

Research Article

A Robust and Lightweight Detector for Ship Target with Complex Background in SAR Image

Hengtao Wang  and Shang Zhang 

College of Computer and Information, China Three Gorges University, Yichang 443002, China

Correspondence should be addressed to Shang Zhang; 3011408157@qq.com

Received 21 June 2022; Revised 12 July 2022; Accepted 26 July 2022; Published 13 August 2022

Academic Editor: Yuxing Li

Copyright © 2022 Hengtao Wang and Shang Zhang. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Accurate target detection technology on ships can improve the comprehensive perception ability of weapon equipment. For SAR ship target detection in complex environments, false and missing alarms are serious. We design a new real-time ship target detection algorithm 3S-YOLO in SAR images. Firstly, reconstruct the network structure, adjust the relationship between receptive field and multiscale fusion, and realize the lightweight processing of feature extraction network and feature fusion network. Then, the network is pruned and compressed by the FPGM pruning algorithm to accelerate the reasoning speed. Finally, the Varifocal-ElIoU loss function is designed to balance the positive and negative samples and overlapping losses and highlight the contribution of positive samples. To verify the effectiveness of the 3S-YOLO algorithm, verification is carried out in public datasets SSDD and HRSID. The results show that the accuracy of the model can be improved to 99.2% and 95.6%, respectively, after optimization. After pruning, the model volume decreased significantly and could be compressed to 190 KB. Model reasoning time can be reduced to less than 3 ms. Compared with the current mainstream algorithms, 3S-YOLO has achieved good results in all aspects to meet the real-time ship target detection in SAR images.

1. Introduction

SAR (synthetic aperture radar) is an active earth observation system that can be installed on aircraft, satellites, and other flight equipment. SAR is widely used in ocean supervision, resource mapping, geographic mapping, and military observation because of its wide coverage, fast mapping speed, and high resolution.

Accurate ship detection in SAR images is conducive to military information acquisition and accurate deployment of marine firepower. It can monitor relevant sea areas all day and help the development of the modern high-tech war. Traditional SAR ship detection algorithms, such as CFAR [1] (constant false alarm rate), two-parameter CFAR based on the Gaussian model [2], template matching [3], wake detection [4], and detection methods based on wavelet transform [5], mainly rely on artificial classifier design. Although the calculation speed is fast, the detection effect is poor, and the design process of the detection algorithm is complex.

Signal feature extraction and processing is crucial in object detection. Signal feature extraction for ships is becoming more and more mature; based on VMD and slope entropy [6] and based on wavelet packet decomposition and energy entropy [7] have achieved good results in signal feature extraction for ships. Dispersion entropy-based Lempel-Ziv complexity entropy [8] provides us a new metric for signal analysis. This all greatly promotes the development of ship feature extraction.

With the explosive development of deep learning, more and more target detection algorithms based on deep learning are applied to SAR images. Target detection algorithms based on deep learning are roughly divided into two categories: single-stage and double-stage target detection algorithms. The single-stage target detection algorithm is widely used because of its fast detection speed, high efficiency, and simple edge transplantation, such as SSD [9], YOLO9000 [10], YOLOv3 [11], YOLOv4 [12], and YOLOv5. The two-stage target detection algorithm has the advantages of high detection accuracy and excellent

detection effect. However, the two-stage algorithm separates the generation of candidate regions from target classification, resulting in large computation and slow detection speed, such as Fast R-CNN [13], Faster RCNN [14], and Mask-RCNN [15]. Many scholars have studied SAR ship target detection based on deep learning.

The difficulty of ship detection in SAR images is as follows:

- (1) Image noise is large. SAR image imaging technology can cause a lot of speckle noise in the image, as shown in Figure 1. On the nearshore, the coastal background is complex, which seriously affects the ship's target detection. Complex reef background, smoke, corner reflector, and other interference information seriously affect the improvement of detection accuracy
- (2) The target scale span is large, and the small target is numerous. Ship targets near the coast have larger imaging, more targets, and serious overlap. The ship targets in the far sea are small and numerous, which are vulnerable to the complex marine environment
- (3) Unbalanced dataset and poor generalization ability. Each kind of SAR shooting angle and height is different, resulting in different imaging effects, and the general degree between different datasets is relatively weak
- (4) The detection algorithm based on deep learning has a large amount of calculation. Due to the limitation of edge computing, the conventional deep learning algorithm is difficult to deploy on the edge

Aiming at the difficulties of ship target detection in SAR images, the contributions of this paper are as follows:

- (1) The network structure 3S-YOLO is designed for small detection targets and large amounts of network calculation. The network is reconstructed, and the feature extraction network and feature fusion network are lightweight processed, respectively
- (2) FPGM pruning algorithm is used to prune the reconstructed network, which greatly reduces the calculation amount and reasoning speed of the model
- (3) For data imbalance, Varifocal loss is introduced to train the network to make IACS regression, balance the proportion of positive and negative samples, and highlight the contribution of positive samples
- (4) For the weak generalization ability of the model, the Varifocal-EIoU loss function is designed to change the border regression loss function, balance positive, and negative samples and improve the detection accuracy of the detection box
- (5) To verify the effectiveness of the algorithm, the proposed lightweight algorithm is verified on SSDD and

HRSID datasets. The detection accuracy of our method on SSDD and HRSID datasets reached 99.2% and 95.6%, respectively

The paper is arranged as follows: Section 2 introduces the model lightweight processing and loss function. Section 3 introduces the design of the 3S-YOLO algorithm. Section 4 verifies the algorithm by experiments. Finally, the summary is presented in Section 5.

2. Related Work

2.1. Lightweight Model Processing. The lightweight network model can be divided into three types, namely, model width, model depth, and model length. The comparison of various lightweight technologies is shown in Table 1. The redundant information is removed from the network structure, gradient, and module based on length, depth, and width, respectively, so as to realize the lightweight processing of the network.

The depth and width of the model will affect the detection effect in different aspects. The performance of multiple versions of YOLOv5 is shown in Table 2. The increase of depth and width can improve the detection accuracy, but the model volume and calculation amount will increase. Therefore, the design of a lightweight model and small size, fast calculation speed has become a hot issue.

2.2. Loss Function. The multitask loss function formula is as follows:

$$\mathcal{L} = \mu_{\text{box}} \mathcal{L}_{\text{box}} + \mu_{\text{cls}} \mathcal{L}_{\text{cls}} + \mu_{\text{obj}} \mathcal{L}_{\text{obj}}. \quad (1)$$

2.2.1. \mathcal{L}_{box} Boundary Loss Function. The boundary loss function is used to calculate the boundary regression loss between the predicted boundary and the real boundary of each layer. Location in the network model provides gradient changes. Common border losses are IoU, GIoU, DIoU, and CIoU. The formula is as follows:

$$\mathcal{L}_{\text{box}} = \sum_{l=1}^L \left(1 - \text{mean} \left(f_{\text{xIoU}} \left(\text{box}_{\text{pre}}^{l,1}, \text{box}_{\text{gt}}^{l,1} \right), \dots, f_{\text{xIoU}} \left(\text{box}_{\text{pre}}^{l,M}, \text{box}_{\text{gt}}^{l,M} \right) \right) \right), \quad (2)$$

$$\text{IoU} = \frac{\text{box}_{\text{pre}}^{l,m} \cap \text{box}_{\text{gt}}^{l,m}}{\text{box}_{\text{pre}}^{n,m} \cup \text{box}_{\text{gt}}^{n,m}}, \quad (3)$$

$$\mathcal{L}_{\text{GIoU}} = 1 - \text{IoU} + \frac{\left| C - \left(\text{box}_{\text{pre}}^{n,m} \cup \text{box}_{\text{gt}}^{n,m} \right) \right|}{|C|}, \quad (4)$$

$$\mathcal{L}_{\text{DIoU}} = 1 - \text{IoU} - \frac{\rho^2 \left(\text{box}_{\text{pre}}^{l,m}, \text{box}_{\text{gt}}^{l,m} \right)}{c^2}, \quad (5)$$

$$\mathcal{L}_{\text{CIoU}} = 1 - \text{IoU} - \frac{\rho^2 \left(\text{box}_{\text{pre}}^{l,m}, \text{box}_{\text{gt}}^{l,m} \right)}{c^2} + \alpha \vartheta, \quad (6)$$

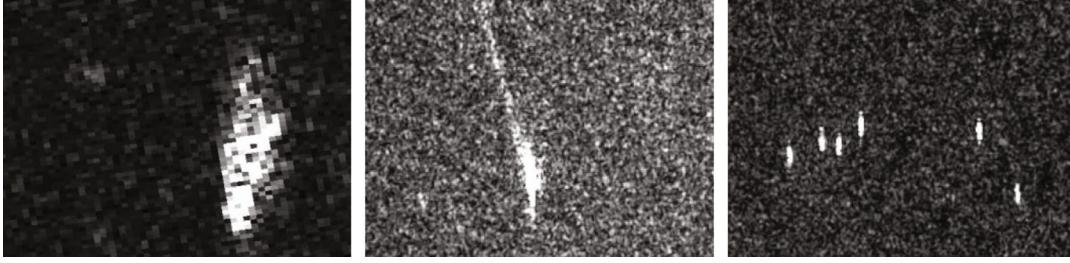


FIGURE 1: SAR noise display.

TABLE 1: Statistical table of model lightweight processing characteristics.

Model	Feature	Adaptive environment
Length-based	Removing redundant modules based on network structure	Redesign of network structure based on actual datasets
Depth-based	Removal of redundant feature information based on gradient	Adjust the feature extraction method of the receptive field to reduce gradient disappearance
Based on width (channel)	Removing redundant parameter based on network module	Wide range of applications, all networks available. A variety of implementations are implemented by specifying widths or using pruning techniques. Common pruning techniques include structured and unstructured pruning

TABLE 2: YOLOv5 version introduction tables (COCO dataset for example).

Model	Width	Depth	mAP/ %	Speed/ ms	params/ M	FLOPs
YOLOv5n	0.25	0.33	36.7	3.5	1.9	4.5
YOLOv5s	0.5	0.33	45.1	4.8	7.2	16.5
YOLOv5m	0.75	0.67	51.7	7.4	21.2	49.0
YOLOv5l	1.0	1.0	53.2	11.9	46.5	109.1
YOLOv5x	1.25	1.33	54.9	20.5	86.7	205.7

$$\mathcal{L}_{obj} = \sum_{l=1}^L \left(\text{mean} \left(\text{loss}_{obj}^{l,1}, \text{loss}_{obj}^{l,2}, \dots, \text{loss}_{obj}^{l,M} \right) \right). \quad (10)$$

3. 3S-YOLO

YOLOv5 has been widely used because of its low power consumption and fast detection speed. It shows remarkable performance in COCO dataset detection tasks. However, under the background of high resolution and wide field of view of SAR images, the original YOLOv5 algorithm is not suitable. Therefore, this paper designs 3S-YOLO (simplified SAR image ship detector).

The core idea of 3S-YOLO is to reconstruct the network structure and lightweight the network model according to the characteristics of small targets in SAR images. Starting from the length and width of the network structure, the network redundancy module and parameters are reduced, so that the network structure is more suitable for ship detection in SAR images. 3S-YOLO is composed of a reconstructed lightweight network structure, FPGM pruning, and Varifocal-EIoU loss function. Firstly, the feature extraction network is pruned by adjusting the weight of the small target receptive field. Adjust feature fusion network architecture, and eliminate redundant feature modules. Then, the width of the network model is adjusted by using FPGM pruning to remove redundant parameters. Finally, Varifocal-EIoU is designed to highlight the contribution of positive samples while balancing positive and negative samples and overlapping losses so that IACS regression accelerates the convergence speed of model training and improves the accuracy of the model. 3S-YOLO network architecture as shown in Figure 2, through the lightweight processing of feature extraction network and feature fusion network to improve the detection ability of model SAR ship target, reduces the amount of calculation and facilitates the deployment of edge.

$$\vartheta = \frac{4}{\pi^2} \left(\arcsin \frac{\omega^{gt}}{h^{gt}} - \arcsin \frac{\omega}{h} \right)^2, \quad (7)$$

$$\alpha = \frac{\vartheta}{(1 - \text{IoU}) + \vartheta}. \quad (8)$$

The characteristics of each type of loss function are analyzed as shown in Table 3, which presents the advantages and disadvantages of the current mainstream loss functions, respectively.

2.2.2. \mathcal{L}_{obj} Confidence Loss Function. The confidence loss function is used to calculate the probability of objects in the prediction box. The confidence between the preselected box and the real box is calculated using the sigmoid cross-entropy loss function. The \mathcal{L}_{obj} formula is as follows:

$$\text{loss}_{obj}^{l,1} = - \left(\text{obj}_{gt}^{l,m} \times \log \left(\delta \left(\text{obj}_{pre}^{l,m} \right) \right) + \left(1 - \text{obj}_{gt}^{l,m} \right) \times \left(1 - \log \left(\delta \left(\text{obj}_{pre}^{l,m} \right) \right) \right) \right), \quad (9)$$

TABLE 3: Characteristic analysis of loss function.

Loss	Merit	Defect
IoU [16]	Scale invariance, nonnegative, identity, symmetry, triangular inequality	If the anchor frames do not intersect, the distance between the two frames cannot be reflected. Cannot accurately reflect the coincidence of the two boxes
GIoU [17]	A minimum bounding box with zero loss without overlap	When the two boxes intersect, the horizontal and vertical convergence is slow
DIoU [18]	Direct regression of the Euclidean distance between the center points of two frames to accelerate convergence	The aspect ratio is not considered in the regression process
CIoU [19]	Increase detection frame size loss and width loss	The aspect ratio is described as a relative value Unconsidering difficult sample equilibrium problem

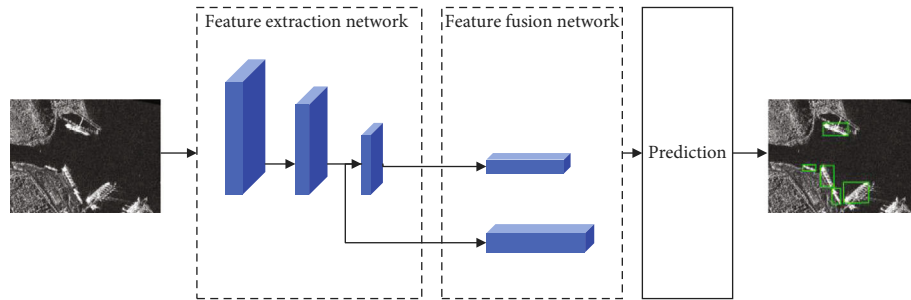


FIGURE 2: 3S-YOLO system architecture.

After the above method adjustment, the network structure of 3S-YOLO reduces the redundant modules and parameters, making the calculation amount, model volume, and detection speed greatly improved.

3.1. System Algorithm Details. 3S-YOLO is mainly composed of reconfigured post-network, FPGM pruning and Varifocal-EIoU. As shown in Algorithm 1, the training pruning process pseudocode is as follows.

3.2. Network Structure Reconstruction. The original YOLOv5 uses 80×80 , 40×40 , and 20×20 feature maps, and 3S-YOLO adjusts the receptive field and feature map to adapt to the feature information of ship targets in SAR images. The large receptive field size 20×20 is eliminated, and 40×40 and 80×80 feature maps are retained, so that it is more suitable for ship small target detection in SAR images.

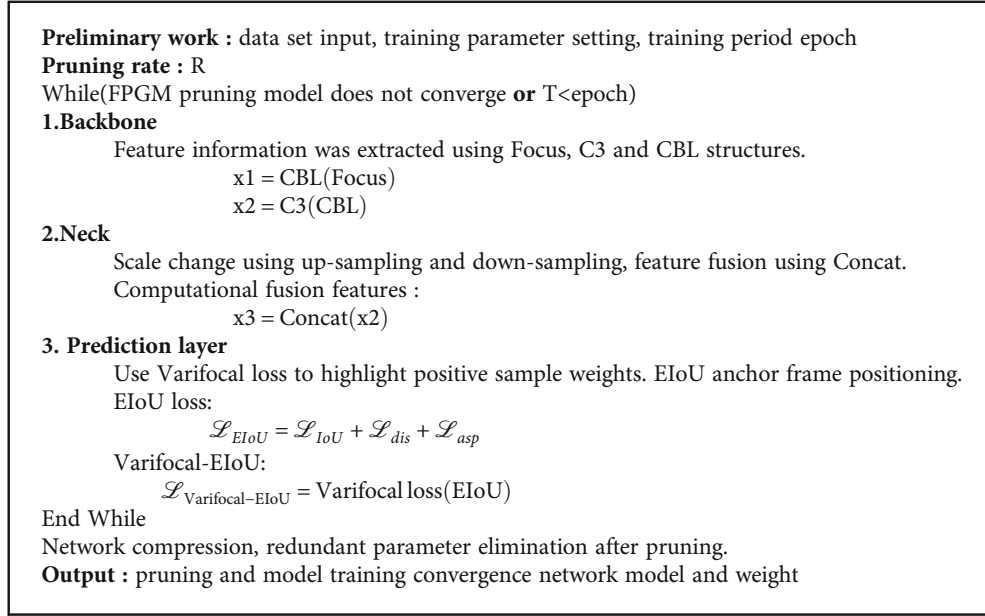
In SAR images, the ship's target area only occupies a small part of the image. After convolution layer-by-layer iteration and downsampling information fusion, the ship feature information will be lost. Therefore, 3S-YOLO reduces the loss of underlying feature information by reducing the number of downsampling and convolution. The main network structure is shown in Table 4. The improved feature extraction network has experienced four downsampling and three convolution operations, which reduces the amount of calculation and parameter while reducing the loss of small target feature information.

3S-YOLO feature map size is 40×40 and 80×80 ; the original feature fusion network is no longer suitable for the new feature extraction network, so it is improved.

The improved feature fusion network structure is shown in Figure 3, and the fusion position of the receptive field is adjusted. Firstly, the feature fusion of shallow semantic and deep semantics is carried out, followed by 80×80 and 40×40 feature maps. Then, the feature fusion of deep semantics and shallow semantics is carried out, and the feature maps of 40×40 and 80×80 are fused in turn. The weight of shallow semantic features is added on the original basis to reduce the loss of target feature information caused by convolution.

The 3S-YOLO algorithm structure is shown in Figure 4. The 3S-YOLO network structure reconstructs the feature extraction network and the feature fusion network, respectively. Firstly, the feature extraction network CSPDarkNet is reconstructed to remove the 20×20 feature map extraction module, improve the weight of the small receptive field of the network, and reduce the calculation amount and volume of the model. Then, a modified version of CSPDarkNet feature fusion is performed to achieve the fusion of feature information of deep and shallow semantics.

3.3. FPGM Pruning. The conventional channel pruning algorithm uses the principle of "small norm less important," which uses norm to measure the importance of each filter. The conventional channel pruning algorithm is limited by the filter contribution with a large norm standard deviation range and small norm. FPGM [20] (convolution neural network filtering pruning based on geometric median) prunes the network according to the substitutability of the filter. The schematic diagram of the FPGM algorithm is shown



ALGORITHM 1: 3S-YOLO detection algorithm.

TABLE 4: Reconstructing CSPDarkNet structure.

Module	Parameter	Filters	Size	Output
Focus	3520	32	(1, 1)	320 × 320
Conv	18560	64	(3, 3)	160 × 160
C3	18816	64		160 × 160
Conv	73984	128	(3, 3)	80 × 80
C3	146928	128		80 × 80
Conv	295424	256	(3, 3)	40 × 40
SPP	164608	256	(1, 1) (5, 5), (9, 9), (13, 13)	40 × 40
C3	296448	256		40 × 40

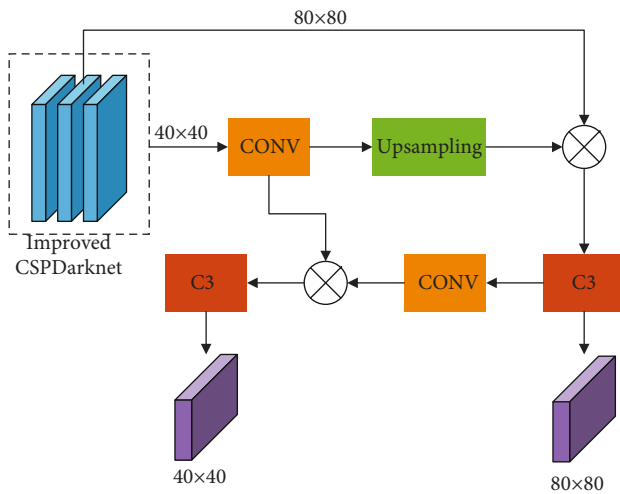


FIGURE 3: Feature fusion network structure diagram.

in Figure 5. Break through the limitation of norm-based pruning and realize the transformation from “relatively insignificant” to “replaceable.” FPGM can retain more abundant information features and cut more redundant information.

The central idea of the geometric median pruning algorithm is to select the lowest elimination by calculating the sum of Euclidean distances of each point. First, calculate the Euclidean distance of each point $a^{(i)}$ compared to other points:

$$f(x) = \sum_{i \in [1, n]} \|x - a^{(i)}\|_2. \quad (11)$$

Select a relatively small sum of Euclidean distance filter:

$$x^* = \operatorname{argmin} f(x). \quad (12)$$

Based on the robustness of the geometric median in the convolution space, the common distance of all filters in the first layer is calculated by using the geometric median

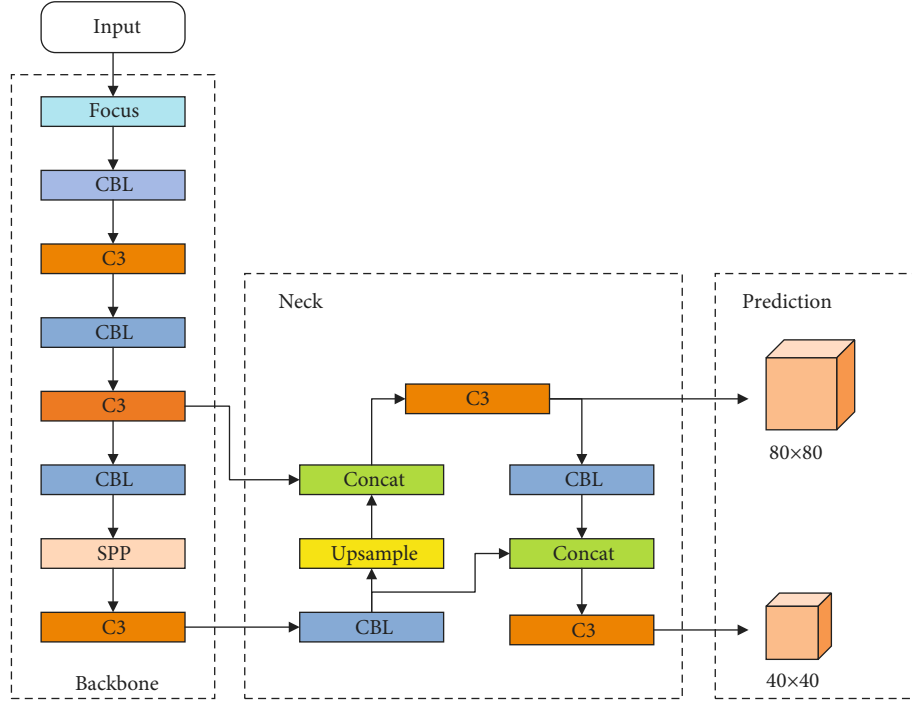


FIGURE 4: 3S-YOLO network structure.

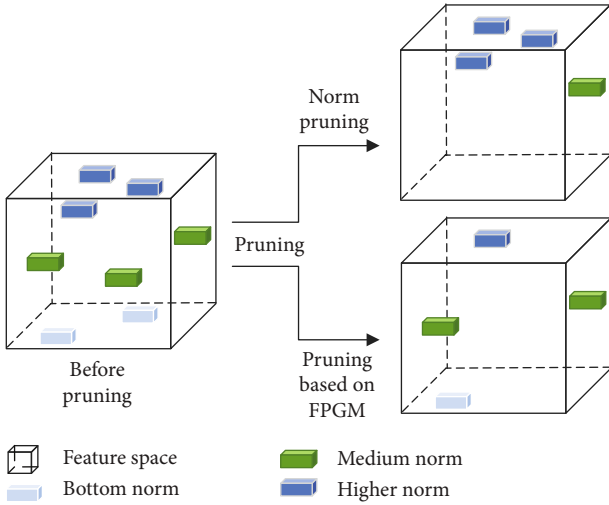


FIGURE 5: FPGM pruning schematic.

calculation method, where each convolution kernel parameter dimension can be expressed as $F_i \in \mathbb{R}^{N_{i+1} \times N_i \times K \times K}$.

$$x^{\text{GM}} = \underset{x \in \mathbb{R}^{N_i \times K \times K}}{\operatorname{argmin}} \sum_{j=1}^{N_{i+1}} \|x - F_{i,j'}\|_2. \quad (13)$$

The pruning of FPGM is based on the fact that the filter from the geometric center is easily replaced, so it can be pruned according to the distance from the geometric center to find the filter closer to the geometric center:

$$F_{i,j_*} = \underset{F_{i,j'}}{\operatorname{argmin}} \|F_{i,j'} - x^{\text{GM}}\|_2^2. \quad (14)$$

The calculation of geometric center is large and cumbersome, and the distance between filters is relatively simple, so only the filter with the smallest sum of distances is calculated:

$$F_{i,x_*'} = \underset{x}{\operatorname{argmin}} g'(x), \quad (15)$$

$$g' = \sum_{j \in [1, N_{i+1}], F_{i,j'} \neq x} \|x - F_{i,j'}\|_2. \quad (16)$$

Among them, $x \in \{F_{i,1}, \dots, F_{i,N_{i+1}}\}$.

The FPGM algorithm is described as Algorithm 2, where YOLOv5 is fused with FPGM. First, complete the preparation before training, and set the pruning rate R ; then, the filter gradient with a small sum of Euclidean distances is iteratively set to zero during training; finally, after the model converges, the zero removal operation is carried out in the reasoning process, and the full zero convolution kernel, redundant channels, and BN redundant parameters are removed. Get the final pruning model.

3.4. Varifocal-EIoU. In target detection, sample imbalance seriously affects the detection effect. High-quality samples are crucial to the training process of network models. Recent research designs Focal-EIoU [21] to solve these problems, but it ignores the contribution of high quality samples in balanced positive and negative samples. Therefore, Varifocal-EIoU is designed in this section to increase the contribution of high-quality samples while improving the accuracy of the model test box.

Preparations:	input training data: X given pruning rate R.
1: Initialize:	model parameter W=1
2: for epoch =1; epoch ≤ epoch_max; epoch ++ do	
3:	Update the model parameter W based on X
4:	for i =1; i ≤ L; i ++ do
5:	Calculate: the sum of N Euclidean distances
6:	end for
7:	Find the relatively small number of N * R filters
8:	Zeroize the selected filter gradient
9:	end for
10:	Obtain the pruning model with zeroing model W [#] from W
11:	Remove zero parameters from the model W*
Output:	The pruning model and its parameters W*

ALGORITHM 2: FPGM description.

Focal loss is commonly used to solve the problem of positive and negative sample imbalance. The focal loss function is as follows:

$$FL(p, y) = \begin{cases} -\alpha(1-p)^\beta \log(p) & \text{if } y = 1, \\ -(1-\alpha)p^\beta \log(1-p) & \text{otherwise,} \end{cases} \quad (17)$$

where p is the prediction probability of the target class, and the range is $[-1, 1]$; y is the real positive and negative sample category, with the value of 1 or -1 ; α is an adjustable proportional factor; $(1-p)^\beta$ is the target modulation factor; and p^β is the background modulation factor. The two types of modulation factors can reduce the contribution of simple samples, increase the importance of false samples, and effectively increase the attention to difficult samples. This allows focal loss to use weighted methods to solve the class imbalance problem in IACS regression training. Focal loss uses an equal way to deal with positive and negative samples, and in the actual detection, the contribution of positive samples is more important. Therefore, the focal loss is improved. Varifocal loss is based on binary cross entropy loss, and the focal loss weighting method is used to deal with the category mismatch problem in the IACS training regression process.

$$VFL(p, q) = \begin{cases} -q(q \log(p) + (1-q) \log(1-q))^\beta & q > 0, \\ -\alpha p^\beta \log(1-p) & q = 0, \end{cases} \quad (18)$$

where p is the predicted value of IACS, representing the target score, and q is the classification condition. For the target class, the q value of the positive sample class is set to the IoU value between the check box and ground truth (gt_IoU); otherwise, it is set to 0. For the background category, the target q values of all categories are 0. As shown in the formula, Varifocal loss uses the p^β scaling factor to process the negative sample, rather than the positive sample. This highlights the contribution of positive samples.

YOLOv5 uses CIoU as the loss calculation. CIoU takes into account the overlapping area, center distance, and

aspect ratio of the border, but there is a vague description of aspect ratio, not considering the difficult sample balance problem, and ignoring the loss caused by height and width. So 3S-YOLO uses EIou to calculate the loss. EIou loss function is as follows:

$$\mathcal{L}_{EIou} = \mathcal{L}_{IoU} + \mathcal{L}_{dis} + \mathcal{L}_{asp}. \quad (19)$$

EIoU divides the loss into three parts based on CIoU: overlapping loss, center distance loss, and width-height loss.

$$\mathcal{L}_{EIou} = 1 - IoU + \frac{\rho^2(b, b^{gt})}{c^2} + \frac{\rho^2(w, w^{gt})}{c_w^2} + \frac{\rho^2(h, h^{gt})}{c_h^2}. \quad (20)$$

Among them, c_w and c_h are the minimum box width between the target box and the real box. In the frame loss, the width-height loss of EIou converges faster, which makes the frame accuracy higher.

By integrating EIou loss and Varifocal loss, we obtain the final Varifocal-EIoU loss, which highlights the contribution of positive samples while balancing positive and negative samples with overlapping losses.

$$\mathcal{L}_{Varifocal-EIoU} = \text{Varifocal loss}(EIou). \quad (21)$$

Figure 6 shows the prediction architecture of EIou-Varifocal network. In the head structure, EIou-Varifocal mainly realizes the regression and optimization of boundary frames. By changing the loss function and adjusting the distance vector, the precise positioning of the boundary frame is realized.

4. Experiment

The configuration of this experiment is as follows: operating system: Ubuntu 16.04 and CUDA10.2; GPU configuration: NVIDIA RTX1660ti, 6GB memory, and call GPU training; and framework: PyTorch.

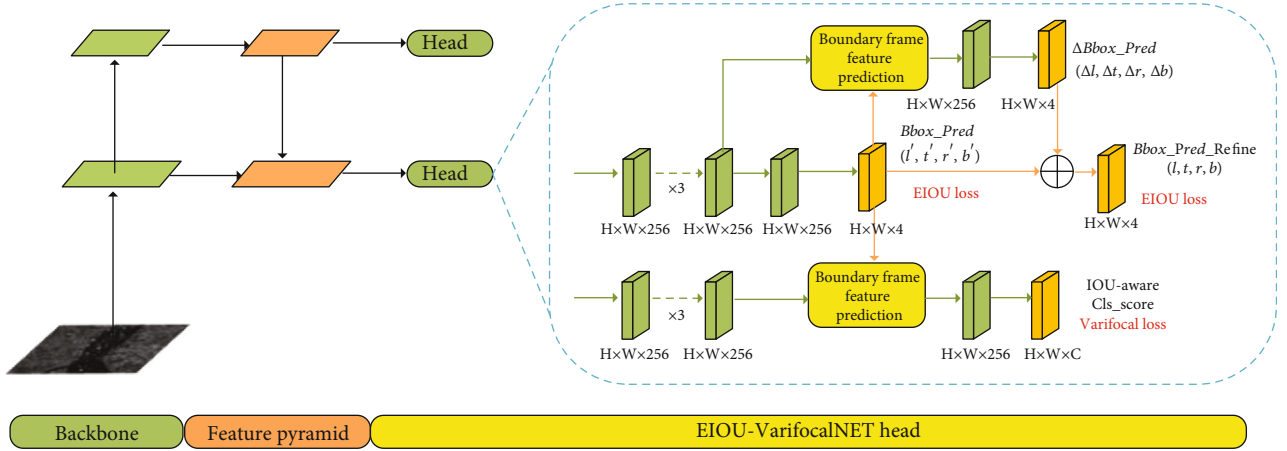


FIGURE 6: The network architecture of EIoU-Varifocal.

TABLE 5: Setting anchor boxes on different datasets.

Dataset	40 × 40 Medium receptive field	80 × 80 Little receptive field
SSDD	(29, 70)	(18, 14)
	(125, 49)	(16, 38)
	(58, 144)	(39, 21)
HRSID	(49, 16)	(9, 13)
	(37, 33)	(18, 11)
	(146, 149)	(18, 36)

4.1. SAR Ship Dataset. In order to verify the detection effect of the SAR ship target, the dataset used in this paper is SSDD (SAR ship detection dataset) and HRSID. A total of 1160 SSDD datasets marked 2456 ships. The dataset contains a variety of resolution images, and the marine environment is diverse. It has rich characteristic information on offshore and large scale. It is widely used in ship target detection of multiresolution imaging. HRSID dataset constructed 5604 images based on Sentinel-1B, TerraSAR-X, and TanDEM-X images, marking 16951 ships. Since the original dataset only contains one category of ships and the generalization ability is relatively weak, it is expanded, and 356 background SAR images are added. In this experiment, SSDD and HRSID datasets were randomly divided into a training set, verification set, and test set according to the ratio of 3:1:1.

4.2. Evaluating Indicator. The main indicators used to evaluate the algorithm by computer vision are mAP (mean average precision), P (precision), and R (recall). Among them, the accuracy rate is the proportion of samples predicted to be positive and actually positive; the recall rate is the proportion of correct detection in positive samples. The calculation formulas of accuracy, recall, and average accuracy are as follows:

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}, \quad (22)$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}, \quad (23)$$

$$\text{AP} = \int_0^1 P(R), \quad (24)$$

$$\text{mAP} = \frac{\sum_{j=1}^c \text{AP}_j}{c}, \quad (25)$$

$$F1 = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}, \quad (26)$$

where TP (true positive) is the positive sample with true test results, FN (false negative) is a positive sample whose result is not true, FP (false positive) is a true negative sample, and AP is the accuracy of a single category. The mAP is the average accuracy for all categories; c is the number of categories.

4.3. Parameter Optimization of Anchor Frame. The size of the preselection box is controlled by human factors. This algorithm reconstructs the network structure, so the original anchor frame size is not suitable for the reconstructed network. In order to improve the training accuracy and regression speed, K-means clustering and genetic algorithm are used to analyze and cluster the size of the preselection box, and the most suitable size is obtained.

The allocation of anchor frames is shown in Table 5. The 20×20 feature map is deleted on the original basis, and the 40×40 and 80×80 feature maps are retained to realize the adjustment of the receptive field. Make it more suitable for ship detection in SAR images.

4.4. Model Pruning Comparison. The comparison of the calculation amount of model pruning is shown in Table 6. From the data point of view, with the increase in pruning rate, the network calculation amount, the number of parameters, the model volume, and the reasoning time all decrease. The amount of calculation, parameters, and model volume decrease along the line; when the pruning rate is less than 50%, the reasoning time decreases significantly; when more than 50%, reasoning time changes slowly. Compared with

TABLE 6: Statistical table of model pruning calculation.

Model	Based on YOLOv5s				Based on 3S-YOLO			
	FLOPs (GFLOPs)	Parameters	Volume/MB	Runtime/ms	FLOPs (GFLOPs)	Parameters	Volume/MB	Runtime/ms
0.1 prune	13.9	5861553	11.6	6.510	9.1	1335541	2.8	4.486
0.2 prune	11.6	4740130	9.22	5.626	7.6	1081299	2.2	3.967
0.3 prune	9.4	3752608	7.53	5.141	6.1	851234	1.8	3.648
0.4 prune	7.5	2863963	5.63	4.618	4.9	650480	1.3	3.265
0.5 prune	5.6	2085510	4.27	4.375	3.7	471860	1.0	2.982
0.7 prune	2.8	916402	1.98	4.303	1.9	210467	0.5	2.922
0.8 prune	1.7	499328	1.16	4.273	1.1	113218	0.3	2.881
0.9 prune	0.8	193115	0.56	4.249	0.5	44848	0.2	2.732

TABLE 7: Comparison of performance indexes of datasets with different pruning rates (pruning model is network reconstructed model).

Model	SSDD				HRSID			
	Precision (%)	Recall (%)	AP (%)	F1 (%)	Precision (%)	Recall (%)	AP (%)	F1 (%)
0.1 prune	96.7	95.6	98.8	96.1	93.8	88.0	94.1	90.8
0.2 prune	96.5	95.2	98.7	95.8	93.2	87.6	94.0	90.3
0.3 prune	96.6	94.9	98.6	95.7	92.6	87.8	93.8	90.1
0.4 prune	96.3	95.2	98.1	95.7	92.9	87.2	93.3	90.0
0.5 prune	96.1	94.0	97.5	95.0	91.7	85.9	92.7	88.7
0.6 prune	95.1	92.9	97.1	94.0	91.1	85.1	92.1	88.0
0.7 prune	91.9	95.5	96.9	93.7	91.2	83.5	90.7	87.2
0.8 prune	94.7	89.0	96.7	91.8	88.9	79.3	88.0	83.8
0.9 prune	92.8	82.4	92.4	87.3	86.0	72.9	82.1	78.9

TABLE 8: Comparison of performance indexes of different pruning rates after Varifocal-EIoU fine-tuning.

Model	SSDD				HRSID			
	Precision (%)	Recall (%)	AP (%)	F1 (%)	Precision (%)	Recall (%)	AP (%)	F1 (%)
0.1 prune	97.9	97.0	99.0	97.4	93.9	90.1	95.0	92.0
0.2 prune	97.7	97.0	98.8	97.3	93.3	88.9	94.6	91.0
0.3 prune	95.4	94.7	98.7	95.0	93.0	88.9	94.3	91.0
0.4 prune	95.3	94.7	98.4	95.0	93.3	86.6	93.8	89.8
0.5 prune	95.9	94.1	97.9	95.0	92.9	86.4	93.3	89.5
0.6 prune	95.5	92.8	97.6	94.1	92.3	86.4	92.9	89.3
0.7 prune	93.6	93.8	97.5	93.7	92.1	85.1	92.4	88.5
0.8 prune	94.7	92.0	97.1	93.3	89.0	82.7	89.3	85.7
0.9 prune	93.3	87.7	95.5	90.4	89.4	76.0	85.1	82.2

YOLOv5s, 3S-YOLO has lower resource consumption and faster reasoning speed under the same pruning rate. The minimum volume of the 3S-YOLO model can be compressed to 0.2 MB, the minimum amount of calculation can be reduced to 0.5 GFLOPs, and the minimum reasoning speed can be reduced to 2.732 ms.

The comparison of model pruning performance is shown in Table 7. As the pruning rate increases, the detection performance of the model decreases. The detection results on SSDD and HRSID datasets show that the detection accuracy AP of the model decreases with the increase of the pruning rate. When the pruning rate is 90%, the performance of the model decreases most.

As the pruning rate increases, the detection performance decreases. We propose to design Varifocal-EIoU for fine-tuning after pruning the model. As shown in Table 8, the performance indicators of the model are partially recovered after fine-tuning. Among them, when the pruning rate is 90%, the performance recovery is the most. Among them, the average detection accuracy is increased by 3.1% in the SSDD dataset and 3% in the HRSID dataset. This also verifies the effectiveness of our Varifocal-EIoU algorithm.

4.5. Experimental Results and Analysis. To verify the effectiveness of each improvement point in 3S-YOLO in SAR image ship detection, ablation experiments are conducted

TABLE 9: Ablation experiment.

Algorithm	Network reconfiguration	Varifocal-EIoU	Prune	SSDD			HRSID		
				P/%	R/%	mAP/%	P/%	R/%	mAP/%
YOLOv5s				93.9	92.8	97.2	91.5	85.0	92.8
A	√			97.2	95.7	98.9	94.4	88.9	94.5
B		√		96.7	96.9	98.9	93.8	89.7	94.1
C	√	√		98.1	97.1	99.2	94.2	91.5	95.67
D	√		50%	96.1	94.0	97.5	91.7	85.9	92.7
E	√	√	50%	95.9	94.1	97.9	92.9	86.4	93.3
F	√		90%	92.8	82.4	92.4	86.0	72.9	82.1
G	√	√	90%	93.3	87.7	95.5	89.4	76.0	85.1

TABLE 10: Performance comparison of SSDD with different algorithms.

Method	Backbone	Precision (%)	Recall (%)	AP (%)	FLOPs (GFLOs)	params	Runtimes(ms)
Libra R-CNN [23]	ResNet 101-FPN	88.6	88.6	89.9	83.0	60.4	30.2
Cascade R-CNN [24]	ResNet 101-FPN	94.3	89.9	89.5	110.4	87.9	38.8
Faster R-CNN	ResNet 101-FPN	90.9	87.6	88.3	82.7	60.1	30.2
CR2A-Net [25]	ResNet 101-FPN	94.0	87.8	89.8	112.0	88.6	67.2
DAPN [26]	ResNet 101-FPN	87.6	91.4	90.1	117.2	63.8	34.5
RetinaNet [27]	ResNet-101-FPN	81.6	92.3	89.6	71.8	55.1	30.2
SSD	SSD-VGG	92.9	88.0	94.0	87.7	24.4	30.2
YOLOv3	Darknet-53	90.7	94.7	95.0	49.6	61.5	10.4
YOLOv4	CSPDarknet-53	93.6	94.0	96.1	45.3	64.3	12.9
YOLOv5	CSPDarknet-53	93.9	92.8	97.2	16.4	7.1	8.8
[22]	Darknet-53	95.1	94.5	94.8	50.6	65.8	16.4
FCOS [28]	ResNet 101-FPN	94.4	85.6	88.7	69.8	50.8	25.9
CenterNet [29]	DAL-34	93.3	94.5	93.5	25.9	20.2	21.5
CenterNet++ [30]	DAL-34	92.6	94.5	92.7	26.6	20.3	21.5
Our	CSPDarknet-53	98.1	97.1	99.2	11.3	1.9	5.1
Our(0.8 prune)	CSPDarknet-53	94.7	92.0	97.1	0.5	0.1	2.9

TABLE 11: Performance comparison of different algorithms for HRSID.

Method	Backbone	Precision (%)	Recall (%)	AP (%)	FLOPs (GFLOs)	params	Runtimes (ms)
Libra R-CNN [23]	ResNet 101-FPN	83.1	77.9	77.5	182.6	60.4	57.6
Cascade R-CNN [24]	ResNet 101-FPN	89.9	79.3	79.2	209.7	87.9	64.7
Faster R-CNN	ResNet 101-FPN	88.8	77.5	78.2	181.9	60.1	56.1
CR2A-Net [25]	ResNet 101-FPN	88.5	78.9	80.9	212.5	88.6	77.3
DAPN [26]	ResNet 101-FPN	88.9	77.6	79.8	266.1	63.8	74.9
RetinaNet [27]	ResNet-101-FPN	69.8	83.8	82.5	175.4	55.1	55.0
SSD512	SSD-VGG	87.4	85.3	88.8	87.7	24.4	44.8
YOLOv3	Darknet-53	90.6	78.2	87.2	121.0	61.5	26.0
YOLOv4	CSPDarknet-53	90.6	84.0	90.1	110.5	64.3	22.4
YOLOv5	CSPDarknet-53	91.5	85.0	92.8	16.4	7.1	9.8
[22]	Darknet-53	92.7	88.1	90.3	123.5	65.8	37.3
FCOS [28]	ResNet 101-FPN	91.9	79.5	86.6	170.6	50.8	50.9
CenterNet [29]	DAL-34	81.8	87.4	86.3	63.3	20.2	55.0
CenterNet++ [30]	DAL-34	82.2	87.3	86.3	64.9	20.3	54.5
Our	CSPDarknet-53	94.2	91.5	95.7	11.3	1.9	5.1
Our (0.7 prune)	CSPDarknet-53	92.1	85.1	92.4	1.9	0.2	2.9

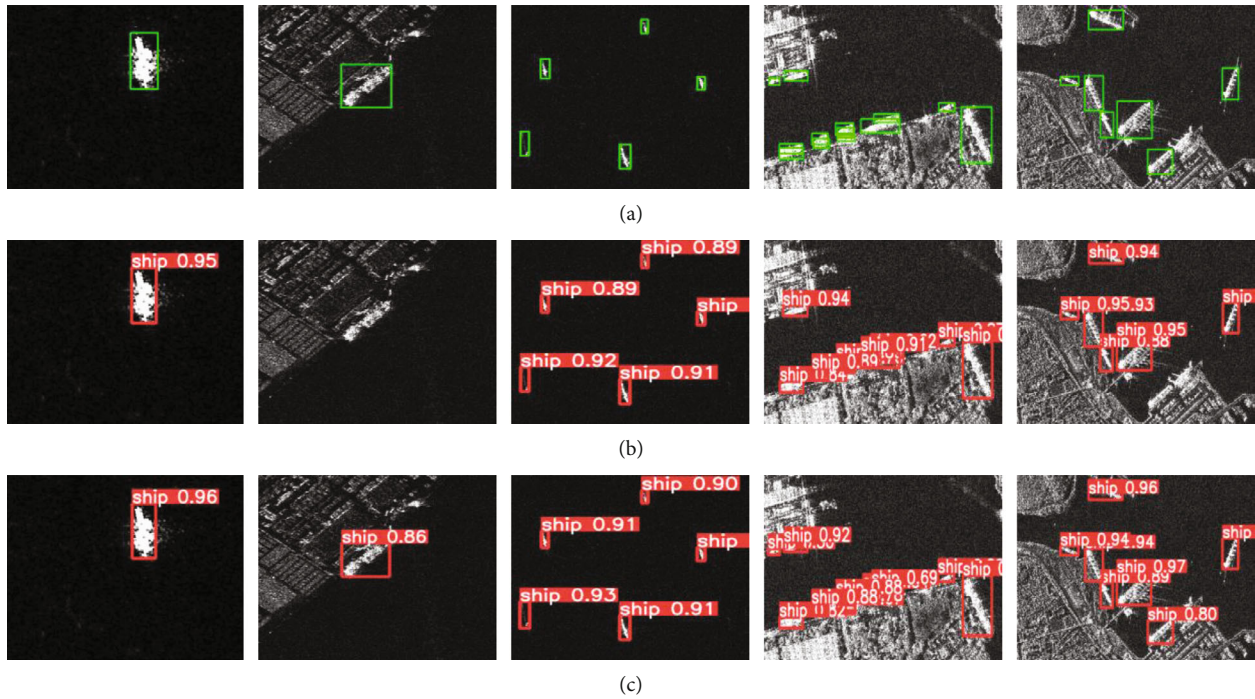


FIGURE 7: Detection effect under different models.

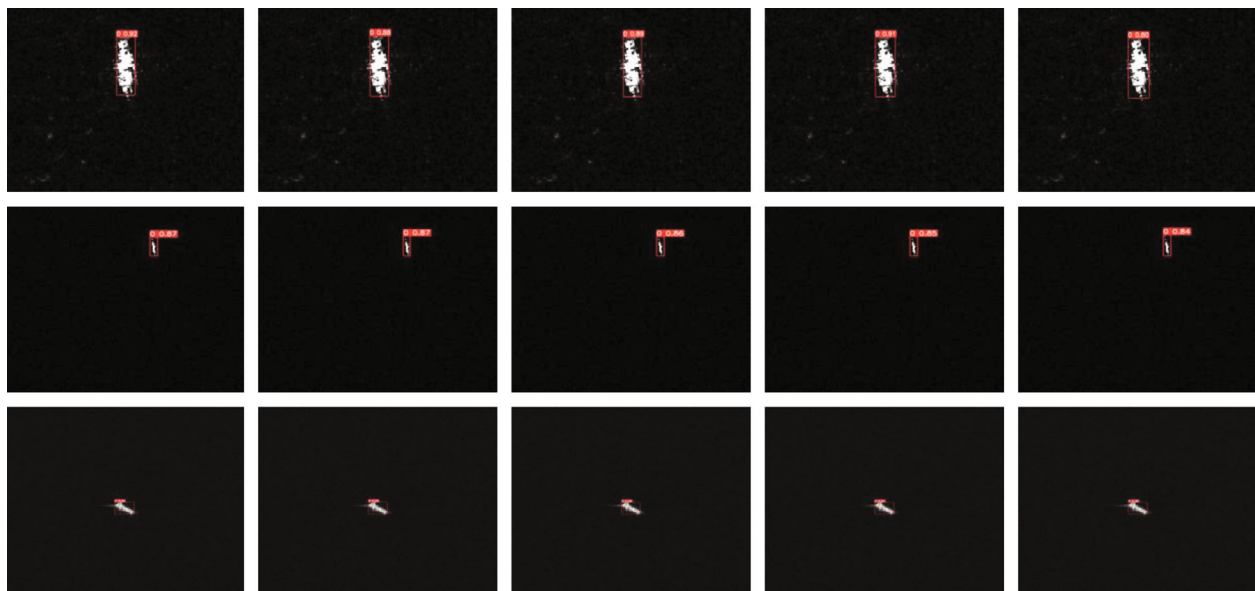


FIGURE 8: Detection effect of different pruning rates on offshore single vessel (the pruning rates (a–e) were 10%, 30%, 50%, 70%, and 90%).

on SSDD and HRSID datasets. The ablation experiments are shown in Table 9, and the results show that 3S-YOLO improves the detection accuracy of the model after both network reconstruction and Varifocal-EIoU optimization, and at the same time, the accuracy and recall rate are also improved.

On the SSDD dataset, model reconstruction can improve the accuracy of the model by 3.3%, and the average detection

accuracy can be improved to 98.9%. Varifocal-EIoU can greatly improve the recall rate by about 4.1%, and the average accuracy can be increased to 98.9%. After 3S-YOLO network reconstruction and Varifocal-EIoU, the average accuracy of the model can be improved to 99.2%, and the recall and accuracy are increased by 2.8% and 4.1%, respectively. After 50% pruning, the average detection accuracy of the model decreased by about 1.5%. At a 90% pruning rate,

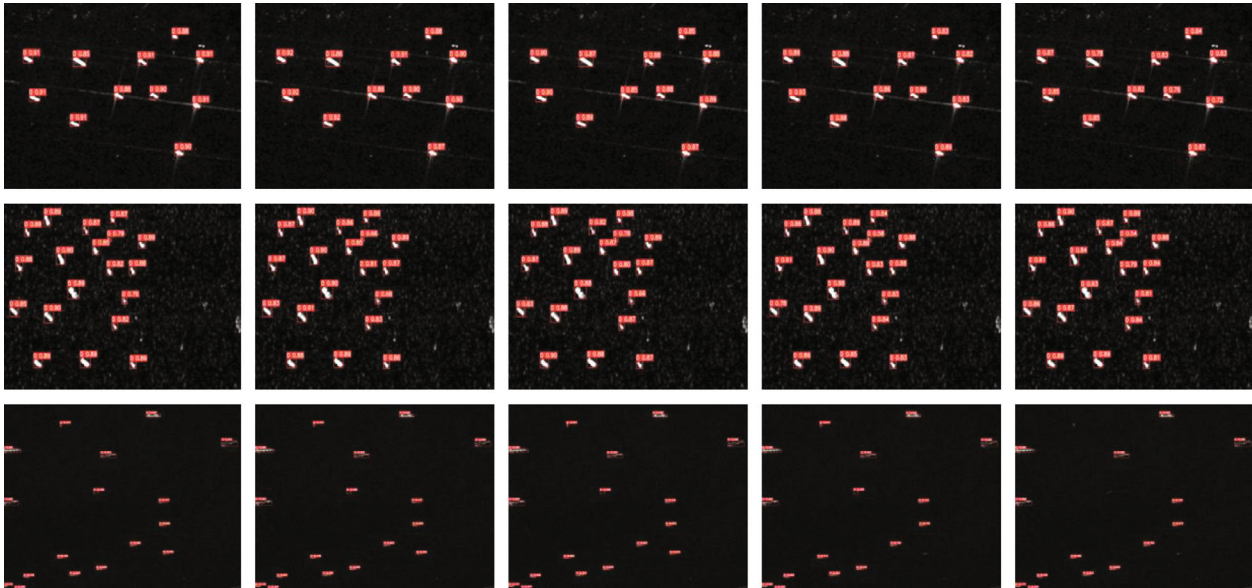


FIGURE 9: Detection effect of different pruning rates on offshore small target ships.

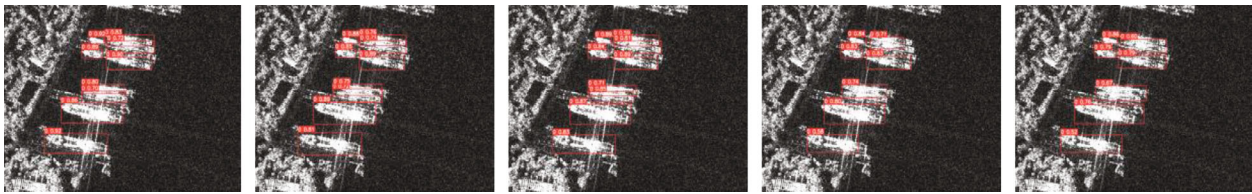


FIGURE 10: Effect of different pruning rates on detection of offshore dense ships.

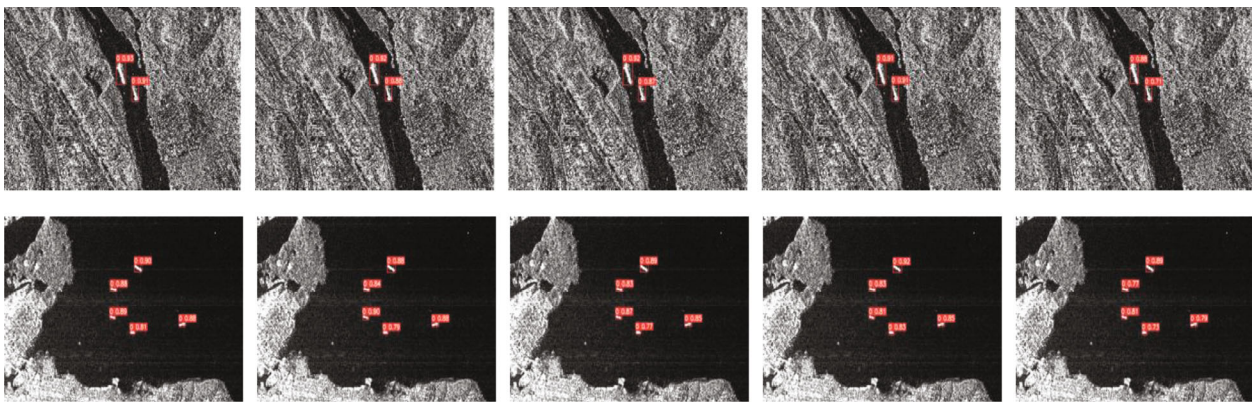


FIGURE 11: The detection effect of different pruning rates on offshore sparse ships.

the average accuracy of the model decreases by 6.5%, and the detection accuracy can be restored by 3.1% through Varifocal-EIoU.

On the HRSID dataset, model reconstruction can improve the accuracy of the model by 2.9%, and the average detection accuracy can be improved to 94.5%. Varifocal-EIoU can greatly improve the recall rate by 4.7%, and the average accuracy can be improved to 94.1%. After 3S-

YOLO network reconstruction and Varifocal-EIoU, the average accuracy of the model can be improved to 95.67%, and the recall and accuracy are increased by 2.7% and 6.5%, respectively. When the pruning rate was 50%, the average detection accuracy of the model decreased by about 3.0%. At a 90% pruning rate, the average accuracy of the model decreased by 10.5%, and it could be restored to 85.1% by Varifocal-EIoU.

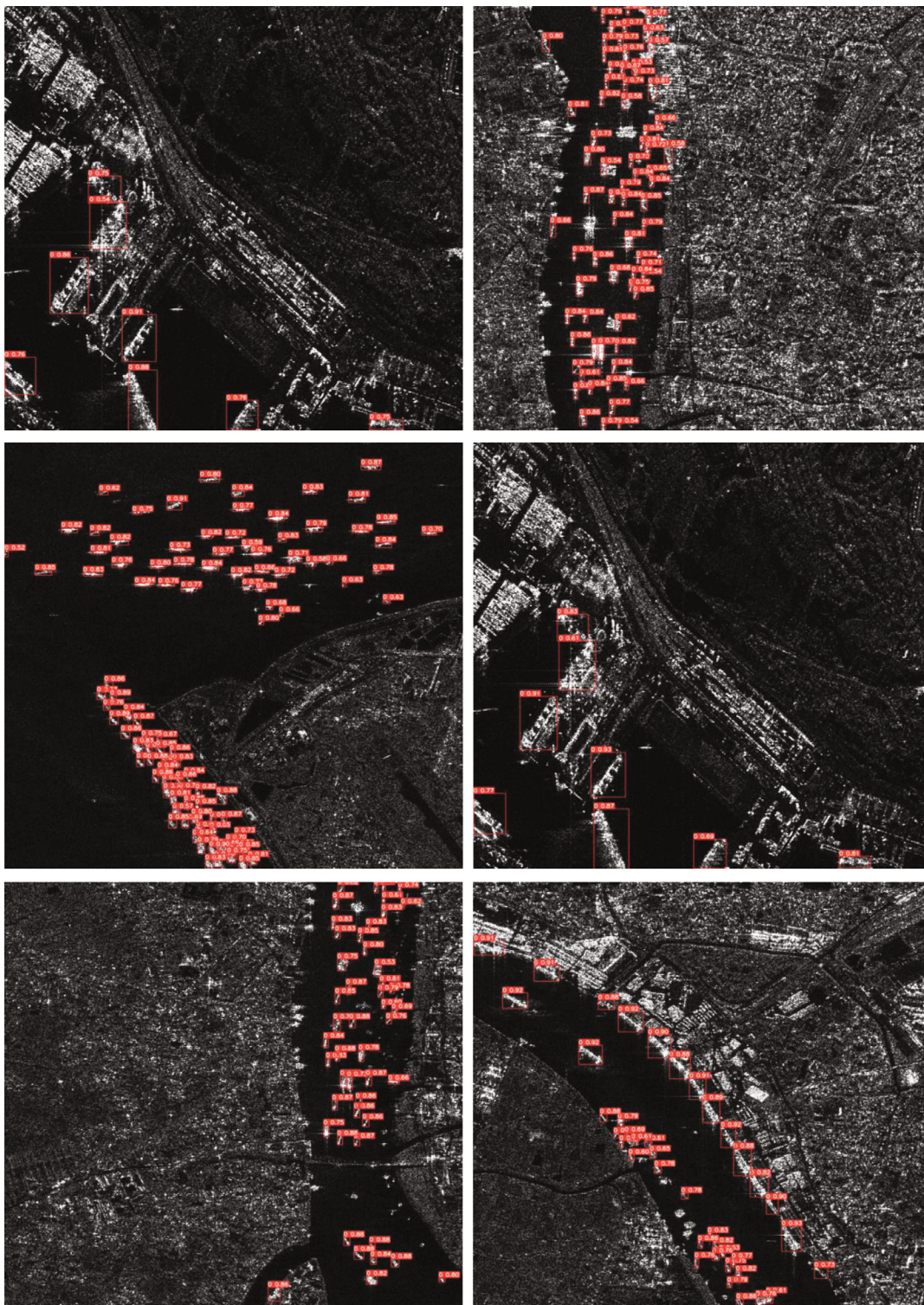


FIGURE 12: Detection effect of 3S-YOLO algorithm in complex background (detection model is reconstructed network +0.1prune + Varifocal-EIoU, model only 2.8 MB).

4.6. Comparison with Advanced Algorithms. In order to verify the advancement of the 3S-YOLO algorithm, a comparative experiment is carried out on the algorithm, and it is compared with the mainstream algorithm.

In the SSDD dataset, the algorithm comparison is shown in Table 10. When only the average accuracy (without prun-

ing) is considered, the average accuracy of 3S-YOLO can be improved to 99.2%, which is better than all methods. Compared with classical algorithms based on R-CNN, namely, Libra R-CNN, Cascade R-CNN, and Faster R-CNN, our method improves by about 9.3%-10.9%. Compared with classical single-stage algorithms, namely, RetinaNet, SSD,

YOLOv3, YOLOv4, and YOLOv5, our method improves by about 2%–9.6%. Compared with FCOS and CenterNet without anchor frames, it increased by 10.5% and 5.7%. It is improved by 4.4%–9.1% compared to other improved ship detection algorithms.

In the HRSID dataset, the algorithm comparison is shown in Table 11. When only the average accuracy (no pruning) is considered, the average accuracy of 3S-YOLO can be improved to 95.7%, which is better than all methods. Compared with the classical algorithm based on R-CNN, namely, Libra R-CNN, Cascade R-CNN, and Faster R-CNN, our method increases by about 16.5%–18.2%. Compared with single-stage classical algorithms, namely, RetinaNet, SSD, YOLOv3, YOLOv4, and YOLOv5, our method improves about 2.9%–13.2%. Compared with FCOS and CenterNet without anchor frames, it is increased by 9.1 and 9.4. It is improved by 5.4%–15.9% compared to other improved ship detection algorithms [22]. Our method is competitive. In addition, for different use environments, we can reduce the amount of calculation by adjusting the pruning rate to meet the requirements of different environments and various real-time detection requirements.

Overall, the 3S-YOLO algorithm has great advantages in model volume, average accuracy, and reasoning time, which can meet the basic needs of real-time ship monitoring in SAR images.

4.7. Experimental Effect and Analysis. In order to verify the actual detection effect of the algorithm, images are randomly selected in the verification set for verification. These images include far-sea and near-sea, multiple ships and single ships, small targets and large targets, and overlapping and non-overlapping categories. Figure 6 shows the comparison and visualization results with the baseline model. Figures 7–11 show the single ship the offshore, small target the offshore, dense ship the offshore, and sparse ship the offshore with different pruning rates. Figure 11 shows the visual result of the detection effect in complex background.

Figure 7(a) is the real test result, Figure 7(b) is the YOLOv5s baseline test result, and Figure 7(c) is the 3S-YOLO test result. 3S-YOLO has higher detection confidence than YOLOv5s and a lower missed detection and false detection rate.

Figures 8–11(a)–(e) are pruning rates 10%, 30%, 50%, 70%, and 90%, respectively. The detection results of a single ship and a small target ship in the sea are excellent; with the increase of pruning rate, the miss detection rate of offshore ship target detection will not increase, and the confidence is high. In offshore detection, the detection effect of sparse targets is excellent, but there is a problem of increasing missed detection rate in offshore dense target detection. When the pruning rate is less than 50%, there is no missing detection problem for offshore dense targets. When the pruning rate is 70% and 90%, a ship's missing detection occurs in the model.

In order to increase the display of the model detection effect, the detection effect diagram under complex background is shown in Figure 12. The detection model is the reconstructed network +0.1prune + Varifocal-EIoU, and

the model is only 2.8 MB. The test results show that the model can maintain high detection ability in complex backgrounds, and the missed detection rate and false detection rate of the model are low.

To sum up, 3S-YOLO has excellent detection effect in different complex environments at sea. When the pruning rate is high, the calculation amount and volume of the model are significantly reduced, and the detection effect will not be reduced in the detection of sparse targets in the offshore and offshore areas. However, in the face of dense targets in the offshore area, the missed detection rate will increase.

5. Conclusions

The ship attack effect of weapon equipment is greatly affected by the target detection effect. Accurate detection of ships can help strengthen science and technology and improve maritime combat and security capabilities. In order to better detect ship targets, this paper designs a 3S-YOLO algorithm, which takes into account the detection effect, and the model is more lightweight. Meet real-time detection in different scenarios. 3S-YOLO firstly designs the network structure according to the ship target characteristics of the SAR image; then, the redundant information is pruned by a pruning algorithm; finally, the Varifocal-EIoU loss function is designed to change the border regression loss function, balance positive, and negative samples and improve the detection accuracy of the detection box. The algorithm verification was carried out on SSDD and HRSID datasets, respectively, and the indicators of the 3S-YOLO algorithm were better than those of the original model. Compared with the current mainstream SAR image ship inspection algorithm, it has achieved excellent results. The accuracy of the 3S-YOLO algorithm can reach 99.2% and 95.67%, respectively. The minimum model volume can be reduced to 190.3 KB, and the minimum reasoning speed can reach 2.732 ms. Detection speed and model volume have great advancements.

Data Availability

The publicly available datasets used in this paper are SSDD (SAR ship detection dataset) and HRSID. 1160 data are available in SSDD, tagging 2456 ships. The dataset contains images of various resolutions, and the ocean environment is diverse. The HRSID dataset is constructed based on Sentinel-1B, TerraSAR-X, and TanDEM-X images with 5604 images, tagging 16951 ships.

Conflicts of Interest

The authors declare no conflict of interest.

Authors' Contributions

H.W. and S.Z. conceptualized the study. H.W. and S.Z. were responsible for the methodology. H.W. and S.Z. were responsible for the software. H.W. and S.Z. validated the study. H.W. and S.Z. were responsible for the formal

analysis. H.W. and S.Z. investigated the study. H.W. and S.Z. were responsible for the resources. H.W. and S.Z. were responsible for the data curation. H.W. and S.Z. were responsible for the writing—original draft preparation. H.W. and S.Z. were responsible for the writing—review and editing. H.W. and S.Z. visualized the study. H.W. and S.Z. supervised the study. H.W. and S.Z. were responsible for the project administration. H.W. and S.Z. were responsible for the funding acquisition. All authors have read and agreed to the published version of the manuscript.

Acknowledgments

This study was supported by the National Innovation and Entrepreneurship Training Program (202011075013).

References

- [1] B. O. Steenson, "Detection performance of a mean-level threshold," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 4, pp. 529–534, 1968.
- [2] L. M. Novak, M. C. Burl, and W. Irving, "Optimal polarimetric processing for enhanced target detection," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 29, no. 1, pp. 234–244, 1993.
- [3] H. Lim, D. Chae, J. H. Yoo, and K.-I. Kwon, "Template matching-based target recognition algorithm development and verification using SAR images," *Journal of the Korea Institute of Military Science and Technology*, vol. 17, no. 3, pp. 364–377, 2014.
- [4] E. Grosso and R. Guida, "A new automatic ship wake detection for Sentinel-1 imagery," in *IGARSS 2020-2020 IEEE International Geoscience and Remote Sensing Symposium*, pp. 1259–1262, IEEE, Waikoloa, HI, USA, 2020.
- [5] L. Debnath and F. A. Shah, *Wavelet Transforms and Their Applications*, Springer, Boston: Birkhäuser, 2002.
- [6] Y. Li, B. Tang, and Y. Yi, "A novel complexity-based mode feature representation for feature extraction of ship-radiated noise using VMD and slope entropy," *Applied Acoustics*, vol. 196, article 108899, 2022.
- [7] Y. Li, F. Ning, X. Jiang, and Y. Yi, "Feature extraction of ship radiation signals based on wavelet packet decomposition and energy entropy," *Mathematical Problems in Engineering*, vol. 2022, Article ID 8092706, 2022.
- [8] Y. Li, B. Geng, and S. Jiao, "Dispersion entropy-based Lempel-Ziv complexity: a new metric for signal analysis," *Chaos, Solitons & Fractals*, vol. 161, article 112400, 2022.
- [9] W. Liu, D. Anguelov, D. Erhan et al., "In SSD: single shot multibox detector," *European conference on computer vision*, vol. 9905, pp. 21–37, 2016.
- [10] J. Redmon and A. Farhadi, "In YOLO9000: better, faster, stronger," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 7263–7271, IEEE, Honolulu, HI, USA, 2017.
- [11] J. Redmon and A. Farhadi, "YOLOv3: an incremental improvement," 2018, <http://arXiv.1804.02767>.
- [12] A. Bochkovski, C.-Y. Wang, and H.-Y. M. Liao, "YOLOv4: optimal speed and accuracy of object detection," 2020, <http://arXiv.2004.10934>.
- [13] R. Girshick, "Fast R-CNN," in *Proceedings of the IEEE international conference on computer vision*, pp. 1440–1448, Santiago, Chile, 2015.
- [14] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: towards real-time object detection with region proposal networks," *Advances in Neural Information Processing Systems*, vol. 28, 2015.
- [15] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," *Proceedings of the IEEE international conference on computer vision*, pp. 2961–2969, Venice, Italy, 2017.
- [16] J. Yu, Y. Jiang, Z. Wang, Z. Cao, and T. Huang, "UnitBox: an advanced object detection network," in *Proceedings of the 24th ACM international conference on Multimedia*, pp. 516–520, ChengDu, SiChuan, China, 2016.
- [17] G. I. O. Union, "A metric and a loss for bounding box regression," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 658–666, Long Beach, CA, USA, 2019.
- [18] Z. Zheng, P. Wang, W. Liu, J. Li, R. Ye, and D. Ren, "Distance-IoU loss: faster and better learning for bounding box regression," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 7, pp. 12993–13000, 2020.
- [19] Z. Zheng, P. Wang, D. Ren et al., "Enhancing geometric factors in model learning and inference for object detection and instance segmentation," *IEEE transactions on Cybernetics*, vol. 52, no. 8, pp. 8574–8586, 2021.
- [20] Y. He, P. Liu, Z. Wang, Z. Hu, and Y. Yang, "Filter pruning via geometric median for deep convolutional neural networks acceleration," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4340–4349, Long Beach, CA, USA, 2019.
- [21] Y. F. Zhang, W. Ren, Z. Zhang, Z. Jia, L. Wang, and T. Tan, "Focal and efficient IOU loss for accurate bounding box regression," <http://arXiv.2101.08158>.
- [22] X. Yang, X. Zhang, N. Wang, and X. Gao, "A robust one-stage detector for multiscale ship detection with complex background in massive SAR images," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, pp. 1–12, 2021.
- [23] J. Pang, K. Chen, J. Shi, H. Feng, W. Ouyang, and D. Lin, "Libra R-CNN: towards balanced learning for object detection," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 821–830, Long Beach, CA, USA, 2019.
- [24] Z. Cai and N. Vasconcelos, "Cascade R-CNN: delving into high quality object detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 6154–6162, IEEE, Salt Lake City, UT, USA, 2018.
- [25] Y. Yu, X. Yang, J. Li, and X. Gao, "A cascade rotated anchor-aided detector for ship detection in remote sensing images," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, pp. 1–14, 2020.
- [26] Z. Cui, Q. Li, Z. Cao, and N. Liu, "Dense attention pyramid networks for multi-scale ship detection in SAR images," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 57, no. 11, pp. 8983–8997, 2019.
- [27] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *Proceedings of the IEEE international conference on computer vision*, pp. 2980–2988, Venice, Italy, 2017.
- [28] Z. Tian, X. Chu, X. Wang, X. Wei, and C. Shen, "Fully convolutional one-stage 3D object detection on LiDAR range images," 2022, <http://arXiv.2205.13764>.

- [29] X. Zhou, D. Wang, and P. Krähenbühl, "Objects as points," 2019, <http://arXiv.1904.07850>.
- [30] H. Guo, X. Yang, N. Wang, and X. Gao, "A CenterNet++ model for ship detection in SAR images," *Pattern Recognition*, vol. 112, article 107787, 2021.