ISSN: 2472-3703 DOI: 10.6911/WSRJ.202311_9(11).0010

Algorithm Improvement of Lightweight YOLOV4 Integrating Multiple Attention Mechanisms

Dongmei Ma[1, 2, a], Xiaoyun Luo[1, 2, b, *], Zhihao Guo[1, 2, c]

[1]School of Physics & Electronic Engineering, Northwest Normal University, Lanzhou 730070, China

[2]Gansu Province Intelligent Information Technology and Application Engineering Research Center, China

[a]madongmei@nwnu.edu.cn, [b]1844322635@qq.com, [c]751392570@qq.com

Abstract

With the development of science and technology, target detection technology has become more and more mature and has achieved great success in terms of accuracy. However, as the number of parameters increases exponentially, the algorithm training time becomes longer and longer. In order to further solve the problem of slow model training speed in target detection tasks, This paper studies the target detection task from a lightweight perspective. Based on the target detection algorithm of YOLOv4, a lightweight network is used as the backbone network. The efficient attention mechanism CBAM is added to the upsampling module to enhance the efficiency of information extraction and improve model accuracy. Compared with the original YOLOv4 algorithm model, the improved model has achieved better improvements in training time. Ablation experiments were conducted on the PASCAL VOC 2012 data set. Compared with the original network, the number of parameters was reduced by 38.2%, the model time was shortened by 43.3%, and the training speed was greatly improved. The improved algorithm model proposed in this article achieves better accuracy while reducing the complexity of the model and greatly reducing the number of parameters in the model.

Keywords

Target detection; YOLOv4; GhostNetv2; CBAM; Model complexity.

1. INTRODUCTION

As a breakthrough research in modern technology, computer vision has always attracted scientific researchers. Computer vision constructs artificial intelligence models to extract effective information from pictures or videos. Then the task of target detection is to locate and classify target locations in pictures or videos. Traditional target detection extracts features through manually designed methods, resulting in a low level of robustness and recognition accuracy, and uses a sliding window as the extraction target frame, which has many limitations. For example, weak target detection and recognition for different illumination, angles, scale changes, etc. are time-consuming and have high window redundancy.

Convolutional neural networks (CNN) have played a key role in the emerging field of deep learning. Looking back at the history of development, in 2012, the AlexNet model designed by a team led by Professor Hinton of the University of Toronto set a historical record in the ImageNet challenge. This breakthrough result triggered the craze of deep learning and brought convolutional neural networks into people's vision. In 2014, the VGG-16 model not only improved the accuracy but also reduced the number of parameters by increasing the number of layers and using small convolution kernels. In 2015, Google's GoogleNet introduced the

Inception structure, which uses multi-scale convolution kernels in parallel to make the network deeper but with fewer parameters and improve performance.

However, as the network deepens, problems of vanishing gradient and model degradation occur [2]. In order to overcome these problems, in 2016, Microsoft's He Kaiming team proposed ResNet, which introduced skip connections, allowing information to flow better in the deep network, and the accuracy rate was greatly improved. In 2017, SENet was proposed to further improve model performance by optimizing the weight distribution of feature channels.

These CNN models have not only achieved great success in the field of image classification, but have also been introduced into the field of target detection and are divided into two types of algorithms: candidate area-based and regression-based algorithms. From R-CNN to Fast R-CNN, Faster R-CNN, as well as YOLO and SSD [3], representatives of one-stage algorithms, these algorithms have improved the speed and accuracy of target detection and promoted the development of the field of computer vision.

YOLO (You Only Look Once) algorithm is a typical one-stage object detection method. This is a representative one-stage method that has attracted much attention due to its high efficiency. Its name means that the neural network processes the image only once and outputs object detection results. YOLO divides the image into multiple areas and generates multiple anchor boxes for each area, and then obtains the classification and location information of these anchor boxes through the network at one time. The advantage of this method is that it is fast because it does not require an additional region proposal step and instead completes the object detection task in one go. In general, two-stage object detection and one-stage object detection are two different methods, and there are some differences in the process and efficiency of target detection.

YOLOv1 is the first version, proposed by Joseph Redmon et al. in 2015. It is a single-stage object detection algorithm known for its high speed because it divides the image into grids in one go and generates multiple anchor boxes on each grid. The network then simultaneously predicts the presence of an object within each anchor box as well as the object's location and category. Although fast, YOLOv1 has some problems in the detection of small objects and dense object areas.

YOLOv2 improves on YOLOv1 and proposes better network architecture and training techniques. Among them, YOLO9000 supports more object categories, making it an important algorithm for multi-category object detection.

YOLOv3 further improves the network architecture based on YOLOv2 and introduces multi-scale detection to improve the detection performance of objects of different sizes. YOLOv3 has better accuracy, but slightly sacrifices in speed.

YOLOv4 is a major breakthrough, released in 2020. It employs a series of innovations, including CIOU loss function, PANet, SAM blocks, etc., resulting in significant improvements in accuracy.

2. TARGET DETECTION RELATED THEORIES AND TECHNOLOGIES

2.1. YOLOv4

YOLOv4 also uses some engineering optimizations to improve inference speed, making it a powerful object detector. In addition, YOLOv4 introduces a new backbone network structure containing five CSP modules, which significantly enhances the model's learning ability. It also introduces Droblock technology to mitigate the impact of overfitting. In YOLOv4, the neck structure uses the SPP (Spatial Pyramid Pooling) activation function and is used in conjunction with the SPP module. This innovative feature aggregation method can effectively process feature maps of different sizes and better realize the fusion of feature information.

By adopting Mosaic technology, YOLOv4 is able to randomly combine four different images together for training, which significantly improves the robustness of the model and provides the model with more information for effective learning. In the prediction part of target detection, YOLOv4 uses CIOU_Loss to replace the traditional IOU_Loss, and DIOU_NMS to replace the standard non-maximum suppression (NMS) method. This innovation takes into account factors such as target boundary non-overlapping, target centroid distance, and target boundary aspect ratio, helping to improve detection accuracy and stability.

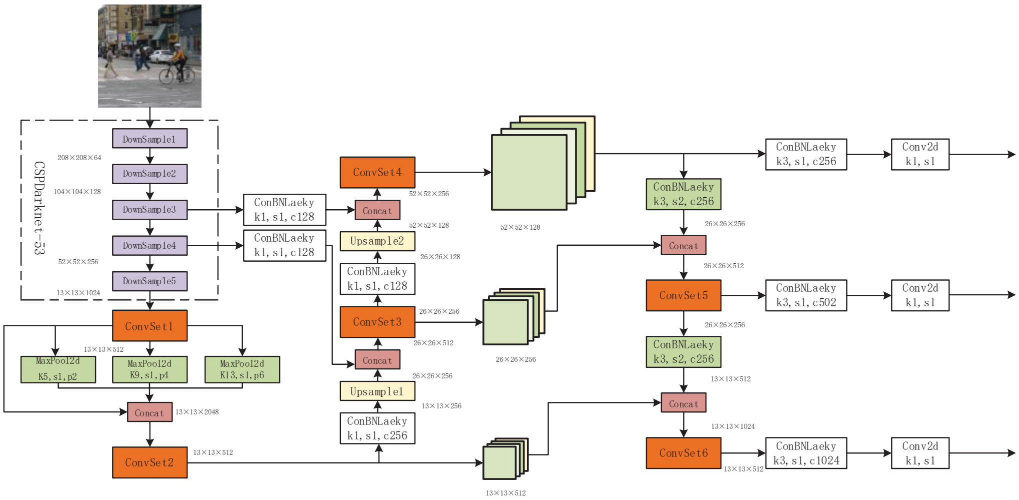


Figure 1. Network structure diagram of YOLOv4

2.2. Freeze-thaw experiment

Freeze-thaw training is a commonly used technique in deep learning, aiming to improve the training speed and performance of the model. The core idea of this method is to lock (freeze) the parameters of some layers of the neural network during the training process, and only train the parameters of other layers. Once the latter part of the layer is trained, the previously frozen layer is unlocked and the entire model continues to be trained until it reaches the optimal state of training. This process can be repeated, and different layers can be selected to freeze and defrost as needed to obtain the best training results.

The background for the application of this technique lies in the problem of gradient vanishing that can occur during the training process of deep neural networks, especially when the number of network layers increases [4]. By freezing the parameters of some layers, the gradient update of these layers can be avoided, thus stabilizing the training process. In addition, deep neural networks usually require a large amount of training data and computational resources [5][6], and by freezing the parameters of some layers, the number of parameters to be trained can be reduced, thus improving the training speed.

In practical applications, it is common practice to use the first few layers of the model as feature extractors and freeze them, and then only train subsequent layer parameters of the network. This can take advantage of the feature extraction capabilities of the pre-trained model to accelerate model training. As training progresses, you can unfreeze the parameters of the first few layers and continue training the entire model to further improve performance.

2.3. Focal Loss function

Focal Loss is a loss function used to solve the class imbalance problem, especially in computer vision tasks such as target detection and image segmentation. Its core idea is to give greater weight to samples that are difficult to classify (that is, samples that are easily misclassified) so that the model can pay more attention to the training of these samples. This helps alleviate training difficulties caused by class imbalance in the dataset.

The key parameter of Focal Loss is an adjustment factor (called the focus parameter), which is used to control the degree of attention paid to difficult-to-classify samples. When the focus parameter is low, the model pays more even attention to all samples and is suitable for class-balanced situations. But when the focus parameter increases, the model will pay more attention to samples that are difficult to classify, thereby improving its ability to adapt to class imbalance situations.

In general, Focal Loss helps the model better handle the category imbalance problem by introducing focus parameters, allowing it to pay more attention to difficult-to-classify samples during the training process, thus improving the performance of the model. This method has been widely used to improve the performance of various deep learning models in vision tasks. The calculation method is as follows:

$$CE(\mathrm{p},\ \mathrm{y}) = \begin{cases} -\log(p) & \textit{if } y=1 \\ -\log(1-p) & \text{otherwise} \end{cases} \tag{1}$$

The above is the formula of binary cross entropy, where y takes values 1 and -1, representing the foreground and background respectively. p takes a value from 0 to 1, which is the probability of the model predicting the prospect.

Next define a function p_t:

$$p_t = \begin{cases} p & \text{if y=1} \\ 1-p & \text{otherwise} \end{cases} \tag{2}$$

The deformed binary cross-entropy loss function is obtained as follows:

$$CE(p,y) = CE(p_t) = -\log(p_t) \tag{3}$$

Focal Loss adds an adjustment factor based on the balanced cross-entropy loss function to reduce the weight of easy-to-classify samples and focus on training difficult-to-classify samples [7]. It is defined as follows:

$$FL(p_t) = -\alpha_t (1-p_t)^{\gamma} \log(p_t) \tag{4}$$

$(1-p_t)^{\gamma}$ is the adjustment factor, $\gamma \geq 0$ is the adjustable focus parameter, and α weight helps to deal with the unevenness of the class.

2.4. Depthwise separable convolution

Depthwise Separable Convolution is a lightweight convolution operation, especially suitable for model lightweight and acceleration in deep neural networks. It is evolved from standard convolution operations and aims to reduce the number of parameters and computational complexity of the model while maintaining model performance.

Depthwise separable convolution is mainly divided into two steps: depthwise convolution (Depthwise Convolution) and pointwise convolution (Pointwise Convolution) [8].

Depthwise Convolution: In this step, each input channel is convolved with a separate filter (convolution kernel), instead of convolving across channels like standard convolution. This means each input channel has its own filter, which helps capture features within the channel.

Pointwise Convolution: After depth convolution, pointwise convolution is a 1x1 convolution operation [9], which is used to fuse features of different channels together. This step typically uses fewer filters to reduce the number of channels and reduce computational effort. The goal of point-wise convolution is to integrate information between channels to generate the final output feature map.

The advantage of depthwise separable convolution over standard convolution is that it significantly reduces the number of parameters and computational overhead [10]. Since depth convolution only performs convolution within channels and does not involve combinations between channels, the number of parameters is greatly reduced. Pointwise convolution further compresses the number of channels, thereby reducing the computational burden. This makes depthwise separable convolutions ideal for deploying lightweight deep learning models on mobile devices and embedded systems for high-performance real-time inference.

In short, depthwise separable convolution is a lightweight convolution operation, through the combination of deep convolution and point-wise convolution, the number of parameters and computational complexity are effectively reduced [11], while maintaining the performance of the model, making it an efficient deep learning model for building in resource-constrained environments. By using depthwise separable convolutions, we can increase the complexity of the neural network while keeping the input constant. The computational intensity of this method is only about 1/3 of traditional convolution.

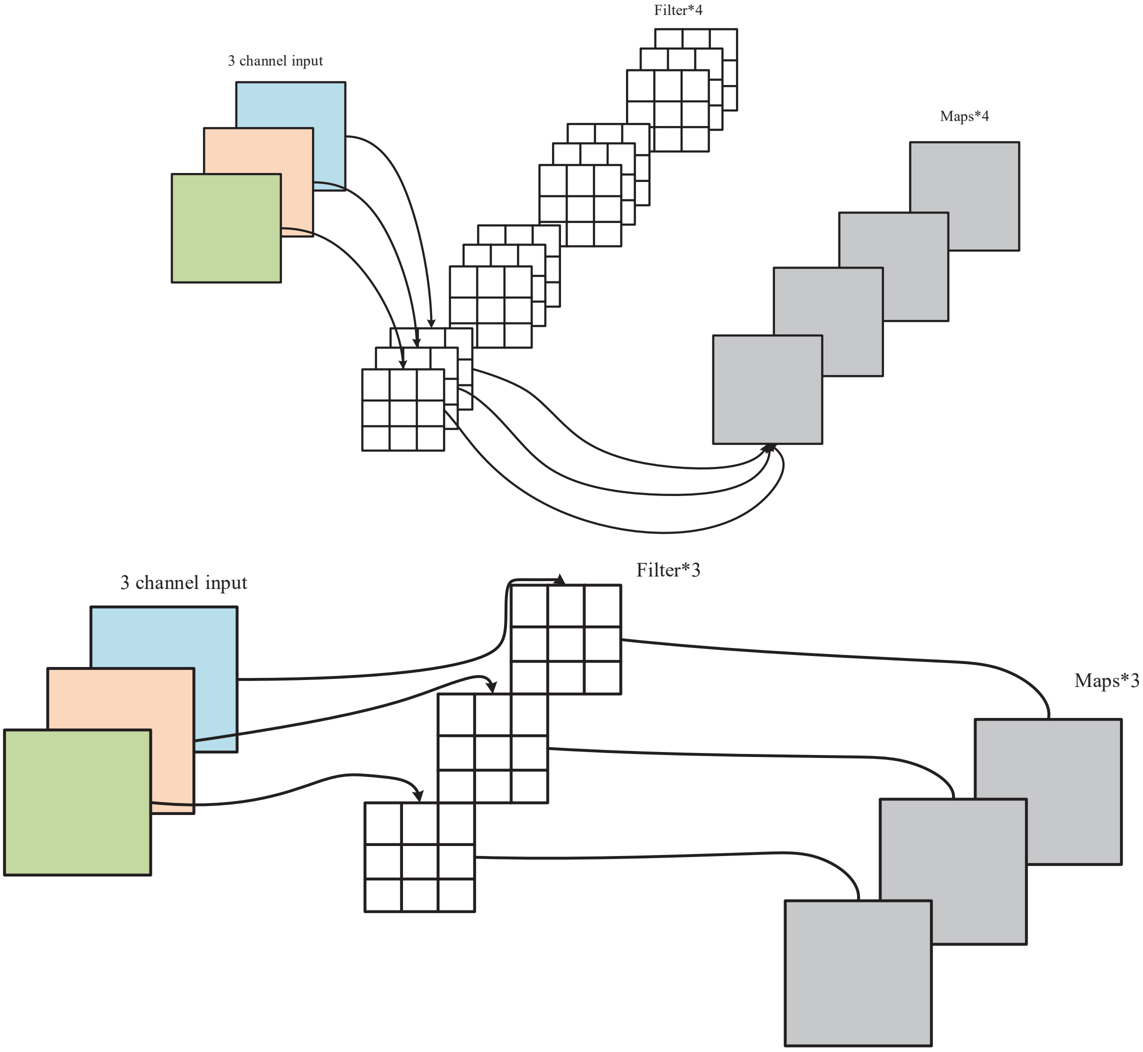


Figure 2. Comparison of ordinary convolution and depth-separable convolution

3. IMPROVE YOLOV4 ALGORITHM

3.1. Improvement strategies

This research is based on YOLOv4 and aims to create a lightweight target detection network structure to balance speed and accuracy in target detection tasks. By optimizing the original network and reducing the network size and number of parameters, we successfully reduced information redundancy. At the same time, while maintaining accuracy, we adopted an optimization strategy to enhance the performance of the feature extraction module, thereby improving the accuracy of target detection. The main contributions are: (1) Freeze-thaw training is used to simplify the calculation amount, and the lightweight backbone network GhostNetv2 is used instead of the CSPDarknet53 network for feature extraction. Depth-wise separable convolution is used in the feature extraction network to reduce the number of model parameters [1] . (2) Add depthwise separable convolution to ASPP to increase the receptive field of multi-scale fusion of the model. The loss function uses the Focal Loss loss function to accelerate the network convergence process. (3) After the model is lightweight, the CBAM attention mechanism is introduced in upsampling to improve accuracy without increasing computational overhead and effectively improve information extraction efficiency. The above three improvements have greatly improved the accuracy and successfully completed the requirements of model balance accuracy, calculation amount and speed.

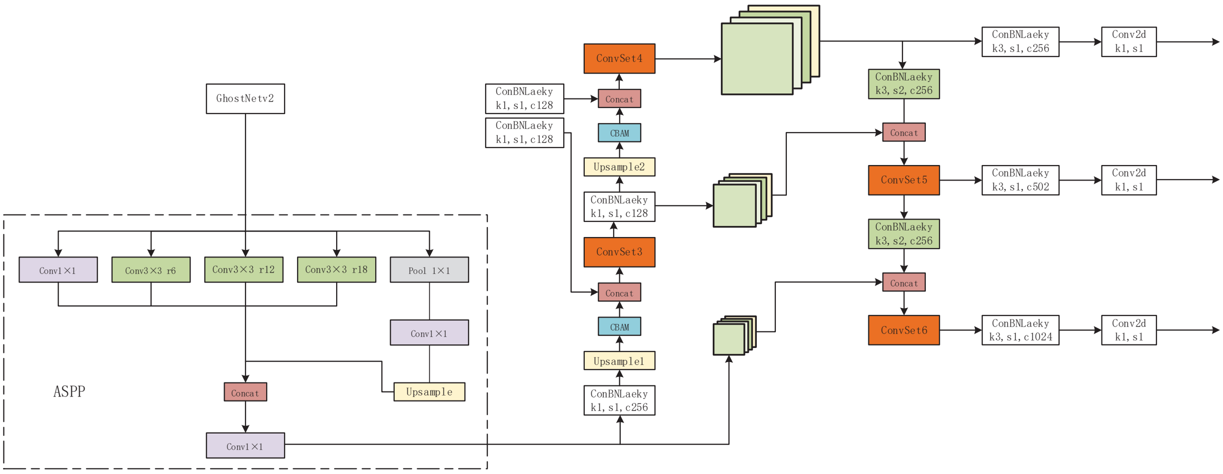


Figure 3. Improved YOLOv4 structure diagram

3.2. Lightweight network GhostNetv2

GhostNetv2 is a lightweight deep neural network architecture that is a further improved version of GhostNet. The main goal of GhostNetv2 is to further improve the performance of the model while remaining lightweight and efficient. GhostNetv2 uses a series of innovative designs and techniques to achieve better image classification and target detection performance under limited computing resources.

One of the key features of GhostNetv2 is the introduction of two new components: Ghost Bottleneck and Ghost Module. Ghost Bottleneck is a convolutional block that, while retaining the main feature channel, introduces a "ghost" channel for learning secondary features. Such a design can improve model performance with almost no increase in computational overhead. Ghost Module is used to replace the standard convolution layer, which uses a more efficient way to extract and fuse features.

Another innovation of GhostNetv2 is to build a deeper network through the combination of Ghost Bottleneck and Ghost Module to improve feature representation capabilities. At the same time, GhostNetv2 also uses an adaptive channel selection mechanism that can dynamically select appropriate channels based on different input data, thereby further improving the flexibility and performance of the model.

In general, GhostNetv2 is a lightweight and efficient deep neural network. It introduces innovative designs such as Ghost Bottleneck, Ghost Module and adaptive channel selection， It achieves better image classification and target detection performance under limited computing resources. This makes GhostNetv2 a strong choice for deploying deep learning models on mobile devices and embedded systems.

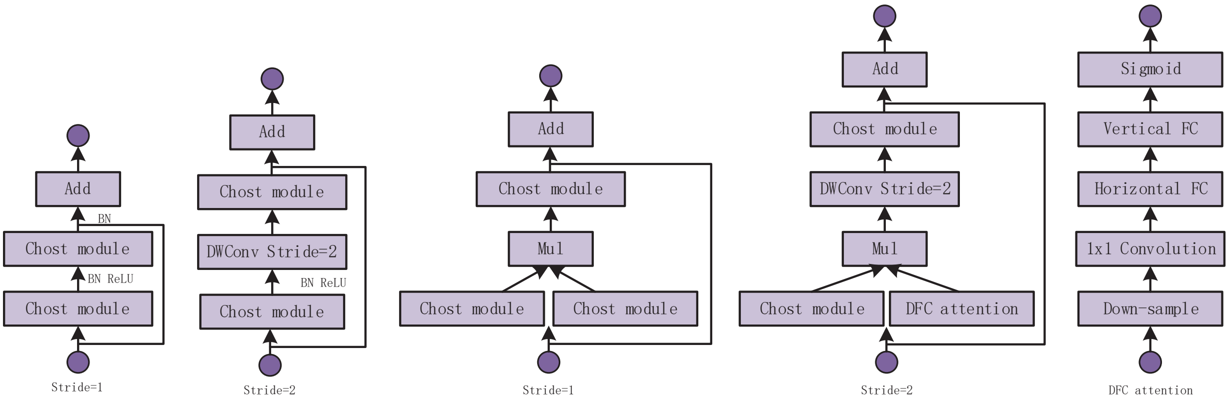


Figure 4. GhostNetv2 structure diagram

3.3. CBAM attention mechanism

The introduction of attention mechanism can improve the performance of neural networks, effectively manage computing resources, and cope with the challenge of information overload. As the model parameters increase, the representation can be improved and the available information can be increased, thereby reducing the burden of information processing. By using attention allocation technology, important information can be focused on key parts of the current task, reducing attention to unnecessary information, reducing interference from external resources, improving task execution efficiency, accelerating task completion and improving accuracy. This process helps to fully utilize resources and address information processing challenges more effectively.

CBAM (Convolutional Block Attention Module) is an attention mechanism based on convolutional neural networks [12]. Its working principle involves both channel and space aspects to improve the feature representation quality of the model.

The core goal of CBAM is to improve the feature representation capabilities of the model so that it can better understand and utilize the input data. Through the combination of channel and spatial attention, CBAM enables the model to focus on important channels and regions, reducing interference with irrelevant information, thereby improving the performance and generalization ability of the model.

3.4. Confidence in target detection

In the target detection algorithm of this article, the confidence of each prediction box is calculated by the following formula:

$$Confidence = \Pr(object) * loU\_truth\_pred \tag{5}$$

$\Pr(object)$ represents the probability of the presence of an object in the predicted frame, loU_truth_pred Represents the intersection ratio between predicted frames and real frames.

The specific calculation is as follows:

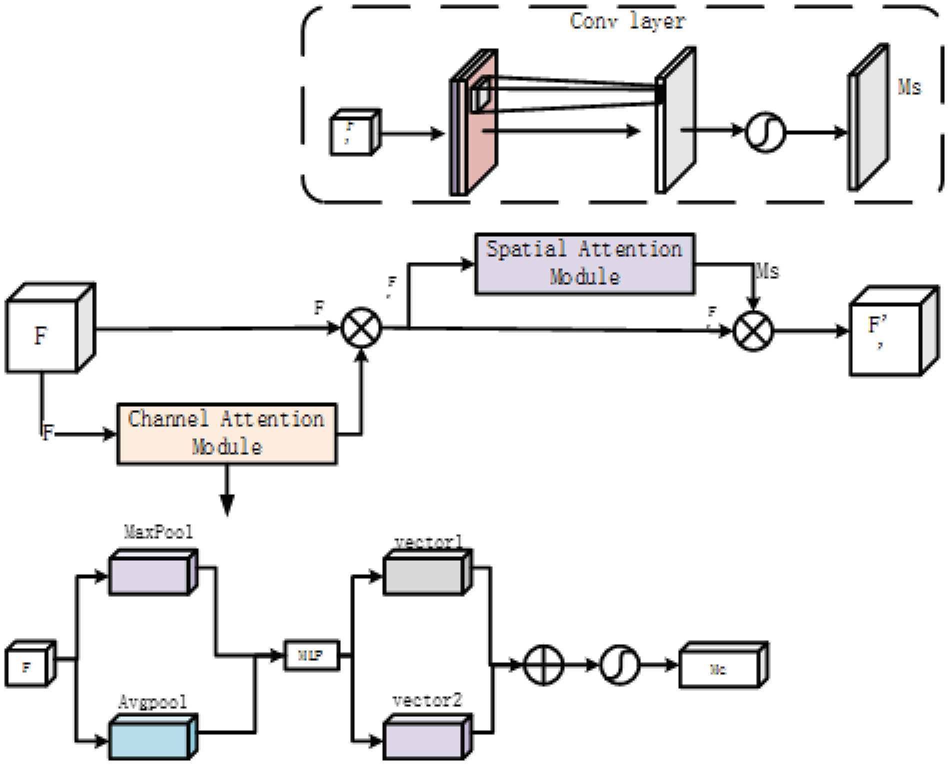


Figure 5. CBAM attention mechanism flow chart

$$\Pr(object) = 1/(1+\exp(-tx)) \tag{6}$$

where tx is a scalar value output by the network indicating the confidence that the object is present in the frame. The larger tx is, the closer it is to $\Pr(object)$ 1, indicating a higher probability that an object exists in the frame; the smaller tx is, the closer it is to $\Pr(object)$ 0, indicating a higher probability that an object does not exist in the frame.

The calculation of loU_truth_pred is as follows:

$$loU\_truth\_pred = Area\_of\_overlap / Area\_of\_union \tag{7}$$

Where $Area\_of\_overlap$ represents the area of the intersection of the predicted frame and the real frame, and $Area\_of\_union$ represents the area of the union of the predicted frame and the real frame. Multiplying by $\Pr(object)$ and lou_truth_pred can get the confidence of the prediction box [18].

3.5. Add ASPP

The ASPP module is added to the target detection algorithm in this article to increase the receptive field of the model and improve the network detection capability. ASPP module, as an advanced convolutional neural network structure, has made significant breakthroughs in the field of target detection.The core idea is to capture semantic information at different scales through multi-scale atrous convolution operations. By introducing a pyramid pooling technology, target objects of different sizes and proportions can be effectively processed. Since the ASPP module has multi-scale characteristics, the introduction of the ASPP module into the algorithm of this article enables the model to better adapt to diversity.

In addition, ASPP also uses atrous convolution to expand the receptive field, which effectively enhances the model's ability to understand image context information. In object detection, contextual information is crucial to correctly determine the location and category of an object. ASPP provides the model with more information about the surrounding environment by expanding the receptive field range, helping to improve the accuracy and robustness of detection. The ASPP module can also reduce false detection rates and improve detection accuracy. By fusing features of different scales, it helps the model better understand the appearance of the target and the surrounding background information. This comprehensive feature processing makes the model more resistant to interference and reduces the misjudgment of the background as a target, thus improving the credibility of the detection results.

4. ANALYSIS OF EXPERIMENTAL RESULTS

4.1. Basic configuration of the experiment and introduction to the data set

The algorithm in this article is re-engineered using the pytorch 1.8.0 framework and the Windows operating system. The CPU model is the 12th generation Intel(R) Core(TM) i5-12400F 2.50 GHz, the GPU model is NVIDA GeForce RTXTM 3060Ti, and the video memory is 8G. A total of 300 epochs were trained, 50 epochs were frozen training with a batch size of 8, and the remaining 250 epochs were unfrozen training with a batch size of 4. Finally gradient descent with a learning rate of 1e-2 is used [13].

The data set used in this experiment is PASCAL VOC 2012. PASCAL VOC 2012 includes a total of 20 types of objects. There are 11530 images in train and val, and a total of 27450 object detection labels. The experiment will select 90% of the images in the data set for training, and the remaining 10% of the pictures are used to verify the detection effect.

4.2. Evaluation indicators

This comparison experiment uses Mean Average Precision as the experimental evaluation standard. Before calculating mAP, we need to calculate IoU, precision, recall and AP. IoU represents the correlation between the prediction box and groundtruth， reflects the relationship between the prediction box and groundtruth. By testing positive prediction boxes, you can determine which prediction boxes predict more accurately, leading to higher accuracy. By setting the IoU threshold, we can accurately count the accuracy of True Positives and False Positives for different classifications of images. In this way we are able to identify different types of precision, which are calculated as follows:

$$precision = \frac{TP}{TP+FP} \tag{8}$$

Since we get accurate True Positives (TP), we can easily count the number of undetected objects (False Negatives, FN), which allows us to calculate the Recall:

$$\text{Recall} = \frac{TP}{TP+FN} \tag{9}$$

AP is the average precision that measures all recalls. AP is calculated as follows:

$$AP = \sum \left(r_{n+1} - r_n\right) p_{\text{int}erp}\left(r_{n+1}\right) \tag{10}$$

By calculating the AP of each category and calculating the average, mAP is obtained. The calculation formula is:

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

4.3. Ablation experiment

To verify the effectiveness of our improved lightweight YOLOv4 algorithm, we conducted a series of ablation experiments on PASCAL VOC 2012. The purpose is to analyze and evaluate the impact of each added module on model performance to confirm that they have an optimization effect on the algorithm.

Table 1. Results of ablation experiments of each module on PASCAL VOC 2012

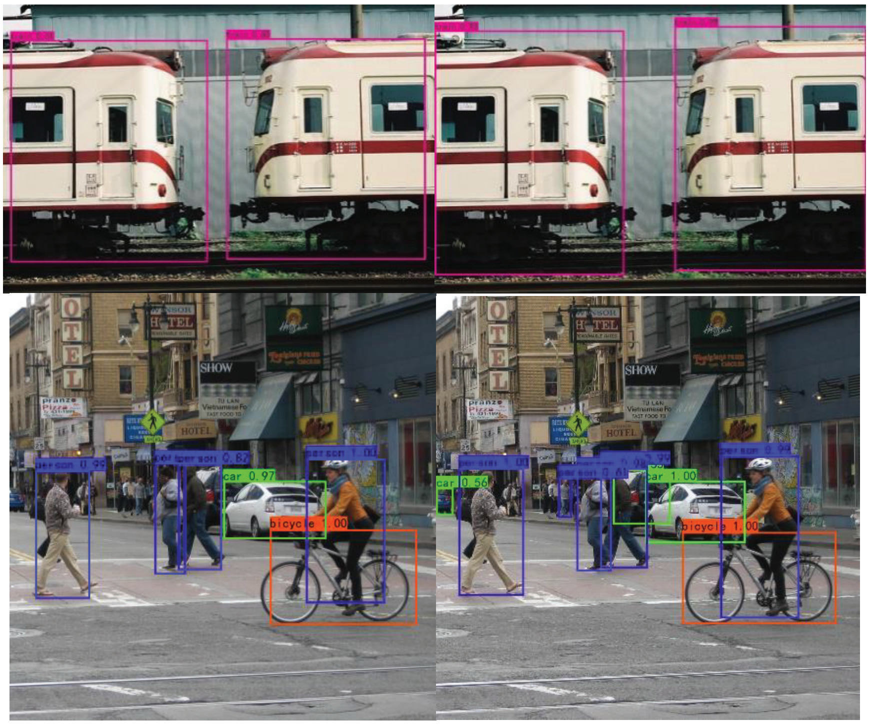
Model	mAP/%	Number of parameters	Model size /MB	FPS
YOLOv4	83.64	64035002	244.29	7.26
YOLOv4+GhostNetv2	78.42	38002010	153.72	12.39
YOLOv4+GhostNetv2+ freeze-thaw	80.64	38249012	155.37	12.11
YOLOv4+GhostNetv2+ freeze-thaw +dw	75.36%	17297033	83.14	14.63
YOLOv4+GhostNetv2+ freeze-thaw +dw+Focalloss	76.21%	19893157	106.46	14.25
YOLOv4+GhostNetv2+ freeze-thaw +dw+Focalloss+CBAM	80.90%	39549489	120.61	12.80

It can be concluded from this table that by introducing the GhostNetv2 network and depthwise separable convolution to optimize our model, the network is lightweight. The lightweight feature extraction network GhostNetv2 was introduced, which reduced the number of model parameters by 41%, reduced the model size by 34%, greatly improved the training speed, and reduced mAP by 5.7% compared with the basic model YOLOv4. This optimization work significantly reduced the parameters of the model. quantity and complexity, while significantly reducing model size, thereby increasing frames per second (FPS) and greatly accelerating the training process. However, it is worth noting that this lightweight approach also has a certain impact on the model's average accuracy (mAP), reducing it to 76.21%.

Through ablation experiments, we clarified the key role of GhostNetv2 and the use of depthwise separable convolution in achieving model lightweighting. Using the freeze-thaw and focal loss pre-training strategies, mAP will be slightly improved, the model size will be slightly increased, and the training speed will be slightly reduced. The above data proves the effectiveness of the pre-training experiment. By introducing the CBAM attention mechanism, we successfully increased the average precision (mAP) of the network to 80.90%. However, this improvement also slightly increases the number of parameters of the network and the size of the model, while slightly reducing the frames per second (FPS). Ablation experiments clearly show that utilizing attention mechanisms can effectively improve model accuracy, but there is also a trade-off between model size and speed. The results show that the improved model in this article can achieve better results in training speed and accuracy.

4.4. Experimental results

Although our experimental results show that the mAP of our improved algorithm decreases by 3.22% compared with the original YOLOv4, this algorithm successfully achieves the goal of significantly reducing the model size and parameter amount. The model size is only 120.61MB. This improvement is of significant significance to the lightweight model. In addition, it also greatly speeds up network training and saves valuable time resources. The algorithm in this paper uses less data and shorter time to obtain good object detection results. The figure below shows some verification results of object detection using the improved algorithm and the original YOLOv4 model.



(a) Test results of this algorithm (b) YOLOV4 test results

Figure 6. Comparison of test results

4.5. Comparison of the improved algorithm with other algorithms

In order to verify the effectiveness of the improved model, the model in this paper was compared with some commonly used target detection algorithms on the PASCAL VOC 2012 data set [14]. Table 2 shows the comparison results.

Table 2. Comparison between this algorithm and other common object detection results

Model	core network	mAP/%	Model size /MB	FPS
YOLOv4	CSPDarknet53	83.64%	244.29	7.26
YOLOv4-Tiny	CSPDarknet53-Tiny	68.61%	22.58	20.18
YOLOv4-ResNet	ResNet50	84.24%	210.35	8.25
YOLOv4-vgg	VG16	80.58%	187.81	9.92
YOLOv5s	SPP+CSP2_X	57.52%	14.45	22.70
Algorithm	GhostNetv2	80.90%	120.61	12.80

From the tabular data, it is observed that the lightweight networks YOLOv4-Tiny and YOLOv5s, although the model accuracy is very low, still perform well in terms of model size and training speed. Compared with YOLOv4 and YOLOv4-ResNet, the improved algorithm in this article has a slight decrease in accuracy, but our algorithm has a significant reduction in model size, while also having higher FPS performance and shorter training time. Compared with YOLOv4-vgg network, our algorithm shows better performance in terms of both accuracy and model size. Taken together, our algorithm shows excellent performance in the field of target detection.

5. CONCLUSION

The main goal of this research is to improve the YOLOv4 target detection model to achieve model lightweight and increase detection speed while maintaining high accuracy. To achieve this goal, we implemented several optimization strategies, resulting in significant performance improvements.

First, we replaced the backbone network of YOLOv4, selected the lightweight GhostNetv2, and introduced depthwise separable convolution. This move significantly reduces the number of parameters and complexity of the model, thereby achieving lightweighting. At the same time, we introduced the attention mechanism and improved the accuracy of the model through attention allocation technology. Especially when dealing with key tasks, the model performed better.

In addition, we also added the ASPP module, which uses the pyramid pooling method to perform Dilated Convolution on feature maps with different sampling rates, and fuses these feature maps to obtain a more comprehensive and refined semantic information representation. This operation expands the receptive field of the model and significantly enhances the target detection capability of the network.

Experimental results show that although the accuracy of the algorithm decreases slightly relative to the original YOLOv4, we achieve significant performance improvements. FPS increased from 7.26 to 12.8, and model size was significantly reduced. Compared with current mainstream object detection algorithms, our algorithm shows obvious advantages in many aspects such as accuracy, model size and training speed.

In summary, the algorithm of this study successfully achieves high performance in the field of target detection through a series of optimization measures. This research provides strong support for the development of lightweight target detection models and has broad application prospects.

The algorithm in this paper also has the problem of missing some objects. In the future, the FPN module and Soft-NMS module will be used to improve the model to further optimize the network, so that the corresponding objects can be output on different layers at a faster speed, while improving the algorithm's detection of small targets or dense targets. capabilities so that the model can be applied in daily life.

REFERENCES

[1] Liu Dongdong: Research on insulator fault detection based on improved YOLOv4 algorithm. Electrical Technology, (2022) No.2, p.151-155.

[2] Li Quanhong, Li Lei, Li Chunbin, Wu Jing, Chang Xiuhong; Automatic segmentation of land cover types in remote sensing images based on residual U-Net, Vol.35 (2021) No.1, p.98-106.

[3] Cao Pei: Dual-sensor information fusion target detection and attitude estimation for autonomous driving (MS., Harbin Institute of Technology, China 2019), p.24.

[4] Zhang Huanhuan: Research on three-dimensional model target detection method based on deep learning (MS., Chang'an University, China 2019), p.31.

[5] Lan Kankan: Research on progressive kernel width ensemble classification (MS., South China University of Technology, China 2020), p.34.

[6] Tian Tengfei: Image feature representation and application based on generative adversarial network (MS., Southeast University, China 2018), p.42.

[7] Xie Xueli, Li Chuanxiang, Yang Xiaogang, et al: Aerial image target detection algorithm based on dynamic receptive fields, Vol. 40 (2020) No.4, p.107-119.

[8] Guo Qinyi: Review of research on malicious code detection technology, Vol. 19 (2023) No.13, p.79-81+93.

[9] Ju Cong, Li Tao: Research on facial expression recognition based on deep separable convolution structure, (2020) No.6, p.1-5.

[10] Ma Jingmin, Feng Jie, Zhang Jing: Research on intrusion detection methods based on convolutional neural networks (Nanning, Guangxi, December 20, 2020). Vol. 1, p.14-23.

[11] Jiang Xiaoyu, Li Zhongbing, Zhang Junhao, et al: Vehicle detection method based on NCS2 neural computing rod, Vol. 47 (2021) No.3, p.298-303.

[12] Long Jiehua, Guo Wenzhong, Lin Sen, et al: Improved YOLOv4 method for strawberry growth period identification in greenhouse environment, Vol. 3 (2021) No.4, p.99-110.

[13] Li Yawen, Sun Haoran, Hu Yueming, et al: Electrode defect YOLO detection algorithm based on attention mechanism and multi-scale feature fusion, Vol. 38 (2023) No.9, p.2578-2586.

[14] Zhang Li, Sun Kelei: Multi-scale target detection algorithm based on improved regional recommendation network, Vol. 27 (2021) No.2, p.26-31.