Academic Journal of Science and Technology
ISSN: 2771-3032 | Vol. 10, No. 1, 2024

Research on Target Detection Algorithm Based on Aerial Video

Xiangchen Liu, Lin Zhang

North China University of Science and Technology, Hebei, Tangshan, 063210, China

Abstract: For the Yolov5 target detection algorithm, firstly, the DPFEM network is proposed to replace the original BottleneckCSP and C3 network structure for the problem of inadequate feature extraction for small targets, secondly, the MAFFEM module is proposed to alleviate the conflict of feature fusion due to the fusion conflicts brought by the different scales of the feature maps during the feature fusion, finally, the training The results show that the improved Yolov5 algorithm mAP0.5 and mAP0.5:0.95 are improved by 2.2% and 1.7%, respectively, and have certain application potential.

Keywords: Deep learning; UAV aerial photography; YOLOv5; attention module; target detection algorithm.

1. Introductory

Deep learning based target detection method through deep learning convolutional neural network and other technologies, so that the computer autonomously on the detection of the target feature extraction, autonomous learning to update the weight parameters, so as to realize the detection and classification. According to the algorithm steps are divided into two-stage algorithm and one-stage algorithm.

1.Two-stage algorithm

The two-stage algorithm, as the name suggests, divides the detection process into two stages, the first step is to screen the image information first, select the region containing the detection target, sieve out the redundant background information, and then the second step is to regress the position of the target in these selected regions and classify them. Therefore the two-stage algorithm is also known as candidate region (Region proposal) based target detection. It is the forerunner of deep learning based detection algorithms, represented by algorithms such as RCNN series and SPPNet.

R-CNN algorithm[1] In 2014, Girshick R et al. proposed the RCNN algorithm inspired by the idea of ImageNet algorithm.The workflow of RCNN algorithm is to first obtain the input image, extract the candidate regions, and then scale the image of each candidate region to a fixed size of 224x224, input into the CNN network, and then input the results into the classifier for category determination.The CNN network task includes feature classification and edge regression. The CNN network tasks include feature classification and edge regression.The RCNN algorithm mainly has the following problems, for example, each candidate region needs to be repeated in the subsequent judgment, this step greatly wastes the arithmetic power, so the detection speed of this algorithm is very slow, and due to the fixed size of the output, the detection algorithm is often unsatisfactory for images with various types of targets.

Aiming at the above problems of R-CNN, Fast R-CNN algorithm [2]proposed by Girshick R in 2015 has been improved in many aspects. For example, the backbone network adopts the VGG16 network to realize the lightweight calculation, and compared with R-CNN, the training speed of Fast R-CNN is 10 times faster than that of R-CNN, and the accuracy of the VOC2012 dataset can reach 68.4%.

The Faster R-CNN algorithm[3]proposed by Ren S et al. in 2016 was further improved by introducing an additional RPN structure and adding an Anchor, i.e., there will be some prior frames before learning, and the subsequent learning can go according to these prior frames, thus reducing the difficulty of learning. Fast R-CNN analyzes the region of interest after convolutional operation for classification and regression, but it contains a lot of useless operations, so the rate is low, while Faster R-CNN improves this problem, so that the speed of the operation is greatly improved, and the accuracy in the VOC2012 dataset reaches 70.4%.

Many scholars in China have studied the RCNN series of algorithms, Wang Xinze, He Chao[4] and others have improved the Faster RCNN model, which integrates the advantages of the Transformer model by increasing the convolution kernel to increase the receptive field, replacing the ordinary convolution with the DW convolution to enhance the detection performance of the model and improving the loss parameter, which improves the model's ability of detecting objects at different scales.

Zhou Shaohong, Fang Xinxin[5] et al. used ResNet50 network instead of the original VGG16 network for feature extraction of the target and used the pre-trained weights as the initial weights to improve the training effect, and also increased the number of anchor frames of the original algorithm to obtain richer feature map information, and the improved Faster RCNN model improved the detection effect and stability.

Du Yunyan[6]et al. proposed a Faster RCNN model for a small number of target samples in response to the problem of a small amount of model training samples, by incorporating the CBAM attention mechanism into the RPN module, screening out a portion of the candidate frames, and proposing a global and local relationship detector to obtain the relationship between a small number of labeled samples and features of the samples to be detected, which reduces the interference of useless information and improves the model detection accuracy.

2. One-stage algorithm

Single-stage algorithms directly input image through the convolutional neural network directly on the target detection, to achieve the detection of the target position regression and classification. The representative algorithms are Yolo series algorithm and SSD algorithm.In 2016, YOLO[7]proposed by Redmon et al. combines target recognition and determination

by segmenting a 448 × 448 image into grids at the same intervals, each of which predicts the position of the two candidate frames as well as the confidence of whether they contain an object or not, and then obtains detection results by Non-Maximum Suppression (NMS)[8]obtain the detection results, this approach makes YOLO detection of an image takes only 20ms, and the inference speed of the related lightweight model can reach 155fps. Although this approach leads to a slightly lower detection accuracy than the Faster R-CNN, but it is still much higher than the traditional algorithms, and its high detection speed to better meet real-time demand. In the same year, SSD[9]was proposed to improve the generalization ability of YOLO, using different resolutions of feature maps to calculate the classification information and regression results, in order to optimize the scale-insensitive problem of YOLO.YOLO algorithms have also been followed up to address this problem, and multi-scale detectors have been added to improve the feature extraction ability of the network, which led to the derivation of YOLOv2[10]and YOLOv3[11]and algorithms. However, these improvements did not change the phenomenon that the detection accuracy of the single-stage algorithm is lower than that of the two-stage algorithm.Lin et al. argued that the single-stage algorithm is trained to learn the whole map, while the two-stage algorithm only needs to train the positive samples, and therefore there is a sample imbalance problem, to solve this problem, Lin et al. proposed Focal Loss, to improve the detector's learning for difficult samples, and used this to established the RetinaNet [12]algorithm to further improve the detection accuracy, thus narrowing the accuracy difference between the single-stage algorithm and the two-stage algorithm.

The target detection algorithms proposed since then have focused more on the improvement of feature fusion, training techniques, and the improvement of the network infrastructure module. YOLOv4[13] proposed in 2020 was oriented in this direction, and listed the latest technologies and achievements at that time, and conducted a series of ablation experiments by permutation and combination, and selected the optimal combination of the YOLOv3 to form YOLOv4, which further improved the overall progress. In 2020, YOLOv5 code was open-sourced and upgraded in engineering applications, becoming the first choice for major projects. In recent years, YOLOv6, YOLOv7, and YOLOv8 have been proposed to further develop the YOLO family.

2. Introduction to the YOLOv5 network

YOLOv5 consists of four parts: input, backbone network, Neck network and Prediction. The network structure is shown in Figure 1.

(1) Input

The input side includes three parts: mosaic data enhancement, adaptive anchor frame calculation and adaptive image scaling. Mosaic data enhancement is done by enhancing the data of four randomly selected images and finally stitching and combining them. The advantage of this method is that it increases the amount of data, which makes the network more robust and reduces the burden of GPU computing. Adaptive anchor frame computation is based on the characteristics of different datasets and outputs predicted frames based on the initial anchor frames, and then compares them with the real frames to compute the differences between them. Adaptive Image ScalingFor images with different aspect ratios, adaptive scaling fills the image according to the standard size to meet the training requirements, reduce the amount of computation and improve the detection speed.

(2) Backbone

The Backbone network consists of three main structures: the CBS convolution module, the CSP feature extraction network, and the spatial pyramid pooling SPPF.The CBS consists of Convolution, Batch Normalization, and SiLU activation functions.The SPPF concatenates multiple MaxPool layers to achieve feature fusion at different scales. to achieve feature fusion at different scales.

(3) Neck

Neck feature fusion network is mainly composed of two parts: feature pyramid network and path aggregation network.FPN is up-sampling the high-level features from bottom to top, and enhances the semantic information by fusing the information with the low-level features.PAN adds top-to-bottom feature fusion on the basis of FPN, and adopts down-sampling method to transfer the low-level features to the high-level for information fusion, which strengthens the ability of bottom information localization. The The combination of the two enhances the sensitivity of the model to small targets.

(4) Head

The Head detection layer consists of three detection heads that detect feature maps of different sizes for final detection of the target. In order to make the prediction frame regression faster and more accurate, the loss function in YOLOv5 uses CIOU-Loss, and in order to prevent the repetition of similarly sized frames, the final prediction results are filtered by non-maximum suppression method.

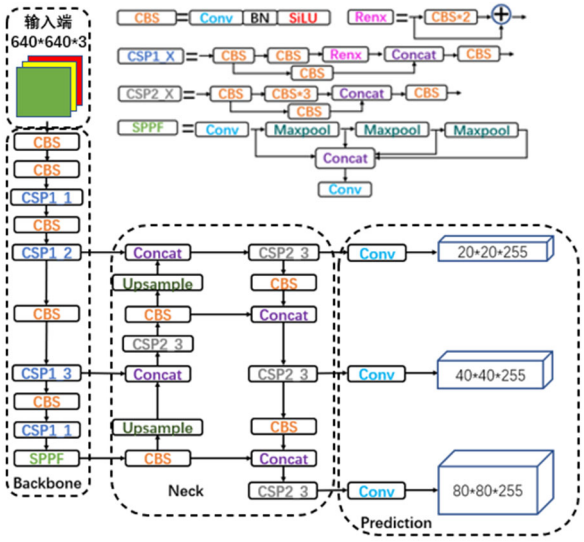


Figure 1. YOLOv5 network architecture

YOLOv5 contains a variety of models, such as YOLOv5s and YOLOv5m models, whose main difference lies in the fact that the depth and width of the network can be controlled to get different sizes of models. In order to speed up the detection and achieve real-time detection, this paper uses YOLOv5s as the benchmark model.

3. Improvement of YOLOv5 Algorithm

3.1. Convolutional Block Attention Module

CBAM, as an effective convolutional attention module, can be seamlessly integrated into CNN architectures and trained

end-to-end with basic CNNs with high applicability.CBAM is divided into two independent sub-modules, Channel Attention Module and Spatial Attention Module, which perform attentional feature fusion in channel and spatial dimensions, respectively.The structure of CBAM is shown in Figure 2. Module, respectively, for the fusion of attention features in the channel and spatial dimensions, the structure of CBAM is shown in Figure 2.

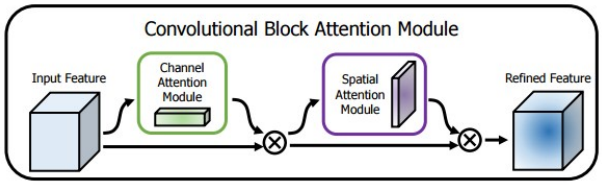


Figure 2. CBAM structure

(1) Channel Attention Module

The channel attention firstly performs Max Pooling and Average Pooling operations on the input feature map F to obtain the corresponding two feature vectors, which are respectively summed up to obtain the feature vector of 1x1xC through the shared fully-connected layer, and then performs sigmoid activation operations on it to obtain the channel attention feature . The channel attention feature map is obtained by multiplying with the input feature map F . The structure of the channel attention module (CAM) is shown in Fig. 3.

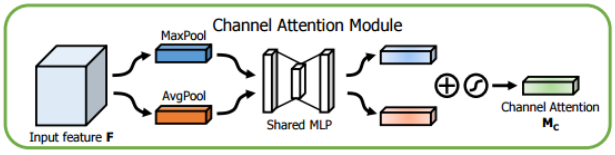


Figure 3. Channel Attention Module

The mathematical expression for the channel attention characteristic is shown in equation (1):

$$\begin{aligned} Mc(F) &= \sigma(MLP(MaxPool(F)) + MLP(AvgPool(F))) \\ &= \sigma(W_1(W_0(F^c_{\max})) + W_1(W_0(F^c_{\text{avg}}))) \end{aligned} \quad (1)$$

The mathematical expression for the channel attention feature map is shown in equation (2):

$$Fc = Mc(F) \otimes F \quad (2)$$

(2) Spartial Attention Module

Spatial attention focuses on the location information of the target in the input image. Take the final output feature map Fc from the channel attention module as input, perform Max Pooling and Average Pooling operations on it, splice the two output feature vectors, and then go through a 7x7 convolutional layer and sigmoid activation to get the spatial attention features . The input feature map is multiplied with the input feature map to obtain the channel attention feature map. The structure of the spatial attention module (SAM) is shown in Fig. 4.

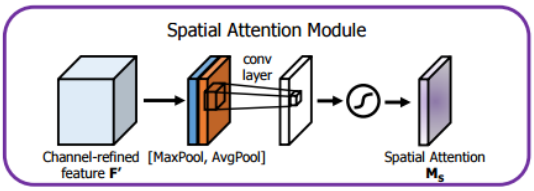


Figure 4. Spatial attention module

The mathematical expression for the spatial attention feature is shown in equation (3):

$$\begin{aligned} Ms(F) &= \sigma(f^{7x7}([MaxPool(Fc); AvgPool(Fc)])) \\ &= \sigma(f^{7x7}([F^s_{\max}; F^s_{avg}])) \end{aligned} \quad (3)$$

The mathematical expression of the spatial attention feature map is shown in equation (4):

$$Fs = Ms(Fc) \otimes Fc \quad (4)$$

YOLOv5 uses the same weighting for all features of different sizes and has no attentional bias in the feature extraction process. While the background of UAV aerial images is complex, there is a large amount of redundant information affecting the network to extract vehicle features. Therefore, in this paper, by introducing the CBAM module, the network can suppress the background information interference to pay more attention to the UAV aerial vehicle in the detection process.

3.2. DPFAM networks

Our algorithm model uses CSP darknet as the backbone network, and DPFEM (Double path feature enhancement module) is used in the neck part instead of the traditional bottleneck CSP and C3 network structure, which combines the advantages of ResNet and DenseNet, and can adapt better to the The structure of DPFEM is shown in Fig. 5. This module is improved by bottleneck CSP, inspired by Dual Path Networks (DPN)[14], we divide the input features into two parts f[:i] and f[i:] for residual learning and dense connectivity respectively. Where i is a hyperparameter. In the main path, 1×1 convolution is first used to reduce the number of channels and thus the number of parameters. In order to extend the sensory field, as well as to better adapt to the shape and orientation changes of different targets, we use the residual unit ResUnit instead of the traditional 3×3 convolutional layer to improve the feature extraction ability of the network for small-sized targets. Second Order Response Transform ,SORT [15]enhances the nonlinear fitting ability of the network and allows the network to adapt to more complex feature distributions. Therefore, in the process of residual feature fusion, we adopt SORT to replace the direct summation of features, while further optimizing the fusion method, which is represented by the following equation:

$$S_{out} = N(f)[:i] + f[:i] + \sqrt{N(f)[:i] \otimes f[:i] + \varepsilon} \quad (5)$$

Where $N(f)$ denotes the output features of the main path, and ε is a very small constant to ensure the stability of the gradient computation, which is taken as 0.00001. Next, we densely concatenate f[i:] with N(f)[i:] in order to mine new features. Finally, we adjust the number of output feature channels by a 1 × 1 convolutional layer.The operation of DP-

FEM is computed as follows:

$$N(f) = C_{1x1}\left(ResUnit\left(C_{1x1}(f)\right)\right) \quad (6)$$

$$f_{out} = C_{1x1}(concat[f[i:], N(f)[i:], S_{out}]) \quad (7)$$

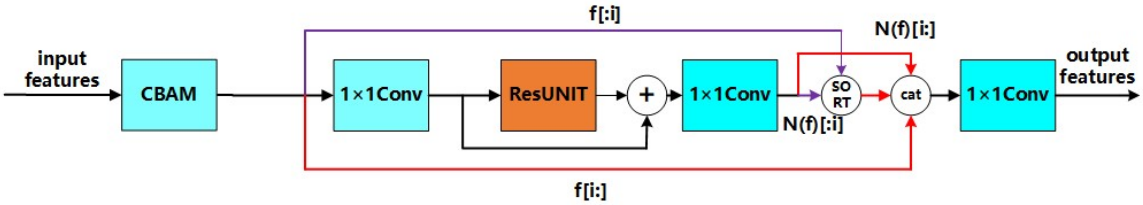


Figure 5. DPFEM network structure

3.3. MAFFEM

Since PANet often directly adopts the splicing approach when performing feature fusion of different layers, this practice will cause the obtained features of three different scales to have feature inconsistency problems. To address this problem, we improve a (Multi-layer attention feature fusion enhancement module) MAFFEM to be added before the head detector layer, which allows each feature layer to autonomously learn its desired features while retaining the key features of the layer, thus improving the scale of the features. Invariance. We denote the feature maps of each stage of PANet as {F3,F4,F5}, and realize feature fusion in two ways. First, we extract the channel attention from each of the three feature maps via the CBAM (Spatial Channel Attention Module) module, and realize Adaptive Channel Attention Fusion (ACAF) via network autonomous learning. Similarly, we directly perform adaptive feature fusion on the three feature maps. These two processes can be expressed by the following equation:

$$ACAF_{ij}^{l} = x_{ij}^{3\to l} \cdot a_{ij}^{l} \cdot CBAM(x_{ij}^{3\to l}) + x_{ij}^{4\to l} \cdot \beta_{ij}^{l} \cdot CBAM(x_{ij}^{4\to l}) + x_{ij}^{5\to l} \cdot \gamma_{ij}^{l} \cdot CBAM(x_{ij}^{5\to l}) \quad (8)$$

where x_{ij} denotes the (i,j) vector of the feature map, α_{ij}, β_{ij}, γ_{ij} are the weights corresponding to the three feature maps, whose values are learned by the network autonomously, and $\alpha_{ij} + \beta_{ij} + \gamma_{ij} = 1$. Finally, we further adaptive fusion the feature fusion maps obtained from these two processes. Thanks to the above improvements, MAFFEM fuses features at different levels while allowing the network to autonomously select and retain the most appropriate features.The structure of the MAFFEM module is shown in Figure 6 below:

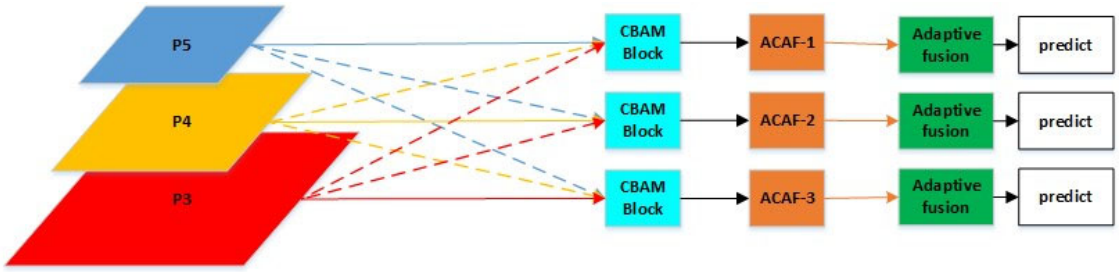


Figure 6. Structure of MAFFEM network

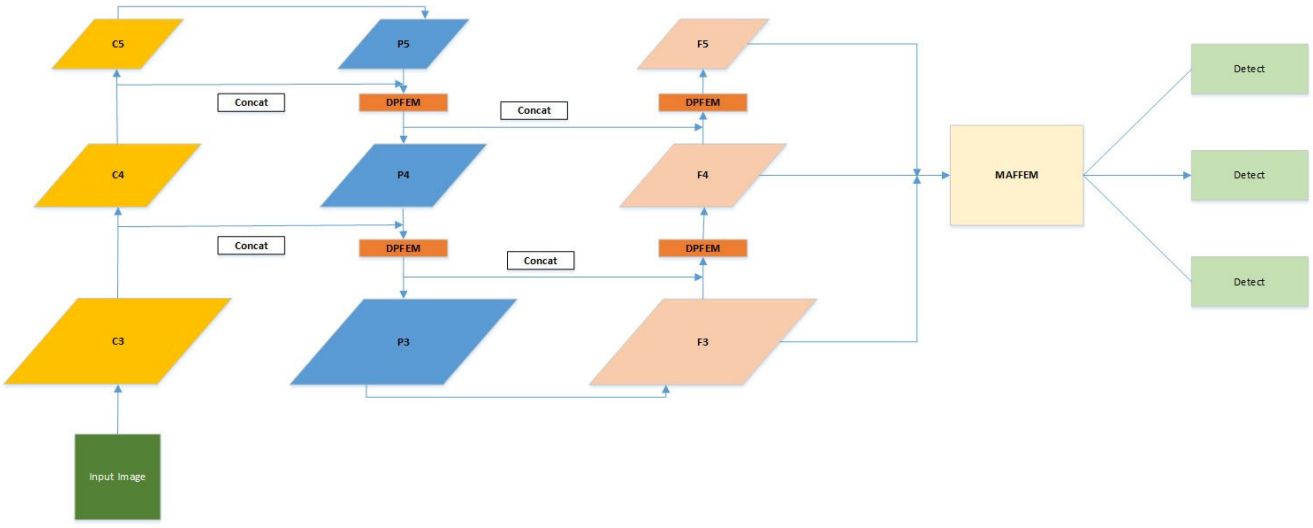


Figure 7. Improved yolov5 network structure

4. Experimental Analyses

4.1. Experimental environment and training setup

(1) The experimental configuration environment is shown in Table 1, and the training parameter settings are shown in Table 2:

Table 1. Experimental configuration environment

experimental environment	parameters
Deep Learning Framework	Pytorch
operating system	Windows 10
GPU	NVIDIA GeForce GTX 2060
Programming Tools	Pycharm
programming language	Python3.8

Table 2. Training settings

training parameter	parameter size
Input image size	640×640
pre-training weight	YOLOv5s
Initial learning rate	0.01
Weight decay factor	0.0005
Category Confidence Threshold	0.5
epochs	100
Batchsize	16

4.2. Evaluation indicators

In this paper, we validate the model detection effect by three commonly used performance metrics, namely precision rate (Precision,P), recall rate (Recall,R) and average precision mean[16]. Precision rate indicates the ratio of the number of samples predicted as positive to the number of true positive samples, as shown in Equation (9). Recall rate indicates the proportion of the number of correctly predicted samples to the number of positive samples of the original sample[17], as shown in Equation (10). Mean precision mean indicates the average precision averaged over all categories[18], as shown in Equation (11). The precision rate verifies the effectiveness of a classifier, and usually, the precision rate is negatively correlated with the recall rate, the higher the precision rate, the lower the recall rate. The average precision mean represents a comprehensive evaluation of the average precision of the detected targets, which can more intuitively show the performance of the classifier[19].

$$P = \frac{TP}{TP + FP} \tag{9}$$

$$R = \frac{TP}{TP + FN} \tag{10}$$

$$mAP = \frac{\sum_{i=1}^{N} AP_i}{N} \tag{11}$$

where TP denotes the number of correctly detected frames; FP denotes the number of incorrectly detected frames; FN denotes the number of falsely detected frames; AP denotes the area enclosed by the curve of precision P and recall R, i.e., the average precision; and denotes the total number of categories[20].

4.3. Experimental analyses

(1) Dataset creation

The detection effect of deep learning target detection model relies on rich dataset, in this paper, on the basis of Visdrone2019 aerial photography open source dataset produced by Machine Learning and Data Mining Laboratory of Tianjin University, we shoot aerial video to collect the data and annotate it by ourselves, and then finally train the model by combining with Visdrone2019 aerial photography dataset.

In order to enhance the generalization ability of the model, a high-precision UAV aerial photography dataset is established, and data collection is carried out by using UAVs to shoot on various roads in Tangshan city area, and the shooting scenes include evening, cloudy and rainy days and sunny days with strong light, which are manually annotated by using LabelImg annotation tool, and fused with the Visdrone2019 dataset into a high-precision UAV aerial photography dataset. The detection effect of the improved yolov5 model is evaluated on this dataset. Finally, 5000 images with labels are selected from VisDrone2019 dataset and fused with 1000 images manually labeled to form a detection dataset. The dataset division is performed with 80% as training set and 20% as testing set.

(2) Comparative analysis of performance test results

The performance test results of the improved YOLOv5 algorithm in this paper and the classical YOLOv5 algorithm on divided datasets of different sizes are shown in Table 3.

Table 3. Comparison of performance test results

	P	R	mAP0.5	mAP0.5:0.95
Original	0.487	0.344	0.346	0.185
DPFEM	0.483	0.391	0.353	0.199
MAFFEM	0.505	0.38	0.361	0.196
ALL	0.523	0.396	0.368	0.202

The DPFEM module adopts a dual-path structure, which is able to mine new features while realizing feature extraction, and improves the feature extraction capability of the network, thus improving mAP0.5 and mAP0.5:0.95 by 0.7% and 1.4%, respectively.MAFFEM fuses different levels of features by autonomously learning to select the key features of different feature layers required by the network, and at the same time allows the network to autonomously select and retain the most appropriate features.The use of the MAFFEM module resulted in an improvement of 1.5% and 1.1% for mAP0.5 and mAP0.5:0.95, respectively. When both DPFEM and MAFFEM modules are added to the network, mAP0.5 and mAP0.5:0.95 are improved by 2.2% and 1.7%, respectively. It can be seen that compared with the classical YOLOv5 algorithm, the precision, recall and mean average precision of the improved algorithm have been improved significantly, and the generalization ability of the model has been improved to some extent.

Table 4. Comparison table of detection results of main detection targets

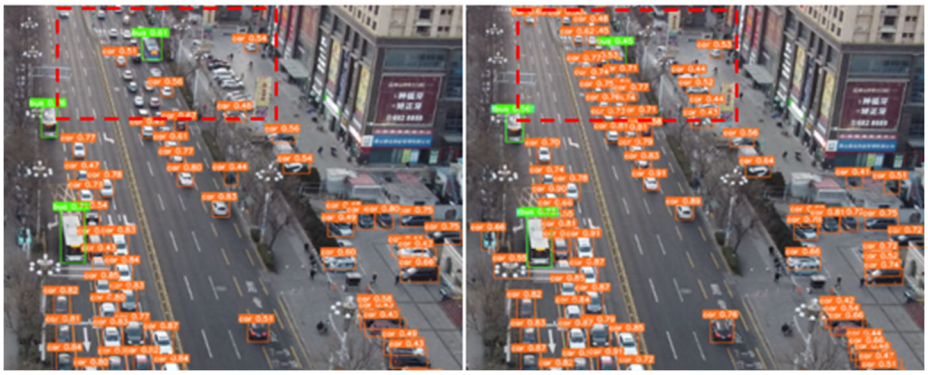
	detection target	P	R	mAP0.5	mAP0.5:0.95
Original	Pedestrians	0.488	0.393	0.376	0.152
	Car	0.634	0.736	0.716	0.462
	Bus	0.577	0.418	0.417	0.251
Revised	Pedestrians	0.511	0.474	0.453	0.192
	Car	0.62	0.788	0.763	0.505
	Bus	0.613	0.446	0.45	0.284

Due to the dataset aerial photography target scale is small, labeling different categories of training samples is not balanced and other issues that lead to greater difficulty in training the model, training to get all the categories of mAP value is low, but the training samples are sufficient categories of pedestrians, vehicles, public transport, and other detection of detection of the target detection effect to achieve a certain degree of accuracy, in particular, the detection of the vehicle recognition improved mAP value can reach 76.3%.

(3) Comparative Analysis of Aerial Photography Example Detection

The weight parameter matrix generated after training is saved in the runs/train/exp/weights folder, best.pt represents the weight matrix with the best training effect, and last.pt represents the weight matrix of the last training. The best.pt weight parameter matrix trained with the model before and after the improvement is used to validate the detection results, as shown below. From the comparison chart of different height detection, it can be seen that with the increase of aerial photography height, the detection target scale becomes smaller resulting in increased detection difficulty, and problems such as missed detection and misdetection obviously increase, but the improved model algorithm reduces the misdetection and misdetection and other problems to a certain extent, and the confidence level of the detection target is increased, which can be seen that the improved model detection effect is improved.

(a) Drone flying at 100 meters



(b) Drone flying at 110 meters

(c) Drone flying at 120 meters

Figure 8. Comparison chart of detection effect

5. Conclusion

Some problems of Yolov5 detection algorithm are optimized and improved to enhance the detection accuracy of the algorithm. Aiming at the problem of insufficient feature extraction, an improved DPFEM network is proposed to

replace the original C3 network structure to strengthen the feature extraction ability of the model. Aiming at the problem that the extracted features have different scales and thus cause feature fusion conflicts, the MAFFEM module is proposed to enable the computer to autonomously learn and regulate the key features of the features, so as to improve the problem of feature conflicts. Finally, the model is trained on a homemade training set, and the mAP0.5 and mAP0.5:0.95 of the model are improved by 2.2% and 1.7%, respectively, and the results show that the detection accuracy is improved. As a phase study, the improved algorithm proposed in this paper has great potential for practical application, especially providing algorithmic support for subsequent vehicle counting, vehicle tracking and vehicle trajectory prediction, while it can be further applied to the detection of non-motorized vehicles and pedestrians in future transportation fields.

Bibliography

[1] Girshick R, Donahue J, Darrell T, et al. Rich feature hierarchies for accurate objectdetection and semantic segmentation [C]//Proceedings of the IEEE conference on computervision and pattern recognition. 2014: 580-587.

[2] Girshick R. Fast r-cnn[C]//Proceedings of the IEEE international conference on computervision. 2015: 1440-1448.

[3] Ren S, He K, Girshick R, et al. Faster r-cnn: Towards real-time object detection with regionproposal networks[J].Advances in neural information processing systems, 2015, 28: 91-99.

[4] X. Z. Wang,C. He. Vehicle detection in complex environments based on Transformer improved Faster RCNN[J/OL]. Electromechanical Engineering Technology. https://link.cnki.net/urlid/44.1522.TH.20240312.1557.002.

[5] SHAO-HONG ZHOU, XIN-JIN FANG, XIN-YI LIU, EDDIE ZHANG, SHENG YAN. Aircraft target detection based on migration learning and improved Faster-RCNN remote sensing imagery[J/OL]. Mechatronics.https://link.cnki.net/urlid/44.1522.TH.20240222.1433.002.

[6] Du Yunyan,Yang Jinhui,Li Hong,et al. Sample less target detection algorithm based on improved Faster RCNN[J]. Electro-Optics and Control,2023,30(5):44-51.

[7] Redmon J, Divvala S, Girshick R, et al. You only look once: Unified, real-time object detection[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2016: 779-788.

[8] Rothe R, Guillaumin M, Van Gool L. Non-maximum suppression for object detection by passing messages between windows[C]//Asian Conference on Computer Vision. Springer, Cham, 2014: 290-306.

[9] Liu W, Anguelov D, Erhan D, et al. Ssd: Single shot multibox detector[C]. European conference on computer vision. Springer, Cham, 2016: 21-37.

[10] Redmon J, Farhadi A. YOLO9000: better, faster, stronger[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2017: 7263-7271.

[11] Redmon J, Farhadi A. Yolov3: An incremental improvement[J]. arXiv preprint arXiv:1804.02767, 2018.

[12] Lin T Y, Goyal P, Girshick R, et al. Focal loss for dense object detection[C] //Proceedings of the IEEE International Conference on Computer Vision. 2017: 2980-2988.

[13] Lin T Y, Dollár P, Girshick R, et al. Feature pyramid networks for object detection[C]. Proceedings of the IEEE conference on computer vision and pattern recognition. 2017:2117-2125.

[14] Y . Chen et al., “Dual path networks,” in Proc. Adv. Neural Inf. Process.Syst., 2017, pp. 1–9.

[15] Wang Y, Xie L, Liu C, et al. Sort: Second-order response transform for visual recognition[C]//Proceedings of the IEEE International Conference on Computer Vision. 2017: 1359-1368.

[16] ZHAO Lulu, WANG Xueying, ZHANG Yi, et al. Research on vehicle target detection technology based on YOLOv5s fusion SENet[J/OL]. Journal of Graphics:1-8[2022-05-23].http://kns.cnki.net/kcms/detail/10.1034.T.20220513.1447.004.html

[17] Kuang Xianxiang,Liu Ping. Vehicle detection method for complex scenes based on improved YOLOv5s[J]. Modern Computer,2022,28(07):47-52.

[18] Feng Z, Xie ZJ, Bao ZW, et al. Real-time dense small target detection algorithm for UAV based on improved YOLOv5[J/OL]. Journal of Aeronautics:1-15[2022-05-16].http://kns.cnki.net/kcms/detail/11.1929.V.20220509.2316.010.html

[19] LONG Sai,SONG Xiaofeng,ZHANG Su,et al. Research on vehicle detection in aerial images with improved YOLOv5s[J/OL]. Laser Journal:1-9[2022-05-23].http://kns.cnki.net/kcms/detail/50.1085.TN.20220331.1716.026.html

[20] LIU Chaoyang,QU Jinshuai,FAN Jing,et al. Vehicle target detection based on improved YOLOv5 algorithm[J/OL]. Journal of Yunnan University for Nationalities(Natural Science Edition):1-9[2022-05-23].http://kns.cnki.net/kcms/detail/53.1192.N.20220421.0940.015.html