Hindawi
Complexity
Volume 2021, Article ID 9167116, 10 pages
https://doi.org/10.1155/2021/9167116

Research Article

Research on Multiscene Vehicle Dataset Based on Improved FCOS Detection Algorithms

Fei Yan[1,2] **Hui Zhang**[2] **Tianyang Zhou**[3] **Zhiyong Fan**[1,2] **and Jia Liu**[1,2]

[1]*College of Automation, Nanjing University of Information Science & Technology, Nanjing 210044, China*
[2]*Jiangsu Collaborative Innovation Center of Atmospheric Environment and Equipment Technology (CICAEET), Nanjing 210044, China*
[3]*College of Energy and Electrical Engineering, Hohai University, Nanjing 21100, China*

Correspondence should be addressed to Jia Liu; liujia@nuist.edu.cn

Received 25 July 2021; Revised 26 September 2021; Accepted 19 October 2021; Published 19 November 2021

Academic Editor: Cheng Lu

Whether in intelligent transportation or autonomous driving, vehicle detection is an important part. Vehicle detection still faces many problems, such as inaccurate vehicle detection positioning and low detection accuracy in complex scenes. FCOS as a representative of anchor-free detection algorithms was once a sensation, but now it seems to be slightly insufficient. Based on this situation, we propose an improved FCOS algorithm. The improvements are as follows: (1) we introduce a deformable convolution into the backbone to solve the problem that the receptive field cannot cover the overall goal; (2) we add a bottom-up information path after the FPN of the neck module to reduce the loss of information in the propagation process; (3) we introduce the balance module according to the balance principle, which reduces inconsistent detection of the bbox head caused by the mismatch of variance of different feature maps. To enhance the comparative experiment, we have extracted some of the most recent datasets from UA-DETRAC, COCO, and Pascal VOC. The experimental results show that our method has achieved good results on its dataset.

1. Introduction

In recent years, with the rapid development of the automobile industry, the number of motor vehicles in the city has also developed rapidly. By the end of 2020, the number of motor vehicles in the country has reached more than 300 million, followed by the colossal traffic pressure and many traffic regulation problems. When the traffic pressure and traffic control problems become more serious, it will bring many inconveniences to the production and life of urban residents and will also restrict the rapid development of cities and towns. On the other hand, with the gradual maturity of artificial intelligence, the application of artificial intelligence in vehicles has also received extensive attention, and the application of vehicle detection is the focus of this article. Moreover, vehicle detection covers more and more areas, such as road traffic monitoring, the automatic lifting of the entrance guard of the community, some charging pile parking places, and automated driving.

Pedestrians or private cars are often parked in unguarded charging areas, causing inconvenience to new energy owners. When the vehicle is fully charged, you need to remind the owner to move the car out of the charging pile. There are still many problems with vehicle detection in autonomous driving. The vehicle will be seriously obscured or cause misidentification in different scenarios, which is often one of the leading causes of accidents in self-driving cars. And, before the license plate recognition, it is often necessary to recognize the body to narrow the identification range, reduce the interference of the surrounding environment, and improve accuracy. Based on these conditions, the vehicle detection is still a challenging task.

The primary purpose of vehicle detection can be divided into two. One is to determine whether a vehicle is detected (such as a bus, truck, and car) in the video or image. During the detection process, it needs to determine its location and mark it. Second, the specific category needs to be

determined. The particular sort of vehicle needs to be determined by analyzing the semantic information (e.g., [1]) of the car in the frame to complete the vehicle detection task.

Neural network design often follows several elements. The main one is to reduce the information path and enhance the information dissemination; for example, the residual connection [2] and dense connection [3] have played a perfect role. It is also effective in improving the flexibility and diversity of information channels, typically using a split-transform-merge strategy [4]. There are also many methods [5–7] to combine high-resolution image information with high-level semantic information.

Driven by these cutting-edge algorithms, we propose an improved-FCOS algorithm to detect vehicles, consisting of mainly three points. Firstly, the car belongs to a rigid structure and changes laterally under different shooting angles and scene occlusions. The receptive field of the standard convolution kernel is often rectangular, extending to the surroundings, which may not completely cover the entire car. We introduce Deformable ConvNet [8] which enables the convolution kernel to adaptively learn the response's position deviation according to the target's deformation. Adding a bottom-up module to the original FPN further enhances the flow of information between different feature layers, reducing the distance between the bottom layer and the top layer. Pang et al. [9] put forward a new idea called the Libra R-CNN before this, thinking that today's detectors all follow the region selection, feature extraction, and then the gradual convergence under the guidance of multitask loss, which directly affects the effect of model training. Based on the former balance concept, we add a balanced module after the improved FPN, which integrates feature maps of different resolutions, uses this feature to strengthen the previous pyramid, and then adds nonlocal attention to enhance the contextual network connection. This operation can reduce the inconsistency of bbox head recognition due to different variances, and the whole process is cost-free through interpolation and pooling.

The work of this paper is as follows:

(1) The introduction of DCN [8] can make the receptive field of the convolution kernel to adaptively change

(2) We added a bottom-up module behind the traditional FPN [6] to reduce the distance from bottom to top

(3) We added a balanced module after the improved FPN to reduce the inconsistency of the bbox head prediction

2. Related Works

This section mainly introduces the method of research used for vehicle detection and FCOS algorithm.

2.1. Detection Algorithm. Many methods based on combining the feature extraction and classifiers have been proposed to achieve vehicle detection formerly. For example, HOG [10] feature detection and then HOG [10] and LBP [11] features are combined to improve the accuracy of the vehicle detection further; Li Xiangfeng et al. put forward the Haar [12] feature algorithm. Although these algorithms achieve better detection results in simple scenarios, they are challenging to deal with complex systems.

After 2012, the convolutional neural network-based algorithm in deep learning became popular, extracting semantic information about vehicles from different feature layers and solving the problem of insufficient robustness of traditional algorithms when the dataset is sufficient.

Detection can be done in three ways. One is a two-stage regression algorithm (e.g., [9, 13–21]). These algorithms all propose an anchor as a prior to further improve the accuracy and speed up the convergence of the network model, which is often slower than single-stage detection algorithms in speed. The second is the single-stage detection algorithm (e.g., [22–27]). These algorithms do not use a prior method such as anchor, so there are many differences in regression.

The last one is the direction proposed by Facebook, represented by a transformer [28], which introduces the transformer in NLP into CV, greatly simplifying the network model. However, it has some shortcomings in accuracy, and it is still far away in the engineering deployment, such as [29–32].

2.2. FCOS Algorithm. We choose the anchor-free network because the anchor-based algorithm has many limitations:

(1) The dataset is susceptible to the size, number, and aspect ratio of anchors, and different tasks need to be readjusted, which is not conducive to generalization

(2) To better match the GT box, many anchors need to be generated, most of which are marked as negative samples, which will cause an imbalance between positive and negative examples

(3) It is necessary to calculate the value of IoU, which consumes a lot of computing power, which slows down the detection speed and increases the cost

FCOS [22] is an anchor-free detection algorithm. The accuracy of the previously proposed anchor-free algorithm is quite different from that of the anchor-based algorithm. And, FCOS [22] successfully surpassed the anchor-based detection algorithm and became the SOTA of the year through alternative solutions.

The definition of the positive and negative samples in the FCOS [22] algorithm is quite different from before. If a location (x, y) falls into any GT box, it is a positive sample and regresses the distance between this point and the bounding box $l^*, t^*, r^*,$ and b^*, as shown in the following equation:

$$\begin{aligned} l^* &= x - x_0^{(i)}, \\ t^* &= y - y_0^{(i)}, \\ r^* &= x_1^{(i)} - x, \\ b^* &= y_1^{(i)} - y. \end{aligned} \tag{1}$$

The previous anchor-free algorithm has no optimal solution for overlapping the GT box regions, so the point

regression has ambiguity. In FCOS [22], the ambiguity is significantly reduced through the feature pyramid. As we all know, the shallow layer of the neural network is rich in more detailed features, which is beneficial to small target detection [33]. Higher level has more semantic features, which are used to detect large targets. To reduce the overlapping of objects with significant differences, the parameter m_i refers to the maximum distance of the feature map i. If a location (x, y) satisfies $\max(l^*, t^*, r^*, b^*) > m_i$ or $\max(l^*, t^*, r^*, b^*) < m_{i-1}$, the point to a negative sample is set without regression. Among them, m_i is set to 0, 64, 128, 256, 512, $+\infty$, respectively, divided into five intervals to reduce the overlapping area. If there is an overlapping area in a layer, it directly returns to the smallest area.

To further constrain those prediction boxes far away from the center of the GT box, FCOS [22] samples the centerness method to solve this problem (Equation (2)) and uses BCE loss [34] to optimize the centerness branch.

$$\text{centerness}^* = \sqrt{\frac{\min(l^*, r^*)}{\max(l^*, r^*)} \times \frac{\min(t^*, b^*)}{\max(t^*, b^*)}}. \tag{2}$$

3. Methods

This chapter will give a detailed overview of the improved part of the algorithm (the complete structure of the algorithm in Figure 1). The deformable convolution [8] is added to the suppression factor that can reduce the influence of noise and background. The added bottom-up module could significantly reduce the loss of data reaching the top. The integrated feature map is put into the nonlocal attention structure by the balanced module and then redivided to obtain a new feature map. In this paper, the improved FCOS dramatically increases the accuracy but does not increase too much calculation.

3.1. Deformable Convolution. At present, the convolution unit commonly used in the CNN is a fixed geometric structure. In the same feature layer, the receptive field size of all activation units is the same. Considering its characteristics, the vehicle changes horizontally under different shooting angles and scene occlusions. The receptive field of the standard convolution kernel is often rectangular, extending to the surroundings, and may not completely cover the entire car. Therefore, to allow the detection algorithm to adapt to the scale, posture, and geometric changes of the vehicle target, we introduced deformable convolution [8], which adaptively determines the size of the receptive field to improve the accuracy of detection and positioning. Compared with standard convolution, the DCN [8] is more in line with the actual situation. Its principle is shown in Figure 2.

We introduced the deformable convolution of the DCN [8] (it is illustrated in Figure 3). Deformable convolution is mainly composed of two parts: (1) the previous feature map is convolved to obtain the deviation; (2) according to the deviation value, the new sampling coordinates are obtained and convoluted to generate a new feature map. The model will focus on the area outside the target in actual training, introducing noise and not conducive to detection, adding a suppression factor to make the model more focused on the target we need.

For example, $R = \{(-1,-1), (-1,0), \ldots, (0,1), (1,1)\}$ denotes the coordinates of the 3×3 convolutional kernel and the convolution calculation is shown in Equation (3). Among them, Δp_k refers to the deviation value, where Δm_k denotes the suppression factor, which mainly assigns different weights to the target area and the noise background area. The sampling coordinates of the convolution kernel on the original feature map are $p_k + \Delta p_k$. In the actual calculation, the coordinates of the former have a decimal point. We use bilinear interpolation to solve this problem, as shown in Equation (4).

In the experiment, the DCN [8] was added to the C3–C5 layer of the backbone, which brought a considerable increase in accuracy, and we added the C2 feature map to improve the learning ability of the model further.

$$y(p_0) = \sum_{p_n \varepsilon R} w_k \cdot x(p_0 + p_k + \Delta p_k) \cdot \Delta m_k, \tag{3}$$

$$x(p) = \sum_q G(q, p) \cdot x(q). \tag{4}$$

3.2. Improved FPN. We modified the FCOS [22] neck module. The previous FPN [6] mainly improves the target detection effect by fusing high- and low-level features, especially for small-size targets. As we all know, high-level features contain semantic information, while low-level features contain more specific descriptions of detailed information. Driven by PAN [7] (the champion of instance segmentation competition that year), this paper adds a bottom-up path augmentation module as in Figure 4 after the traditional FPN [6]. However, in the FPN [6] algorithm, a top-down process is required. The transfer of shallow features to the top layer requires dozens or more than one hundred network layers. Obviously, after such a multilayer transfer, the superficial feature information will be seriously lost. The bottom-up path augmentation added in this article can connect the shallow features to the P2 through the lateral connection of the original FPN underneath and then pass from P2 to the top layer along with the bottom-up path augmentation. The number of layers passed is less than 10, which can better retain the shallow feature information.

3.3. Balance Module. The previous improvement makes the original image go through the FPN [6] layer to perform multiscale feature extraction from top to bottom and then go through the bottom to top to enhance the positioning feature information. The information of adjacent resolution feature maps is aggregated and strengthened, but there still exist some problems. The former does not consider the aggregation relationship of the hierarchical feature information between different resolutions, and the variance of each feature map is different. When sent to the bbox head, there

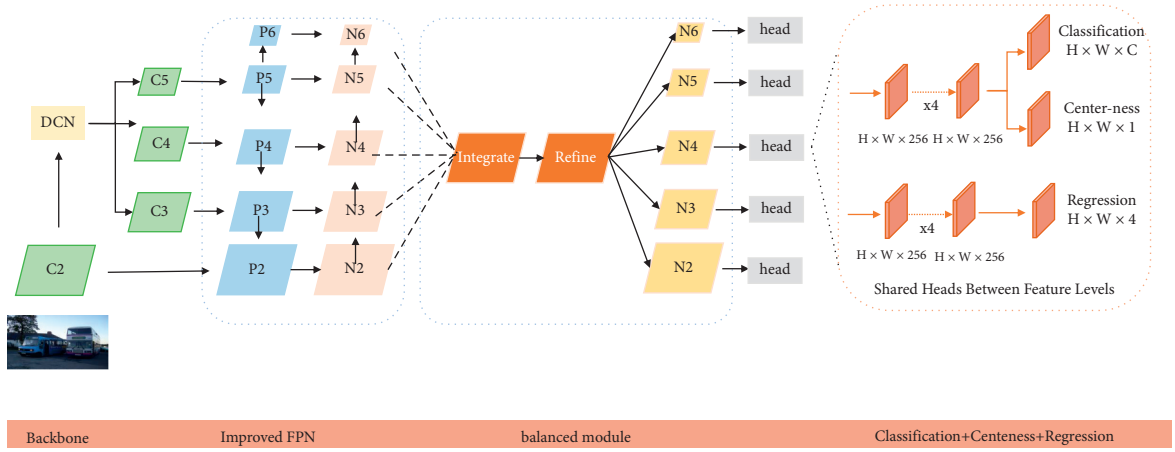


Figure 1: Improved FCOS algorithm structure: we clearly show all the tricks in the original picture for easy reading. Among them, C2~C5 are the feature layers output by the backbone, the first N2~N6 are the improved FPN, and the second is the result of adding local attention.

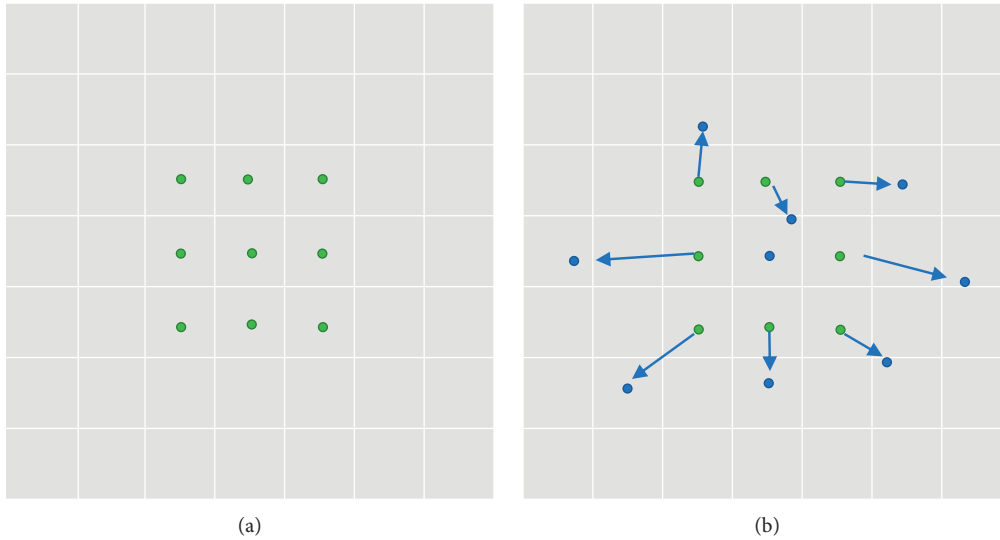


Figure 2: All examples use a 3×3 convolutional kernel. (a) Standard convolution: green dots are the sampling points. (b) Deformable convolution sampling locations (dark blue points) with augmented offsets (light blue arrows) in the deformable convolution.

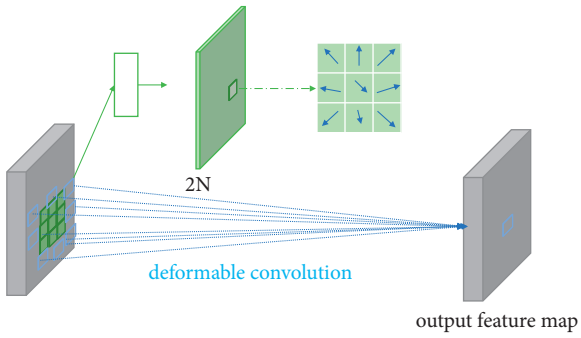


Figure 3: Illustration of 3×3 deformable convolution. 2N represents the deviation in the x and y directions, respectively.

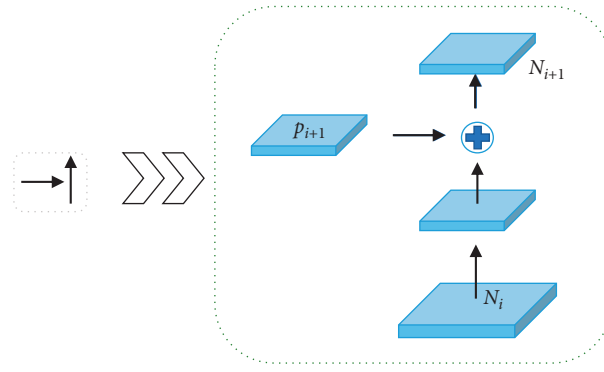


Figure 4: Block diagram of bottom-up path augmentation.

will be inconsistency problems. Thanks to Libra RCNN's [9] success, we add a balance module after the improved FPN (Figure 5). It resizes feature maps of different resolutions to the same size, then adds the feature elements together, and gets divided by the number of levels to achieve aggregation. The feature information of different scales is aggregated in the N4 feature map and then sent to the following refine structure (Equation (5)). The refine structure introduces a nonlocal attention mechanism, which is used to capture long-distance dependence, that is, how to build the connection between two nonadjacent pixels on the image. When calculating the response of an arbitrary position, nonlocal attention will consider the relationship of the feature map context to assign weights adaptively.

$$y_i = \frac{1}{C(x)} \sum_{\forall j} f\left(x_i, x_j\right) g\left(x_j\right), \tag{5}$$

where i is the location of the output feature map, j is the location of other different feature maps, x is the input feature map, f is the pairing calculation function for the two feature maps to calculate the correlation between the ith position and all other positions, g is the unary input function for transform information, and $C(x)$ is the normalization function.

Figure 6 is the specific form of nonlocal attention. First, $g(x_i)$ convolves the input feature map three times to obtain the θ, φ, and g features, and then it calculates the correlation between the two positions through the f function (3 parts in the structure diagram).

$$f\left(x_i, x_i\right) = e^{\theta\left(x_i\right)^T \phi\left(x_j\right)}. \tag{6}$$

Equations (7) and (8) calculate the Gaussian distance in the embedding space by the corresponding parts 1 and 2 in the structure diagram.

$$\theta\left(x_i\right) = W_\theta x_i, \tag{7}$$

$$\Phi\left(x_j\right) = W_\Phi x_j. \tag{8}$$

Then, the dimensions of the above three features are reshaped except for the number of channels, and the correlation is calculated by matrix point multiplication of θ and φ. Finally, the weights are 0~1 by the softmax operation, as follows:

$$C(x) = \sum_{\forall j} f\left(x_i, x_j\right), \tag{9}$$

$$\text{weight} = \frac{1}{C(x)} f\left(x_i, x_j\right). \tag{10}$$

Equations (9) and (10) are sorted out to obtain the following equation:

$$y = \text{soft}\max\left(x^T W_\theta^T W_\phi x\right) g(x). \tag{11}$$

Finally, the attention coefficient is correspondingly multiplied back to the feature matrix g, plus the number of extended channels. The result and the original input feature map are used for the residual operation (4 in the structure diagram) to obtain the refined feature map, enhancing the relationship between the feature maps and balancing the variance.

Figure 5: Balanced module flow.

3.4. Loss Function. The final loss function is

$$L\left(\left\{P_{x,y}\right\}, \left\{t_{x,y}\right\}\right) = \frac{1}{N_{\text{pos}}} \sum_{x,y} L_{\text{cls}}\left(P_{x,y}, c^*_{x,y}\right) + \frac{\lambda}{N_{\text{pos}}} \sum_{x,y} 1_{\left\{c^*_{x,y} > 0\right\}} L_{\text{reg}}\left(t_{x,y}, t^*_{x,y}\right). \tag{12}$$

In order to highlight the improvements, we have not changed the loss function of the original algorithm. Here, L_{cls} is the focal loss as in [26], and it can greatly reduce the problem of imbalance between positive and negative samples. L_{reg} is the IoU loss as in UnitBox [35], which considers that the correlation between the coordinates is different from the weighted sum of L1 and L2 loss. N_{pos} denotes the number of positive samples, and λ being 1 in this paper is the balance weight for L_{reg}. The summation is calculated over all locations on the feature maps F_i. $1_{\{c_i^* > 0\}}$ is the indicator function, being 1 if $c_i^* > 0$ and 0 otherwise.

4. Experiment

Our experiment performed detection on three diverse datasets, including UA-DETRAC, MSCOCO2017, and Pascal VOC, for joint training (where each dataset only uses pictures in the category of car, bus, and truck).

4.1. Experimental Details. UA-DETRAC (a multitarget tracking dataset, taken on different roads in Beijing and Tianjin, China) contains various weather conditions, such as cloudy, night, sunny, and rainy. Occlusion is divided into unoccluded and heavy occlusions; the video per second recording had 25 frames and about 130,000 pictures. The pictures are highly similar, and this article takes a sample every 40 frames by increasing the contrast to prevent overfitting. The second dataset selected part of the vehicle pictures in Pascal VOC2012. The third dataset uses MSCOCO2017. The full name of COCO is "Common Objects in Context," a dataset provided by the Microsoft team for image recognition. The images in the dataset are divided into training, verification, and test sets. This article samples all vehicle pictures in the training and validation sets, and the total dataset is about 20,000 (the specific information is in Table 1). All the annotation information is represented by the VOC format, 90% is used for training and verification, and the remaining 10% is used for the testing.

The experiments used in this article are all based on the mmdetection [36] framework developed by Shangtang,

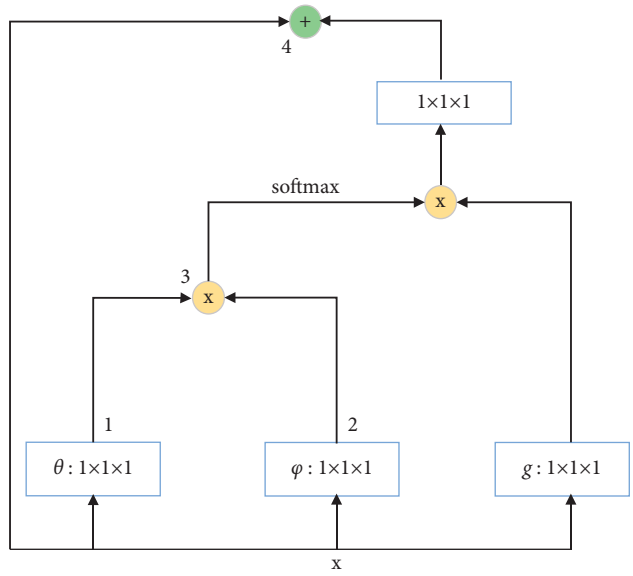


Figure 6: Nonlocal attention structure (refine).

which is developed based on the PyTorch framework. It divides the target detection algorithm into several significant modules: backbone, neck, head, bbox, encode, decode, and loss, decoupling the connection between the modules. This article uses 6 Nvidia TITAN Xp to train the network, and all the parameter settings are consistent with mmdetection official, where the learning rate and the number of GPUs have a linear scaling relationship.

4.2. Accuracy Experiment. We report our main results on the test (approximate 2K images) by uploading detection results to the server. We firstly forward the input image through the network and obtain the predicted bounding boxes with a predicted class. Unless specified, the postprocessing and data enhancement of the algorithm will use the official default of mmdetection. We hypothesize that the performance of our detector may be improved further if we carefully tune the hyperparameters. We compare the mainstream algorithms in recent years, and the results are in Table 2. Compared with other algorithms in current years, our method achieves the best performance on this dataset.

4.3. Model Complexity. We also tested the complexity of each model on the dataset as shown in Table 3. It can be seen from the table that the one-stage network often has fewer parameters than the two-stage network and our model has dramatically improved the accuracy and reduced them. For the GFLOPs indicator, the parameter has only risen a little.

4.4. Ablation Experiment. This section analyzes the performance of ablation experiments on the improved network (Table 4). The DCN [8] can expand the receptive field of the convolution kernel, which can change the sampling points with the deformation of the object. Its visualization result is shown in Figure 7, and we can see that the feature focus area

Table 1: The distribution of the dataset.

DETRAC	Pascal-VOC2012	COCO
3457	467	16977

Table 2: Our proposed algorithm vs. other algorithms with ResNet-50 and FPN as default.

Model	mAP
Faster-RCNN [13]	67.4
ATSS [23]	67.2
GFL [37]	66.8
FCOS [22]	66.8
YOLOF [38]	67.1
YOLOV3 [16]	58.9
SSD [39]	57.6
DETR [29]	61.0
Centernet [40]	51.3
Retinanet [9]	66.0
Ours	**70.1**

Table 3: Model complexity comparison.

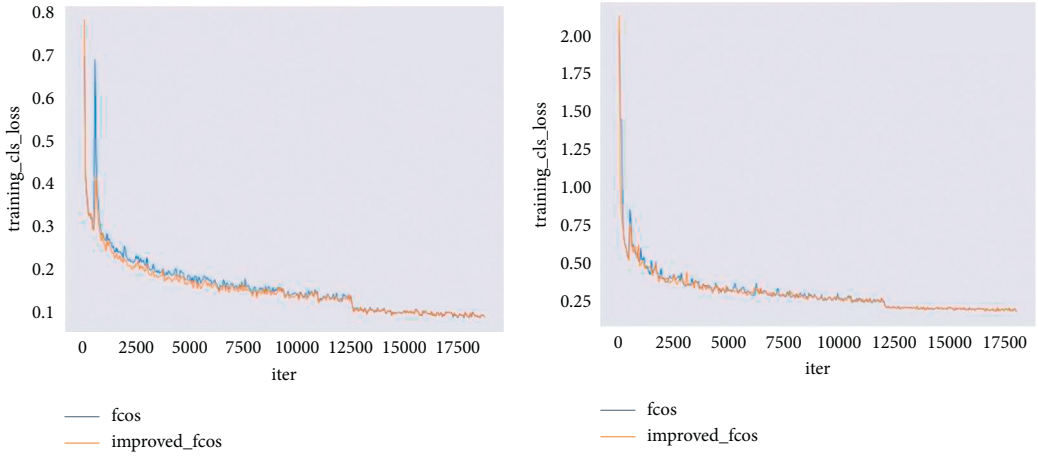
Model	Input shape	GFLOPs	Parameters (M)
Faster-RCNN [13]	1280 × 800	206.67	41.13
ATSS [23]	1280 × 800	201.51	31.89
GFL [37]	1280 × 800	204.61	32.04
FCOS [22]	1280 × 800	196.76	31.84
YOLOF [38]	1280 × 800	98.21	42.11
YOLOV3 [16]	1280 × 800	193.89	61.53
SSD [39]	1280 × 800	343.77	24.68
DETR [29]	1280 × 800	101.34	20.09
Centernet [40]	1280 × 800	51.02	14.21
Retinanet [9]	1280 × 800	205.24	36.15
Ours	1280 × 800	174.39	35.1

Table 4: Ablation study for the proposed methods.

Method	mAP
FCOS [22]	66.8
+Improved FPN	67.2
+DCN [8]	68.4
+Balanced module	67.9
+All	**70.1**

(a) (b)

Figure 7: Ablation experiment example. (a) Figure without DCN, where its network attention is relatively distracted, and the sampling area cannot cover the overall target. (b) After adding DCN, the effect figure shows that the DCN sampling area is more comprehensive and can learn more features.



(a) (b)

Figure 8: The linechart of cls and bbox loss, where the earlier loss_cls converges faster and the fluctuation is minor.

is very different. The experimental results are shown in Table 4. It is evident that the combined performance of the three improvement methods is the best. Figure 8 shows the linechart of cls and bbox loss. Compared with FCOS, our method converges faster and more stably on cls.

4.5. Visualization of Results. Figure 9 is a visualization of the effect of the algorithm. It can be seen that vehicles can be detected no matter whether they are at a distance or in some unique scenes, even when there is little picture information, which proves the superiority of our algorithm.

(a) (b) (c) (d)

Figure 9: Examples of visualizations. (a, c) The original diagram contains the vehicle at various angles, far and near perspective, and even includes the reflective glass vehicle. (b, d) The result of the visualization of the improved model.

5. Conclusions

We improved the detection algorithm based on the anchor-free FCOS [22] and introduced the DCN [8] based on the original backbone to broaden the receptive field of the convolution kernel and also introduced a bottom-up module to improve the FPN and reduce the loss between the information transmission. A balance module is added because the variance of the feature pyramid affects the accuracy, which has a good effect and proves the superiority of the improved algorithm.

Data Availability

The codes used in this paper are available from the corresponding author upon request (fyan@nuist.edu.cn).

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (No. 61605083) and Jiangsu Provincial Key Research and Development Program (China).

References

[1] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3431–3440, Boston, MA, USA, June 2015.

[2] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 770–778, Las Vegas, NV, USA, June 2016.

[3] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4700–4708, Honolulu, HI, USA, July 2017.

[4] C. Szegedy, W. Liu, Y. Jia et al., "Going deeper with convolutions," in *Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1–9, Boston, MA, USA, June 2015.

[5] H. Zhu, X. Li, P. Zhang et al., "Learning tree-based deep model for recommender systems," in *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 1079–1088, London, UK, August 2018.

[6] T. Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2117–2125, Honolulu, HI, USA, July 2017.

[7] W. Wang, E. Xie, X. Song et al., "Efficient and accurate arbitrary-shaped text detection with pixel aggregation network," in *Proceedings of the 2019 IEEE/CVF International Conference*

on Computer Vision, pp. 8440–8449, Seoul, South Korea, October 2019.

[8] J. Dai, H. Qi, Y. Xiong et al., "Deformable convolutional networks," in *Proceedings of the 2017 IEEE International Conference on Computer Vision*, pp. 764–773, Venice, Italy, October 2017.

[9] J. Pang, K. Chen, J. Shi, H. Feng, W. Ouyang, and D. Lin, "Libra r-CNN: towards balanced learning for object detection," in *Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 821–830, Long Beach, CA, USA, June 2019.

[10] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, pp. 886–893, IEEE, San Diego, CA, USA, June 2005.

[11] Z. Guo, L. Zhang, and D. Zhang, "A completed modeling of local binary pattern operator for texture classification," *IEEE Transactions on Image Processing*, vol. 19, no. 6, pp. 1657–1663, 2010.

[12] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, IEEE,Kauai, HI, USA, December 2001.

[13] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: towards real-time object detection with region proposal networks," *Advances in Neural Information Processing Systems*, vol. 28, pp. 91–99, 2015.

[14] Z. Cai and N. Vasconcelos, "Cascade R-Cnn: delving into high quality object detection," in *Proceedings of the 2018 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 6154–6162, Salt Lake City, UT, USA, June 2018.

[15] J. Redmon and A. Farhadi, "YOLO9000: better, faster, stronger," in *Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 7263–7271, Honolulu, HI, USA, July 2017.

[16] J. Redmon and A. Farhadi, "Yolov3: an incremental improvement," 2018, http://arxiv.org/abs/1804.02767.

[17] J. Wang, K. Chen, S. Yang, C. Change Loy, and D. Lin, "Region proposal by guided anchoring," in *Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2965–2974, Long Beach, CA, USA, June 2019.

[18] Y. Cao, J. Xu, S. Lin, F. Wei, and H. Hu, "Gcnet: non-local networks meet squeeze-excitation networks and beyond," in *Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision Workshops*, Long Beach, CA, USA, June 2019.

[19] C. Zhu, Y. He, and M. Savvides, "Feature selective anchor-free module for single-shot object detection," in *Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 840–849, Long Beach, CA, USA, June 2019.

[20] X. Lu, B. Li, Y. Yue, Q. Li, and J. Yan, "Grid R-CNN," in *Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7363–7372, Long Beach, CA, USA, June 2019.

[21] K. Hu, Y. Zhang, C. Weng, P. Wang, Z. Deng, and Y. Liu, "An underwater image enhancement algorithm based on generative adversarial network and natural image quality evaluation index," *Journal of Marine Science and Engineering*, vol. 9, no. 7, p. 691, 2021.

[22] Z. Tian, C. Shen, H. Chen, and H. Tong, "Fcos: a simple and strong anchor-free object detector," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020.

[23] S. Zhang, C. Chi, Y. Yao, Z. Lei, and S. Z. Li, "Bridging the gap between anchor-based and anchor-free detection via adaptive training sample selection," in *Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9759–9768, Seattle, WA, USA, June 2020.

[24] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: unified, real-time object detection," in *Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 779–788, Las Vegas, NV, USA, June 2016.

[25] S. Zhang, L. Wen, X. Bian, Z. Lei, and S. Z. Li, "Single-shot refinement neural network for object detection," in *Proceedings of the 2018 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4203–4212, Salt Lake City, UT, USA, June 2018.

[26] T. Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *Proceedings of the 2017 IEEE International Conference on Computer Vision*, pp. 2980–2988, Venice, Italy, October 2017.

[27] B. Zhu, J. Wang, Z. Jiang et al., "Autoassign: differentiable label assignment for dense object detection," 2020, http://arxiv.org/abs/2007.03496.

[28] A. Vaswani, N. Shazeer, N. Parmar et al., "Attention is all you need," in *Proceedings of the 31st Conference on Neural Information Processing Systems (NIPS 2017)*, pp. 5998–6008, Long Beach, CA, USA, 2017.

[29] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, "End-to-End object detection with transformers," in *Proceedings of the Computer Vision-ECCV 2020*, pp. 213–229, Springer, Glasgow, UK, August 2020.

[30] X. Zhu, W. Su, L. Lu, B. Li, X. Wang, and J. Dang, "Deformable detr: deformable transformers for end-to-end object detection," 2020, http://arxiv.org/abs/2010.04159.

[31] A. Dosovitskiy, L. Beyer, A. Kolesnikov et al., "An image is worth 16 x 16 words: transformers for image recognition at scale," 2020, http://arxiv.org/abs/2010.11929.

[32] Y. Qu, M. Xia, and Y. Zhang, "Strip pooling channel spatial attention network for the segmentation of cloud and cloud shadow," *Computers & Geosciences*, vol. 157, Article ID 104940, 2021.

[33] M. Xia, X. Zhang, W. A. Liu, L. Weng, and Y. Xu, "Multi-stage feature constraints learning for age estimation," *IEEE Transactions on Information Forensics and Security*, vol. 15, no. 1, pp. 2417–2428, 2020.

[34] Z. Zhang and M. R. Sabuncu, "Generalized cross entropy loss for training deep neural networks with noisy labels," in *Proceedings of the 32nd Conference on Neural Information Processing Systems (NeurIPS)*, Montreal, Canada, December 2018.

[35] G. Brazil and X. Liu, "M3d-rpn: monocular 3d region proposal network for object detection," in *Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision*, pp. 9287–9296, Seoul, South Korea, October 2019.

[36] K. Chen, J. Wang, J. Pang et al., "MMDetection: open mmlab detection toolbox and benchmark," 2019, http://arxiv.org/abs/1906.07155.

[37] X. Li, W. Wang, L. Wu et al., "Generalized focal loss: learning qualified and distributed bounding boxes for dense object detection," 2020, http://arxiv.org/abs/2006.04388.

[38] Q. Chen, Y. Wang, T. Yang, X. Zhang, J. Chen, and J. Sun, "You only look one-level feature," in *Proceedings of the 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 13039–13048, Nashville, TN, USA, June 2021.

[39] W. Liu, D. Anguelov, D. Erhan et al., "SSD: single shot multibox detector," in *Proceedings of the Computer Vision-ECCV 2016*, pp. 21–37, Amsterdam, The Netherlands, October 2016.

[40] X. Zhou, D. Wang, and P. Krähenbühl, "Objects as points," 2019, http://arxiv.org/abs/1904.07850.