

基于改进YOLOv4的低慢小无人机实时探测算法

吴璇 张海洋 赵长明 李志朋 王元泽

Improved YOLOv4 for real-time detection algorithm of low-slow-small unmanned aerial vehicles

WU Xuan, ZHANG Haiyang, ZHAO Changming, LI Zhipeng, WANG Yuanze

引用本文:

吴璇, 张海洋, 赵长明, 等. 基于改进YOLOv4的低慢小无人机实时探测算法[J]. 应用光学, 2024, 45(1): 79–88. DOI: 10.5768/JAO202445.0102002

WU Xuan, ZHANG Haiyang, ZHAO Changming, et al. Improved YOLOv4 for real-time detection algorithm of low-slow-small unmanned aerial vehicles[J]. Journal of Applied Optics, 2024, 45(1): 79–88. DOI: 10.5768/JAO202445.0102002

在线阅读 View online: <https://doi.org/10.5768/JAO202445.0102002>

您可能感兴趣的其他文章

Articles you may be interested in

多尺度融合超分辨率算法在无人机探测中的应用

Application of multi-scale fusion super-resolution algorithm in UAV detection

应用光学. 2021, 42(3): 462–473 <https://doi.org/10.5768/JAO202142.0302003>

激光供能无人机的一种优化跟踪算法

Optimal tracking algorithm for laser powered unmanned aerial vehicles

应用光学. 2020, 41(1): 194–201 <https://doi.org/10.5768/JAO202041.0107002>

无人机载多光谱侦察效能研究

Study on performance of multi spectral reconnaissance of unmanned aerial vehicle

应用光学. 2017, 38(5): 685–688 <https://doi.org/10.5768/JAO201738.0501001>

基于结构光的植保无人机障碍物在线检测系统

Obstacle detection system of plant protection UAVs based on structural light

应用光学. 2018, 39(3): 343–348 <https://doi.org/10.5768/JAO201839.0301008>

基于嵌入式GPU的红外弱小目标检测算法

Infrared weak small target detection algorithm based on embedded GPU

应用光学. 2020, 41(5): 1089–1095 <https://doi.org/10.5768/JAO202041.0506004>

基于改进SSD的车辆小目标检测方法

Detecting method of small vehicle targets based on improved SSD

应用光学. 2020, 41(1): 150–155 <https://doi.org/10.5768/JAO202041.0103004>



关注微信公众号, 获得更多资讯信息

文章编号: 1002-2082 (2024) 01-0079-10

基于改进 YOLOv4 的低慢小无人机实时探测算法

吴璇, 张海洋, 赵长明, 李志朋, 王元泽

(北京理工大学 光电学院, 北京 100081)

摘要: 针对低慢小无人机探测任务中精度不高、在嵌入式平台上部署实时性能差的问题, 提出了一种基于改进 YOLOv4 的小型无人机目标检测算法。通过增加浅层特征图、改进锚框、增强小目标, 提高网络对小目标的检测性能, 通过稀疏训练和模型修剪, 大大缩短了模型运行时间。在 1080Ti 上平均精度 (mAP) 达到 85.8%, 帧率 (FPS) 达 75 frame/s, 实现了网络轻量化。该模型部署在 Xavier 边缘计算平台上, 可实现 60 frame/s 的无人机目标检测速度。实验结果表明: 与 YOLOv4 和 YOLOv4-tiny 相比, 该算法实现了运行速度和检测精度的平衡, 能够有效解决嵌入式平台上的无人机目标检测问题。

关键词: 低慢小无人机; 目标检测; YOLOv4; 剪枝; 嵌入式

中图分类号: TN201

文献标志码: A

DOI: 10.5768/JAO202445.0102002

Improved YOLOv4 for real-time detection algorithm of low-slow-small unmanned aerial vehicles

WU Xuan, ZHANG Haiyang, ZHAO Changming, LI Zhipeng, WANG Yuanze

(School of Optics and Photonics, Beijing Institute of Technology, Beijing 100081, China)

Abstract: In order to solve the low accuracy in low-slow-small unmanned aerial vehicles (UAVs) mission on embedded platform and deployment problem of poor real-time performance, a small UAV target detection algorithm based on improved YOLOv4 was proposed. By increasing the shallow characteristic figure, improving the anchor, enhancing the small target, and the detection performance of network for small target was improved, through sparse training and model pruning, the model running time was greatly reduced. The average accuracy (mAP) reaches 85.8% on the 1080Ti, and the frame rate (FPS) reaches 75 frame/s, which achieving network lightweight. This lightweight model was deployed on the Xavier edge computing platform, which could achieve the UAV target detection speed of 60 frame/s. Experimental results show that, in compared with YOLOv4 and YOLOv4-TINY, this algorithm achieves the balance of running speed and detection accuracy, and can effectively solve the problem of UAV target detection on embedded platform.

Key words: low-slow-small unmanned aerial vehicles; target detection; YOLOv4; pruning; embedded

Introduction

With the rapid development of aviation technology and the upgrade of communication technology, the unmanned aerial vehicles (UAVs) have been widely used in fire fighting^[1], agricultural monitoring^[2] and other fields. UAV has low flying altitude, uncertain

flight trajectory and high flexibility^[3], which will pose a threat to public security and privacy when used by criminals. It is necessary to take countermeasures against UAVs, and UAV target detection is the key to interfere and strike them.

Common UAV target detection methods include

收稿日期: 2022-12-30; 修回日期: 2023-03-06

基金项目: 冬季项目场景三维感知及重建技术 (2018YFF0300802)

作者简介: 吴璇 (1999—), 女, 硕士研究生, 主要从事目标检测与识别研究。E-mail: 3120200583@bit.edu.cn

通信作者: 张海洋 (1981—), 男, 博士, 副教授, 主要从事光电成像、目标智能感知研究。E-mail: ocean@bit.edu.cn

classical moving target detection based on optical flow method and frame difference method^[4]. Since AlexNet network was proposed, deep learning has been gradually applied to object detection^[5]. Although the detection accuracy of two-stage algorithms such as R-CNN(region with convolutional neural network feature)^[6], Fast R-CNN^[7], Faster R-CNN^[8], SPP-Net(spatial pyramid pooling network)^[9] is significantly improved compared with traditional algorithms, they cannot meet the real-time requirements of engineering. In one-stage algorithm, SSD(singleshot multibox detector) algorithm^[10] adopts multi-scale feature map combined with anchor mechanism to improve the detection accuracy as much as possible while ensuring the speed. For small target detection, WANG Ruoxiao et al. reduced the channels of VGG16 to meet the real-time detection of UAV on the embedded platform^[11]. LIN T Y et al. proposed RetinaNet, which uses focal loss to overcome the class imbalance problem caused by high foreground to background ratio^[12]. RAZA M A et al. proposed BV-RNet, which can effectively detect small scale targets by extracting dense features and optimizing predefined anchor points^[13]. SUN Han et al. proposed a lightweight detection network for UAVs: TIB-Net^[14]. In view of the lack of texture and shape features of infrared UAVs, DING Lianghui et al. enhanced the high-resolution network layer and adopted the adaptive pipeline filter (APF) based on temporal correlation and motion information to correct the results^[15]. FANG H et al. transformed the infrared small UAV target detection into nonlinear mapping from infrared image space to residual image and got better detection performance in complex background^[16]. YOLO(you only look once) algorithm uses whole-process convolution for target discrimination and candidate box prediction^[17], which has high detection accuracy and fast detection speed. HU Y et al. used feature maps of 4 scales to predict bounding boxes in YOLOv3 to obtain more texture and contour information, the mAP was increased by about 4.16%^[18]. LI Zhipeng et al. used the super-resolution algorithm to reconstruct high-resolution UAV images, and used YOLOv3 to realize the effective detection of low-slow-small UAVs^[19].

The lack of semantic information in small UAV target imaging will reduce the detection accuracy, and the memory and computing power of the embedded platform are limited, which cannot meet the real-time requirements of UAV detection tasks, there is a lack of high-precision real-time target detection algorithms for small UAV. Aiming at the above problems, this paper improves the mAP(mean average precision) by 6.2% and the FPS(frame per second) by 22 frame/s on the basis of YOLO4 through model improvement and pruning, and achieves 85.6% mAP and nearly 60 frame/s detection performance with half-precision deployment on the embedded platform. Experiments have verified the effectiveness of this method for high-precision real-time detection of low-slow-small UAV targets.

1 Algorithm design for low-slow-small UAV target detection

YOLOv4 algorithm was proposed in 2020. Compared with YOLOv3, it has been optimized in backbone network, multi-scale fusion, activation function, loss function and other aspects^[20], its structure is shown in Figure 1. The backbone network part refers to the idea of jump connection of CSPNet^[21], and forms CSPDarkNet53 on the basis of DarkNet53(as shown in the residual part in Fig. 1), which enhances the network feature extraction ability and speeds up the network training speed. The neck part uses the SPP structure (see SPP structure diagram in Fig. 1) to improve the size of the receptive field, and then PANet is used to achieve the fusion of feature maps of different scales and sizes. Through repeated feature extraction, the feature extraction capability of network for objects of different sizes is effectively enhanced. In the position loss function, CIOU(complete intersection over union) is used to comprehensively evaluate the overlap area, aspect ratio, distance of the center position and other factors between the ground truth box and the predicted box. The Mish activation function is used to avoid gradient saturation.

Since YOLOv4 performs well in the field of traditional target detection and has made some optimization for small target detection, this paper improved the YOLOv4 algorithm according to the

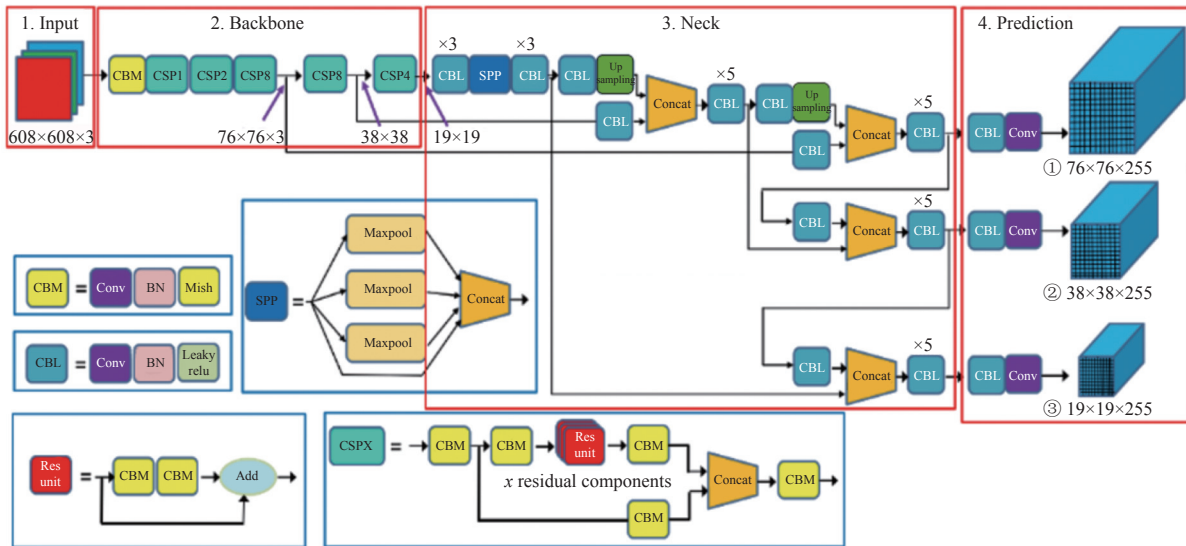


Fig. 1 Network structure diagram of YOLOv4

characteristics of low-slow-small UAV targets.

1.1 Improvements to YOLOv4

There are still some problems in YOLOv4 algorithm for the detection of low-slow-small UAV targets: the feature maps extracted by YOLO4 have fewer small target features; deep feature extraction network makes UAV features easy to be lost; the generalization ability of anchor adopted in YOLOv4 algorithm for small targets is weak^[22]. This paper improves YOLOv4 from the aspects of network

structure, small target enhancement and candidate box adjustment.

1.1.1 Network structure improvement

As shown in Figure 2, this work improves the feature fusion part of YOLOv4 by up-sampling the shallow feature map and splicing it with the shallow UAV feature image, adding the output branch with a scale of 104×104 pixel. Figure 3 shows the feature maps output from neck and head of the improved YOLOv4. More details of UAV are obtained in the

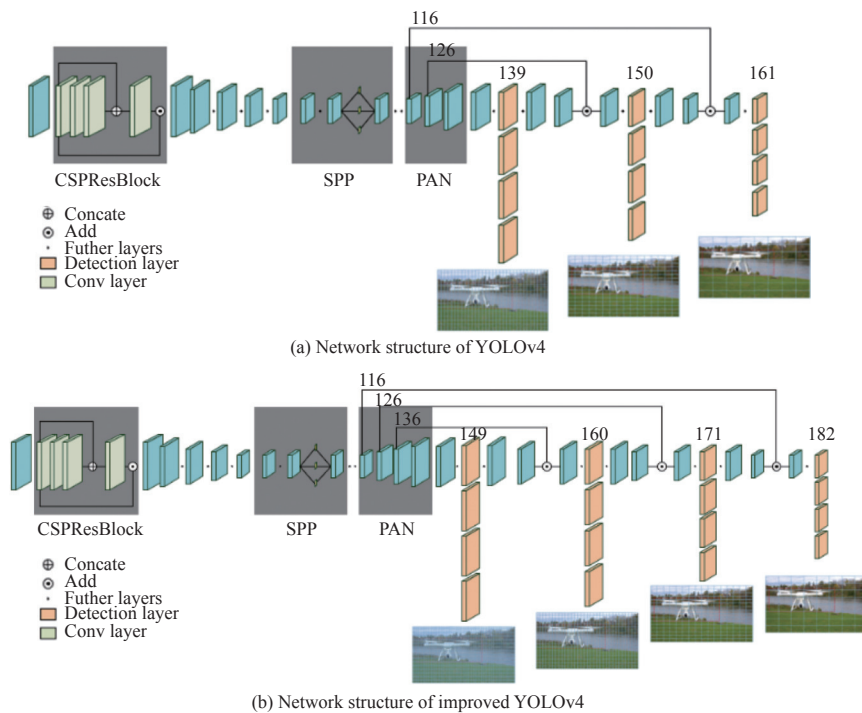


Fig. 2 Comparison of YOLOv4 network structure before and after improvement

newly added scale, which is conducive to the improvement of UAV detection accuracy. The improved network makes full use of the low-level and high-level information, and achieves the detection of small object scale through the new detection layer.



(a) Partial feature maps output from neck module



(b) Four scale feature maps output from head module

Fig. 3 Feature maps of improved YOLOv4

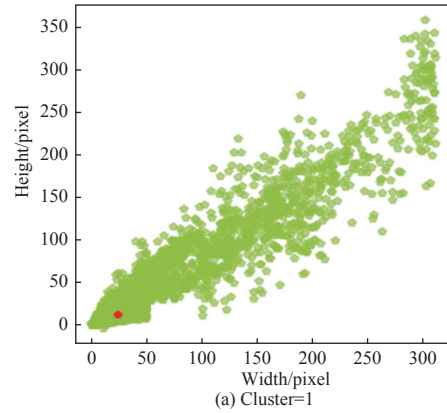
1.1.2 Adjustment of anchor boxes

YOLOv4 adopts k -means clustering, k is the number of clusters, the higher the value of k , the better the quality of the preset anchor box, which is conducive to the convergence of the model in the training process^[23]. YOLOv4 allocates 3 anchor boxes to each scale, and gets 9 anchor boxes in total. k -Means randomly selects k initial cluster centers, which can greatly affect the results when not initialized properly.

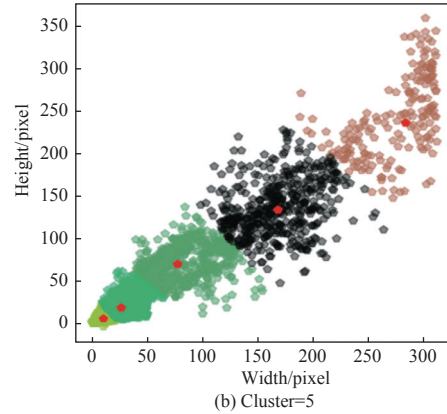
The improved YOLOv4 adopts k -means ++ to cluster UAV samples, k -means ++ randomly selects a cluster center and calculates the distance with other samples. The sample with larger distance is more likely to become the next cluster center, until k cluster centers are obtained. Euclidean distance is used to measure the distance between the sample and the cluster center, and the objective function of clustering is expressed as follows:

$$f = \min \sum_{i=1}^k \sum_{x \in k_i} \text{dist}(c_i, x)^2 \quad (1)$$

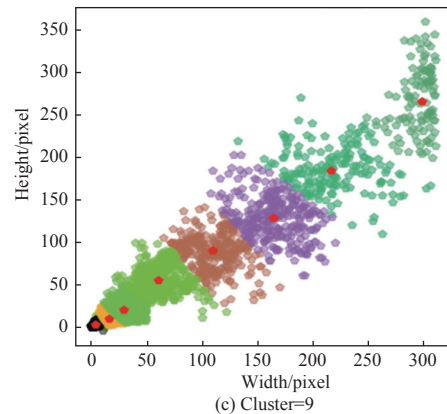
where k is the number of clusters, k_i is the i_{th} cluster, and $\text{dist}(c_i, x)^2$ is the squared distance from sample x to the i_{th} cluster center c_i . For the improved YOLOv4, resize the image to 416×416 pixel, 3 anchor boxes are assigned to the feature maps of each scale, resulting in a total of 12 anchor boxes. The clustering process for anchor boxes is shown in Fig. 4. The clustering results are shown in Table 1. The k -means ++ makes the anchor frame of clustering pay more attention to small targets, and the clustering result is more consistent with the real label.



(a) Cluster=1



(b) Cluster=5



(c) Cluster=9

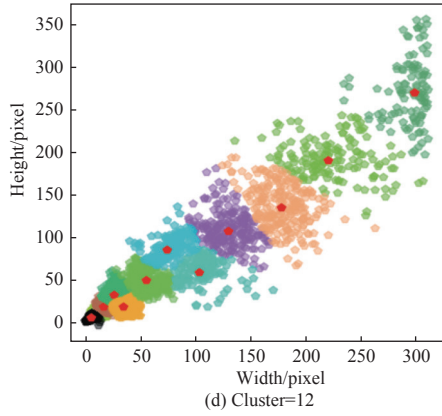


Fig. 4 Process of obtaining anchor boxes by k-means++ clustering

Table 1 Clustering results of different clustering methods in training set

Algorithm	Layer 1	Layer 2	Layer 3	Layer 4
YOLOv4	(282,242)	(73,58)	(15,24)	
	(203,160)	(38,32)	(11,15)	
	(110,95)	(21,16)	(8,11)	
Improved YOLOv4 (k-means)	(331,301)	(227,250)	(99,86)	(20,24)
	(292,238)	(195,159)	(74,57)	(12,17)
	(252,186)	(144,118)	(40,36)	(8,11)
Improved YOLOv4 (k-means++)	(298,269)	(130,108)	(55,48)	(18,20)
	(220,190)	(103,60)	(35,19)	(12,17)
	(178,135)	(74,85)	(26,31)	(8,11)

1.1.3 Data augmentation for small UAV targets

The mosaic data enhancement used in YOLOv4 will randomly scale the target, possibly resulting in serious loss of drone target information. This paper adopts the method of copying multiple UAVs into one image to increase the number of UAVs (as shown in Figure 5), so that the model pays more attention to small UAVs and improves the contribution of small UAVs to the loss function^[24].



Fig. 5 UAV data augmentation

1.2 Model pruning of improved YOLOv4 algorithm

Network pruning reduces network parameters and computational complexity by removing a large number of unimportant channels to improve inference speed, its

general process includes sparse training, network pruning, and model fine-tuning^[25].

1.2.1 Sparse training

The scale factor γ of the batch normalization (BN) layer is used as the index to evaluate the importance of the channel, and $L1$ regularization is used to train γ , the loss function is expressed as:

$$L(\gamma) = l(\gamma)_{\text{YOLOv4}} + \rho \|\gamma\|_1 \quad (2)$$

where $L(\gamma)$ is the total loss function, $l(\gamma)_{\text{YOLOv4}}$ is the loss function of YOLOv4, $\|\gamma\|_1$ is the penalty term, and ρ represents the parameter factor of $L1$ norm.

1.2.2 Network pruning

Channel pruning is carried out according to the sparsely trained γ value, the channel corresponding to a small γ value has a small contribution to the network inference results. Sort the value of s and set the pruning rate to remove unimportant channels in the network. The channel pruning of shortcut structure refers to the practice of SlimYOLOv3^[26], as shown in Figure 4. Assuming that layer A retains channels 1 and 2, layer C retains channels 2 and 3, and layer F retains channels 2 and 4, layer A, C, D, F and G retains channels 1, 2, 3 and 4.

Layer pruning is based on the γ value of the convolution module before the shortcut layer. The two convolution modules before the shortcut layer are pruned together with it. As shown in the red box of Fig. 6, when layer D is cut, layer B and layer C are also cut.

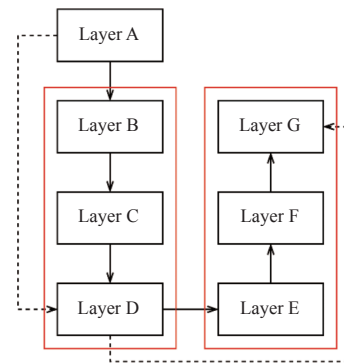


Fig. 6 Structure diagram of shortcut layer

2 Experiments

2.1 Experimental setting

A large number of UAV images (with the size of 1 920×1 080 pixel) collected by the camera were com-

binned with the UAV Dataset (Drone Dataset, Drone-data-2021) to form an experimental dataset containing 20 000 UAV images, of which 80% were used as the training set and the rest were used as the testing set.

The comparison experiments of model improvement and pruning were carried out on windows10 operating system, equipped with i7-7700 processor and NVIDIA GeForce GTX 1080Ti. The network was implemented by Pytorch1.6-GPU. The input image was resized to 416×416 pixel, batch size was set to 8, initial learning rate was set to 0.002324, and Adam optimization strategy was used. The network was trained using a fine-tuning approach to reduce training time, first on the COCO dataset and then on the UAV training set. Finally, the embedded computing performance was verified on a Jetson AGX Xavier (16GB).

2.2 Evaluation index

In object detection, mAP and FPS are commonly used for evaluation, where FPS represents the model inference speed, and mAP needs to be calculated by confusion matrix (see Table 2).

Table 2 Confusion matrix

Annotated results	Predicted result	
	True	False
True	T_P	F_N
False	F_P	T_N

The average precision (AP) is the area enclosed by the PR curve plotted with precision (P) and recall (R). See Formula (3) and Formula (4) for the calculation of accuracy and recall, and AP is calculated by Formula (5):

$$P = \frac{T_P}{T_P + F_P} \quad (3)$$

$$R = \frac{T_P}{F_N + T_P} \quad (4)$$

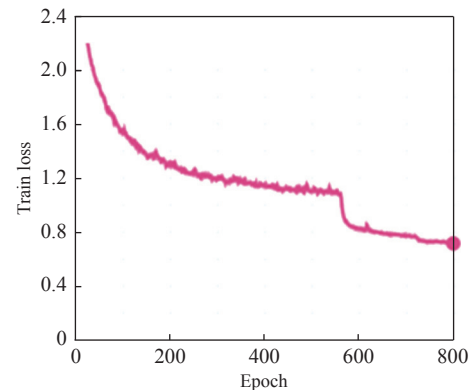
$$P_{AP} = \sum_{i=1}^n P(i) \Delta R(i) \quad (5)$$

mAP is the average accuracy of all categories, which can be calculated by Formula (6):

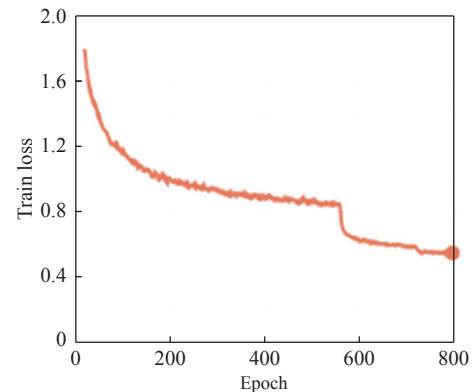
$$P_{mAP} = \frac{\sum C P_{AP}^c}{C} \quad (6)$$

2.3 Performance comparison before and after algorithm improvement

The IoU(intersection over union) threshold is set to 0.5 to test the algorithm before and after improvement. Fig. 7 shows the loss curve, the improved YOLOv4 has better convergence effect on the UAV data set, and the loss is reduced to below 0.6 after training. The PR curve plotted against recall and precision is shown in Fig. 8, where the curve closer to the top right corner indicates better detection performance. The PR curve of the improved YOLOv4 completely enveloped the curve of the original YOLOv4, proving its stronger detection ability. Fig. 9 shows the detection results for low-slow-small UAVs. Compared with the original YOLOv4, the improved YOLOv4 adds a small UAV target prediction branch and adjusts the candidate box, which can reduce missed detection and false detection, improve the prediction accuracy of the size and position of the bounding box.



(a) Loss curve of YOLOv4



(b) Loss curve of improved YOLOv4

Fig. 7 Comparison of loss curves during training of YOLOv4 and improved YOLOv4

The comparison results of mAP and FPS of different algorithms are shown in Table 3 and Fig.10.

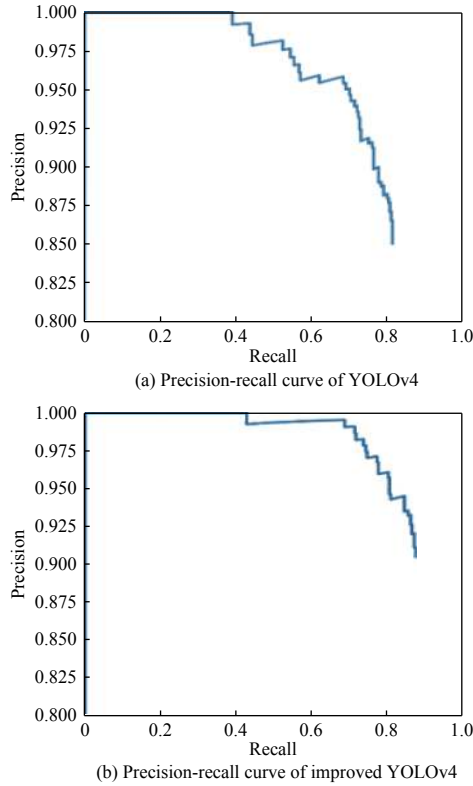


Fig. 8 Comparison of precision-recall curves of YOLOv4 and improved YOLOv4

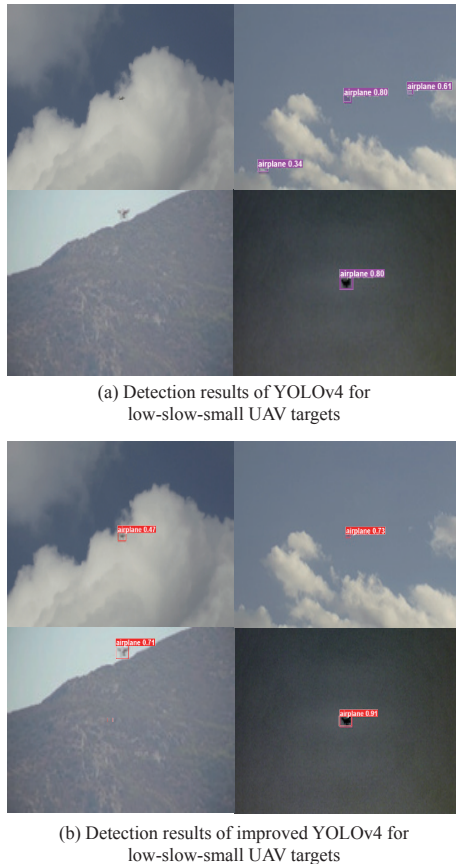


Fig. 9 Comparison of detection results of YOLOv4 and improved YOLOv4 on UAV dataset

Table 3 Detection effects of different algorithms

Algorithm	Faster R-CNN	SSD	YOLOv3	YOLOv4	Improved YOLOv4
mAP/%	74.5	76.1	76.5	79.6	86.7
FPS	21.70	28.93	43.57	53.11	49.26

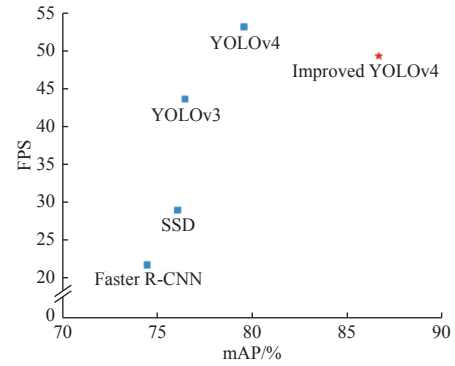


Fig. 10 Detection effects of different algorithms

The mAP of the improved YOLOv4 is 7.1% higher than that of the original YOLOv4, the FPS of the improved algorithm is 49 on NVIDIA GeForce GTX 1080Ti, which is slightly lower than the original YOLOv4. The experiment proves that the improved YOLOv4 algorithm performs better in mAP and FPS than YOLOv3, SSD and other algorithms, it achieves a good balance in FPS and mAP, and has a stronger detection ability for small UAV targets.

2.4 Comparison of model performance before and after pruning

Sparse training was performed by local sparsity rate decay, setting sparsity rate to 0.001 and epoch to 300. In the last 50% training phase, 85% of the network channels are sparsified normally, and the remaining 15% channels are sparsified with a sparsity rate of 1% to prevent the model accuracy from decreasing sharply.

The weights of BN(batch normalization) layers before and after sparse training are shown in Fig.11. During the training process, the weight distribution gradually approaches 0, and channels with weights close to 0 will be pruned preferentially. The pruning rate is set to 0.8, the number of pruning layers is 16, and a total of 27 137 channels and 48 layers are pruned. The pruning result is shown in Fig.12. The weights of the middle part of the network are closer to 0, so most channels are pruned, and the clipped layers are the layers with fewer remaining channels after channel

pruning. Table 4 shows the network performance before and after pruning. After pruning, the number of model parameters is reduced to 7.5% of the original, the model size is reduced from 248 MB to 18.5 MB, and the detection speed is improved from 49 frame/s to nearly 75 frame/s (1080Ti). Finally, we recover the mAP to 85.8% with 100 epochs of model fine-tuning.

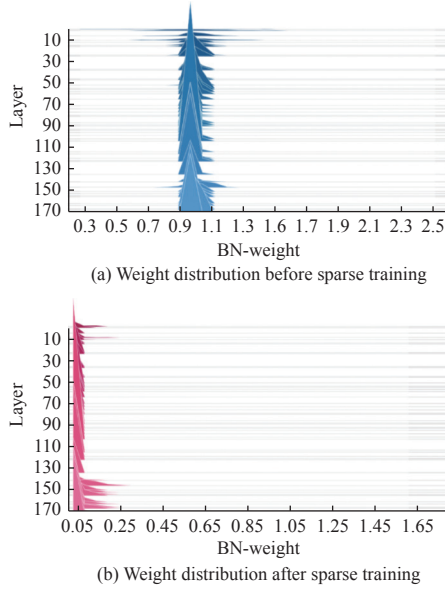


Fig. 11 Weight distribution of BN layers in sparse training

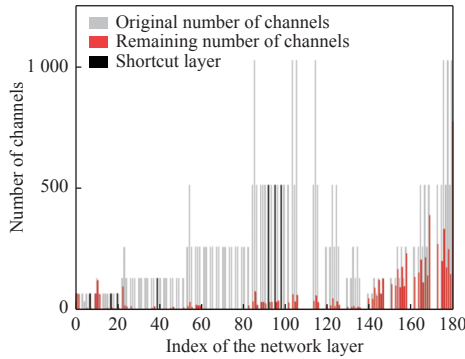


Fig. 12 Comparison of network layers and channels in improved YOLOv4 before and after pruning

Table 4 Comparison of network pruning results with different parameters

Channel pruning rate	Number of pruning layers	mAP/%	Number of parameters	Inference time /s
0	0	86.2495	64967464	0.0203
0.8	16	80.7483	4854054	0.0132

Figure 13 shows the comparison between the pruned model and YOLOv4-tiny. Although the detection speed of the lightweight model based on the

improved YOLOv4 is slightly lower than that of YOLOv4-tiny, it has greater advantages than YOLOv4-tiny in terms of precision, recall and mAP. Experiments show that the network is suitable for low-slow-small UAV detection.

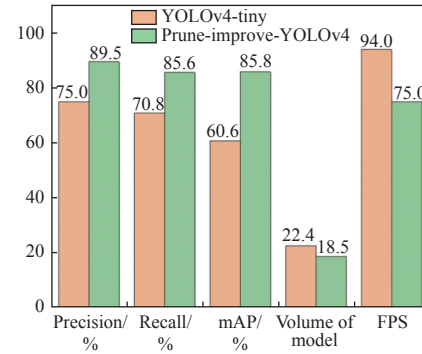


Fig. 13 Performance comparison of prune-improve-YOLOv4 and YOLOv4-tiny

2.5 Deployment experiment on embedded platform

The data of the model is stored in the form of 32-bit double floating point precision (FP32), which will occupy a large memory space and increase the inference time. Using low precision data (FP16, INT8) for inference can reduce the storage space and bandwidth required for calculation and improve the inference speed. In this experiment, the prune-improve-YOLOv4 is accelerated and quantized by TensorRT on the Xavier platform. The Xavier running power is 30 W, the test image size is $3 \times 416 \times 416$ pixel, the mAP and inference time under different precision are shown in Table 5.

Table 5 Inference performance of prune-improve-YOLOv4 deployed with different precision on Xavier

Precision	Torch	Trt FP32	Trt FP16	Trt INT8
mAP	85.8	85.8	85.8	68.4
Inference time/ms	58.50	27.59	16.83	11.46

The mAP shows almost no drop for half-precision (FP16) inference, but shows a large drop for INT8 inference. In this paper, semi-precision (FP16) deployment is adopted to achieve 85.8% mAP and a detection speed of nearly 60 frame/s on Xavier, the detected image is shown in Figure 14. Experiments show that the prune-improve-YOLOv4 can meet the

real-time and high-precision requirements of detecting low-slow-small UAV targets using embedded platform.



Fig. 14 Detection result of prune-improve-YOLOv4 deployed on Xavier platform

3 Conclusion

Aiming at the difficulties of small UAV target semantic information is less and the target is not obvious, this paper firstly improves the original YOLOv4 network from three aspects: algorithm network structure, small target enhancement and anchor box allocation. The improved YOLOv4 network can detect most UAVs and the position of the boundary box is more accurate. Recall, accuracy and mAP have been greatly improved. Then, by setting the channel pruning rate of 0.8 and the layer pruning number of 16, the network parameters and inference time are greatly reduced. Finally, 85.8% mAP and nearly 60 FPS real-time detection are achieved on Xavier. Experimental results show that the proposed algorithm achieves higher accuracy and speed than YOLOv4 in small target detection, and can be used for embedded equipment to carry out real-time detection of low-slow-small UAV.

参考文献:

- [1] NIHEI K, KAI N, MARUYAMA Y, et al. Forest fire surveillance using live video streaming from UAV via multiple LTE networks[C]//2022 IEEE 19th Annual Consumer Communications & Networking Conference (CCNC). New York: IEEE, 2022: 465-468.
- [2] ZHANG H, WANG L, TIAN T, et al. A review of unmanned aerial vehicle low-altitude remote sensing (UAV-LARS) use in agricultural monitoring in China[J]. *Remote Sensing*, 2021, 13(6): 1221.
- [3] WANG Z G, KANG Q, XUN Y J, et al. The security threats of low-slow and small drones to military areas and their countermeasures[J]. *Defense Technology*, 2021, 42(3): 65-71.
- [4] WANG S, SU T, LAI S. Detecting moving objects from dynamic background with shadow removal[C]//2011 IEEE International Conference on Acoustics, Speech and Signal Processing. New York: IEEE, 2011: 925-928.
- [5] KRIZHEVSKY A, SUTSKEVER I, HINTON G. ImageNet classification with deep convolutional neural networks[C]. New York: ACM, 2017: 84-90.
- [6] GIRSHICK R, DONAHUE J, DARRELL T, et al. Rich feature hierarchies for accurate object detection and semantic segmentation[C]//2014 IEEE Conference on Computer Vision and Pattern Recognition. New York: IEEE, 2014: 580-587.
- [7] GIRSHICK R. Fast R-CNN[C]//Proceedings of the IEEE International Conference on Computer Vision. New York: IEEE, 2015: 1440-1448.
- [8] REN S, HE K, GIRSHICK R, et al. Faster R-CNN: towards real-time object detection with region proposal networks[J]. *Advances in Neural Information Processing Systems*, 2015, 28: 1137-1149.
- [9] HE K M, ZHANG X Y, REN S Q, et al. Spatial pyramid pooling in deep convolutional networks for visual recognition[J]. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2017, 39(6): 1137-1149.
- [10] LIU W, ANGUELOV DR, ERHAN D, et al. SSD: single shot multibox detector[C]//European conference on computer vision. Cham: Springer International Publishing, 2016: 21-37.
- [11] WANG R X, XU Z Y, ZHANG J L. Real-time lightweight UAV detection method based on SSD algorithm[J]. *Semiconductor Optoelectronics*, 2020, 41(2): 296-307.
- [12] LIN T Y, GOYAL P, GIRSHICK R, et al. Focal loss for dense object detection[J]. *IEEE Transactions on Pattern Analysis Machine Intelligence*, 2020, 42(2): 318-327.
- [13] RAZA M A, BINT-E-NAEEM H, YASIN A, et al. Bird-View Retina-Net: small-scale object detector for unmanned aerial vehicles[C]//2021 16th International Conference on Emerging Technologies (ICET). New York: IEEE, 2021: 1-6.
- [14] SUN H, YANG J, SHEN J Q, et al. TIB-Net: drone detection network with tiny iterative backbone[J]. *IEEE Access*, 2020, 8: 130697-130707.
- [15] DING L H, XU X, CAO Y, et al. Detection and tracking

- of infrared small target by jointly using SSD and pipeline filter[J]. *Digital Signal Processing*, 2021, 110: 102949.
- [16] FANG H, XIA M, ZHOU G, et al. Infrared small UAV target detection based on residual image prediction via global and local dilated residual networks[J]. *IEEE Geoscience and Remote Sensing Letters*, 2021, 19: 1-5.
- [17] REDMON J, DIVVALA S, GIRSHICK R, et al. You only look once: unified, real-time object detection[C]//2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). New York: IEEE, 2016: 779-788.
- [18] HU Y, WU X, ZHENG G, et al. Object detection of UAV for anti-UAV based on improved YOLOv3[C]//2019 Chinese Control Conference (CCC) on Technical Committee on Control Theory. Guangzhou: Chinese Association of Automation, 2019: 8386-8390.
- [19] 李志朋, 赵长明, 张海洋, 等. 多尺度融合超分辨率算法在无人机探测中的应用[J]. *应用光学*, 2021, 42(3): 462-473.
- LI Zhipeng, ZHAO Changming, ZHANG Haiyang, et al. Application of multi-scale fusion super-resolution algorithm in UAV detection[J]. *Journal of Applied Optics*, 2021, 42(3): 462-473.
- [20] BOCHKOVSKIY A, WANG C Y, LIAO H Y M. YOLOv4: optimal speed and accuracy of object detection[EB/OL]. [2022-12-20]. <https://arxiv.org/abs/2004.10934>.
- [21] WANG C Y, LIAO H Y, YEH I H, et al. CSPNet: a new backbone that can enhance learning capability of CNN[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops. New York: IEEE, 2020: 390-391.
- [22] HU Xinyi. Intelligent traffic monitoring system based on small target[D]. Hangzhou: Hangzhou Dianzi University, 2021.
- [23] CAI Yingfeng, LUAN Tianyu, GAO Hongbo, et al. YOLOv4-5D: an effective and efficient object detector for autonomous driving[J]. *IEEE Transactions on Instrumentation and Measurement*, 2021, 70: 1-13.
- [24] LIU Hansen, FAN Kuangang, OUYANG Qinghua, et al. Real-Time small drones detection based on pruned YOLOv4[J]. *Sensors*, 2021, 21(10): 3374-3389.
- [25] LIU Z, SUN M G, ZHOU T H, et al. Rethinking the value of network pruning[EB/OL]. [2022-12-20]. <https://link.csdn.net/?target=https%3A%2F%2Fgithub.com%2Feric-mingjie%2Frethinking-network-pruning>.
- [26] ZHANG P Y, ZHONG Y X, LI X Q. SlimYOLOv3: narrower, faster and better for real-time UAV applications[C]//Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops. New York: IEEE, 2019.