IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received July 9, 2019, accepted July 20, 2019, date of publication July 24, 2019, date of current version August 14, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2930939

A Deep Neural Network Based on an Attention Mechanism for SAR Ship Detection in Multiscale and Complex Scenarios

CHEN CHEN[1], CHUAN HE[1,2], CHANGHUA HU[1], HONG PEI[1], AND LICHENG JIAO[2], (Fellow, IEEE)

[1]Department of Automation, Xi'an Institute of High-Technology, Xi'an 710025, China
[2]School of Artificial Intelligence, Xidian University, Xi'an 710071, China

Corresponding author: Chuan He (hechuan8512@163.com)

This work was supported by the National Natural Science Foundation of China under Grant 61773389, Grant 61833016, and Grant 61573365.

ABSTRACT Synthetic aperture radar (SAR) ship detection based on deep learning has been widely applied in recent years. However, two main obstacles are hindering SAR ship detection. First, the identification of ships in a port is seriously disrupted by the presence of onshore buildings. It is difficult for the existing detection algorithms to effectively distinguish the targets from such a complex background. Additionally, it appears more complicated to accurately locate densely arranged ships. Second, the ships in SAR images exist at a variety of scales due to multiresolution imaging modes and the variety of ship shapes; these pose a much greater challenge to ship detection. To solve the above problems, this paper proposes an object detection network combined with an attention mechanism to accurately locate targets in complex scenarios. To address the diverse scales of ship targets, we construct a loss function that incorporates the generalized intersection over union (GIoU) loss to reduce the scale sensitivity of the network. For the final processing of the results, soft nonmaximum suppression (Soft-NMS) is also introduced into the model to reduce the number of missed detections for ship targets in the presence of severe overlap. The experimental results reveal that the proposed model exhibits excellent performance on the extended SAR ship detection dataset (SSDD) while achieving real-time detection.

INDEX TERMS Ship detection, synthetic aperture radar (SAR), deep neural network, attention mechanism.

I. INTRODUCTION

With the advancement of the ocean industry, ships are playing an increasingly essential role in marine development and transportation. Suitable means of monitoring and controlling ships can effectively improve the efficiency of marine transportation and reduce maritime traffic accidents [1]-[3]. Synthetic aperture radar (SAR) is widely used in marine ship detection because of its advantageous independence from solar illumination and ability to provide images of the ocean in all-weather operating conditions [4], [5]. In recent years, the rapid development of TerraSAR-X, RADARSAT-2 and Sentinel-1 has promoted research on ship detection in SAR images [6], [7].

Due to the strong feature extraction capabilities of convolutional neural networks (CNNs), deep learning has achieved great success in object detection tasks. Object detection methods based on deep learning can be divided into two main categories: two-stage detection algorithms, including Faster R-CNN [8], and single-stage detection algorithms such as SSD [9], RFBNet [10], and YOLO [11]-[13]. Two-stage detection algorithms offer high positioning accuracy, whereas single-stage detection algorithms have an absolute advantage in terms of speed. Both classes of algorithms are widely applied in automated driving, intelligent security, remote sensing detection and other fields. For SAR image object detection tasks, compared with traditional constant false alarm rate (CFAR) algorithms [14], [15], ship detection algorithms based on deep learning do not require complex modeling processes; consequently, they have attracted considerable research interest from scholars. Li *et al.* applied the various training strategies to improve the Faster R-CNN detection algorithm for ship detection in SAR images [16]. Kang *et al.* developed a detection algorithm combining

The associate editor coordinating the review of this manuscript and approving it for publication was Wei Liu.

traditional CAFR and Faster R-CNN. The region generated by Faster R-CNN was used as the guard window of CAFR to obtain the location information of targets [17]. Jiao *et al.* proposed a densely connected multiscale neural network based on a Faster R-CNN framework. The method leveraged the densely connected network as its main trunk to detect ship targets [18]. Kang *et al.* proposed a region-based CNN that combines context information and shallow location features with deep semantic features to improve the accuracy of ship target location [19]. However, some problems still arise for ship detection in SAR images based on deep learning. First, the background for ships adjacent to a port is complex and is seriously disturbed by the wharf and onshore buildings. The above algorithms cannot effectively distinguish targets from such complex scenes. In particular, when the ships are densely arranged, the above algorithms cannot accurately locate them. Furthermore, the areas of overlap between the detection boxes for different ships will be quite large; unfortunately, however, a target with a large overlap region will be discarded after the operation of nonmaximum suppression (NMS). Second, SAR ship targets exhibit a broad diversity of scale due to the multiresolution imaging modes and the variety of ship shapes, making them difficult for existing algorithms to effectively detect and locate, especially for small-scale ship targets. Finally, most SAR ship detection algorithms based on deep learning adopt a two-stage detection framework based on Faster R-CNN, which emphasizes detection accuracy while ignoring the detection speed. This results in failure to detect the targets in real time.

Visual attention models have been widely applied in object detection, object recognition, object tracking and other fields [20]. The core idea of an attention mechanism is to help a model learn to focus on key information while ignoring irrelevant information [21]. An object detection algorithm based on a visual attention mechanism usually obtains a saliency feature map by means of the attention model and then calibrates the targets in the image by analyzing the saliency map. For the task of ship detection, Song *et al.* combined the sparse saliency of the targets obtained through an attention model with the local binary pattern (LBP) features and proposed an automatic ship detection algorithm for optical satellite images. This algorithm exhibits good robustness against interference from clouds and varying lighting conditions [22]. With the development of deep learning in the field of computer vision, it is becoming increasingly important to build neural networks equipped with attention mechanisms. On the one hand, such a neural network can independently learn the attention mechanism. On the other hand, the attention mechanism can in turn contribute to the understanding of the neural network [23]. In the context of combining visual attention mechanisms and neural networks, the work reported in Reference [24] imitated the characteristics of human attention and learned the corresponding weights through a CNN. Then, the weights were reassigned to the feature matrix, and features with high weights were selected as the focus of attention. Wang *et al*. proposed a residual network combined with an attention model that achieved good results in image classification [25]. Zheng *et al*. proposed a component learning method using a CNN based on a multiattention model, which enabled the network to obtain more fine-grained image features [26].

When building a SAR ship detection model based on a CNN, it is necessary to fully consider the differences between optical images and SAR images and to design the CNN model accordingly. In this paper, we propose a single-stage object detection algorithm combined with an attention mechanism to solve the current problems arising in the context of ship detection in SAR images. The attention module proposed in [25] employs a hybrid attention mechanism of spatial attention and channel attention that can effectively extract the salient features of the target. Inspired by this attention mechanism, this paper designs an object-detection network that additionally combines an attention mechanism. We have integrated the attention module proposed in [25] into the detection network to extract a salient-feature map and enhance the difference between the target and background. The network proposed in this paper is different from that in [25], where the attention mechanism at a single level is used to highlight features that are more advantageous to classification. In contrast, this paper constructs a multilevel feature pyramid, uses the attention model to obtain the salient features of different levels, and fuses the salient features of different levels. As a consequence, the proposed network has more accurate feature expression ability for targets in complex scenarios. Another important difference is that the network proposed in [25] focuses more on the semantic information of the target than on the multiscale characteristics. However, the ships in SAR images exist at an obvious variety of scales due to multiresolution imaging modes and the variety of ship shapes. In this paper, the Inception module [29] is employed at different levels of the network to address the multiscale problem of the target. Feature information of different scales is activated on different branches, which improves the information transmission between upper and lower levels. In addition, at the end of the network, we assign different sizes of feature maps to predict different scales of targets, which improves the adaptability of the network to different scales. The main contributions of this paper are as follows.

1) In view of the complex scenarios encountered in SAR images containing ships, an end-to-end network structure for SAR ship detection is proposed. We integrate an attention mechanism into the network to obtain salient feature maps at different depths and fuse corresponding multiscale features, thereby improving the accuracy of the network in detecting and locating densely arranged ship targets against complex backgrounds.
2) A loss function that incorporates the generalized intersection over union (GIoU) loss [27] is constructed to reduce the scale sensitivity of the network in order to address the multiscale characteristics of ships in SAR images.

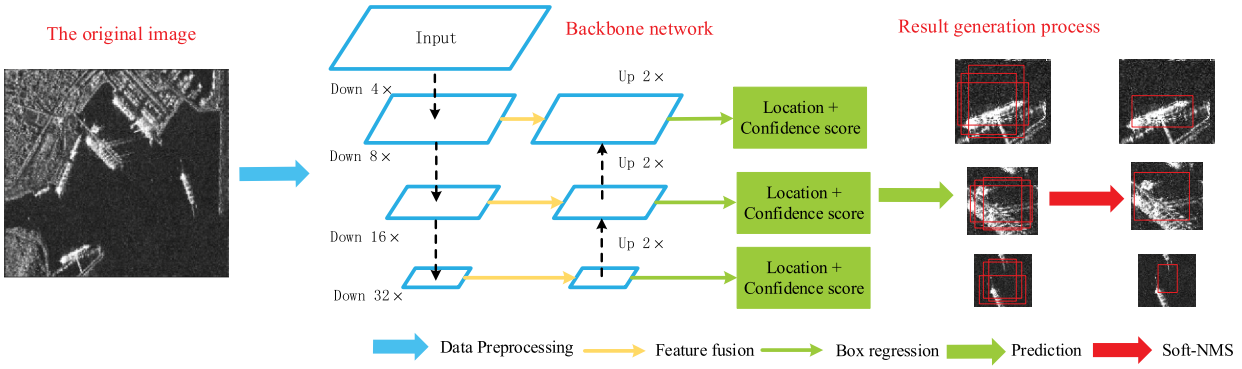


FIGURE 1. The main flow of the algorithm. The blue boxes represent the salient-feature maps at different depths enhanced by the attention mechanism. We use differently colored arrows to represent operations such as data preprocessing.

3) Soft nonmaximum suppression (Soft-NMS) [28] is integrated into the final processing of the results to improve the detection rate for ship targets with large overlaps in their detection regions.
4) The proposed model is based on a single-stage object detection algorithm and thus can achieve a good detection effect while maintaining a fast detection speed. Therefore, it can support real-time ship target detection.

The rest of the paper is organized as follows. Section II illustrates our proposed method and network structure. Section III introduces the dataset used in our experiments and describes the experimental details. The results and possibilities for future work are discussed in Section IV. Section V presents the conclusions.

II. METHODS

This paper proposes a method of ship detection in SAR images based on an attention mechanism. The main flow of the algorithm is shown in Fig. 1. First, after the original image is preprocessed, it is used as the input to the network. Second, a backbone network is constructed from Inception modules [29] to obtain multilevel target mapping features. Third, the saliency of the mapping features is enhanced by means of the attention mechanism to obtain saliency feature maps; then, the saliency features expressed at different depths are fused via a feature fusion method. On the fused feature maps, the locations and confidence scores of the targets are predicted. Finally, the predicted boxes are filtered via Soft-NMS [28], and the final detection results are obtained.

A. CONSTRUCTING THE FEATURE EXTRACTION NETWORK

To handle the characteristics of ship targets in SAR images, we select Inception-ResNet modules [29] as the basic units for constructing the feature network and acquiring the image feature pyramid. The network structure used in the algorithm is shown in Fig. 2. The residual part of the ResNet architecture [32] is replaced with an Inception module in this network. With this extension of the Inception module, the ability of the network to transmit higher-level information is enhanced. A shortcut method is introduced to overcome

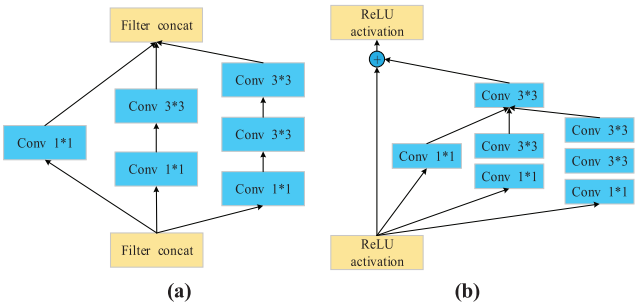


FIGURE 2. An Inception-ResNet module. (a) The original Inception module. (b) Inception combined with ResNet.

the phenomenon of gradient disappearance and increase the depth of the network. We superimpose two convolutions with dimensions of 3∗3 in the Inception branch to obtain receptive fields (RFs) with dimensions of 5∗5 [10], [30]. Larger RFs can capture a wider range of information, which is beneficial for distinguishing ship targets from complex backgrounds. Inception modules are introduced to form a multibranch convolution structure, and the convolution cores of different sizes in each branch increase the diversity of the feature information obtained. In this network, a 1∗1 convolution channel is adopted for dimension reduction, thereby reducing the number of parameters of each Inception module. At the same time, linear convolution is used for dimension stitching to match the input and output dimensions. After each convolution layer, a batch normalization (BN) layer and a leaky rectified linear unit (ReLU) layer are applied to accelerate the convergence of the network and avoid overfitting.

B. OBJECT DETECTION NETWORK WITH AN ATTENTION MECHANISM

In this section, we propose a detection network integrated with an attention module in [25] to extract the salient-feature map and enhance the difference between the target and background.

The attention module is mainly composed of two branches: a convolution branch and a mask branch. The mask branch has a symmetrical, hourglass-like network structure mainly

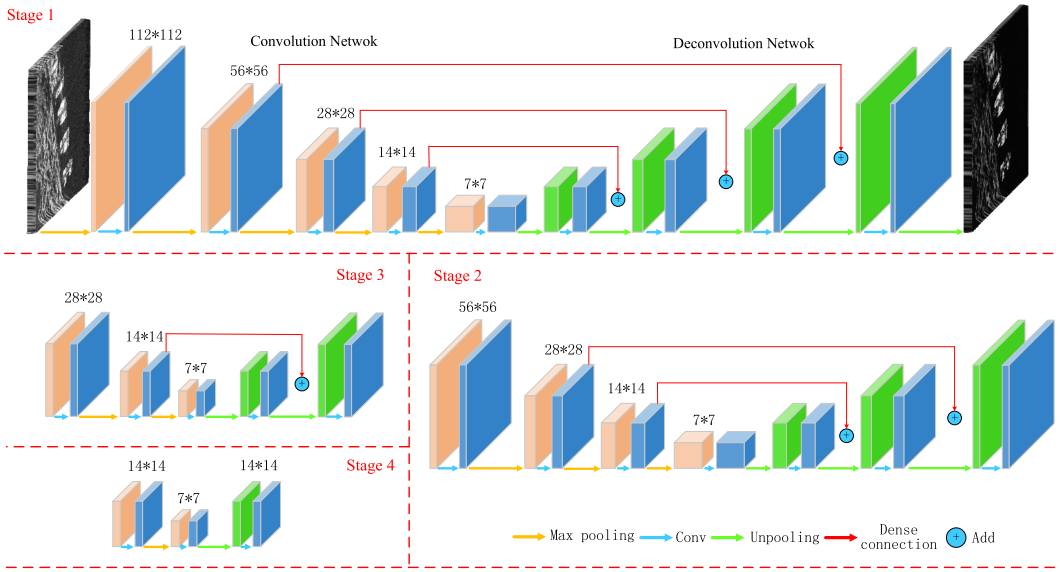


FIGURE 3. Different stages of the mask branch of the attention model. The yellow and green boxes represent the feature maps after pooling, and the blue boxes represent the feature maps acquired from the convolution layers; the numerical values represent the feature-map dimensions. We use arrows to represent operations such as convolution and pooling. Through the depicted process, the mask map of the targets can be learned.

composed of a convolution and a deconvolution network [31] as shown in Fig. 3.

During the process of convolution, max pooling is performed several times to extract the representative activation values in the RFs; then, the high-dimensional features of the targets are obtained through the convolution layers; and finally, the corresponding mask is learned through the deconvolution network. Repeated pooling operations result in the loss of location information, which is detrimental to the accurate localization of the targets in the detection task. Therefore, in the deconvolution network, unpooling is introduced to recover the original feature map dimensions [31]. In addition, we use a dense connection approach to fuse the features from different layers, thereby further highlighting the information characteristics of the mask maps. The mask maps act on the convolutional feature maps through Eq. (1). The saliency features are obtained by multiplying the corresponding elements of the mask and feature maps. In this way, the elements in the mask map play a role similar to that of weights for the feature map, enhancing regions of interest and suppressing nontarget regions.

$$A_{i,n}(x) = M_{i,n}(x) * C_{i,n}(x). \tag{1}$$

where A is the output of the attention model, M is the mask generated by the mask branch, C represents the features generated by the convolution branch, i an index representing the positions of different points in space, and n is an index representing the different convolution channels.

The network structure of an attention module is shown in Fig. 4(a). The output M of the mask branch is used as a set of control gates for the neurons of the convolution branch. To avoid differences between features maps at

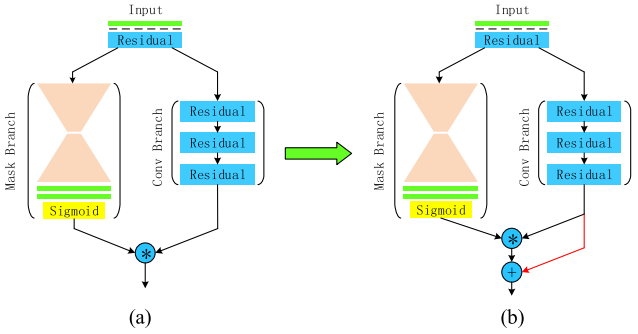


FIGURE 4. An attention module. The hourglass structure represents the mask branch. The green bars represent 1∗1 convolutions, which are used to adjust the dimensions. The red arrow represents an identical mapping branch. (a) An attention module. (b) An attention module combined with identical mapping.

different levels caused by the attention model, the sigmoid activation function is used to normalize the values of the pixels in the mask maps to the range of [0,1]. It should be noted that attention mode can be selected by changing the normalization step in the activation function. Reference [25] shows that hybrid attention can achieve better results than spatial attention or channel attention alone. Therefore, this paper adopts the mode of hybrid attention, which can be realized by the basic sigmoid activation function. As shown in Eq. (2).

$$f(x_{i,n}) = \frac{1}{1 + e^{-x_{i,n}}}. \tag{2}$$

However, in the process of constructing the network, multiple attention modules are stacked and multiplied, thus repeatedly reducing the element values in the feature maps. This process will ultimately destroy the original characteristics

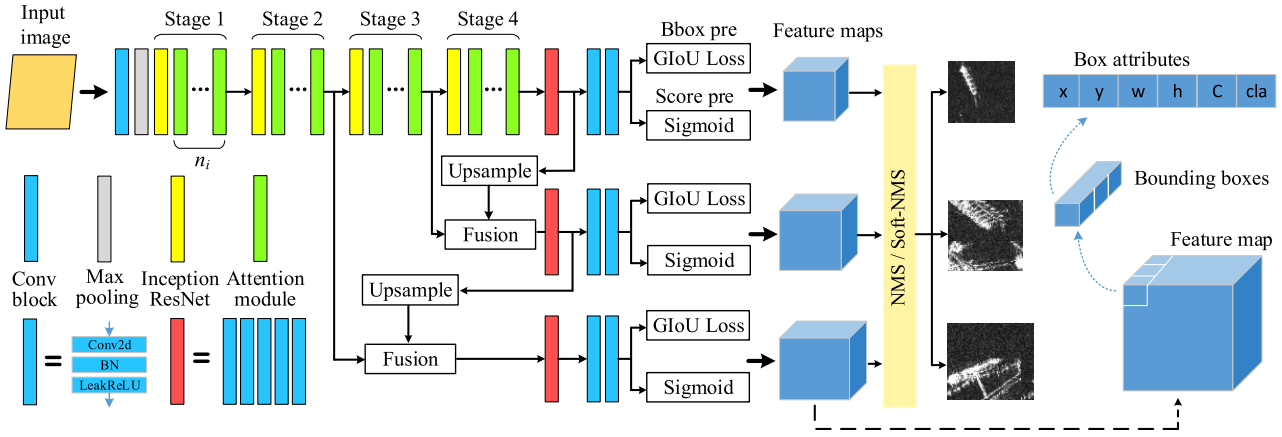


FIGURE 5. Structure of the proposed object-detection network with an attention mechanism.

of the network. In deeper network layers, it is easier to fall into local optima. Therefore, the concept of identical mapping that is used in ResNet [32] is also introduced into the attention model. In the convolution branch, an identical mapping branch is added to the original output, as shown in Fig. 4(b). On the one hand, the identical mapping concept solves the problem of gradient disappearance in the network, allowing the network depth to be increased; on the other hand, through the addition operation, the salient features of the model output become more obvious, and the discrimination of target features is enhanced. The corresponding network can be described as follows:

$$\begin{aligned} A_{i,n}(x) &= M_{i,n}(x) * C_{i,n}(x) + C_{i,n}(x) \\ &= \left(1 + M_{i,n}(x)\right) * C_{i,n}(x). \end{aligned} \tag{3}$$

Most existing algorithms combine low-level location information with high-level semantic information to solve the problem of missing target location information in the process of network downsampling [13], [33]. However, the identification of ships in SAR images is seriously disrupted by complex backgrounds, and it is difficult to obtain effective location features in a low-level network. If effective location information cannot be obtained in a low-level network, then the fusion of features of different depths will also be meaningless. Therefore, a new feature fusion method is proposed in this paper. First, the attention model is integrated into the detection network to enhance the saliency of the target location information in the shallow features. Then, saliency features of different depths are fused using the structure mode of the Feature Pyramid Network (FPN) feature extractor [33] to retain more semantic information while also ensuring the accuracy of the location information.

The overall network structure is shown in Fig. 5. First, the dimensions of the input image are adjusted by a 7∗7 convolution layer, and then, downsampling is performed by a max pooling layer. The backbone network consists of four stages. In each stage, an Inception-ResNet module is used as the basic unit to construct the feature pyramid, thereby enhancing the ability to acquire higher-level information. Moreover, the convolution layer in each Inception-ResNet module can connect different stages by changing the size and dimensions of the feature map. In each stage, saliency feature maps at different depths are obtained by concatenating several attention modules in series, and the features from different depths are fused to highlight the location information. The details of the proposed backbone network are shown in Table 1.

The output of the network is a set of feature maps of three different scales, which are obtained by downsampling the dimensions of the input image by factors of 32, 16 and 8. The algorithm makes predictions based on these three-scale feature maps. In detail, the tensors of the feature maps are divided into different numbers of grid cells according to their scales. Each grid cell includes the location attributes of the corresponding bounding boxes and the confidence scores of the corresponding objects. After filtering by means of a confidence threshold and NMS, the final prediction results are obtained.

C. LOSS FUNCTION

The mean squared error (MSE) loss is used as a loss function in most detection algorithms to evaluate the effect of bounding box regression. However, for SAR ship targets, the target sizes vary greatly with different resolutions, and the MSE loss is sensitive to scale [27]. Therefore, using the MSE loss as the loss function will affect the positioning accuracy for ship targets. In this paper, we integrate the GIoU [27] mechanism into the loss function to reduce the scale sensitivity of the loss function. The GIoU is defined as follows:

$$GIoU = IoU - \frac{|C \setminus (A \cup B)|}{|C|}, \quad IoU = \frac{|A \cap B|}{|A \cup B|}. \tag{4}$$

where A and B are arbitrary convex shapes and C is the smallest convex shapes enclosing both A and B. $|C \setminus (A \cup B)|$ is the area occupied by C that excludes $A \cup B$. The GIoU loss can be defined as $L_{GIoU} = 1 - GIoU$. In our scheme, the following method is used to calculate L_{GIoU}. First, the coordinates of

TABLE 1. The details of the proposed backbone network.

Stage	Layer	Input	Kernel Size	Stride	Output	Memory
	Conv	3@448×448	7×7	2	64@224×224	64×224×224
	Max pool	64@224×224	2×2	2	128@112×112	128×112×112
1	Inception-ResNet-1	128@112×112	3×3, 1×1	1	128@112×112	128×112×112
	Attention model	128@112×112	3×3	1	128@112×112	128×112×112
2	Inception-ResNet-2	128@112×112	3×3, 1×1	2	256@56×56	256×56×56
	Attention model	256@56×56	3×3	1	256@56×56	256×56×56
	Attention model	256@56×56	3×3	1	256@56×56	256×56×56
3	Inception-ResNet-3	256@56×56	3×3, 1×1	2	512@28×28	512×28×28
	Attention model	512@28×28	3×3	1	512@28×28	512×28×28
	Attention model	512@28×28	3×3	1	512@28×28	512×28×28
	Attention model	512@28×28	3×3	1	512@28×28	512×28×28
	Attention model	512@28×28	3×3	1	512@28×28	512×28×28
4	Inception-ResNet-4	512@28×28	3×3, 1×1	2	1024@14×14	1024×14×14
	Attention model	1024@14×14	3×3	1	1024@14×14	1024×14×14
	Attention model	1024@14×14	3×3	1	1024@14×14	1024×14×14
	Attention model	1024@14×14	3×3	1	1024@14×14	1024×14×14
	Attention model	1024@14×14	3×3	1	1024@14×14	1024×14×14

the predicted bounding box and the ground-truth bounding box are obtained from the location information x, y, w, and h predicted by the network.

$$\begin{aligned} x_1 &= w/2 - x, \quad x_2 = w/2 + x \\ y_1 &= h/2 - y, \quad y_2 = h/2 + y. \end{aligned} \tag{5}$$

In Eq. (5), x_1, y_1, x_2, and y_2 are the coordinates of the predicted bounding box, and the corresponding area is denoted by S_p. The coordinates of the ground-truth bounding box, x_1^*, y_1^*, x_2^*, and y_2^* can also be calculated according to the object label, and the corresponding area is denoted by S_g. We can then calculate the intersection area S_I and the union area S_U between two boxes, as shown in Eq. (6) and Eq. (7), respectively:

$$S_I = \begin{cases} \left(x_2^I - x_1^I\right) * \left(y_2^I - y_1^I\right) & x_2^I > x_1^I, y_2^I > y_1^I \\ 0 & otherwise. \end{cases} \tag{6}$$

$$S_U = S_p + S_g - S_I. \tag{7}$$

where $x_1^I = \max\left(x_1, x_1^*\right)$, $x_2^I = \min\left(x_2, x_2^*\right)$ $y_1^I = \max\left(y_1, y_1^*\right)$, and $y_2^I = \min\left(y_2, y_2^*\right)$. Furthermore, the smallest enclosing box B_C can be determined, and the corresponding area S_C can be formulated as follows:

$$S_C = \left(x_2^C - x_1^C\right) * \left(y_2^C - y_1^C\right). \tag{8}$$

where $x_1^C = \min\left(x_1, x_1^*\right)$, $x_2^C = \max\left(x_2, x_2^*\right)$, $y_1^C = \min\left(y_1, y_1^*\right)$, and $y_2^C = \max\left(y_2, y_2^*\right)$. Based on the above derivations, $GIoU$ and L_{GIoU} can be calculated as follows:

$$GIoU = IoU - \frac{S_C - S_U}{S_C}, \tag{9}$$

$$L_{GIoU} = 1 - GIoU. \tag{10}$$

It can be seen that the GIoU is invariant with respect to the scale. Thus, integrating L_{GIoU} into the loss function in place of the original MSE loss for bounding box regression can effectively reduce the scale sensitivity of the algorithm. The loss function of the network consists of two terms: a confidence score loss, $Loss_{conf}$, and a location loss, $Loss_{coord}$. Finally, the loss function that incorporates $GIoU$ can be expressed as follows:

$$\begin{aligned} Loss &= Loss_{conf} + Loss_{coord} \\ &= \alpha \sum_{i=1}^{S^2} \sum_{j=1}^{B} \left[l_{ij}^{obj} f\left(C_{ij}, C_{ij}^*\right) + l_{ij}^{noobj} f\left(C_{ij}, C_{ij}^*\right) \right] \\ &\quad + \beta \sum_{i=1}^{S^2} \sum_{j=1}^{B} l_{ij}^{obj} L_{GIoU}. \end{aligned} \tag{11}$$

where S denotes the number of grid cells into which the feature map predicted by the network is partitioned and B represents the number of bounding boxes contained in each grid cell. l_{ij}^{obj} represents the number of bounding boxes predicted to contain objects, whereas l_{ij}^{noobj} is the number of bounding boxes predicted to contain no objects. α is the balance factor for the confidence score, expressed as $\left(C_{ij} - C_{ij}^*\right)^2$, where C and C^* are the confidence score and the corresponding label, respectively. β is the balance factor for the location loss, expressed as $[2 - (w * h) / (w_{in} * h_{in})]$, where w and h represent the width and height, respectively, of the object and w_{in} and h_{in} represent the dimensions of the network input. β is negatively correlated with the object area. When the size of a target is small, the weight of this target in the loss function can be increased to improve the detection effect of the algorithm. Since the output of the network is normalized by the sigmoid activation function, using the cross-entropy loss function shown in Eq. (12) as the confidence score loss

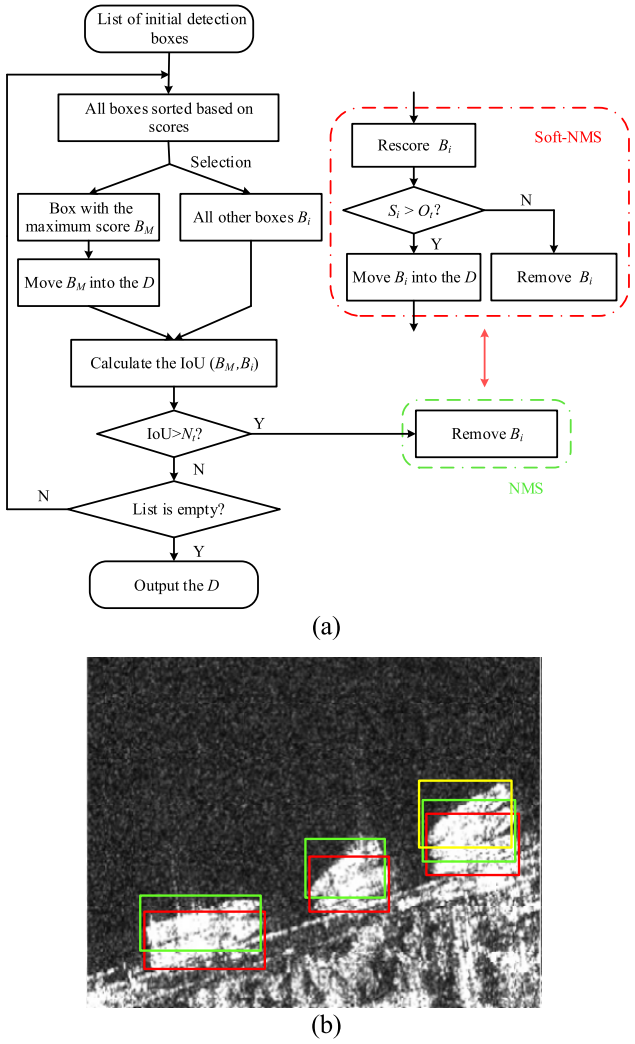


FIGURE 6. The algorithm flow of NMS and an illustration of the problem with NMS for the SAR ship detection task. (a) The algorithm flow of NMS. (b) An illustration of the problem.

can lead to a convergence effect.

$$f\left(C_{ij}, C_{ij}^{*}\right) = C_{ij}^{*}\log\left(C_{ij}\right) + \left(1 - C_{ij}\right)\log\left(1 - C_{ij}^{*}\right). \quad (12)$$

D. SOFT-NMS FOR FINAL PROCESSING OF THE RESULTS

NMS is applied in most state-of-the-art detectors to obtain the final results because it significantly reduces the number of false positives [8]-[10], [13]. The algorithm flow of NMS is shown in Fig. 6(a). First, the initial detection boxes in the list are sorted by their confidence scores S_i. Second, the detection box with the maximum score, B_M, is moved to the final detection list, D, and all other detection boxes are assigned unique identifiers B_i. Third, any box B_i that has an overlap area with B_M that is greater than some threshold N_t is removed. This process is repeated for the remaining boxes B_i until the initial list is empty. However, ships near a harbor are typically densely arranged; therefore, using the NMS algorithm to process the results will result in missed detections. This problem is illustrated in Fig. 6(b). The red and green rectangles represent detection results for different targets and correspond to different confidence scores. We assume that the red rectangles have the highest scores, followed by green and yellow. If the results are processed by the NMS algorithm, the green rectangles will be deleted because of their large overlaps with the red rectangles. Hence, the ship targets marked with green rectangles will be ignored, even if they actually exist. This will reduce the average precision of ship detection. Therefore, Soft-NMS [28] is introduced in place of the original NMS algorithm to process the results of ship detection.

The original NMS algorithm can be expressed as a rescoring function, as shown in Eq. (13):

$$S_i = \begin{cases} S_i, & IoU\left(B_M, B_i\right) < N_t \\ 0, & IoU\left(B_M, B_i\right) \geq N_t. \end{cases} \quad (13)$$

where S_i is the score of detection box B_i, B_M is the detection box with the maximum score, and N_t is the overlap threshold. In NMS, a hard threshold is set to decide which boxes should be kept and which should be removed in the neighborhood of B_M. If an object actually exists but has an overlap rate with B_M that is greater than N_t, its detection will be missed. The core idea of Soft-NMS is to attenuate the scores of detection boxes that have large overlaps with B_M by means of a penalty function instead of directly setting those scores to zero. Soft-NMS can be expressed as follows:

$$S_i = \begin{cases} S_i, & IoU\left(B_M, B_i\right) < N_t \\ S_i e^{-\frac{IoU\left(B_M, B_i\right)^2}{\sigma}}, & IoU\left(B_M, B_i\right) \geq N_t. \end{cases} \quad (14)$$

where $e^{-\frac{IoU\left(B_M, B_i\right)^2}{\sigma}}$ is a Gaussian penalty function and σ is an empirically selected hyperparameter. It is clear that the scores for detection boxes that have larger overlaps with B_M will be more strongly reduced, whereas detection boxes that are far away from B_M will not be affected. If the score of such a penalized box is still higher than the evaluation threshold O_t, then that box will be retained rather than discarded. By integrating this soft method into the model, the missed detection rate for ship targets in a harbor can be reduced.

III. EXPERIMENTS

In this section, we describe the experiments carried out in this study, including the data preprocessing, network training, experimental details, and analysis of the experimental results.

A. INTRODUCTION TO THE EXPERIMENTAL PLATFORM AND DATASET

All experiments were implemented on a workstation with an Intel(R) Xeon Silver 4114@2.20 Hz×40 CPU, an NVIDIA GTX TITAN-XP GPU, 128 GB of memory and the Keras framework. The initial learning rate of the network was set to 0.001. The optimization algorithm used stochastic gradient descent (SGD), with a momentum parameter of 0.9 and an attenuation coefficient of 0.00004. Warm-up [34] was

TABLE 2. The details of SSDD.

Polarization	Resolution	Scenario	Number
HH, VV HV, VH	1 m-15 m	near-shore offshore	1706

introduced during the initial training stage to avoid gradient explosion; the corresponding number of epochs was 3. To evaluate the detection performance of the model, the open SAR Ship Detection Dataset (SSDD) [16] was utilized in the experiments. This dataset includes ship objects at different resolutions (1 m to 15 m) and of different sizes against different scenarios (near-shore and offshore). The scene diversity of the samples ensured that the trained model would have a strong generalization ability. In addition, because the ship targets are too small to detect in low-resolution images, only ship targets consisting of more than three pixels are labeled. In summary, the dataset contains 1160 images of different scenes with multiscale ship targets. To make the trained model more robust, we extended the dataset as follows. A total of 14 SAR images containing ship targets were cut into small slices and labeled in the PASCAL VOC format [35]. Finally, the number of images in the dataset was expanded to 1706. We divided the data into a training set, a verification set, and a test set at a ratio of 7:1:2. The details of SSDD are shown in Table 2.

B. DATA PREPROCESSING

1) ANCHOR BOX SELECTION BASED ON K-MEANS++

If the network directly predicts the width and height of each target bounding box, this will affect the stability of the gradient for the object detection task [12]. Consequently, most existing object detection algorithms instead predict the offset between the ground-truth box and an anchor box. The implementation is shown in Fig. 7(a), where P_w and P_h denote the width and height, respectively, of the anchor box. Since the anchor box represents prior knowledge obtained from the statistics of the training samples, selection of a reasonable anchor box size can effectively improve the ability of the model to detect some objects of unknown size and shape. In this study, the dataset was reclustered using the k-means++ algorithm. Nine anchor boxes were obtained, three for each scale (138∗64, 121∗191, 81∗43, 50∗148, 43∗83, 31∗27, 21∗68, 12∗17, and 12∗38), with an average intersection over union (IoU) of 0.67. On the one hand, compared with manual anchor box selection [8], [16], [18], clustering can achieve better results. On the other hand, k-means++ reduces the impact of the initial value selection on the clustering results compared with the k-means algorithm [36], [37] and makes the loss function converge faster. The distance metric used in the clustering process is as follows:

$$D\left(box, centroid\right) = 1 - IoU\left(box, centroid\right). \quad (15)$$

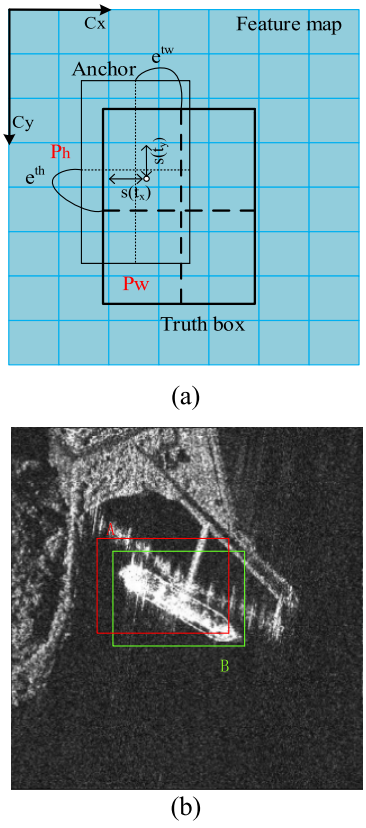


FIGURE 7. Illustrations of an anchor box and the IoU. (a) An anchor box. (b) A diagram illustrating the IoU algorithm.

where *box* denotes the bounding box area and *centroid* represents the cluster centroid. *IoU* illustrated in Fig. 7(b) and is calculated as follows:

$$IoU = (A \cap B) \big/ (A \cup B). \quad (16)$$

The use of prior information will make the training of the neural network more meaningful and improve the performance of the algorithm to some extent. In future work, we will consider some other more effective prior-information mining methods.

2) DATA AUGMENTATION

It is more difficult to obtain labeled SAR image data than it is to obtain common labeled images [39]. Therefore, it was necessary to expand the existing labeled images to achieve a sufficient number of samples for model training in this study. Data augmentation [34] is implemented via a series of transformations in order to make full use of a limited amount of training data, which is beneficial for preventing overfitting and enhancing the generalization ability of a model. The introduction of noise can simulate the interference encountered in the actual process and increase the robustness of the model.

The data augmentation results are shown in Fig. 8. The images were processed as follows: (1) original image, (2) random flipping, (3) affine transformation (rotation, translation, and scaling), (4) adding additive noise, (5) adding

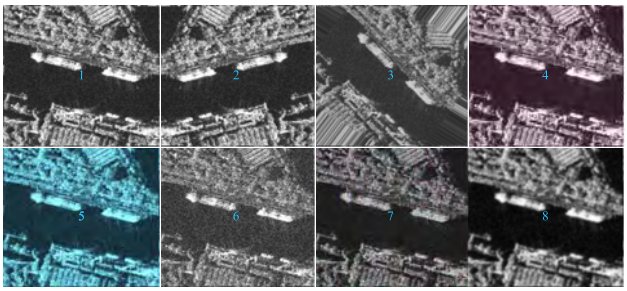


FIGURE 8. The results of data augmentation.

multiplicative noise, (6) adding Gauss noise, (7) randomly inactivating pixels and (8) blurring.

C. EVALUATION METRICS

To quantitatively evaluate the detection effect of the model, the detection performance was assessed in terms of the following criteria:

$$precision = \frac{N_{tp}}{N_{tp} + N_{fp}}. \tag{17}$$

$$recall = \frac{N_{tp}}{N_{tp} + N_{fn}}. \tag{18}$$

where N_{tp} is the number of correctly detected ship targets, N_{fp} the number of incorrectly detected targets and N_{fn} is the number of missing ship targets. We calculated the F1 score and the average precision (AP) as shown in Eqs. (19) and (20), respectively, to represent the comprehensive performance of the algorithm [19], [35], [38]. We considered a predicted bounding box to be correct when it had an IoU greater than 0.5 with a single ground-truth box.

$$F1 = \frac{2 \times precision \times recall}{precision + recall}. \tag{19}$$

$$AP = \sum_{k=1}^{n} precision\,(k) \times \Delta recall\,(k)\,. \tag{20}$$

where n is the total number of images in the dataset, $precision\,(k)$ is the precision at a cutoff point of k images, and $\Delta recall\,(k)$ is the difference in $recall$ between the cutoff point $k-1$ and the cutoff point k [35].

D. EXPERIMENTAL DETAILS

1) OPTIMIZATION OF THE NETWORK STRUCTURE

The detection effect of a CNN mainly depends on the underlying backbone network [40]. Therefore, the detection performance can be greatly improved by optimizing the network structure. In our proposed model, the backbone network consists of four stages, each of which consists of different number of attention modules, as shown in Fig. 5. We denote the numbers of attention modules in stages 1-4 by $\{n_1, n_2, n_3, n_4\}$, respectively, and treat these quantities as hyperparameters. The values of the hyperparameters were empirically selected based on a number of experiments to adjust the network depth. This structure not only facilitates depth adjustment of the network but also makes it easy to compare the effects of the numbers of attention modules used in different stages. Various network structures were evaluated on the test set from SSDD [16]. The experimental results are shown in Table 3.

TABLE 3. The performances of different backbone networks.

Backbone Structure	Training Time/Epoch	Testing Time/Image	AP
{1,1,1,2}	455s	24.60 ms	69.75%
{1,1,2,4}	464s	26.65 ms	73.93%
{1,2,4,4}	480s	28.40 ms	**77.74%**
{2,2,3,4}	494s	29.79 ms	76.69%
{2,2,6,4}	508s	31.13 ms	76.07%

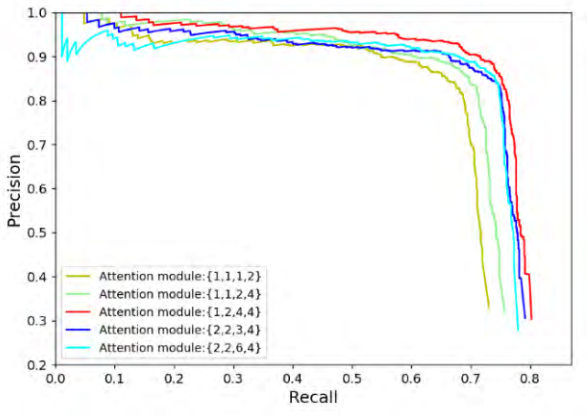
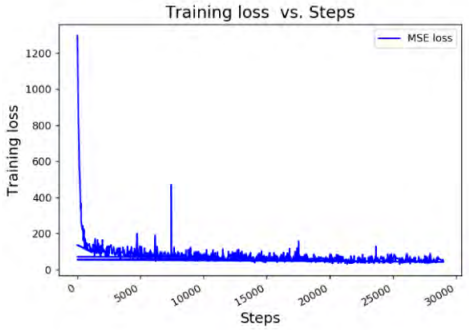


FIGURE 9. The P-R curves for different backbone networks.

By comparing the different network structures, it can be seen that with an increase in the network depth, the AP of the model increases, and the corresponding detection time for a single image also increases. However, for the network structure of {2, 2, 3, 4}, the AP begins to show a downward trend, indicating that the network structure has fallen into a local optimum. Continuing to deepen the network will no longer be advantageous in terms of the AP, and it will increase the detection time. Moreover, we find that increasing n_2 and n_3 can yield better detection results than adjusting n_1. Considering the detection time and AP results, we choose $\{1, 2, 4, 4\}$ as the optimal network structure. Finally, we draw the P-R curves [13], [35], [38] for several network structures to intuitively illustrate the influence of different network structures on the AP, as shown in Fig. 9.

2) COMPARISON OF DIFFERENT LOSS FUNCTIONS

To compare the effects of different loss functions on network training, we used the MSE and GIoU losses as loss functions for bounding box regression and trained the network on the basis of the optimal network structure proposed in the last section. The same training set and training mode were used in both cases. The variation of the loss function during training is shown in Fig. 10. In the initial stage of training, the value of the GIoU loss was higher, and the convergence



(a)

(b)

FIGURE 10. The training loss versus the number of steps when network was trained using the MSE loss and the GIoU loss.

TABLE 4. Comparison between the performances of networks trained using the MSE loss and the GIoU loss across multiple evaluation thresholds.

Evaluation/Loss	MSE Loss	GIoU Loss	Improvement (%)
AP(0.5)	77.74	79.37	1.63
AP(0.6)	54.29	55.91	1.62
AP(0.7)	27.74	30.77	3.03
AP(0.75)	16.02	19.11	3.09

rate was slower. In contrast, the MSE loss converged faster in the initial stage. After 20,000 steps of backpropagation (BP), the training loss curves of both methods tended toward stability. Early stopping [41] was used to terminate training. Therefore, the training process with the MSE loss was stopped at 29,000 steps, and the final convergence loss was 98.47. The training process with the GIoU loss was stopped at 26,000 steps, and the loss was 69.23 at final convergence. Thus, it can be seen that the convergence effect of the GIoU loss is better and more stable in the later stage of training.

We evaluated the models trained with these two loss functions on the test set, and the results are shown in Table 4. For an evaluation threshold of 0.5-0.75, the AP of detection is considerably improved by using the GIoU loss as the regression loss instead of the original MSE loss.

3) INFLUENCES OF NMS AND SOFT-NMS

The AP values under different overlap thresholds can illustrate when a detector achieves its best detection effect [40]. Therefore, in Table 5, we compare the AP values achieved with NMS with those achieved with Soft-NMS under different overlap thresholds.

The left and right sides of the table correspond to the AP values achieved with NMS and Soft-NMS, respectively, under multiple evaluation thresholds for the confidence score (0.5-0.75) and multiple overlap thresholds (0.3-0.8). In addition, the values of the hyperparameter σ are appended to the Soft-NMS results. We can infer that the AP decreases as the evaluation threshold is increased. From a horizontal comparison, it can be seen that when NMS and Soft-NMS have the same overlap threshold, the AP achieved with Soft-NMS is higher than that achieved with NMS under multiple evaluation thresholds. This is because Soft-NMS merely penalizes the scores of detection boxes that overlap considerably with B_M (the box with the maximum score) with penalty functions rather than deleting them directly as NMS does. In this way, detection boxes with large overlaps but actually existing targets are preserved, and thus, the detection rate for densely arranged ship targets is improved. With increasing N_t, the AP shows a stronger decrease with NMS because the high threshold decreases the filtering effect for repeated detection boxes. However, for Soft-NMS, the improvement effect is increasingly obvious. This is because when the overlap between B_M (the box with the maximum score) and B_i (any other detection box) is larger, the probability of B_i being regarded as a repeated detection is greater, and the penalty weight is also greater. When the score of B_i multiplied by the penalty function is lower than the set evaluation threshold, B_i will be deleted, thus guaranteeing the filtering effect of the algorithm on repeated detection boxes. Therefore, when N_t is large, the effect of Soft-NMS is more obvious. Compared with that observed with NMS, the decline in the AP value is delayed with Soft-NMS. Through these experiments, we can clearly identify the differences between the effects of NMS and Soft-NMS on the AP value under different conditions. Moreover, we can also identify the influence of the hyperparameter σ on the behavior of Soft-NMS, thus allowing us to select the value of this hyperparameter more reasonably. The AP values are plotted versus N_t for both NMS and Soft-NMS under multiple evaluation thresholds in Fig. 11. It is clear that Soft-NMS can achieve better performance under different evaluation thresholds.

E. EXPERIMENTAL RESULTS

To test the validity of the network model, the ship detection results obtained under the different environmental conditions represented in the extended SSDD [16] were analyzed, as shown in Fig. 12. In the first row of this figure, we show results for the detection of densely arranged ship targets, which is a difficult problem for ship target detection in SAR images. It can be seen that the algorithm proposed in this

TABLE 5. AP comparison across multiple overlap thresholds N_t and values of the parameter σ for NMS and Soft-NMS. The best performance at each evaluation threshold O_t is marked in bold for each method.

N_t	AP(0.5)	AP(0.6)	AP(0.7)	AP(0.75)	σ	AP(0.5)	AP(0.6)	AP(0.7)	AP(0.75)
0.3	0.7860	0.5522	0.3054	0.1897	**0.1**	0.7930	0.5615	0.3108	0.1909
0.4	0.7936	0.5558	0.3059	0.1901	**0.3**	**0.7978**	**0.5661**	0.3119	0.1912
0.5	**0.7937**	**0.5591**	**0.3077**	0.1911	**0.5**	0.7973	0.5640	**0.3136**	0.1929
0.6	0.7920	0.5579	0.3071	**0.1920**	**0.7**	0.7937	0.5631	0.3112	**0.1990**
0.7	0.7883	0.5528	0.3046	0.1909	**0.9**	0.7924	0.5608	0.3083	0.1984
0.8	0.7328	0.5119	0.2859	0.1815	**1.1**	0.7569	0.5321	0.3021	0.1949

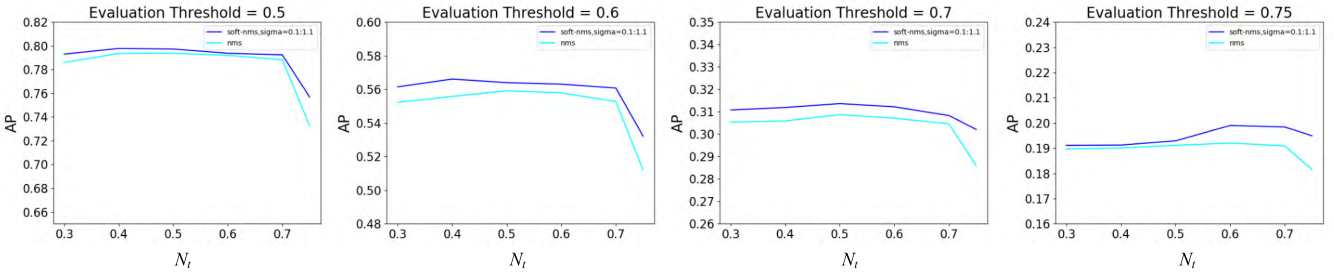


FIGURE 11. AP values versus N_t for NMS and Soft-NMS under multiple evaluation thresholds.

paper can effectively distinguish closely spaced ships and can also effectively segment ship targets close to the coast. The second row shows results for ship target detection against ambiguous backgrounds. Ship targets of this kind are characterized by unclear outlines and unclear boundaries between target and background. We find that the proposed method can effectively distinguish targets from their backgrounds. The third row shows results for the detection of ships of different sizes and orientations in the same image. It is seen that the algorithm can accurately locate these targets. The fourth row shows detection results for small, sparsely distributed targets. It is clear that the proposed algorithm achieves an improved detection effect for small targets, with a lower missed detection rate. Furthermore, note that there might be islands in the ocean that have shapes and sizes similar to ships. Neither the human eye nor the network can discern whether a target is a boat or an island by its brightness or shape. In these cases, the islands and ships can be distinguished by the target's state of motion by combining continuous multiframe images. We will conduct further research in follow-up work.

F. COMPARISON WITH OTHER METHODS

In this section, based on further experiments, the proposed method is quantitatively compared with several mainstream object detection models based on deep learning in terms of the AP and detection speed. The results are shown in Table 6.

It is apparent that the proposed method achieves the best AP of 79.78% on the extended SSDD compared with other single-stage object detection methods based on different backbone networks, including SSD [9] and YOLOv3 [13]. Although our algorithm is not as fast as YOLOv3, its single-image detection time of 28 ms is sufficient for real-time detection. Compared with the two-stage object detection algorithm Faster R-CNN [8], the proposed algorithm is faster specifically, its time cost is only 30% of that of Faster R-CNN. Note that different experimental platforms may have some impact on the detection time. Furthermore, note that for the same network structure, the performance of Faster R-CNN combined with FPN [33] is better than that of the original Faster R-CNN. There are many small-sized ship targets in the experimental dataset. However, the original Faster R-CNN does not combine shallow location information with deep semantic information; instead, it predicts targets only at a single level, which results in missed detections of small ship targets. These findings further demonstrate the importance of the FPN network structure in SAR ship detection.

TABLE 6. The detection performance of four methods.

Method	Backbone	AP (%)	Time (ms)
SSD	VGG16	70.62	30.30
	Darknet-53	71.76	28.06
YOLO v3	Darknet-53	73.70	**24.90**
Faster R-CNN	ResNet-101	70.90	73.28
Faster R-CNN+FPN		**84.26**	93.50
Proposed	Attention-ResNet	79.78	28.40

In most cases, the performance of the four algorithms are similar. To intuitively compare the differences among the different algorithms, we have chosen relatively complex examples for display in Fig. 13. It is clear that all four algorithms can effectively detect ship targets, but in terms

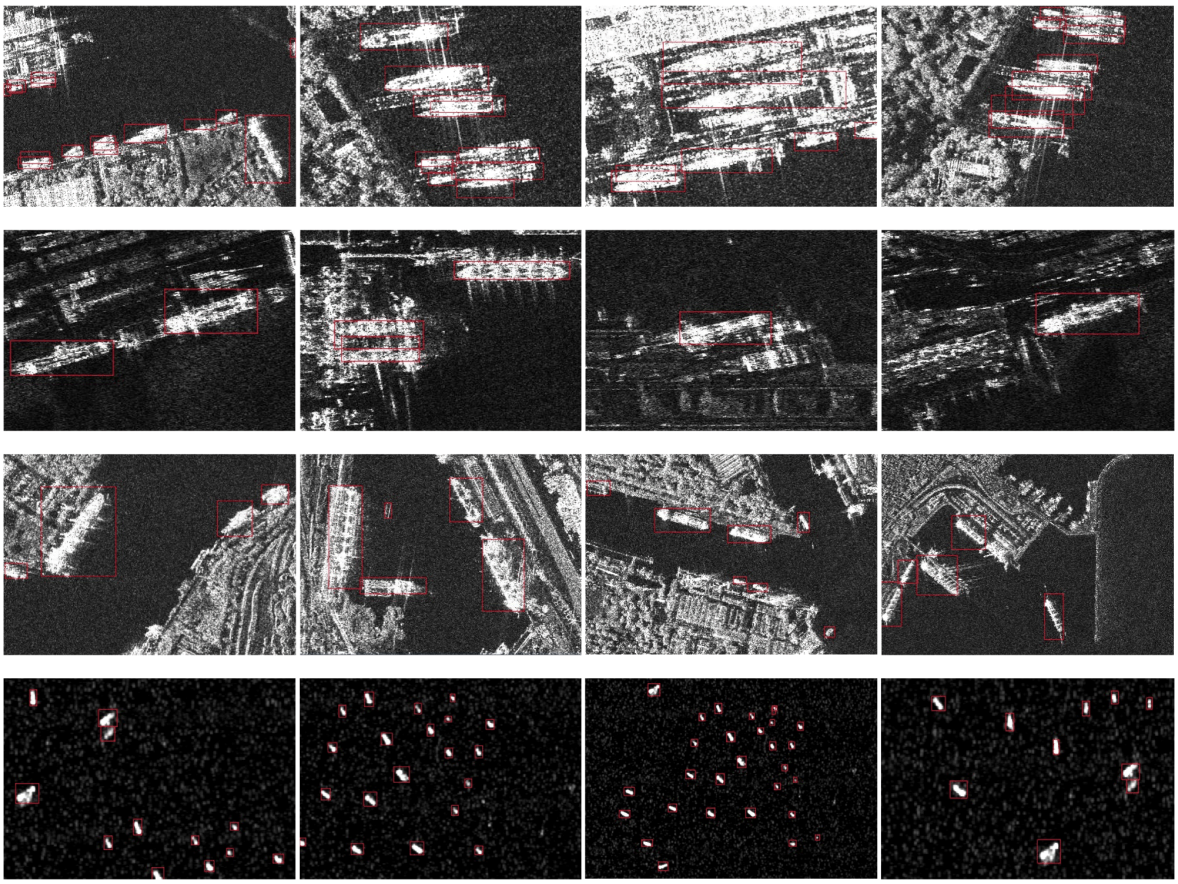


FIGURE 12. Experimental results.

TABLE 7. Detailed information FoR the TerraSAR-X images.

Index	Imaging Time	Resolution	Width	Height	Polarization	Number of Ships
(a)	15 May 2018	5 m	6288	10640	HH	71
(b)	15 May 2018	5 m	6288	10640	HV	71
(c)	26 May 2018	5 m	6456	10224	VH	83
(d)	26 May 2018	5 m	6456	10224	VV	83

of positioning accuracy, the method proposed in this paper has the best performance. Moreover, due to the introduction of the attention mechanism, the proposed network can learn finer features. Compared with the other three algorithms, the proposed algorithm can effectively distinguish densely arranged ships and reduces the number of missed detections for ships with large overlap.

G. GENERALIZATION ABILITY TESTING

The generalization ability is an important criterion for model evaluation [41]. Since the training and testing of the proposed model were carried out on the extended SSDD, it is useful to evaluate the generalization ability of the model on several unseen large SAR images. In this section, TerraSAR-X imaging data from a section of water near Qinhuangdao, China, are used as the test samples. The test samples have different time and polarization characteristics and contain ship targets of different scales against different backgrounds (on the ocean and in the harbor), as shown in Table 7 and Fig. 15. An optical remote sensing image of this area is shown in Fig. 14. This image serves as an important reference for us to determine the position information of the ships in the SAR images. The three major differences between the SAR and optical images of this area can be summarized as follows: First, the SAR images reflect the characteristics of the scattering of electromagnetic waves from the targets. The optical image contains abundant visual scene information, whereas the SAR images are of low resolution, have a low signal-to-noise ratio and contain relatively monotonous information. Second, because the SAR images were obtained in the forward direction from

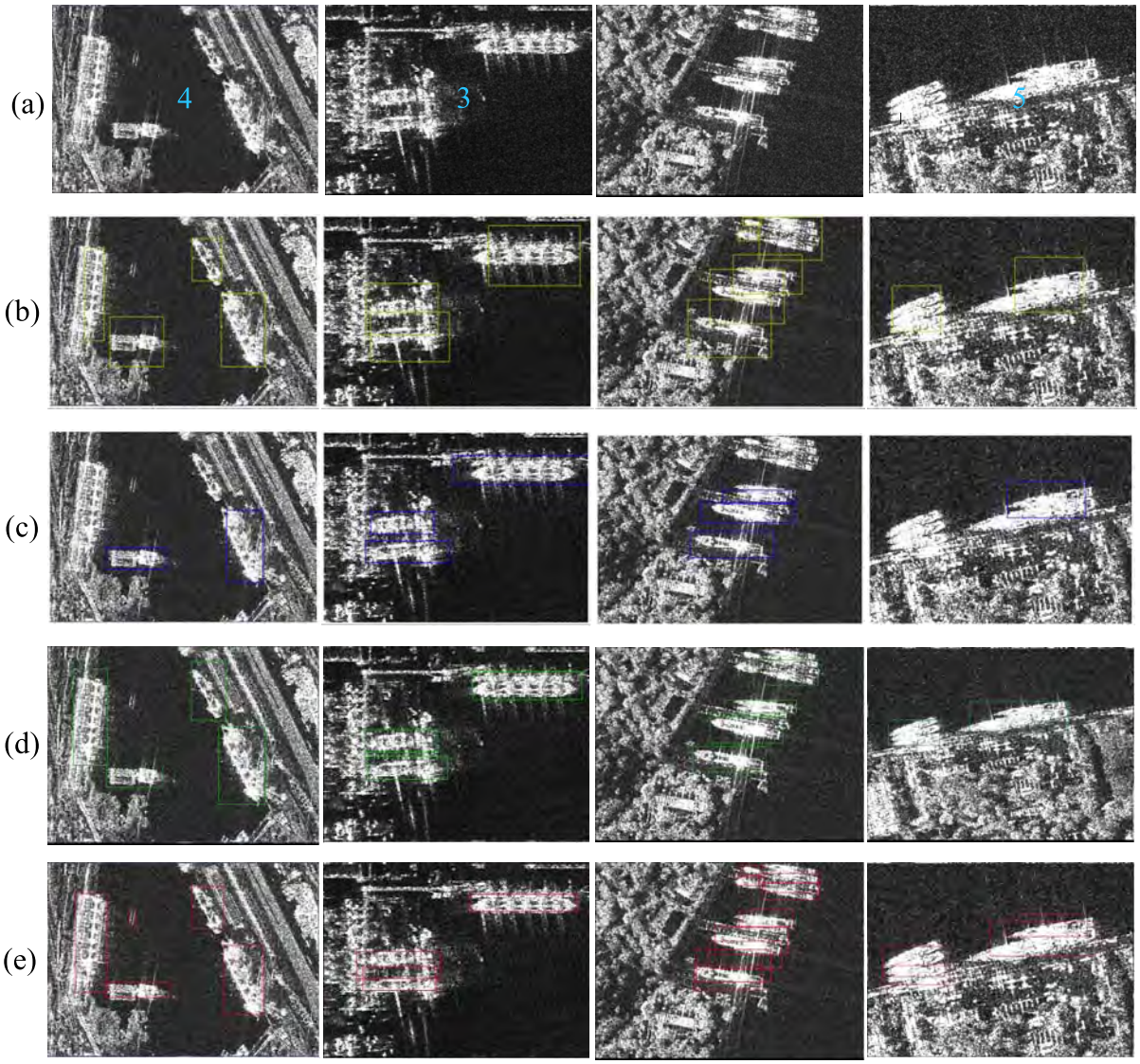


FIGURE 13. Comparison of the results of our proposed method and other methods on SSDD. The numbers in the images are the numbers of ships. (a) Ground truth. (b) The results of SSD. (c) The results of YOLOv3. (d) The results of Faster R-CNN. (e) The results of the proposed method.

a lateral view, they are easily affected by the terrain, resulting in an inversion of the top-bottom image distortion. Third, the SAR images contain severe speckle noise, posing difficulties for object detection. Panels a, b, c, and d of Fig. 15 present the detection results for different regions (ocean and harbor) in SAR images acquired at different times and with different polarizations. It is obvious that the proposed model can effectively detect ship targets on the ocean. Although some instances of false alarms and missed detections are observed for the ship targets in the port, on the whole, the proposed algorithm shows good performance. The validation results for the proposed model are given in Table 8.

IV. DSCUSSION

To better illustrate the effectiveness of the proposed approaches, we compared the effects of different versions of our method on the AP through step-by-step experiments. We considered the optimal network selected in Section III.D as the fundamental model. The various approaches incorporated into our model, as mentioned above, improve the detection performance of the model to varying degrees, as shown in Table 9. By integrating the GIoU loss into the loss function, the sensitivity of the network to different ship target scales is reduced, and the AP of the model is improved by 1.63%. By contrast, using Soft-NMS improves the AP of the model by only 0.44%. The reason for this lesser improvement is that the dataset contains relatively few densely arranged ship samples; therefore, because the primary effect of Soft-NMS is to improve the detection of ship targets with high overlap rates, its improvement effect is relatively small on this dataset. Based on the approaches mentioned above, the final AP of the proposed model is 79.78% on the extended SSDD.

TABLE 8. The ship-detection results obtained to validate our method on TerraSAR-X images from Qinhuangdao, Hebei, China.

Detected Ships	True Ships	False Alarms	Missed Ships	Precision	Recall	F1
313	264	49	44	84.3%	85.7%	84.9%

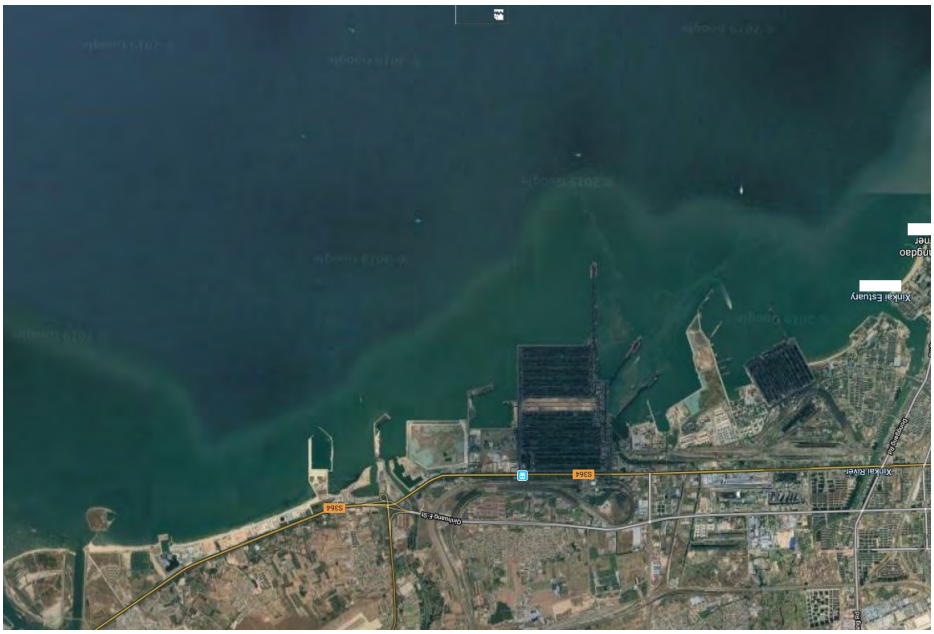


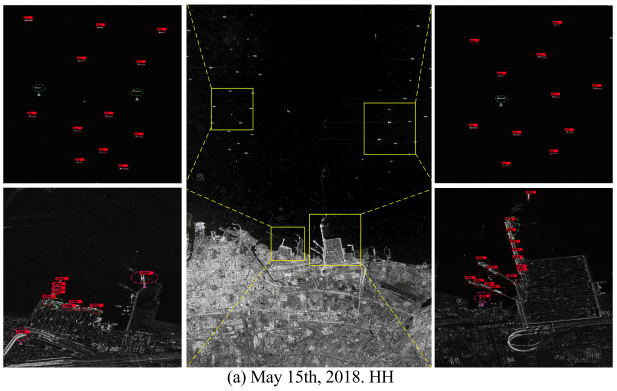
FIGURE 14. An optical remote-sensing image of Qinhuangdao, Hebei, China.

TABLE 9. The detection performances of four versions of our method.

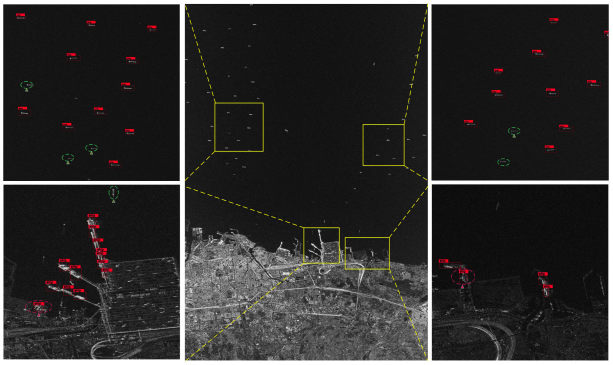
Method	GIoU Loss	Soft-NMS	Data Augmentation	AP (%)
Fundamental model	√	√	√	79.78
	√		√	79.37
		√	√	78.18
			√	77.74

Notably, the detection effect for SAR ships is affected by many factors, including incidence angle, image resolution, polarimetry, metocean parameters, wind speed, ship size, and ship orientation [42]. Fast wind speeds and poor metocean parameters in particular can lead to water turbulence, which may produce volume scattering, resulting in more complex surrounding environments [43]. The detection algorithm designed in this paper is mainly intended to overcome the effects of ship size, complex coastal backgrounds and densely arranged ship targets. In the future, the ship target information will be considered in combination with sea state information to further improve ship target detection.

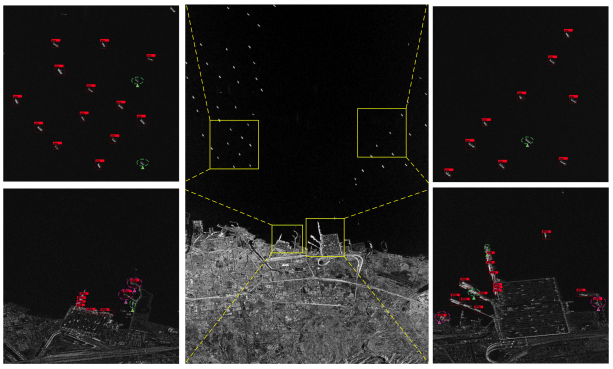
In this paper, we have also reported some research on training from scratch [44], [45]. On the one hand, Reference [45] emphasizes that model convergence can be accelerated in the early stage of training by using a model that has been pretrained on ImageNet, but this approach does not necessarily improve the final target task accuracy. Satisfactory convergence can be achieved by applying an appropriate regularization method and sufficient iterations. On the other hand, since the network proposed in this paper is new, there is no suitable backbone that can be used for transfer learning. Pretraining on ImageNet would require considerable time and computing power, which would obviously be infeasible. Therefore, the network was trained from scratch based on random initialization. BN and leaky ReLU layers were appended after each convolution layer to speed up convergence and avoid overfitting. In addition, we used YOLOv3 both with and without ImageNet pretraining to compare the resulting detection performance. We found that the AP could not be improved by using weights pretrained on ImageNet; in fact, sometimes the AP would even decrease. This is because the SAR ship detection task is more sensitive to localization than to classification and ImageNet pretraining cannot effectively improve localization performance. Moreover, Reference [46] has suggested that training from scratch has great potential in cross-domain scenarios, such as depth images, medical images, and multispectral images. Similarly, for SAR images, training from scratch may result in better performance.



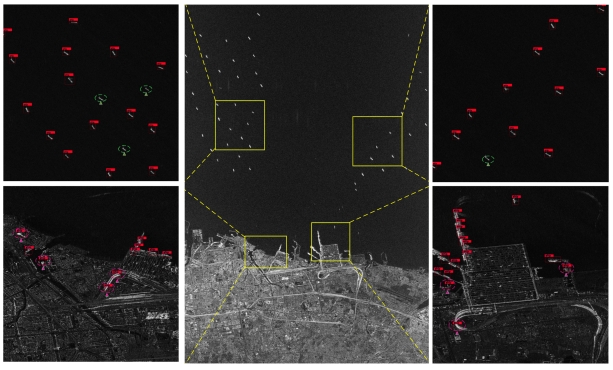
(a) May 15th, 2018. HH



(b) May 15th, 2018. VH



(c) May 26th, 2018. VH



(d) May 26th, 2018. VV

FIGURE 15. Experimental validation of the generalization ability of the proposed model on TerraSAR-X images of Qinhuangdao, Hebei, China. The red rectangles, magenta ellipses, and green ellipses represent the ships detected by the model, false alarms, and missed ship detections, respectively.

V. CONCLUSION

In this paper, a single-stage object detection algorithm based on an attention mechanism is proposed, and the effectiveness of the algorithm is verified on the open dataset SSDD. First, a modified residual module is applied as the basic unit of the feature extraction network, thereby enhancing the network's ability to acquire higher-level target information. At the same time, an attention mechanism is integrated into the backbone network, and the network is redesigned to improve its ability to detect and locate targets and endow it with the ability to effectively distinguish densely arranged ships. To account for the typical multiscale characteristics of ship targets in SAR images, the GIoU loss is integrated into the loss function to reduce the scale sensitivity of the network. In addition, Soft-NMS is introduced to solve the problem of missed detections of ship targets with high overlap rates. Another advantage of our proposed algorithm is its fast detection speed. The detection time for a single image on SSDD is only 28 ms, which is sufficient for real-time ship detection. The continued development of SAR technology will enable us to obtain more high-quality data, which will strongly promote research on the use of deep learning algorithms in the field of SAR image processing.

REFERENCES

[1] K. El-Darymli, P. McGuire, D. Power, and C. R. Moloney, "Target detection in synthetic aperture radar imagery: A state-of-the-art survey," *J. Appl. Remote Sens.*, vol. 7, no. 1, 2013, Art. no. 071598.

[2] J. Zhao, Z. Zhang, W. Yu, and T.-K. Truong, "A cascade coupled convolutional neural network guided visual attention method for ship detection from SAR images," *IEEE Access*, vol. 6, pp. 50693–50708, 2018.

[3] D. Gleich and M. Datcu, "Despeckling and information extraction from SLC SAR images," *IEEE Trans. Geosci. Remote Sens.*, vol. 52, no. 8, pp. 4633–4649, Aug. 2014.

[4] C. H. Gierull and I. Sikaneta, "A compound-plus-noise model for improved vessel detection in non-Gaussian SAR imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 3, pp. 1444–1453, Mar. 2018.

[5] D. J. Crisp, "A ship detection system for RADARSAT-2 dual-pol multi-look imagery implemented in the ADSS," in *Proc. IEEE Int. Conf. Radar*, Adelaide, SA, Australia, Sep. 2013, pp. 318–323.

[6] X. Leng, K. Ji, S. Zhou, X. Xing, and H. Zou, "An adaptive ship detection scheme for spaceborne SAR imagery," *Sensors*, vol. 16, no. 9, p. 1345, 2016.

[7] S. Wang, M. Wang, S. Yang, and L. Jiao, "New hierarchical saliency filtering for fast ship detection in high-resolution SAR images," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 1, pp. 351–362, Jan. 2017.

[8] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 6, pp. 1137–1149, Jun. 2017.

[9] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "SSD: Single shot MultiBox detector," in *Proc. Eur. Conf. Comput. Vis.*, Amsterdam, The Netherlands, Oct. 2016, pp. 21–37.

[10] S. Liu, D. Huang, and Y. Wang, "Receptive field block net for accurate and fast object detection," in *Proc. Eur. Conf. Comput. Vis.*, Munich, Germany, Sep. 2018, pp. 385–400.

[11] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Boston, MA, USA, Jun. 2016, pp. 779–788.

[12] J. Redmon and A. Farhadi, "YOLO9000: Better, faster, stronger," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Honolulu, HI, USA, Jul. 2017, pp. 6517–6525.

[13] J. Redmon and A. Farhadi, "YOLOv3: An incremental improvement," Apr. 2018, *arXiv:1804.02767*. [Online]. Available: https://arxiv.org/abs/1804.02767

[14] S. Song, B. Xu, Z. Li, and J. Yang, "Ship detection in SAR imagery via variational Bayesian inference," *IEEE Geosci. Remote Sens. Lett.*, vol. 13, no. 3, pp. 319–323, Mar. 2016.

[15] S. Liu, Z. Cao, and H. Yang, "Information theory-based target detection for high-resolution SAR image," *IEEE Geosci. Remote Sens. Lett.*, vol. 13, no. 3, pp. 404–408, Mar. 2016.

[16] J. Li, C. Qu, and J. Shao, "Ship detection in SAR images based on an improved faster R-CNN," in *Proc. BIGSARDATA*, Beijing, China, Nov. 2017, pp. 1–6.

[17] M. Kang, X. Leng, Z. Lin, and K. Ji, "A modified faster R-CNN based on CFAR algorithm for SAR ship detection," in *Proc. Int. Workshop Remote Sens. Intell. Process.*, Shanghai, China, May 2017, pp. 1–4.

[18] J. Jiao, Y. Zhang, H. Sun, X. Yang, X. Gao, W. Hong, K. Fu, and X. Sun, "A densely connected end-to-end neural network for multiscale and multiscene SAR ship detection," *IEEE Access*, vol. 6, pp. 20881–20892, 2018.

[19] M. Kang, K. Ji, X. Leng, and Z. Lin, "Contextual region-based convolutional neural network with multilayer fusion for SAR ship detection," *Remote Sens.*, vol. 9, no. 8, p. 860, 2017.

[20] L. Wan-Yi, W. Peng, and Q. Hong, "A survey of visual attention based methods for object tracking," *Acta Automatica Sinica*, vol. 40, no. 4, pp. 561–576, 2014.

[21] K. Xu, J. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhudinov, R. Zemel, and Y. Bengio, "Show, attend and tell: Neural image caption generation with visual attention," in *Proc. Int. Conf. Mach. Learn.*, Lille, France, Jul. 2015, pp. 2048–2057.

[22] Z. Song, H. Sui, and Y. Wang, "Automatic ship detection for optical satellite images based on visual attention model and LBP," in *Proc. IEEE Workshop Electron., Comput. Appl.*, Ottawa, ON, Canada, May 2014, pp. 722–725.

[23] Q. Zhang, Y. N. Wu, and S.-C. Zhu, "Interpretable convolutional neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Salt Lake City, UT, USA, Jun. 2018, pp. 8827–8836.

[24] S. Woo, J. Park, J.-Y. Lee, and I. S. Kweon, "CBAM: Convolutional block attention module," in *Proc. Eur. Conf. Comput. Vis.*, Munich, Germany, Sep. 2018, pp. 3–19.

[25] F. Wang, M. Jiang, C. Qian, S. Yang, C. Li, H. Zhang, X. Wang, and X. Tang, "Residual attention network for image classification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Honolulu, HI, USA, Jul. 2017, pp. 3156–3164.

[26] H. Zheng, J. Fu, T. Mei, and J. Luo, "Learning multi-attention convolutional neural network for fine-grained image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Honolulu, HI, USA, Oct. 2017, pp. 5209–5217.

[27] H. Rezatofighi, N. Tsoi, J. Gwak, A. Sadeghian, I. Reid, and S. Savarese, "Generalized intersection over union: A metric and a loss for bounding box regression," Feb. 2019, *arXiv:1902.09630*. [Online]. Available: https://arxiv.org/abs/1902.09630

[28] N. Bodla, B. Singh, R. Chellappa, and L. S. Davis, "Soft-NMS–improving object detection with one line of code," in *Proc. Int. IEEE Conf. Comput. Vis. (ICCV)*, Venice, Italy, Oct. 2017, pp. 5561–5569.

[29] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. A. Alemi, "Inception-v4, inception-resnet and the impact of residual connections on learning," in *Proc. 21st AAAI Conf. Artif. Intell.*, San Francisco, CA, USA, Feb. 2017, pp. 4278–4284.

[30] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Las Vegas, NV, USA, Jun. 2016, pp. 2818–2826.

[31] H. Noh, S. Hong, and B. Han, "Learning deconvolution network for semantic segmentation," in *Proc. Int. IEEE Conf. Comput. Vis. (ICCV)*, Santiago, Chile, Dec. 2015, pp. 1520–1528.

[32] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Amsterdam, The Netherlands, Oct. 2016, pp. 770–778.

[33] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Honolulu, HI, USA, Jul. 2017, pp. 936–944.

[34] Z. Zhang, T. He, H. Zhang, Z. Zhang, J. Xie, and M. Li, "Bag of freebies for training object detection neural networks," Feb. 2019, *arXiv:1902.04103*. [Online]. Available: https://arxiv.org/abs/1902.04103

[35] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The Pascal visual object classes (VOC) challenge," *Int. J. Comput. Vis.*, vol. 88, no. 2, pp. 303–338, Sep. 2009.

[36] O. Bachem, M. Lucic, H. Hassani, and A. Krause, "Fast and provably good seedings for K-means," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, Barcelona, Spain, Dec. 2016, pp. 55–63.

[37] D. Arthur and S. Vassilvitskii, "K-means++: The advantages of careful seeding," in *Proc. 18th Annu. ACM-SIAM Symp. Discrete Algorithms*, New Orleans, LA, USA, Jan. 2007, pp. 1027–1035.

[38] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Miami, FL, USA, Jun. 2009, pp. 248–255.

[39] A. Moreira, P. Prats-Iraola, M. Younis, G. Krieger, I. Hajnsek, and K. P. Papathanassiou, "A tutorial on synthetic aperture radar," *IEEE Geosci. Remote Sens. Mag.*, vol. 1, no. 1, pp. 6–43, Mar. 2013.

[40] L. Liu, W. Ouyang, X. Wang, P. Fieguth, J. Chen, X. Liu, and M. Pietikäinen, "Deep learning for generic object detection: A survey," Sep. 2018, *arXiv:1809.02165*. [Online]. Available: https://arxiv.org/abs/1809.02165

[41] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. Cambridge, MA, USA: MIT, 2016.

[42] B. Tings, C. Bentes, D. Velotto, and S. Voinov, "Modelling ship detectability depending on TerraSAR-X-derived metocean parameters," *CEAS Space J.*, vol. 11, no. 1, pp. 81–94, 2019.

[43] Y. Wang, C. Wang, H. Zhang, Y. Dong, and S. Wei, "Automatic ship detection based on RetinaNet using multi-resolution Gaofen-3 imagery," *Remote Sens.*, vol. 11, no. 5, pp. 531–544, 2019.

[44] Z. Li, C. Peng, G. Yu, X. Zhang, Y. Deng, and J. Sun, "DetNet: A backbone network for object detection," Apr. 2018, *arXiv:1804.06215*. [Online]. Available: https://arxiv.org/abs/1804.06215

[45] K. He, R. Girshick, and P. Dollár, "Rethinking ImageNet pre-training," Nov. 2018, *arXiv:1811.08883*. [Online]. Available: https://arxiv.org/abs/1811.08883

[46] Z. Shen, Z. Liu, J. Li, Y.-G. Jiang, Y. Chen, and X. Xue, "DSOD: Learning deeply supervised object detectors from scratch," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Venice, Italy, Oct. 2017, pp. 1919–1927.