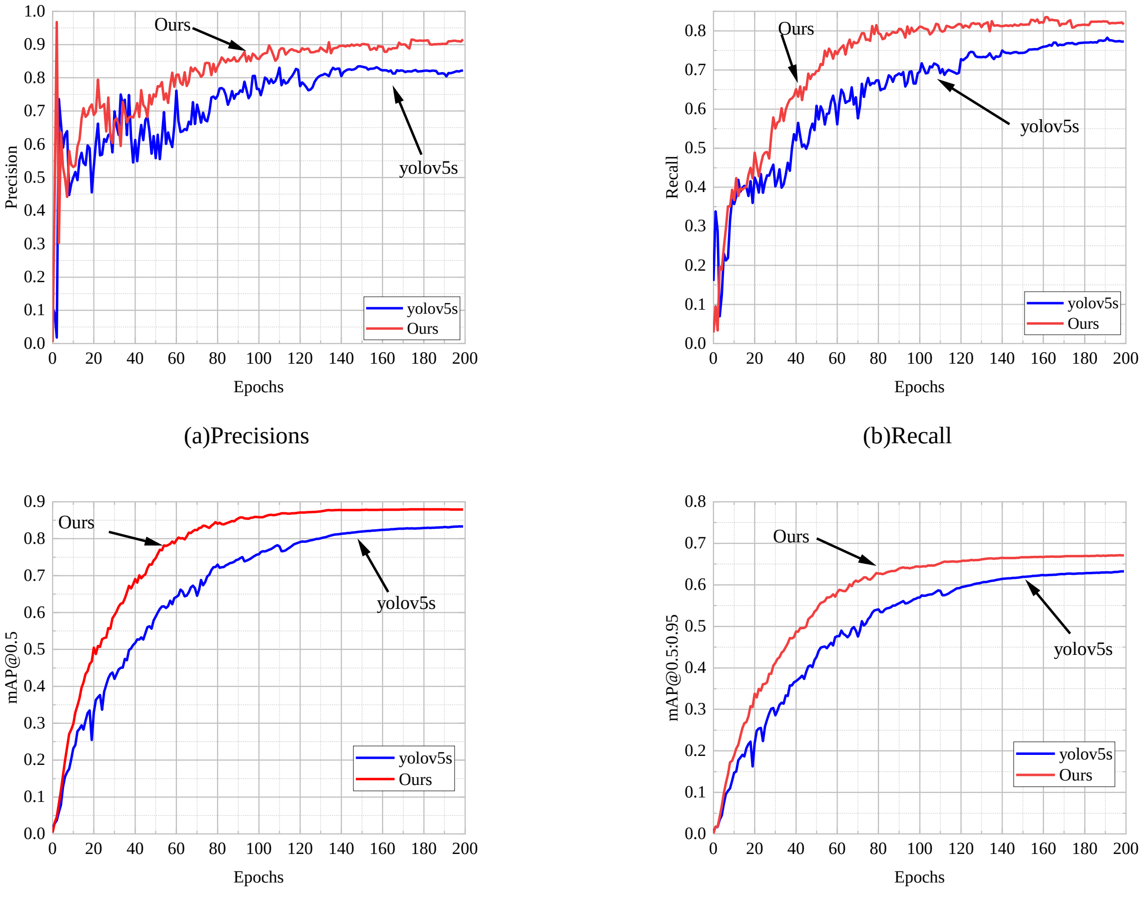
If a positive example is classified as a positive example, it is denoted as TP (true positive).If a positive example is misclassified as a negative example, it is denoted as FN (false negative).If a negative example is misclassified as a positive example, it is denoted as FP (false positive).

D. Experimental Results and Analysis

Fig. 8 presents a training comparison between the YOLOv5s baseline model and the proposed algorithm in this paper on the TT100K dataset, using images of the same size (640×640) as input to the network. The training comparison reveals that the proposed algorithm outperforms the baseline model. The Precision is increased by 7.89%, Recall by 5.05%, mAP@0.5 by 4.36%, and mAP@0.5:0.95 by 3.88%. All the evaluated indicators show improvement compared to the original network model. Furthermore, the improved model demonstrates faster convergence, a smoother training curve, and higher accuracy, exhibiting significant superiority over the baseline network.

E. Ablation Study

In order to demonstrate the effectiveness of the proposed module, ablation experiments are conducted on the TT100K dataset. The experimental design consists of the following variations: (a) Adding only the RFEM module to the baseline model. (b) Adding only the MDH module to the



(a)Precisions

(b)Recall

(c)mAP@0.5

(d)mAP@0.5:0.95

Fig. 8. Comparison diagram between training of YOLOv5s and improved algorithm

TABLE II
ABLATION EXPERIMENT

Models	RFEM	MDH	K-means++	P (%)	R (%)	mAP (%)	#Params(M)	FLOPs(G)	Inference time(ms)
YOLOv5s	×	×	×	82.21	77.31	83.35	7.1	16.1	2.7
YOLOv5s+ RFEM	√	×	×	84.72	77.75	84.37	7.2	18.2	3.0
YOLOv5s+ MDH	×	√	×	88.03	80.80	87.05	7.2	21.0	3.0
YOLOv5s+ MDH+ K-means++	×	√	√	89.14	83.19	87.93	7.2	21.0	2.9
Ours	√	√	√	**90.10**	**82.36**	**87.98**	**7.3**	**23.1**	**3.2**

baseline model. (c) Conducting two experiments to verify the effectiveness of the MDH module, wherein the initial anchors obtained through MDH and K-means++ algorithm re-clustering are added to the baseline model. (d) Adding all the modules proposed in this paper to the baseline model to assess the overall performance. The experimental results are presented in Table II.

From Table II, several conclusions can be drawn. When only the RFEM module is added, the Precision increases by 2.51%, the Recall rate increases by 0.44%, the mAP increases by 1.02%, the number of model parameters increases by 1%, the FLOPs increase by 2.1, and the model inference time increases by 0.3ms. When only the MDH module is added, the Precision increases by 5.22%, the Recall rate increases by 3.49%, the mAP increases by 3.7%, the number of model parameters increases by 1%, the FLOPs increase by 4.9, and the model inference time increases by 0.3ms. When the MDH module is added and the initial frames obtained through the K-means++ algorithm are re-clustered, the Precision increases by 6.93%, the Recall rate increases by 5.88%, the mAP increases by 4.58%, the number of model parameters increases by 1%, the FLOPs increase by 4.9, and the model inference time increases by 0.2ms. Finally, by deploying all the proposed improvement modules to the baseline model, the Precision, Recall, and mAP improve by 7.89%, 5.05%, and 4.63% respectively compared to the baseline model. The total number of model parameters increases by 2%, and the model inference time increases by 0.5ms. Despite a slight increase in model complexity, parameter count, and inference time, the results in Table II indicate that it does not significantly impact real-time inference detection. This demonstrates the practical significance of our proposed improvement modules, and sacrificing a small amount of model inference time has minimal impact on the detection speed in essence. Thus, while ensuring lightweightness, substantial improvements in detection accuracy can be achieved.

F. Feature Map Visual Analysis is used for MDH Module

To thoroughly illustrate the effectiveness of the MDH module in augmenting a small object detection layer within the baseline model, comprehensive research has been conducted. Fig. 9 visualizes the feature maps obtained by the prediction head before and after the enhancement. In Fig. 9, the inputs to the prediction head for the baseline model are represented by (b), (c), and (d), corresponding to the outputs of layer 17, layer 20, and layer 23, respectively. Conversely, the inputs to the prediction head for the proposed model are denoted as (e), (f), (g), and (h). The enhanced model incorporates an additional prediction head, with the feature map obtained by this prediction head integrating the output of the second layer of the backbone network, as indicated by (f). Significantly, it is evident that feature map (f) encompasses more intricate target features, ensuring a comprehensive extraction of target characteristics from the original image. Consequently, utilizing this improved feature map substantially enhances the detection performance for small targets.

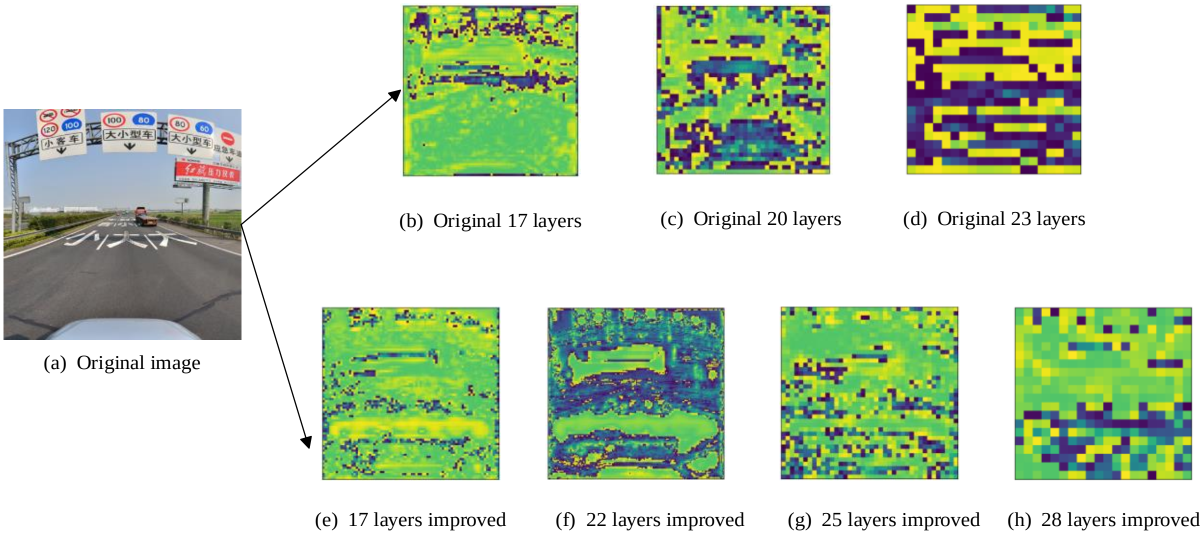
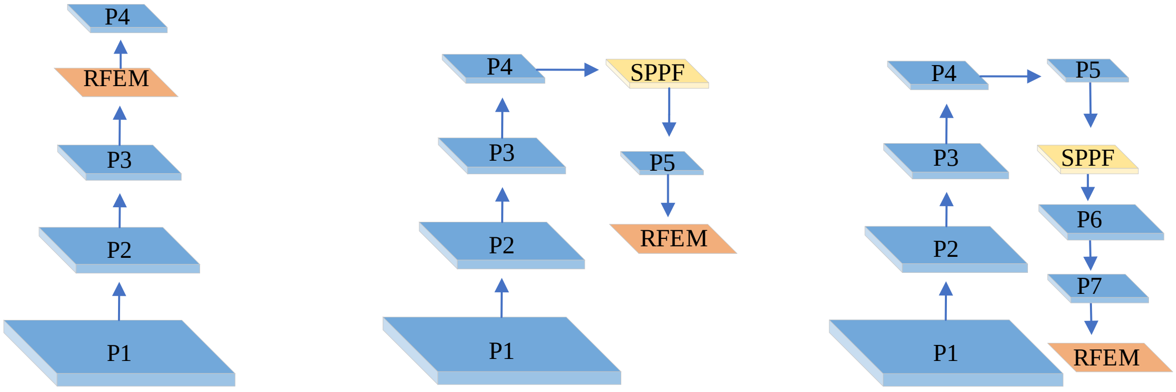


Fig. 9. Feature maps of the input multi-scale prediction head before and after improvement



(a)Backbone1 (b)Backbone2 (c)Backbone3

Fig. 10. Feature maps of the input multi-scale prediction head before and after improvement. We fuse the RFEM module in different positions of the baseline model, and discuss the impact of the three positions in the baseline model fused by the RFEM module on the performance of the baseline model, respectively

G.Comparative experiments of the RFEM Module

We conducted experiments on the three model structures depicted in the Fig. 10, and the results are presented in Table III. Upon analyzing the experimental data, we observed the following findings:

(1) When utilizing the Backbone1 structure, the model achieved a Precision of 79.62%, Recall rate of 70.69%, mAP of 78.63%, with 7.2M parameters, 18.2 FLOPs, and an inference time of 2.7ms.

(2) With the use of the Backbone2 structure, the model achieved a Precision of 83.45%, Recall rate of 74.08%, mAP of 81.70%, with 7.9M parameters, 18.2 FLOPs, and an inference time of 2.0ms.

(3) By employing the Backbone3 structure, the model achieved the highest Precision of 84.72%, Recall rate of 77.75%, mAP of 84.37%, with 7.2M parameters, 18.2 FLOPs, and an inference time of 2.8ms.

Despite the slightly longer inference time of the Backbone3 structure, the additional 0.1ms increase is negligible in practical scenarios. Using the Backbone3 structure allows for maximizing the detection accuracy of the model, while only incurring a minimal increase in the number of parameters and model complexity. This is advantageous for real-world detection scenarios.

The experimental results indicate that the computational complexity of the RFEM module remains consistent regardless of the fusion position, with a fixed complexity of 18.2 FLOPs, irrespective of the fusion method employed. This consistency in computational complexity ensures the reliability and efficiency of the RFEM module, making it a valuable addition to the model, irrespective of the fusion approach used. This contributes to the model's overall stability and performance.

TABLE III
RFEM COMPARATIVE EXPERIMENT

Experimental Grouping	Backbone	P (%)	R (%)	mAP (%)	# Parameters (M)	FLOPs(G)	Inference time(ms)
1	Backbone1	79.62	70.69	78.63	7.2	18.2	2.7
2	Backbone2	83.45	74.08	81.70	7.9	18.2	2.0
3	Backbone3	**84.72**	**77.75**	**84.37**	**7.2**	**18.2**	**2.8**

TABLE IV
PERFORMANCE COMPARISON OF DETECTION ALGORITHMS

Model Name	P (%)	R (%)	mAP (%)	# Parameters (M)	FLOPs(G)	Inference time(ms)
YOLOv3	73.36	72.14	73.79	61.6	77.7	15.2
YOLOv5s	82.21	77.31	83.35	7.1	16.1	2.7
SSD	60.59	70.28	70.46	-	-	-
Faster R-CNN	75.54	75.20	74.58	41.2	91.1	5.4
RetinaNet	78.37	70.56	79.39	-	-	-
FOCS	73.24	71.12	67.21	31.8	78.9	8.2
YOLOv7-tiny	71.12	73.23	72.82	6.2	13.8	2.6
RetinaNet-NeXt	87.45	79.65	86.71	-	-	-
Ours	**90.10**	**82.36**	**87.98**	**7.3**	**23.1**	**3.2**

H.Performance Comparison of Object Detection Algorithms

To further validate the effectiveness of our proposed algorithm, we conducted a comparative analysis with the mainstream object detection algorithms currently available, including YOLOv3, YOLOv5s, SSD, Faster R-CNN, RetinaNet, Retinanet-next, FOCS [29], and YOLOv7-tiny [30]. Precision rate, Recall rate, mAP, parameters, FLOPs, and inference time were utilized as evaluation metrics, and the results are presented in Table IV.

Based on the data presented in the table, our improved algorithm exhibits superior performance compared to other algorithms in terms of Precision rate, Recall rate, mAP, parameters, FLOPs, and inference time. Notably, our proposed algorithm demonstrates an outstanding Precision rate, reaching an impressive value of 90.10%. This indicates that the enhancements made to the baseline model in our paper effectively improve its performance using the same dataset. Furthermore, the improved algorithm only introduces a marginal increase of 0.5ms in inference time while significantly enhancing the detection accuracy of the model. Thus, accepting this minor trade-off in inference speed is justifiable to achieve substantial improvements in detection accuracy. Although the FLOPs metric increases by 7FLOPs compared to the baseline model, this increase remains relatively low compared to specific traditional algorithms. Moreover, the number of model parameters is only augmented by 2%.

I. Detection on Random Images

To intuitively observe the effectiveness of the improved algorithm presented in this paper, we randomly selected three images from the test set of the TT100K dataset to compare the detection results, as shown in Fig. 11. From the comparison images of the detection effects, and it is evident that the improved algorithm in this paper outperforms the baseline model significantly. Specifically, the improved algorithm is depicted in the three lower figures of Fig. 11. In detecting small targets at long distances. The improved model successfully detects the p6 target and p19 target, which the baseline model missed. The detection accuracy for the p19 target reaches 81%. Moreover, the improved model performs better for close-range targets than the baseline model. Overall, our improvement of the baseline model proves to be effective in detecting small targets at a distance, demonstrating practical significance.

V.Conclusion

To tackle the challenge of low detection accuracy and suboptimal performance in existing traffic sign detection models, this paper introduces an enhanced version of the YOLOv5s model, tailored specifically for traffic sign detection. The proposed model builds upon YOLOv5s and incorporates several key improvements. Firstly, a prediction head optimized for detecting small objects is integrated into the multi-scale prediction head section. This addition effectively combines local feature information extracted by the backbone network, thereby enhancing the accuracy of small object detection. Secondly, K-means++ clustering is employed to reconfigure the initial anchor boxes, making them better suited for the specific dataset. This process aligns the prior knowledge with the dataset's characteristics, resulting in reduced prediction box loss and faster model convergence. Lastly, the paper introduces the RFEM (Region-based Feature Enhancement Module) module, with its optimal embedding position determined through comparative experiments. This module is seamlessly integrated into the baseline model, increasing detection accuracy across all categories in the dataset under study.

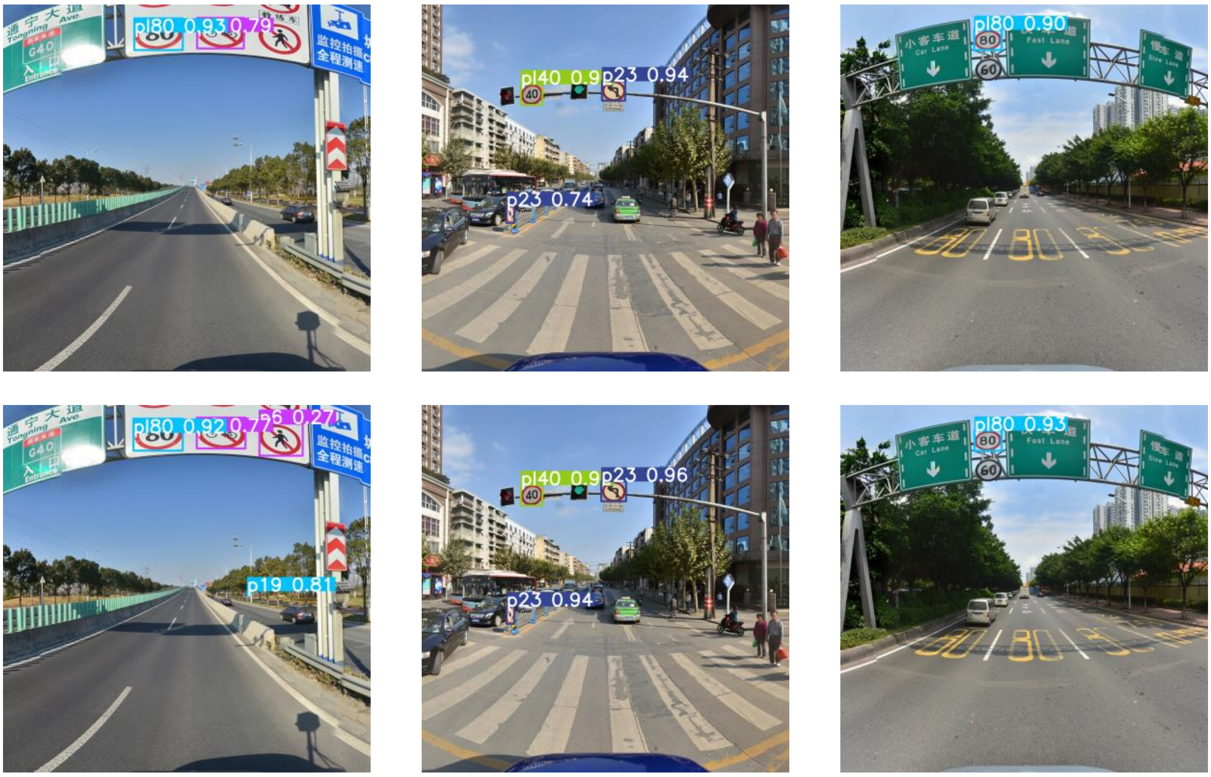


Fig. 11. Comparison of detection effect of detection algorithm before and after improvement

Experimental results showcase significant enhancements in the model's Precision (an increase of 7.98%), Recall (an increase of 5.88%), and mAP (an increase of 4.36%). Notably, these improvements in detection accuracy are not at the cost of a substantial impact on the model's inference speed. The increase in inference time is minimal, with a mere 0.5ms difference compared to the baseline model. This negligible effect on real-time performance ensures the model's practical deployability. Future research explores lightweight model approaches to reduce size while maintaining detection accuracy. This would boost detection speed, facilitate integration into mobile hardware devices, and enhance its suitability for real-world applications.

REFERENCES

[1] Z. Li, Q. Guo, B. Sun, D. Cao, Y. Li, and X. Sun, "Small Object Detection Methods in Complex Background: An Overview," *International Journal of Pattern Recognition and Artificial Intelligence,* vol. 37, no. 2, 2023, doi: 10.1142/S0218001423500027.

[2] D. Miao, Y. Wang, L. Yang, and S. Wei, "Foreign Object Detection Method of Conveyor Belt Based on Improved Nanodet," *IEEE Access,* vol. 11, pp. 23046-23052, 2023, doi: 10.1109/ACCESS.2023.3253624.

[3] L. Yuhai, L. Yuntian, H. Shunhu, Q. Qianlong, X. Pengfei, and F. Youchen, "Visible Light Small Object Detection Based on YOLOv5," in *9th Symposium on Novel Photoelectronic Detection Technology and Applications, April 21, 2023 - April 23, 2023*, Hefei, China, 2023, vol. 12617: SPIE, in Proceedings of SPIE - The International Society for Optical Engineering, pp. Chinese Society for Optical Engineering; Science and Technology on Low-light-level Night Vision Laboratory, doi: 10.1117/12.2664562. [Online]. Available: http://dx.doi.org/10.1117/12.2664562

[4] Yingwei. Li and Xiaoxia. Zhang, "Object Detection for UAV Images Based on Improved YOLOv6," *IAENG International Journal of Computer Science,* vol. 50, no. 2, pp. 759-768, 2023.

[5] T. Huang, M. Cheng, Y. Yang, X. Lv, and J. Xu, "Tiny Object Detection based on YOLOv5," in *5th International Conference on Image and Graphics Processing, ICIGP 2022, January 7, 2022 - January 9, 2022*, Virtual, Online, China, 2022: Association for Computing Machinery, in ACM International Conference Proceeding Series, pp. 45-50, doi: 10.1145/3512388.3512395. [Online]. Available: http://dx.doi.org/10.1145/3512388.3512395

[6] H. Wei, Q. Zhang, Y. Qin, X. Li, and Y. Qian, "YOLOF-F: you only look one-level feature fusion for traffic sign detection," 2023, doi: 10.1007/s00371-023-02813-1.

[7] R. Y. O. Matsumura and A. Hanazawa, "Human detection using color contrast-based histograms of oriented gradients," *International Journal of Innovative Computing, Information and Control,* vol. 15, no. 4, pp. 1211-1222, 2019, doi: 10.24507/ijicic.15.04.1211.

[8] V. Jyothi, G. Sri Priya Reddy, M. Sree Kavya, K. Harshitha Reddy, and P. Shvithi Reddy, "Study of Image Forgery Detection using Scale Invariant Feature Transform," in *2nd International Conference on Sustainable Computing and Data Communication Systems, ICSCDS 2023, March 23, 2023 - March 25, 2023*, Erode, India, 2023: Institute of Electrical and Electronics Engineers Inc., in 2nd International Conference on Sustainable Computing and Data Communication Systems, ICSCDS 2023 - Proceedings, pp. 669-673, doi: 10.1109/ICSCDS56580.2023.10104934. [Online]. Available: http://dx.doi.org/10.1109/ICSCDS56580.2023.10104934

[9] P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, and Y. LeCun, "Overfeat: Integrated recognition, localization and detection using convolutional networks," in *2nd International Conference on Learning Representations, ICLR 2014, April 14, 2014 - April 16, 2014*, Banff, AB, Canada, 2014: International Conference on Learning Representations, ICLR, in 2nd International Conference on Learning Representations, ICLR 2014 - Conference Track Proceedings.

[10] A. Farhadi and J. Redmon, "Yolov3: An incremental improvement," in *Computer vision and pattern recognition*, 2018, vol. 1804: Springer Berlin/Heidelberg, Germany, pp. 1-6.

[11] W. Liu *et al.*, "SSD: Single shot multibox detector," in *14th European Conference on Computer Vision, ECCV 2016, October 8, 2016 - October 16, 2016*, Amsterdam, Netherlands, 2016, vol. 9905 LNCS: Springer Verlag, in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), pp. 21-37, doi: 10.1007/978-3-319-46448-0_2. [Online]. Available: http://dx.doi.org/10.1007/978-3-319-46448-0_2

[12] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollar, "Focal Loss for Dense Object Detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. 42, no. 2, pp. 318-327, 2020, doi: 10.1109/TPAMI.2018.2858826.

[13] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *27th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2014, June 23, 2014 - June 28, 2014*, Columbus, OH, United states, 2014: IEEE Computer Society, in Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 580-587, doi: 10.1109/CVPR.2014.81. [Online]. Available: http://dx.doi.org/10.1109/CVPR.2014.81

[14] K. He, X. Zhang, S. Ren, and J. Sun, "Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. 37, no. 9, pp. 1904-1916, 2015, doi: 10.1109/TPAMI.2015.2389824.

[15] R. Girshick, "Fast R-CNN," in *15th IEEE International Conference on Computer Vision, ICCV 2015, December 11, 2015 - December 18, 2015*, Santiago, Chile, 2015, vol. 2015 International Conference on Computer Vision, ICCV 2015: Institute of Electrical and Electronics Engineers Inc., in Proceedings of the IEEE International Conference on Computer Vision, pp. 1440-1448, doi: 10.1109/ICCV.2015.169. [Online]. Available: http://dx.doi.org/10.1109/ICCV.2015.169

[16] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. 39, no. 6, pp. 1137-1149, 2017, doi: 10.1109/TPAMI.2016.2577031.

[17] J. Dai, Y. Li, K. He, and J. Sun, "R-FCN: Object detection via region-based fully convolutional networks," in *30th Annual Conference on Neural Information Processing Systems, NIPS 2016, December 5, 2016 - December 10, 2016*, Barcelona, Spain, 2016, vol. 0: Neural information processing systems foundation, in Advances in Neural Information Processing Systems, pp. 379-387.

[18] T.-Y. Lin, P. Dollar, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, July 21, 2017 - July 26, 2017*, Honolulu, HI, United states, 2017, vol. 2017-January: Institute of Electrical and Electronics Engineers Inc., in Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, pp. 936-944, doi: 10.1109/CVPR.2017.106. [Online]. Available: http://dx.doi.org/10.1109/CVPR.2017.106

[19] Q. Hou, D. Zhou, and J. Feng, "Coordinate attention for efficient mobile network design," in *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2021, June 19, 2021 - June 25, 2021*, Virtual, Online, United states, 2021: IEEE Computer Society, in Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 13708-13717, doi: 10.1109/CVPR46437.2021.01350. [Online]. Available: http://dx.doi.org/10.1109/CVPR46437.2021.01350

[20] D. Sridhar, N. Quader, S. Muralidharan, Y. Li, P. Dai, and J. Lu, "Class Semantics-based Attention for Action Detection," in *18th IEEE/CVF International Conference on Computer Vision, ICCV 2021, October 11, 2021 - October 17, 2021*, Virtual, Online, Canada, 2021: Institute of Electrical and Electronics Engineers Inc., in Proceedings of the IEEE International Conference on Computer Vision, pp. 13719-13728, doi: 10.1109/ICCV48922.2021.01348. [Online]. Available: http://dx.doi.org/10.1109/ICCV48922.2021.01348

[21] A. Vaswani *et al.*, "Attention is all you need," in *31st Annual Conference on Neural Information Processing Systems, NIPS 2017, December 4, 2017 - December 9, 2017*, Long Beach, CA, United states, 2017, vol. 2017-December: Neural information processing systems foundation, in Advances in Neural Information Processing Systems, pp. 5999-6009.

[22] S. Woo, J. Park, J.-Y. Lee, and I. S. Kweon, "CBAM: Convolutional block attention module," in *15th European Conference on Computer Vision, ECCV 2018, September 8, 2018 - September 14, 2018*, Munich, Germany, 2018, vol. 11211 LNCS: Springer Verlag, in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), pp. 3-19, doi: 10.1007/978-3-030-01234-2_1. [Online]. Available: http://dx.doi.org/10.1007/978-3-030-01234-2_1

[23] Q.-L. Zhang and Y.-B. Yang, "SA-Net: Shuffle attention for deep convolutional neural networks," in *2021 IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP 2021, June 6, 2021 - June 11, 2021*, Virtual, Toronto, ON, Canada, 2021, vol. 2021-June: Institute of Electrical and Electronics Engineers Inc., in ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings, pp. 2235-2239, doi: 10.1109/ICASSP39728.2021.9414568. [Online]. Available: http://dx.doi.org/10.1109/ICASSP39728.2021.9414568

[24] X. Zhu, S. Lyu, X. Wang, and Q. Zhao, "TPH-YOLOv5: Improved YOLOv5 Based on Transformer Prediction Head for Object Detection on Drone-captured Scenarios," in *18th IEEE/CVF International Conference on Computer Vision Workshops, ICCVW 2021, October 11, 2021 - October 17, 2021*, Virtual, Online, Canada, 2021, vol. 2021-October: Institute of Electrical and Electronics Engineers Inc., in Proceedings of the IEEE International Conference on Computer Vision, pp. 2778-2788, doi: 10.1109/ICCVW54120.2021.00312. [Online]. Available: http://dx.doi.org/10.1109/ICCVW54120.2021.00312

[25] Z. Zhu, D. Liang, S. Zhang, X. Huang, B. Li, and S. Hu, "Traffic-Sign Detection and Classification in the Wild," in *29th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, June 26, 2016 - July 1, 2016*, Las Vegas, NV, United states, 2016, vol. 2016-December: IEEE Computer Society, in Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 2110-2118, doi: 10.1109/CVPR.2016.232. [Online]. Available: http://dx.doi.org/10.1109/CVPR.2016.232

[26] J. Zhang, X. Zou, L.-D. Kuang, J. Wang, R. S. Sherratt, and X. Yu, "CCTSDB 2021: A More Comprehensive Traffic Sign Detection Benchmark," *Human-centric Computing and Information Sciences,* vol. 12, 2022, doi: 10.22967/HCIS.2022.12.023.

[27] M. A. L. Montiel, Y. J. Rubio, M. Sánchez, and U. Orozco-Rosas, "Evaluation of algorithms for traffic sign detection," in *Optics and Photonics for Information Processing XIII*, 2019.

[28] Huaixu. Gao and Ying. Tian, "Research on Road-Sign Detection Algorithms Based on Depth Network," *Engineering Letters,* vol. 31, no. 1, pp. 136-142, 2023.

[29] Z. Tian, C. Shen, H. Chen, and T. He, "FCOS: Fully convolutional one-stage object detection," in *17th IEEE/CVF International Conference on Computer Vision, ICCV 2019, October 27, 2019 - November 2, 2019*, Seoul, Korea, Republic of, 2019, vol. 2019-October: Institute of Electrical and Electronics Engineers Inc., in Proceedings of the IEEE International Conference on Computer Vision, pp. 9626-9635, doi: 10.1109/ICCV.2019.00972. [Online]. Available: http://dx.doi.org/10.1109/ICCV.2019.00972

[30] 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.