ISSN 2616-5775 Vol. 7, Issue 12: 36-43, DOI: 10.25236/AJCIS.2024.071205

An improved yolov5s algorithm and its application in object detection

Chenxi Yan[1], Jiafeng Li[2]

[1]*School of Computer Science, Northeast Electric Power University, Jilin, 132011, China*
[2]*College of Software Engineering, Sichuan University, Chengdu, 610207, China*

Abstract: *With the rapid development of artificial intelligence technology in recent years, object detection methods have become a research hotspot in theory and application. However, the existing detection methods generally have the problem of low detection accuracy. To solve this problem, some scholars have proposed deep learning-based models, but this increases the complexity of the model and reduces the training efficiency. To this end, this paper proposes a new improved YOLOv5s algorithm that balances lightweight and performance. First, replace the original C2F module with MobileNetV3-Small to reduce the model complexity. Then, the SE attention mechanism is introduced to obtain global information, learn the correlation between features at different scales and fuse them, enhance the semantic information of features, and use SGD as an optimizer to further improve the accuracy. This paper is verified on the STL-10 public data set. The experimental results show that after the introduction of the MobileNetV3-Small framework, the number of valid parameters of the model is reduced, and the training time is greatly reduced. At the same time, compared with other mechanism attention, the SE attention mechanism has the greatest improvement in performance, and has excellent performance in lightweight and algorithm performance balance. The effectiveness of the optimization strategy has been verified. Compared with the underlying Yolov5 algorithm, the proposed improved Yolov5s algorithm improves the detection accuracy by 0.5, and the superiority of the model is verified.*

Keywords: *yolov5s; object detection; SE attention mechanism; MobileNetV3-Small*

1. Introduction

As one of the core problems of computer vision domain, object detection is widely used in facial recognition, intelligent transportation, industrial detection, autonomous driving and many other domains [1]. With the rapid development of deep learning in recent years, object detection has become a research hotspot in theory and application. However, existing methods generally have the problem of low detection accuracy when the target scale distribution is inconsistent. To solve this problem, some scholars have proposed models based on deep learning, but this also increases the complexity of the model and reduces the training efficiency.

Redmon [2] et al. first proposed the YOLOv2 model, and combined a method of jointly training object detection and classification. At the same time, the YOLOv2 model was trained on the COCO detection data set and the ImageNet classification data set, so that it can predict the class of objects without labeled detection data. However, compared with fast R-CNN, YOLOv2 performs poorly in terms of position error, and has a low recall rate, which is prone to unstable training. Subsequently, Redmon [3] et al. proposed the YOLOv3 model based on this, using a new network for feature extraction, allowing cross-scale prediction, and training on the same data set, resulting in a significant improvement in accuracy. However, this model is not suitable for dealing with larger size targets, and there are certain difficulties in perfectly aligning object bounding boxes.

With the continuous research on deep learning, Bochkovskiy [4] et al. proposed a new YOLOv4 model in 2020, which uses the CSPDarknet53 backbone network and Mosaic data augmentation method to modify the parameters of the model through self-adversarial training, thereby improving the stability and accuracy of the model, and trained on the MS COCO data set, the implementation of 43.5% of the AP, but because the model is more complex and computationally intensive, resulting in a long training time, and prone to overfitting problems when dealing with small-scale data sets. After that, some teams further improved YOLOv4, aiming to improve the training speed and more accurate detection. The final experiment reached 50.7% AP, but it was also difficult to balance the contradiction between model size and detection accuracy. In the same year, Ultralytics [5] launched YOLOv5 algorithm, which not only

ISSN 2616-5775 Vol. 7, Issue 12: 36-43, DOI: 10.25236/AJCIS.2024.071205

reduced the model size, but also accelerated the convergence speed of the training process through adaptive anchor box and other data set-based optimization strategies. Although the first-stage object detection algorithm directly outputs the class probability and position coordinates of the object in parallel classification and positioning, thus speeding up the detection speed, it still lacks accuracy and is not enough to fully meet the needs of high-accuracy detection for autonomous driving.

For the object detection algorithm, its training process can be regarded as a process of hyperparameter optimization. The random gradient descent (SGD) algorithm [6] is an underlying optimization technique that is widely used in many ML (Machine Learning) tasks to solve optimization problems and achieve excellent results. During each iteration, the algorithm selects the steepest descending direction, the direction opposite to the gradient direction, to update the model parameters. Although the algorithm is simple in structure and easy to program implementation, the randomness in the parameter update process may lead to instability in the update direction, which causes continuous fluctuations in the model during the optimization process, resulting in slower convergence speed [7].

As mentioned above, in view of the problems of low detection accuracy and dependency of high-performance optimizer in yolov5 algorithm, this paper proposes an improved yolov5 algorithm. Specifically, MobileNetV3-Small architecture is introduced into the backbone network of YOLOV5 to reduce model complexity and improve training speed. At the same time, a high-performance SE attention mechanism is introduced to improve the model's ability to capture global information. Finally, the SGD optimizer is used for model hyperparameter optimization to improve model training efficiency. The improved YOLOv5 algorithm has the characteristics of light level, high performance, high efficiency and wide applicability, which can meet the application scenarios that require object detection accuracy, real-time performance and resource consumption.

2. Basic yolov5s algorithm

YOLOv5s [8] uses feature pyramid network (FPN) technology to fuse different levels of features to generate a fixed number of feature maps for each grid cell, enabling the model to better identify targets of different scales. In addition, YOLOv5s also adopts anchor box technology, which predefines a series of bounding boxes, and predicts the position and size of objects according to the position and size of these boxes, further improving the detection accuracy. Figure 1 shows the model framework of YOLOv5s. The model is mainly composed of four parts: Input, Backbone, Neck, and Head.

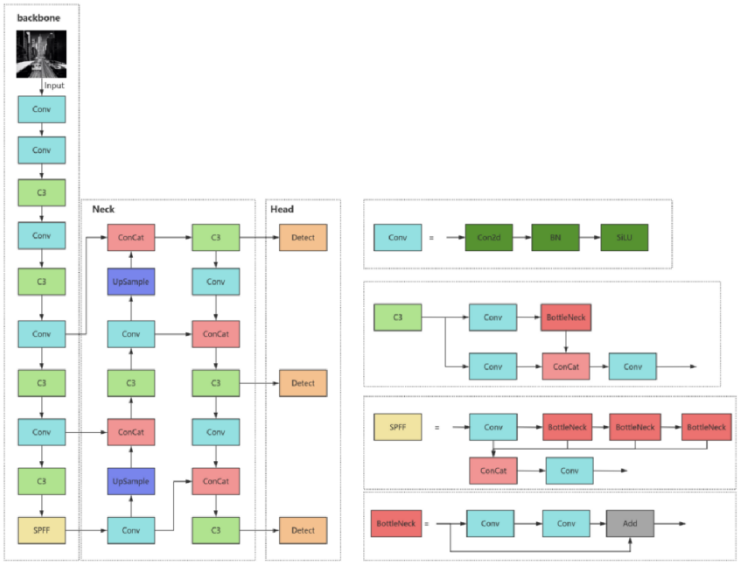

Figure 1: YOLOv5s overall frame diagram

2.1 Head

The main part is the output layer in the model, which is responsible for generating the final detection result. It basically consists of a series of convolution layers that are used to generate the bounding box coordinates and class probability of each feature based on the input feature map.

ISSN 2616-5775 Vol. 7, Issue 12: 36-43, DOI: 10.25236/AJCIS.2024.071205

The bounding box center point coordinates (x,y) and object map coordinates are mapped to the center point coordinates of the predicted bounding box using a linear hover function.

$$\begin{cases} x = sigmoid(x) \\ y = sigmoid(y) \end{cases} \tag{1}$$

Where x and y are coordinate values in the feature map, the sigmoid function maps coordinate values in the range 0 to 1, indicating the relative position of the midpoint of the bounding boxes in the image.

The bounding box width and height (w, h) are also mapped to the predicted bounding box width and height using a linear activation function.

$$\begin{cases} w = \exp(w) * anchor\_width \\ h = \exp(h) * anchor\_height \end{cases} \tag{2}$$

Where W and H are the width and height values on the feature map, and the EXP function plots the values to a positive range representing the width and height of the border. anchor_width and anchor_height are predefined values for the width and height of the anchor box.

Certainty (confidence), confidence represents the degree of confidence that the model has in the presence of objects in the predicted boundary boxes.

$$confidence = sigmoid(conf) \tag{3}$$

where, $conf$ is the confidence value on the feature map, and the sigmoid function maps the value between 0-1, indicating the degree of confidence the model has in the presence of objects in the predicted bounding boxes.

2.2 Backbone

Frameworks typically use deep learning models such as ResNet or CSPNet as the core architecture. These basic architectures usually consist of several folded layers and residual blocks. The convolutional layer is used to extract image features, while the remaining blocks achieve regression of the function [8] by jumping connections to alleviate the fading gradient problem in deep neural networks.

$$F = H(I) + I \tag{4}$$

In the equation, F stands for the indicator image after the stump has been processed, H(I) stands for the input image of the precipitation symbol and link to the rest, while you stand for the input image. The results of this can then be picked up by the rest and reused.

2.3 Neck

Neck is a feature hybrid network in the YOLOv5s model that is tasked with combining features at different levels to provide richer feature information for the final detection task. It consists of a series of convolution layers that can sample features extracted from the Backbone to create a fixed number of feature maps for each network cell, sampling, merging, and bottoming. These feature maps are used as inputs to the Head to produce the final detection results.

First, the feature map is complicated through a series of layers of complexity to extract more features. The feature map then performs sampling operations to create a fixed number of feature maps. Finally, resampling is done on the feature map after the bottom sampling to combine the features at different levels while improving the clarity of the feature map.

The YOLOv5s model shows high accuracy and speed in object detection tasks, but there are also some problems with low accuracy: (1) the detection effect of small targets is weak and the object's local information cannot be fully captured, resulting in a reduced accuracy of models when detecting small target scenes. (2) A small target occupies a small pixel area in the image, making the feature more difficult to display, and the model loses or misrecognizes. (3) The loss function is not accurate enough to predict the position and size of the object, and the detection results in the statistics are biased.

ISSN 2616-5775 Vol. 7, Issue 12: 36-43, DOI: 10.25236/AJCIS.2024.071205

3. Improved YOLOV5s algorithm

By focusing on the problems of low detection accuracy and high performance optimizer dependent on yolov5s algorithm, this paper proposes an improved yolov5 algorithm. More specifically, the mobileNetV3-Small architecture was introduced into the yolov5s's spindle network to reduce model complexity and increase training speed. At the same time, an efficient SE attention mechanism is introduced to improve the ability of the model to collect global information. Finally, SGD optimizer is used for model hyperparameter optimization to improve the efficiency of model training.

3.1 MobileNetV3-Small

The entire vintage enetv structure [9] essentially serves the design of the mobile enetv-v formation. In this way, a depth of light is determined that we can divide and separate. Although the server is still made up of several modules, each module has been improved and improved, including the structure of bottlenecks, arrivals, and NL. Vintage enetv3 improves the accuracy of orders related to ImageNet by 3.2% and reduces calculation by 20%.

Overall, MobileNetV3 has two major innovations: (1) the combination of complementary search technologies. (2) Improve the network structure. Table 1 shows the network structure of MobileNetV3-Small. The first column "input" represents the shape change for each functional layer in mobilenetV3. The second column, the operator, represents the block structure containing the individual feature layers. We can see that the functionality in MobileNetV3 is extracted through a number of Bneck structures. The third and fourth columns represent the number of channels after adding the reverse residual structure in the bneck, and the number of channels in the function layer after input in the bneck. The fifth column SE is about whether an awareness mechanism should be introduced at this level. The sixth column NL represents the type of activation function, HS represents h-swish, and RE represents RELU. The seventh column represents the step size for each block structure.

Table 1: MobileNetV3-Small network architecture

Input	*Operator*	*exp size*	*#out*	*SE*	*NL*	*s*
2242 x 3	*Conv2d,3x3*	*-*	*16*	*-*	*HS*	*2*
112²x 16	*Bneck, 3x3*	*16*	*16*	*√*	*RE*	*2*
56² x 16	*Bneck, 3x3*	*72*	*24*	*-*	*RE*	*2*
28² x 24	*Bneck, 3x3*	*88*	*24*	*-*	*RE*	*1*
28² x 24	*Bneck, 5x5*	*96*	*40*	*√*	*HS*	*2*
14² x 40	*Bneck, 5x5*	*240*	*40*	*√*	*HS*	*1*
14² x 40	*Bneck, 5x5*	*240*	*40*	*√*	*HS*	*1*
14² x 40	*Bneck, 5x5*	*120*	*48*	*√*	*HS*	*1*
14² x 48	*Bneck, 5x5*	*144*	*48*	*√*	*HS*	*1*
14² x 48	*Bneck, 5x5*	*288*	*96*	*√*	*HS*	*2*
7² x 96	*Bneck, 5x5*	*576*	*96*	*√*	*HS*	*1*
7² x 96	*Bneck, 5x5*	*576*	*96*	*√*	*HS*	*1*
7² x 96	*Conv2d 1x1*	*-*	*576*	*√*	*HS*	*1*
7² x 576	*Pool, 7x7*	*-*	*-*	*-*	*-*	*1*
7² x 576	*Conv2d 1x1 , NBN*	*-*	*1280*	*-*	*HS*	*1*
1² x 1280	*Conv2d 1x1 , NBN*	*-*	*k*	*-*	*-*	*1*

3.2 SE attention mechanism

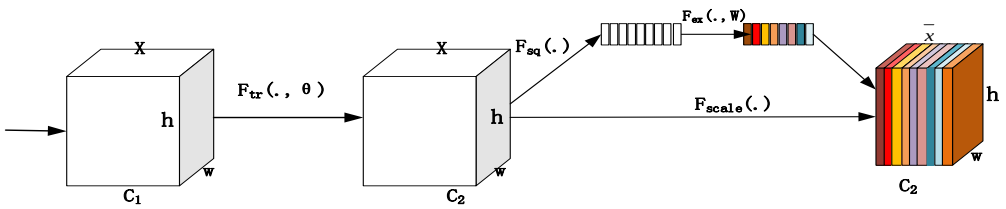

Figure 2: SE attention mechanism structure diagram

Figure 2 shows the structure of the SE [10] attention mechanism, including the Squeeze, Excitation, and Scale modules.

ISSN 2616-5775 Vol. 7, Issue 12: 36-43, DOI: 10.25236/AJCIS.2024.071205

(1) Squeeze: By global average pooling, the two-dimensional feature (H * W) compression of each channel is 1 real number, and the feature map is] ==> [1, 1, c from [h, w, c].

(2) Excitation: Generate a weight value for each feature channel. The correlation between channels is constructed through two fully connected layers. The number of weight values output is the same as the number of channels input to the feature map. [1, 1, c] ==> [1, 1, c].

(3) Scale: Weighting the normalized weights obtained earlier onto the features of each channel. The paper uses multiplication, multiplying channel by channel. [h, w, c] * [1,1,c] ==> [h, w, c].

SE automatically learns feature weights through the loss function loss of FCNN (fully connected neural network), just like copying the final loss of convolution to this place, roughly identifying the channel in advance, and assigning weights; making the weight of the valid feature channel significant.

3.3 Stochastic gradient descent

To improve performance, this article improves evaluation using random steps [11], improving model performance. The goal function for deep learning is usually an average function where the training data focuses on losses per sample.

$$f(x) = \frac{1}{n}\sum_{i=1}^{n} f_i(x) \tag{5}$$

where in the first sample corresponding to the loss function, the target loss function. For stochastic gradient descent, the gradient is calculated as:

$$\nabla f(x) = \frac{1}{m}\sum_{i=1}^{n} f_i(x) \tag{6}$$

Its computational complexity is 1, which can greatly reduce the computational complexity because it does not change significantly as it increases.

4. Experiments

4.1 Data set

The STL-10 dataset [12] is an image recognition dataset for the development of unsupervised, deep learning, and self-learning algorithms inspired by CIFAR-10 data. The STL-10 images come from ImageNet, a total of 113,000 96 x 96 resolution RGB images, of which 5,000 are for the training group and 8,000 are for the test group. The remaining 100,000 images were unlabeled and contained a total of 10 categories (planes, birds, cars, cats, dogs, horses, monkeys, boats, trucks), each containing 500 training samples and 800 test samples. In addition to the ten categories listed above, there are other unlabeled pictures of animals and vehicles.

Table 2: Experimental environment settings

Experimental environment	
Operating System	Windows11
CPU	11th Gen Intel(R) Core(TM) i7-11800H @ 2.30GHz 2.30 GHz
memory	32GB
Use language	Python
experimental platform	Pycharm
framework	Pytorch-CPU 3.80

Table 3: Experimental parameter configuration

Experimental parameters	
Data set STL-10	Data set STL-10
Number of cycles 300	Number of cycles 300
Batch size 64	Batch size 64
Input resolution 9696	Input resolution 9696
Optimizer SE	Optimizer SE
Loss function BCE Loss	Loss function BCE Loss
Learning rate 0.01	Learning rate 0.01

In this paper, model training and validation are performed using the publicly available STL-10

ISSN 2616-5775 Vol. 7, Issue 12: 36-43, DOI: 10.25236/AJCIS.2024.071205

dataset. The training sequence is divided into the following five categories: airplanes, birds, cars, ships and trucks in the main lane 5:4:4, ie. h. The training tape contains 2500 images, the test tape contains 2500 images, and the authentication device contains 25000 images. Enter image size 96px * 96px. The experimental environment settings and parameter settings in this paper are shown in Table.2 and Table.3.

4.2 Experimental results

This paper uses the following evaluation indicators to measure the performance of the network: Precision (P), loss curve (Loss Curve), average precision (mean Average Precision, mAP). The calculation formulas for each indicator are as follows:

$$P = \frac{TP}{TP+FP} \tag{7}$$

$$R = \frac{TP}{TP+FN} \tag{8}$$

$$mAP = \frac{1}{N}\sum_{i=0}^{n} i * \int_0^1 P * Rdr \tag{9}$$

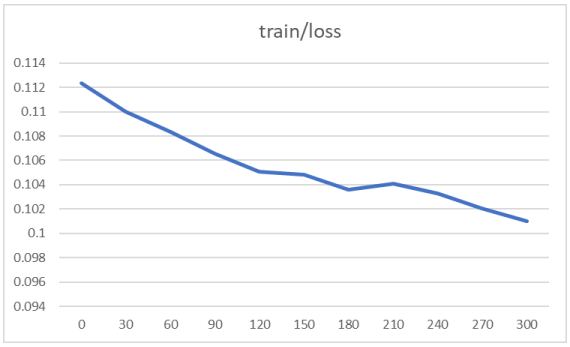

Figure 3: The loss curve of the training set

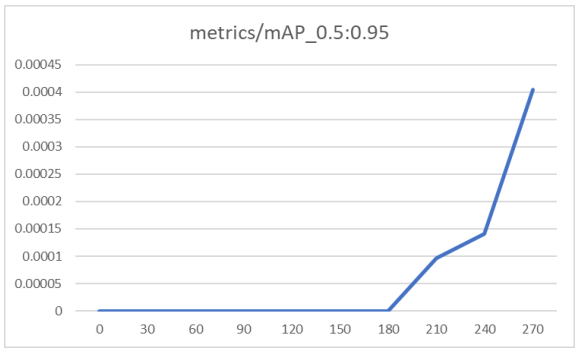

Figure 4: Evaluation index ratio change curve

As shown in Figure 3, with the increase of the number of iterations, the loss function value becomes smaller and smaller, the early downhole amplitude is larger, and gradually convergence to 0. As shown in Figure 4, with the increase of the number of iterations, the map function value becomes larger and larger, the early rise amplitude is larger, and gradually convergence to, indicating that the training process is reasonable and effective.

Table 4: Comparison of light quantization effects of different network architectures

	Paramers	GFLOPs	Training time	mAP	Layers
YOLOv5s	1605509	3.4	2.018	0.745	79
YOLOv5s with MobileNetV3-Small	430769	0.4	0.827	0.705	80
YOLOv5s with SE	1802885	3.6	2.394	0.726	84
YOLOv5s with MobileNetV3-Small with SE	628145	0.6	0.827	0.748	85

It can be seen from Table.4 that MobileNetV3-Small can greatly reduce the amount of model parameters and computation, and the YOLOv5s model complexity added to the SE module increases

ISSN 2616-5775 Vol. 7, Issue 12: 36-43, DOI: 10.25236/AJCIS.2024.071205

slightly, which meets the expected effect. The YOLOv5s model fused with MobileNetV3-Small [13] and SE [14] can effectively balance accuracy and light quantization, greatly reduce the amount of parameters and computation while ensuring accuracy and achieving the expected effect. Verify the effectiveness and applicability of the improvement strategy proposed in this paper.

To further validate the effectiveness of the optimization strategy, this paper conducts ablation experiments and records the detection accuracy and total detection accuracy results for each category in Table.5.

Table 5: Performance comparison of different optimizers for improved YOLOv5

mAP@0.5	airplane	bird	car	ship	truck	Total
Yolov5s	0.70	0.92	0.72	0.74	0.57	0.701
Yolov5s-se	0.76	0.86	0.67	0.68	0.52	0.731
YOlov5s-mobilenet-v3	0.76	0.85	0.68	0.76	0.63	0.737
YOlov5s-mobilenet-v3+se	0.77	0.87	0.68	0.78	0.64	0.748

As shown in the Table.5, the basic yolov5s algorithm obtains a total detection accuracy of 0.701. After the introduction of the SE attention mechanism, the total accuracy is improved by 0.03. The improvement is due to the introduction of the SE attention mechanism, which enables the model to pay more attention to and utilize important feature information more effectively. After the introduction of mobilenet-v3, the total accuracy is improved by 0.036. The improvement is mainly due to its stronger feature extraction ability, optimized network architecture design, more efficient model training, and better adaptability to object detection tasks. Finally, after SE attention mechanism and mobilenet, the overall accuracy is improved by 0.047. The reason for the improvement is that they jointly enhance the ability of the model to extract and utilize features, making the network perform better in object detection tasks. The above analysis results show that compared with the basic yolov5s algorithm, the accuracy is improved by how much, and the superiority of the model is verified.

5. Conclusion

This paper proposes a new improved YOLOv5s algorithm that balances light quantization with performance. Aiming at the problem of high complexity and low accuracy of YOLOv5s model, the original C2F module is replaced by MobileNetV3-Small. In order to further improve the accuracy reduction caused by the decrease of parameter size, this paper adds SE attention mechanism to obtain global information, learn the correlation between different scale features and fuse them, enhance the semantic information of features, and use SGD as optimizer to further improve the accuracy. Experiments on the STL-10 data set show that the introduction of MobileNetV3-Small and SE effectively reduces the number of parameters of the model, the training time is greatly reduced, and the accuracy is considerable. Compared with other attention mechanisms, SE attention mechanism has the greatest improvement in performance, and has excellent performance in light quantization and algorithm performance balance. The effectiveness of the optimization strategy has been verified. The improved YOLOv5s algorithm has the characteristics of lightweight, high performance, high efficiency and wide applicability, which can meet the application scenarios that require object detection accuracy, real-time performance and resource consumption.

References

[1] 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), Las Vegas, NV, USA, 2016.

[2] Redmon J, Farhadi A. YOLO9000: Better, faster, stronger[C]. 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, 2017.

[3] Redmon J, Farhadi A. YOLOv3: An incremental improvement[J]. Computer Vision and Pattern Recognition, 2018, 3: 121-126.

[4] Bochkovskiy A, Wang CY, Liao HY. YOLOv4: Optimal speed and accuracy of object detection[J]. Cornell Universit, 2020, 3(8): 11-16.

[5] Liu S, Zhou X, Wang Y. Research on ship recognition algorithm based on improved YOLOv5[J]. Information Technology and Informatization, 2023, (08): 188-193+198.

[6] Robbins H, Monro S. A stochastic approximation method[J]. Annals of Mathematical Statistics, 1951, 22(3): 400-407.

ISSN 2616-5775 Vol. 7, Issue 12: 36-43, DOI: 10.25236/AJCIS.2024.071205

[7] Li Z, Zhao Y. Face detection with improved Adam optimization algorithm[J]. Journal of Taiyuan Normal University (Natural Science Edition), 2022, 21(04): 58-63.
[8] 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, Columbus, OH, USA, 2014.
[9] Carion N, Massa F, Synnaeve G, et al. End-to-end object detection with transformers[C]. Computer Vision-ECCV 2020, Lecture Notes in Computer Science, 2020: 213-229.
[10] Qin X, Zhang Z, Huang C, et al. U2-Net: Going deeper with nested U-structure for salient object detection [J]. Pattern Recognition, 2020: 107404.
[11] Han X. Quantum multi-class classification support vector machine optimized for stochastic gradient descent [J]. Fujian Computer, 2024, (4): 1-6.
[12] Zeng Q L, Zhou G Y, Wan L R, et al. Detection of coal and gangue based on improved YOLOv8[J]. Sensors, 2024, 24(4): 1246.
[13] Howard A, Sandler M, Chu G, et al. Searching for MobileNetV3[J]. Computer Vision and Pattern Recognition, 2019, 45(6): 589-594.
[14] Lin S, Liu M, Tao Z. Underwater treasure detection using attention mechanism and improved YOLOv5[J]. Chinese Journal of Agricultural Engineering, 2021, (18): 307-314.