symmetry

MDPI

Article

Improved YOLOv4 Marine Target Detection Combined with CBAM

Huixuan Fu, Guoqing Song and Yuchao Wang *

College of Intelligent Systems Science and Engineering, Harbin Engineering University, Harbin 150001, China; fuhuixuan@hrbeu.edu.cn (H.F.); sgq@hrbeu.edu.cn (G.S.)
* Correspondence: wangyuchao@hrbeu.edu.cn

Abstract: Marine target detection technology plays an important role in sea surface monitoring, sea area management, ship collision avoidance, and other fields. Traditional marine target detection algorithms cannot meet the requirements of accuracy and speed. This article uses the advantages of deep learning in big data feature learning to propose the YOLOv4 marine target detection method fused with a convolutional attention module. Marine target detection datasets were collected and produced and marine targets were divided into ten categories, including speedboat, warship, passenger ship, cargo ship, sailboat, tugboat, and kayak. Aiming at the problem of insufficient detection accuracy of YOLOv4's self-built marine target dataset, a convolutional attention module is added to the YOLOv4 network to increase the weight of useful features while suppressing the weight of invalid features to improve detection accuracy. The experimental results show that the improved YOLOv4 has higher detection accuracy than the original YOLOv4, and has better detection results for small targets, multiple targets, and overlapping targets. The detection speed meets the real-time requirements, verifying the effectiveness of the improved algorithm.

Keywords: deep learning; marine targets; YOLOv4; CBAM; target detection

Citation: Fu, H.; Song, G.; Wang, Y. Improved YOLOv4 Marine Target Detection Combined with CBAM. *Symmetry* **2021**, *13*, 623. https://doi.org/10.3390/sym13040623

Academic Editor: Calogero Vetro

Received: 18 March 2021
Accepted: 2 April 2021
Published: 8 April 2021

1. Introduction

As an important carrier of marine resource development and economic activities, accurate monitoring of marine targets has become increasingly important. Carrying out automated research on marine target monitoring is of great significance for strengthening the management of sea areas, and whether illegal targets can be detected and located in time is the focus of marine target monitoring.

Among the traditional marine target detection algorithms, Fefilatyev et al. used a camera system mounted on a buoy to quickly detect ship targets, and proposed a new type of ocean surveillance algorithm for deep-sea visualization [1]. In the context of ship detection, a new horizon detection scheme for complex sea areas has been developed. Shi W. et al. [2] effectively suppressed the noise of the background image and detected the ship target by combining morphological filtering with multi-structural elements and improved median filtering. The influence of sea clutter on the ship target detection was eliminated by using connected domain calculation. Chen Z et al. proposed a ship target detection algorithm for marine video surveillance [3], the purpose is to reduce the impact of clutter in the background and improve the accuracy of ship target detection. In the proposed detector, the main steps of background modeling, model training update, and foreground segmentation are all based on the Gaussian Mixture Model (GMM). This algorithm not only improves the accuracy of the target, but also greatly reduces the probability of false alarms and reduces the impact of dynamic scene changes. Although traditional methods have achieved good results, in the face of complex and changeable sea environments with a lot of noise interference, traditional detection algorithms have problems such as low detection accuracy and poor robustness. Therefore, traditional methods have great limitations in practical applications.

In recent years, deep learning technology [4] has been increasingly used in target detection tasks and video surveillance [5], autonomous driving [6], face recognition [7], etc. The target detection algorithm based on deep learning relies on big data to obtain better detection results than traditional methods, and has stronger robustness.

Target detection algorithms are mainly divided into three categories, which are target detection based on region proposals, target detection based on regression, and target detection based on anchor-free. Based on region proposals, it is also called two-stage target detection, and regression-based is called one-stage target detection. Two-stage detection algorithms mainly include SPP-Net [8], Fast R-CNN [9], Faster R-CNN [10], and R-FCN [11]. The one-stage target detection algorithm does not require the step of generating region proposals. It can directly obtain the position and category information of the target, and the detection speed is greatly improved. One-stage detection algorithms mainly include SSD [12], YOLO [13], YOLOv2 [14], YOLOv3 [15], and YOLOv4 [16]. Algorithms based on anchor-free target detection mainly include CornerNet [17] and CenterNet [18], and no longer use a priori box.

Many researchers have applied deep learning technology to marine target detection. Gu D proposed a maritime ship recognition algorithm based on Faster-R-CNN [19], which added a dropout layer after the first full connection layer of the regional generation network to improve the accuracy. Qi L et al. proposed an improved target detection algorithm based on Faster R-CNN [20], which enhanced the useful information of ship images through image down-sampling, and used scene narrowing technology to improve detection speed. Zou Y et al. proposed an improved SSD algorithm based on the MobilenetV2 convolutional neural network for target detection and recognition in ship images [21]. In order to improve the accuracy and robustness of ship target detection, Huang H et al. proposed an improved target detection algorithm based on YOLOv3 [22]. First, guided filtering and gray enhancement were used to preprocess the input image, and then k-means ++ clustering was used to get the prior box, and finally reduced feature redundancy by reducing part of the convolution operation and adding a jump connection mechanism, and the accuracy of ship identification was improved under the premise of ensuring real-time performance. Chen X et al. proposed a framework based on integrated YOLO to detect ships from ocean images [23]. This method can accurately detect ships and successfully identify ships' historical behavior. Huang Z et al. proposed an improved YOLOv3 network [24] for intelligent detection and classification of ship images/videos.

This paper provides a method for improving the detection accuracy of small, multiple, and overlapping marine targets. The contributions of this paper are described as follows. Aiming at the problem of insufficient detection accuracy of YOLOv4 in the self-built marine target data set, an improved YOLOv4 algorithm combined with the convolutional block attention module (CBAM) network is proposed. By adding a CBAM module to each of the three branches at the end of the feature fusion network, the weights of the channel features, and spatial features of the feature map are allocated to increase the weight of useful features while suppressing the weight of invalid features to improve the accuracy of marine target detection.

2. YOLOv4 Target Detection Algorithm

YOLOv4 introduces path aggregation network (PANet), spatial pyramid pooling (SPP), mosaic data enhancement, Mish activation function, self-adversarial training, CmBN, and many other techniques to greatly improve the detection accuracy. The backbone network uses CSPDarknet53, which integrates the Cross Stage Partial Network (CSPNet), reduces the amount of calculation while maintaining accuracy, achieving a perfect combination of speed and accuracy. Figure 1 shows the YOLOv4 network structure.

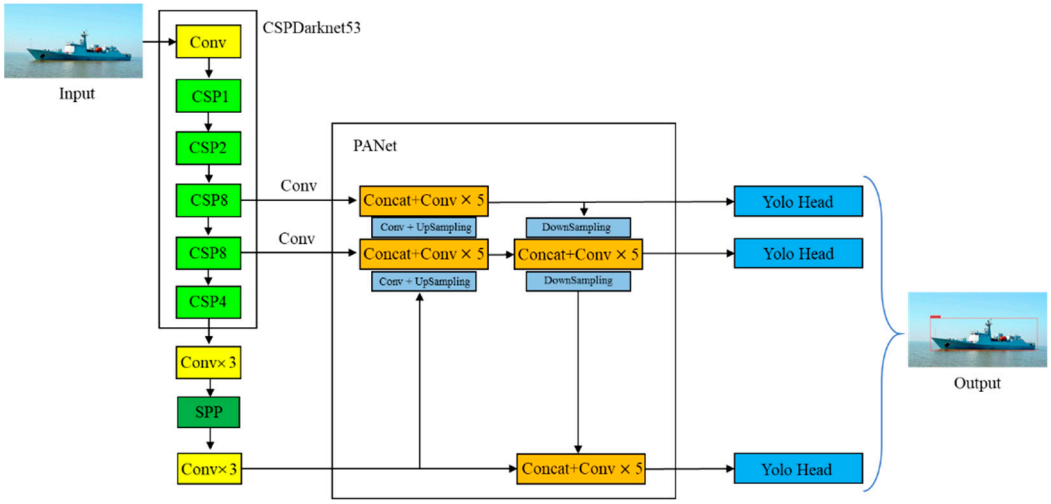

Figure 1. YOLOv4 network structure.

In Figure 1, the input image is sent to the backbone network to complete feature extraction, then through SPP and PANet to complete the fusion of feature maps of different scales, and, finally, output feature maps of three scales to predict the bounding box, category, and confidence, and the head of YOLOv4 is consistent with YOLOv3.

2.1. Feature Extraction Network

YOLOv4 uses a new backbone network CSPDarknet-53 for feature extraction of the input data. CSPDarknet-53 is an improvement of Darknet-53. Darknet-53 is composed of 5 large residual modules, each of residual module is separate corresponding to 1, 2, 8, 8, and 4 small residual units, the residual module solves the problem of gradient disappearance caused by the continuous deepening of the network, greatly reduces network parameters, and makes it easier to train deeper convolutional neural networks. The network structure is shown in Figure 2.

	Type	Filters	Size	Output
	Convolutional	32	3×3	416×416
	Convolutional	64	3×3/2	208×208
	CrossStagePartial			
1 ×	Convolutional	32	1×1	
	Convolutional	64	3×3	
	Residual			208×208
	Convolutional	128	3×3/2	104×104
	CrossStagePartial			
2 ×	Convolutional	64	1×1	
	Convolutional	64	3×3	
	Residual			104×104
	Convolutional	256	3×3/2	52×52
	CrossStagePartial			
8 ×	Convolutional	128	1×1	
	Convolutional	128	3×3	
	Residual			52×52
	Convolutional	512	3×3/2	26×26
	CrossStagePartial			
8 ×	Convolutional	256	1×1	
	Convolutional	256	3×3	
	Residual			26×26
	Convolutional	1024	3×3/2	13×13
	CrossStagePartial			
4 ×	Convolutional	512	1×1	
	Convolutional	512	3×3	
	Residual			13×13

Figure 2. CSPDarknet-53 network structure.

In Figure 2, the Convolution layer consists of a convolutional layer, a batch normalization layer, and a Mish activation function. Cross Stage Partial is a newly added cross-stage local network. The residual layer is a small residual unit. CSPDarknet-53 also removes the pooling layer and the fully connected layer, greatly reducing parameters and improving calculation speed. During training, the image is stretched and scaled to a size of 416×416, then sent it to the convolutional neural network. After 5 times of $3 \times 3/2$ convolution (convolution kernel size is 3×3, step size is 2), the size is reduced to 13×13, and three scales of 52×52, 26×26, 13×13 are selected as the size of the output feature map. Different sizes of feature maps are used to detect different sizes of targets. Small feature maps detect large targets, and large output feature maps detect small targets.

The CSP module in CSPDarknet-53 solves the problem of increased calculation and slower network speed due to redundant gradient information generated in the network deepening process. The structure of the CSP module is shown in Figure 3, and the structure of the small residual unit is shown in Figure 4.

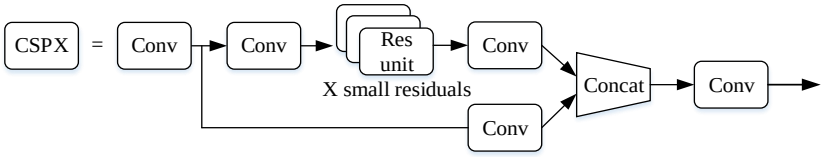

Figure 3. Cross stage partial (CSP) module structure.

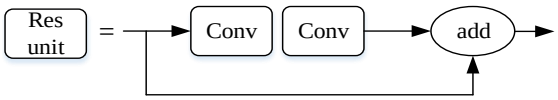

Figure 4. Small residual unit structure.

In Figure 3, the feature map of the base layer is divided into two parts, and then they are merged through a cross-stage hierarchical structure. The CSP module can achieve a richer gradient combination, greatly reduce the amount of calculation, and improve the speed and accuracy of inference.

In Figure 4, there are more shortcuts in the small residual unit than in the normal structure. The shortcut connection is equivalent to directly transferring the input features to the output for identity mapping, adding the input of the previous layer and the output of the current layer.

2.2. Feature Fusion Network

After the feature extraction network extracts the relevant features, the feature fusion network is required to fuse the extracted features to improve the detection ability of the model. The YOLOv4 feature fusion network includes PAN and SPP. The function of the SPP module is to make the input of the convolutional neural network not restricted by a fixed size, and it can increase the receptive field and effectively separate important context features, while not reducing the running speed of the network. The SPP module is located after the feature extraction network CSPDarknet-53, and the SPP network structure is shown in Figure 5.

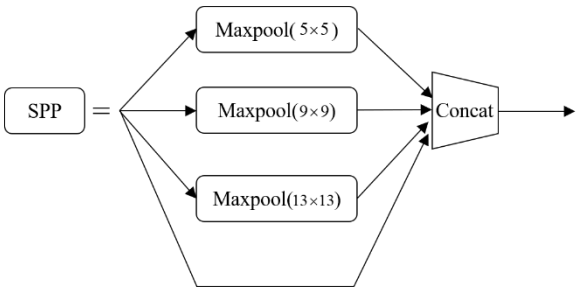

Figure 5. Spatial pyramid pooling (SPP) network structure.

In Figure 5, the SPP network uses four different scales of maximum pooling to process the input feature maps. The pooling core sizes are 1×1, 5×5, 9×9, 13×13, and 1×1 is equivalent to without processing, the four feature maps are subjected to a concat operation. The maximum pooling adopts padding operation, the moving step is 1, and the size of the feature map does not change after the pooling layer.

After SPP, YOLOv4 uses PANet instead of the feature pyramid in YOLOv3 as the method of parameter aggregation. PANet adds a bottom-up path augmentation structure after the top-down feature pyramid, which contains two PAN structures. And the PAN structure is modified. The original PAN structure uses a shortcut connection to fuse the down-sampled feature map with the deep feature map, and the number of channels of the output feature map remains unchanged. The modified PAN uses the concat operation to connect the two input feature maps, and merge the channel numbers of the two feature maps. The top-down feature pyramid structure conveys strong semantic features, and the bottom-up path augmentation structure makes full use of shallow features to convey strong positioning features. PANet can make full use of shallow features, and for different detector levels, feature fusion of different backbone layers to further improve feature extraction capabilities and improve detector performance.

2.3. Predictive Network

YOLOv4 outputs three feature maps of different scales to predict the bounding box position information, corresponding category, and confidence of the target. YOLOv4 continues the basic idea of YOLOv3 bounding box prediction and adopts a prediction scheme based on priori box. YOLOv4 bounding box prediction is shown in Figure 6.

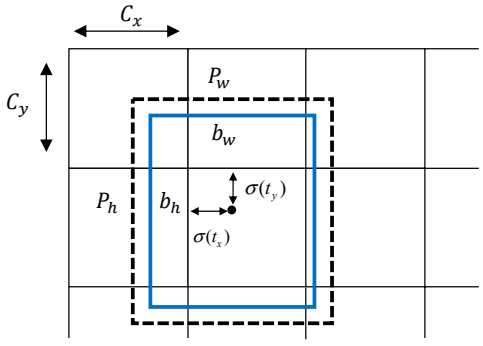

Figure 6. YOLOv4 bounding box prediction.

In Figure 6, (c_x, c_y) are the coordinates of the upper left corner of the grid cell where the target center point is located, (p_w, p_h) is the width and height of the priori box, $(b_w,\ b_h)$ is the width and height of the actual prediction box, and $(\sigma(t_x), \sigma(t_y))$ is the offset value predicted by the convolutional neural network. The position information of the bounding box is calculated by Formulas (1)–(5), where t_w, t_h is also predicted by the convolutional network, and $(b_x,\ b_y)$ is the coordinates of the center point of the actual prediction box. In

the obtained feature map, the length and width of each grid cell are 1, so $(c_x, c_y) = (1, 1)$ in Figure 6, and the sigmoid function is used to limit the predicted offset to between 0 and 1.

$$b_x = \sigma(t_x) + c_x \tag{1}$$

$$b_y = \sigma(t_y) + c_y \tag{2}$$

$$b_w = p_w e^{t_w} \tag{3}$$

$$b_h = p_h e^{t_h} \tag{4}$$

$$\sigma(x) = \frac{1}{1 + e^{-x}} \tag{5}$$

The loss function of YOLOv4 includes regression, confidence, and classification loss functions. Among them, the bounding box regression loss function uses CIOU to replace the mean square error loss, which makes the boundary regression faster and more accurate [25]. By minimizing the loss function between the predicted box and the real box, the network is trained and the weights are constantly updated. Confidence and classification loss still use cross entropy loss.

3. CBAM-Based YOLOv4 Network Structure Improvement

The attention mechanism generates a mask through the neural network, and the values in the mask represent the attention weights of different locations. Common attention mechanisms mainly include channel attention mechanism, spatial attention mechanism, and mixed domain attention mechanism. The channel attention mechanism is to generate a mask for the channel of the input feature map, and different channels have corresponding attention weights to achieve channel-level distinction; the spatial attention mechanism is to generate a mask on the spatial position of the input feature map, and different spatial regions have corresponding weights to realize the distinction of spatial regions; the hybrid attention mechanism is to introduce the channel attention mechanism and the spatial attention mechanism at the same time. In this paper, the mixed attention CBAM module is introduced to make the neural network pay more attention to the target area containing important information [26], suppress irrelevant information, and improve the overall accuracy of target detection.

CBAM is a high-efficiency, lightweight attention module that can be integrated into any convolutional neural network architecture, and can be trained end-to-end with the basic network. The CBAM module structure is shown in Figure 7.

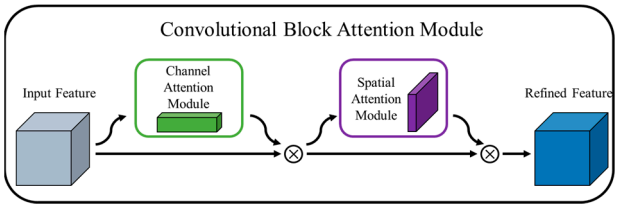

Figure 7. Convolutional block attention module (CBAM) module structure.

In Figure 7, the CBAM module is divided into a channel attention module and a spatial attention module. First, input the feature map into the channel attention module, output the corresponding attention map, then multiply the input feature map with the attention map, the output passes through the spatial attention module, and performs the same operation, and, finally, get the output feature map, the mathematical expression of which is as follows:

$$\begin{aligned} F' &= M_c(F) \otimes F \\ F'' &= M_s(F') \otimes F' \end{aligned} \tag{6}$$

where $\otimes$ represents element-wise multiplication, F is the input feature map, $M_c(F)$ is the channel attention map output by the channel attention module, $M_s(F')$ is the spatial attention map output by the spatial attention module, and F'' is the feature map output by the CBAM.

3.1. Channel Attention Module

Each channel of the feature map represents a feature detector. Therefore, channel attention is used to focus on what features are meaningful. The structure of the channel attention module is shown in Figure 8.

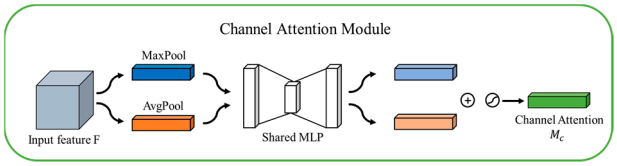

Figure 8. Channel attention module structure.

In Figure 8, the input feature map F is first subjected to global maximum pooling and global average pooling based on width and height, and then a multi-layer perceptron (MLP) with shared weights is passed in. The MLP contains a hidden layer, which is equivalent to two fully connected layers. The two outputs of the MLP are added pixel by pixel, finally, the channel attention map is obtained through the Sigmoid activation function. Its mathematical expression is:

$$\begin{aligned} M_c(F) &= \sigma(MLP(AvgPool(F)) + MLP(MaxPool(F))) \\ &= \sigma(W_1(W_0(F^c_{avg})) + W_1(W_0(F^c_{\max}))) \end{aligned} \tag{7}$$

where σ is the Sigmoid activation function, W_0 and W_1 are the weights of MLP, $W_0 \in R^{C/r \times C}$, $W_1 \in R^{C \times C/r}$, r is the dimensionality reduction factor, and $r = 16$ in this paper.

3.2. Spatial Attention Module

After the channel attention module, the spatial attention module is used to focus on where the meaningful features come from. The structure of the spatial attention module is shown in Figure 9.

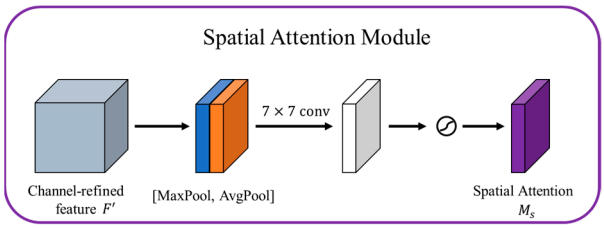

Figure 9. Spatial attention module structure.

In Figure 9, the spatial attention module takes F' as the input feature map, and, respectively, through channel-based global maximum pooling and global average pooling, then merges the two feature maps F^s_{avg} and $F^s_{\max}$ to obtain a feature map with a channel number of 2, it passes a 7×7 convolutional layer to reduce the channel number to 1, and finally gets a spatial attention map $M_s(F)$ through a Sigmoid activation function. Its mathematical expression is as follows:

$$\begin{aligned} M_s(F) &= \sigma(f^{7\times7}([AvgPool(F);\ MaxPool(F)])) \\ &= \sigma(f^{7\times7}([F^s_{avg};\ F^s_{\max}])) \end{aligned} \tag{8}$$

where 7×7 represents the size of the convolution kernel.

3.3. Improved YOLOv4 Algorithm

This paper adds a CBAM module to each of the three branches at the end of the YOLOv4 feature fusion network. Aiming at the characteristics of denseness, mutual occlusion, and multiple small targets of marine targets. By integrating the CBAM module, the weights of the channel features and spatial features of the feature map are assigned to increase the weights of useful features while suppressing the weights of invalid features, paying more attention to target regions containing important information, suppressing irrelevant information, and improving the overall accuracy of target detection. The improved network structure is shown in Figure 10.

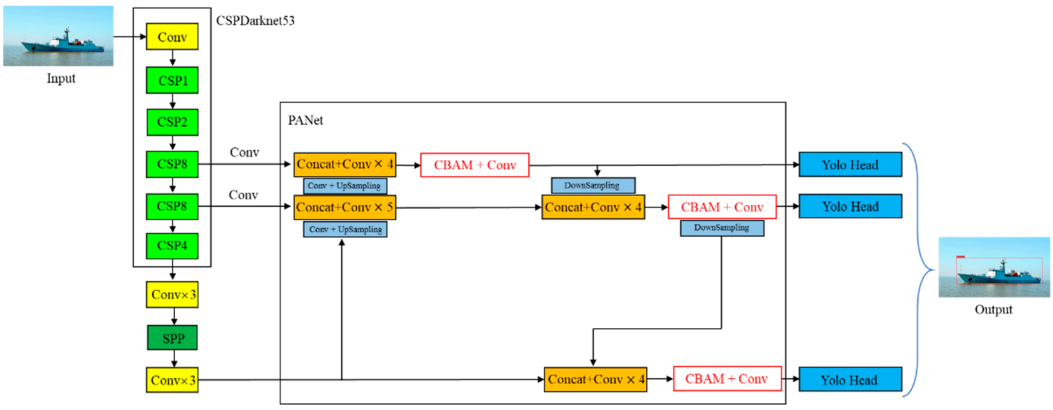

Figure 10. Improved YOLOv4 network structure combined with CBAM.

In Figure 10, assuming that the input image size is $416 \times 416 \times 3$, take the first CBAM module as an example, the input feature map size is $52 \times 52 \times 256$, after global maximum pooling and global average pooling, gets two feature maps of size $1 \times 1 \times 256$ and $1 \times 1 \times 256$, and then passes through a multi-layer perceptron with shared weights. The dimensionality reduces to $1 \times 1 \times 16$, the dimensionality reduction coefficient is 16, and then increases the dimensionality to $1 \times 1 \times 256$, adds the operation to the two feature maps, and gets the channel attention map with a size of $1 \times 1 \times 256$ through the Sigmoid activation function, multiplies the input feature map and the attention map to get the output of $52 \times 52 \times 256$ size. Next, the spatial attention module is entered, through channel-based global maximum pooling and global average pooling respectively. Two feature maps of $52 \times 52 \times 1$ size are obtained, and the number of channels of the two feature maps is combined to obtain a feature map of size $52 \times 52 \times 2$, and then a 7×7 convolution is used to reduce the number of channels to 1. Finally, the Sigmoid activation function is used to obtain a spatial attention map of size $52 \times 52 \times 1$, the input of the spatial attention module and the spatial attention map are multiplied to obtain an output feature map of size $52 \times 52 \times 256$. The output feature map of the CBAM module is consistent with the input feature map.

4. Network Training and Experimental Results

4.1. Marine Target Data Set

The target detection data set in this article is 3000 images of marine targets collected from the Internet. The data format is JPG. The marine targets are divided into 10 categories, namely speedboat, warship, passenger ship, cargo ship, sailboat, tugboat, kayak, boat, fighter plane, and buoy.

LabelImg is used to label the obtained data. The labeled data is divided into the training set, validation set, and test set according to the ratio of 5:1:1. The format of the data set is produced according to the VOC data set. Part of the marine target data is shown in Figure 11.

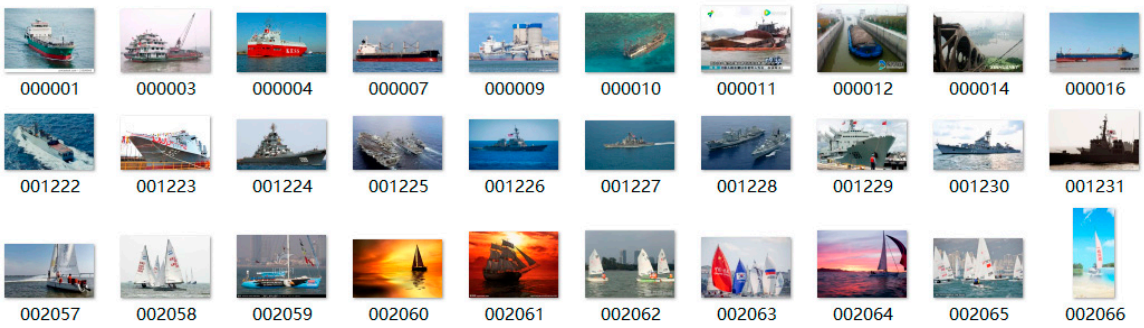

Figure 11. Part of the marine target dataset.

In order to further increase the generalization ability of the model and increase the diversity of samples, offline data augmentation is performed by mirroring, brightness adjustment, contrast, random cropping, etc., and then the enhanced data is added to the training dataset to complete the data expansion, and, finally, get 10,000 pictures.

4.2. Experimental Environment and Configuration

In order to further accelerate the network training speed, this experiment introduces transfer learning technology, loads the pre-trained model on the COCO dataset, and then trains the marine target dataset. The hardware environment and software version of the experiment are shown in Table 1.

Table 1. Hardware environment and software version.

Hardware and Software	Configuration
Desktop computer	Operating System: Ubuntu18.04
	CPU: Intel i7-7700
	RAM: 15.5 G
	GPU: NVIDIA GTX 1080Ti (11G Video memory)
Software version	Pycharm2020 + CMake3.10.2 + Python 3.6.8 + CUDNN7.4.2 + Opencv3.4.3 + CUDA10.0.130

The parameters of the training network are shown in Table 2.

Table 2. Training network parameters.

Parameter	Value
Batch size	16
Momentum parameter	0.9
Learning rate	0.001
Number of iterations	20000
Learning rate decay	0.1
Fixed image size	416 × 416

Among them, the learning rate decay of 0.1 means that the learning rate is reduced to one-tenth of the original after a certain number of iterations. In this experiment, the learning rate was changed at 16,000 iterations and 18,000 iterations.

The convergence curve of the loss function obtained after training the network is shown in Figure 12.

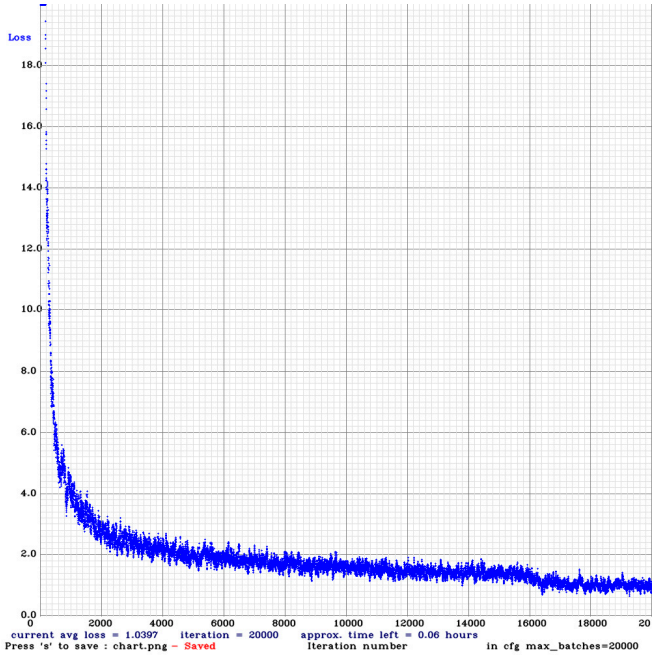

Figure 12. Improved YOLOv4 training loss function curve.

It can be seen from Figure 12 that due to the loading of the pre-trained model, the loss value can be reduced to a lower value in a few iterations. It can also be seen that the loss value maintains a downward trend until it finally converges. The loss value drops to a relatively small value when the number of iterations is 4000, reaches a relatively stable level when iteration 18,000, and, finally, drops to 1.0397. The overall effect of training is ideal.

4.3. Marine Target Detection Performance Comparison

In order to verify the effectiveness of the improved YOLOv4 network, a comparative experiment was conducted between the original YOLOv4 training model and the improved YOLOv4 network model. The original YOLOv4 training parameters are consistent with the improved YOLOv4 training parameters. The commonly used target detection evaluation mAP is used to compare the model before and after the improvement.

In the target detection task, according to the intersection over union (IOU) to determine whether the target is successfully detected, the ratio of the intersection and union between the prediction box and ground truth box of the model is IOU. For a certain type of target in the dataset, assuming the threshold is α, when the IOU of the prediction box and ground truth box is greater than α, it means that the model prediction is correct; when the IOU of the prediction box and ground truth box is less than α, it means that the model predicts incorrectly. The confusion matrix is shown in Table 3 below.

Table 3. Confusion matrix.

Real \ Prediction	Positive	Negative
True	TP	FN
False	FP	TN

In Table 3, TP represents the number of positive samples correctly predicted, FP represents the number of negative samples incorrectly predicted, FN is the number of positive samples incorrectly predicted, and TN represents the number of negative samples correctly predicted. The calculation formulas for precision rate and recall are as follows:

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

$$\text{Re}call = \frac{TP}{TP + FN} \tag{10}$$

AP value is usually used as the evaluation index of target detection performance. AP value is the area under the P-R curve, in which recall is taken as X-axis and precision as Y-axis. AP represents the accuracy of the model in a certain category. mAP represents the average the accuracy of all categories and can measure the performance of the network model in all categories. mAP50 represents the mAP value where the IOU of the prediction box and ground truth box is greater than 0.5, and mAP75 represents the mAP value where the IOU threshold is greater than 0.75. The calculation formula of mAP is as follows:

$$mAP = \frac{1}{N}\sum_{i=1}^{N}\int_{0}^{1} PdR \tag{11}$$

where N represents the number of detected categories.

FPS (Frame Per Second) is used to evaluate the detection speed of the algorithm. It represents the number of frames that can be processed per second. Models have different processing speeds under different hardware configurations. Therefore, this article uses the same hardware environment when comparing detection speed.

The data of the test set is sent to the trained target detection model, and different thresholds are selected for experimental comparison. The model comparison before and after the improvement of YOLOv4 is shown in Table 4.

Table 4. Comparison of original and improved YOLOv4 results.

Algorithm \ Evaluation	$\mathbf{mAP_{50}}$	$\mathbf{mAP_{75}}$	**FPS**	**Model Volume**
YOLOv4	82.04%	66%	53.6	256.2 MB
Improved YOLOv4	84.06%	67.85%	50.4	262.4 MB

It can be seen from Table 4 that both map50 and map75 of the YOLOv4 combined with the CBAM algorithm are improved, in which mAP50 is increased by 2.02%, and mAP75 is increased by 1.85%. Because of the addition of three CBAM modules, the volume of the model becomes larger, from 256.2 MB to 262.4 MB, resulting in a slight decrease in FPS value from 53.6 to 50.4, but the speed still meets the real-time requirements. Figure 12 shows the test results before and after the improvement.

In Figure 13, the first column is the input pictures, the second column is the YOLOv4 detection results, and the third column is the improved YOLOv4 detection results. In the first row, YOLOv4 mis-detected the tug as a warship, and the improved YOLOv4 successfully detected it as a tug; from the second, fifth, sixth, and seventh rows, it can be seen that the improved YOLOv4 detects small targets more accurately than the original algorithm, and more small targets are detected. Among them, the fifth row of ship targets has more background environment interference. Improved YOLOv4 has stronger robustness and detects more targets. In the seventh row, the improved YOLOv4 successfully detects the small target of the occluded cargo ship; from the third and fourth rows, it can be seen that the improved YOLOv4 can detect more mutually occluded targets when the ship targets are dense and mutually occluded. The last line shows that when there is interference in the background, the original algorithm detection box has a position shift, and the improved algorithm detection box is more accurate.

(**a**) (**b**) (**c**)

Figure 13. Comparison of YOLOv4 and improved YOLOv4 combined with CBAM algorithm marine target detection results. (**a**) Original picture; (**b**) YOLOv4 detection results; (**c**) Improved YOLOv4 detection results.

For dense targets, mutual occlusion targets, and small targets, the improved YOLOv4 network can effectively detect and reduce the missed detection rate. In addition, the original YOLOv4 network will cause false detections when detecting targets. The improved YOLOv4 network improves the target false detection situation. According to the experimental results, the improved YOLOv4 combined with the CBAM target detection algorithm proposed in this paper is more effective, improves the accuracy of target detection, can basically meet the needs of marine target detection tasks, and has practical application value.

5. Conclusions

Marine target detection technology is of great significance in the fields of sea surface monitoring, sea area management, ship collision avoidance, etc. and is focused on the problem of insufficient detection accuracy of YOLOv4 in the self-built ship dataset. On the basis of YOLOv4, the CBAM attention module is added to make the neural network pay more attention to the target area that contains important information, suppress irrelevant information, and improve detection accuracy. Experimental results show that the improved YOLOv4 model has higher accuracy in target detection tasks than the original YOLOv4, mAP50 is increased by 2.02%, mAP75 is increased by 1.85%, and the detection speed meets the real-time requirements, which verifies the effectiveness of the improved algorithm. It provides a theoretical reference for further practical applications.

Author Contributions: Conceptualization, H.F. and Y.W.; methodology, H.F. and G.S.; software, G.S. and Y.W.; validation, H.F. and G.S.; writing—original draft preparation, G.S.; writing—review and editing, H.F. and G.S. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Natural Science Foundation of China, grant number 52071112; Fundamental Research Funds for the Central Universities, grant number 3072020CF0408.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

YOLO v4	You Only Look Once (version 4)
CBAM	Convolutional Block Attention Module
GMM	Gaussian Mixture Model
SPP	Spatial Pyramid Pooling
CmBN	Cross mini-Batch Normalization
Fast R-CNN	Fast Region Convolutional Neural Networks
Faster R-CNN	Faster Region Convolutional Neural Networks
R-FCN	Region based Fully Convolutional Network
SSD	Single Shot MultiBox Detector
PANet	Path Aggregation Network
CSPDarknet53	Cross Stage Partial Darknet53
CSPNet	Cross Stage Partial Network
IOU	Intersection Over Union
CIOU	Complete Intersection Over Union
MLP	Multi-Layer Perceptron
VOC	Visual Object Classes
COCO	Common Objects in Context
AP	Average Precision
mAP	mean Average Precision
TP	True Positive

FP	False Positive
TN	Ture Negative
FN	False Negative
P-R curve	Precision-Recall curve
FPS	Frame Per Second

References

1. Fefilatyev, S.; Goldgof, D.; Shreve, M.; Lembke, C. Detection and tracking of ships in open sea with rapidly moving buoy-mounted camera system. *Ocean Eng.* **2012**, *54*, 1–12. [CrossRef]
2. Shi, W.; An, B. Port ship detection method based on multi-structure morphology. *Comput. Syst. Appl.* **2016**, *25*, 283–287.
3. Chen, Z.; Yang, J.; Chen, Z.; Kang, Z. Ship Target Detection Algorithm for Maritime Surveillance Video Based on Gaussian Mixture Model. In Proceedings of the IOP Conference, Bangkok, Thailand, 27–29 July 2018.
4. LeCun, Y.; Bengio, Y.; Hinton, G. Deep learning. *Nature* **2015**, *521*, 436–444. [CrossRef] [PubMed]
5. Chen, J.; Li, K.; Deng, Q.; Li, K.; Yu, P.S. Distributed deep learning model for intelligent video surveillance systems with edge computing. *IEEE Trans. Ind. Inform.* **2019**. [CrossRef]
6. Xu, Y.; Wang, H.; Liu, X.; He, H.R.; Gu, Q.; Sun, W. Learning to See the Hidden Part of the Vehicle in the Autopilot Scene. *Electronics* **2019**, *8*, 331. [CrossRef]
7. Goswami, G.; Ratha, N.; Agarwal, A.; Singh, R.; Vatsa, M. Unravelling Robustness of Deep Learning based Face Recognition Against Adversarial Attacks. *arXiv* **2018**, arXiv:1803.00401.
8. Purkait, P.; Zhao, C.; Zach, C. SPP-Net: Deep absolute pose regression with synthetic views. *arXiv* **2017**, arXiv:1712.03452.
9. Girshick, R. Fast R-CNN. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 11–18 December 2015; pp. 1440–1448.
10. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster r-cnn: Towards real-time object detection with region proposal networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **2016**, *39*, 1137–1149. [CrossRef] [PubMed]
11. Dai, J.; Li, Y.; He, K.; Sun, J. R-fcn: Object detection via region-based fully convolutional networks. *arXiv* **2016**, arXiv:1605.06409.
12. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.; Berg, A.C. Ssd: Single Shot Multibox Detector. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 8–16 October 2016; pp. 21–37.
13. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You Only Look Once: Unified, Real-Time Object Detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 27–30 June 2016; pp. 779–788.
14. Redmon, J.; Farhadi, A. YOLO9000: Better, Faster, Stronger. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 7263–7271.
15. Redmon, J.; Farhadi, A. Yolov3: An incremental improvement. *arXiv* **2018**, arXiv:1804.02767.
16. Bochkovskiy, A.; Wang, C.Y.; Liao, H.Y.M. YOLOv4: Optimal Speed and Accuracy of Object Detection. *arXiv* **2020**, arXiv:2004.10934.
17. Law, H.; Deng, J. Cornernet: Detecting Objects as Paired Keypoints. In Proceedings of the European Conference on Computer Vision, Munich, Germany, 8–14 September 2018; pp. 734–750.
18. Duan, K.; Bai, S.; Xie, L.; Qi, H.; Huang, Q.; Tian, Q. Centernet: Keypoint Triplets for Object Detection. In Proceedings of the IEEE International Conference on Computer Vision, Seoul, Korea, 27 October–2 November 2019; pp. 6569–6578.
19. Gu, D.; Xu, X.; Jin, X. Marine Ship Recognition Algorithm Based on Faster-RCNN. *J. Image Signal Process.* **2018**, *7*, 136–141. [CrossRef]
20. Qi, L.; Li, B.; Chen, L.; Wang, L.; Dong, L.; Jia, X.; Huang, J.; Ge, C.; Xue, G.; Wang, D. Ship target detection algorithm based on improved faster R-CNN. *Electronics* **2019**, *8*, 959. [CrossRef]
21. Zou, Y.; Zhao, L.; Qin, S.; Pan, M.; Li, Z. Ship Target Detection and Identification Based on SSD_MobilenetV2. In Proceedings of the 2020 IEEE 5th Information Technology and Mechatronics Engineering Conference (ITOEC), Chongqing, China, 12–14 June 2020; pp. 1676–1680.
22. Huang, H.; Sun, D.; Wang, R.; Zhu, C.; Liu, B. Ship Target Detection Based on Improved YOLO Network. *Math. Probl. Eng.* **2020**, *2020*, 1–10. [CrossRef]
23. Chen, X.; Qi, L.; Yang, Y.; Luo, Q.; Postolache, O.; Tang, J.; Wu, H. Video-based detection infrastructure enhancement for automated ship recognition and behavior analysis. *J. Adv. Transp.* **2020**, *2020*, 1–12. [CrossRef]
24. Huang, Z.; Sui, B.; Wen, J.; Jiang, G. An intelligent ship image/video detection and classification method with improved regressive deep convolutional neural network. *Complexity* **2020**, *2020*, 1–11. [CrossRef]
25. Zheng, Z.; Wang, P.; Liu, W.; Li, J.; Ye, R.; Ren, D. *Distance-IoU Loss: Faster and Better Learning for Bounding Box Regression*; AAAI: New York, NY, USA, 2020; pp. 12993–13000.
26. Woo, S.; Park, J.; Lee, J.Y.; Queon, I.S. Cbam: Convolutional Block Attention Module. In Proceedings of the European Conference on Computer Vision, Munich, Germany, 8–14 September 2018; pp. 3–19.