P2RNet: Fast Maritime Object Detection From Key Points to Region Proposals in Large-Scale Remote Sensing Images

Yantong Chen, Jialiang Wang, Yanyan Zhang, Yang Liu, and Junsheng Wang

***Abstract*—Due to the long distance and large-scale of satellite imaging, and the high complexity of depth convolutional neural network, common detectors cannot be directly applied to large-scale remote sensing images. Therefore, this article proposes a two-phase object detection network from key points to region proposals, namely P2RNet. In the first phase, the key points of all suspected objects are obtained through the key point extraction network, and then these key points are divided into multiple region proposals using the region proposal generator. In the second phase, these region proposals are input into the lightweight object detection network to achieve fast and accurate maritime object detection. The lightweight object detection network is improved based on YOLOv5. To significantly reduce the number of parameters and computation of the network, the improved MobileNetv2 constructed by the grouped sandglass block is used as the backbone to extract sufficient feature information. To effectively improve the detection accuracy of the network, the simple attention module is embedded in the feature fusion network to strengthen the feature fusion process, and the oriented spatial pyramid pooling-fast is proposed to capture long-distance dependencies. The experimental results on the DOTA Ship dataset show that the average precision and frames per second of the lightweight object detection network reached 80.59% and 112, respectively, achieving a good balance. Moreover, the overall object detection network achieved excellent detection results on large-scale remote sensing images.**

***Index Terms*—Key point extraction, large-scale remote sensing images, lightweight network, maritime object detection.**

I. Introduction

With the rapid development of optical remote sensing technology and computer vision, the study of remote sensing images has received widespread attention. As the most typical maritime objects, ships are the main transportation carriers and important monitoring objects of maritime trade. In the field of ocean remote sensing, ship detection can be used for dynamic harbor monitoring, maritime traffic management, maritime rescue and combating illegal fishing, which has important research value. The example of ship detection in satellite remote sensing images is shown in Fig. 1. The ocean remote sensing images taken by satellite are processed on board and transmitted to the ground receiving station, which can assist maritime traffic safety management through real-time ship detection. Currently, many ship detection methods have been proposed in remote sensing images, which is still a challenging task due to the complexity of the scene and the multiscale of the ship, as well as the large-scale of the image.

Fig. 1. Example of ship detection in satellite remote sensing images.

Traditional ship detection method is divided into three steps. First, the sea–land segmentation is performed to eliminate the interference of the land part [1]. Then, region proposals are generated to locate the areas suspected to contain ships [2]. Finally, feature extraction algorithm is used to extract and classify region proposals to identify ships [3]. Traditional ship detection methods rely on a large amount of prior knowledge for feature design, with high computational complexity and poor performance in complex scenes.

With the wide application of deep learning technology in object detection, arbitrary-oriented ship detection in remote sensing images has been paid more and more attention. Nie et al. [4] improved the mask R-CNN and obtained a ship detection and segmentation network with better detection effect, which uses

Manuscript received 27 December 2023; revised 1 March 2024 and 27 March 2024; accepted 16 April 2024. Date of publication 23 April 2024; date of current version 3 May 2024. This work was supported in part by the National Natural Science Foundation of China under Grant 61901081, in part by the China Postdoctoral Science Foundation under Grant 2020M680927, and in part by the Fundamental Research Funds for the Central Universities under Grant 3132022237. *(Corresponding author: Junsheng Wang.)*

The authors are with the Department of Information Science and Technology, Dalian Maritime University, Dalian 116026, China (e-mail: chenyantong@dlmu.edu.cn; wangjialiang@dlmu.edu.cn; zhangyanyan999@dlmu.edu.cn; ly1120211369@dlmu.edu.cn; wangjsh@dlmu.edu.cn).

Digital Object Identifier 10.1109/JSTARS.2024.3392635

spatial and channel attention mechanisms to adjust the weights of each pixel and channel, respectively. Therefore, the object features can obtain better response in the feature map. In order to improve the detection performance in complex scenes, Ren et al. [5] proposed a ship detection network assisted by saliency information, in which the feature-enhanced structure can accurately obtain the features of foreground objects, and the salient screening mechanism is used to increase the number of positive samples. Qin et al. [6] proposed an arbitrary-oriented ship detection network suitable for offshore scenes, which introduced a context location module in the backbone and a global channel module in the neck to enhance the network's distinction between ships and background interference. Han et al. [7] designed a two-way dense feature fusion network to maximize the use of multilayer features. By refining the fused features through the dual mask attention module, the detection performance of the network in dense scenes has been improved. To reduce the cost of manual annotation, Li et al. [8] constructed a remote sensing object detection network based on weakly supervised learning. It introduces point labels to guide the mining of region proposals and utilizes a progressive mining strategy to improve detection performance.

The above-mentioned deep learning-based ship detection methods all perform detection on small or medium-scale remote sensing images, which is not applicable to large-scale remote sensing images. However, the size of remote sensing images captured by satellites in real situations is large, and common object detection networks cannot directly process them due to their high complexity. At present, there are two main types of methods to perform object detection for large-scale remote sensing images. One type of method is to reduce the image size before performing object detection. Su et al. [9] designed a new feature extraction network based on YOLO by reducing parameters and adding deformable convolutions. Thanks to the advantages of fully convolutional lightweight network, ship detection on large-scale remote sensing images has been basically achieved. Wang et al. [10] proposed a single-shot multiclass object detection method for large-scale remote sensing images, which uses feature pyramid and multiple dilated rates to fuse the context information in multiscale features. In addition, this method defines an area-weighted loss function to pay more attention to small objects during training. This type of methods will further narrow down small objects, seriously affecting detection performance. Another type of method is to first obtain a large number of image blocks from large-scale images using the sliding window method, and then perform block object detection. Yu et al. [11] developed a cascade ship detection network aided by rotating anchor, which uses the data preprocessing module to determine whether the cropped image block contains ships. After the image block containing ships passes through the basic detector and the cascade refinement module, the final detection results are obtained. Shen et al. [12] proposed a fast multiclass remote sensing object detection method, which applies the manhattan-distance intersection of union loss function to YOLOv4 to improve detection accuracy. Moreover, this method utilizes the truncated nonmaximum suppression (NMS) algorithm to filter out duplicate and incorrect detection boxes from the concatenated detection results. This type of methods will generate many overlapping areas during the sliding window process, resulting in a significant decrease in detection efficiency. Although the above-mentioned two methods can partially solve the problems faced by large-scale remote sensing images, they do not take into account the large amount of remote sensing image data, let alone build a lightweight object detection network. In summary, there is currently almost no lightweight ship detection network that can simultaneously balance detection performance and efficiency for large-scale remote sensing images.

In response to the problems of low detection performance and efficiency in large-scale remote sensing image ship detection, as well as the difficulties in lightweight network design, this article proposes a two-phase object detection network P2RNet from key points to region proposals for large-scale remote sensing image ship detection. Specifically, the main work and contributions of this article are as follows.

1) An innovative two-phase detection strategy is proposed for large-scale remote sensing images, which is key to achieve efficient ship detection. In the first phase, the key points of objects are obtained from the heatmap generated by the key point extraction network, and then region proposals are divided based on these key points. In the second phase, these region proposals are input into the lightweight object detection network based on improved YOLOv5 for fast and accurate ship detection.
2) To reduce the number of parameters and computation of the overall network, ResNet18 is selected as the backbone of the key point extraction network, and the improved MobileNetv2 constructed by the grouped sandglass block (G-SB) is used as the backbone of the lightweight object detection network.
3) To improve the detection accuracy of the lightweight network, the simple attention module (SimAM) is first embedded in the feature fusion network to strengthen the feature fusion process and enhance the feature representation of ships. The oriented spatial pyramid pooling-fast (O-SPPF) is then used to capture long-distance dependencies and local context information.

The rest of this article is organized as follows. Section II introduces the related works of remote sensing image object detection and lightweight network. Section III describes the details of the proposed ship detection network P2RNet. The experimental results are reported and analyzed in Section IV. Finally, Section V concludes this article.

II. Related Work

A. Object Detection in Remote Sensing Images

Deep learning technology mines distributed feature representation of input data by learning a deep nonlinear network structure. It combines low-level features to form abstract deep representations, that is, attribute classes or features. Depending on whether region proposals are generated or not, deep learning-based object detection methods can be divided into two types: two-stage and single-stage detection methods. The

two-stage detection method first extracts the region of interest (RoI), and then detects each region. The typical method is faster R-CNN [13]. The single-stage detection method simultaneously predicts the bounding box and class of the object, with typical methods represented by YOLO [14] and SSD [15]. According to the characteristics of remote sensing images, researchers have proposed many improvement strategies for universal object detection networks, which promote the development of deep learning in the field of remote sensing image object detection.

To achieve accurate detection of dense remote sensing objects, Wu et al. [16] proposed a feature fusion module that can aggregate global context with low-level and high-level features, and a feature refinement module that combines multiple branches with different receptive fields. Shi et al. [17] proposed a geometric transformation module to solve the problem of object direction change. Moreover, a global context feature fusion module is designed to learn the association between different locations and obtain the global context information by using spatial attention mechanism. Zhang et al. [18] replaced the residual blocks in the backbone of YOLOv3 with contextual transformer blocks to enhance the visual representation of small objects. Meanwhile, subpixel convolution upsampling is adopted to optimize the feature fusion process. Zhu et al. [19] proposed an arbitrary-oriented ship detection network based on RetinaNet, which utilizes rotating anchors and skew NMS to detect rotating objects. In order to improve detection accuracy, It additionally introduces the feature alignment module and the intersection over union (IoU) constant factor. In view of the mismatch between the RoI and the object, Ding et al. [20] proposed an RoI transformer, which extracts rotation-invariant features through the spatial transformation of RoI. Li et al. [21] developed a single-stage rotating object detector, in which the rotation feature selection module is used to dynamically adjust the receptive field of neurons, and the rotation feature align module can adaptively align features according to the shape and direction of features. Liu et al. [22] constructed an adaptive balanced network, which adds a context enhancement module and uses an enhanced effective channel attention mechanism. The former is used to extract rich semantic information, while the latter can enhance the feature representation of the object.

B. Lightweight Networks

Lightweight network aims to further reduce the parameters and complexity of the model while maintaining accuracy. It has gradually become a research hotspot of computer vision. In terms of lightweight feature extraction network, Xception [23] uses depthwise separable convolution to improve Inception V3, which consists of entry flow, middle flow, and exit flow. Under the condition of equivalent number of parameters, it achieves higher precision. SqueezeNet [24] designs a fire module consisting of a squeeze layer and an expand layer, which is compressed between convolutional layers. This network substantially reduces the model size while maintaining accuracy. As an improved network of MobileNet, MobileNetv2 [25] proposes an inverted residual block with linear bottleneck structure to avoid information loss caused by nonlinear transformations, which significantly improves the accuracy and speed of image classification. To address the computationally intensive problem of pointwise convolution, ShuffleNet [26] adopts two operations of pointwise group convolution and channel shuffle to build an efficient lightweight mobile terminal network. By analyzing the redundancy of feature maps, GhostNet [27] uses a series of low-cost linear operations to generate a large number of feature maps containing existing feature information.

In terms of lightweight object detection network, to achieve a balance between resources and accuracy, Tiny-DSOD [28] designs a novel lightweight feature pyramid network, as well as a depthwise dense block combining depthwise separable convolution and DenseNet. CSL-YOLO [29] proposes a cross-stage lightweight module with two-branch structure where the first branch generates redundant features through linear operations, and the second branch generates necessary features through lightweight operations. ThunderNet [30] is a two-stage lightweight detector that enables real-time object detection thanks to the efficient backbone and detection head. Furthermore, a context enhancement module is designed to enhance the feature representation of the object. LightDet [31] uses detail-preserving modules that can capture more low-level features to build the backbone. In order to optimize the feature fusion network, it introduces a lightweight feature-preserving and refinement module. SlimYOLOv4 [32] takes MobileNetv2 as the feature extraction network. Meanwhile, conventional convolution is replaced by more suitable depthwise overparameterized depthwise convolution, which improves network performance while reducing the computation.

III. Proposed Method

This section describes the structure of P2RNet in detail. P2RNet adopts a two-phase detection strategy from key points to region proposals. In the first phase, the large-scale remote sensing image is input into the key point extraction network to obtain the heatmap containing the rough position information of the object, and all possible key points in the heatmap are extracted. Further, these key points are input into the region proposal generator to obtain multiple region proposals. In the second phase, considering the lightweight requirements in practical applications, these region proposals are successively input into the lightweight object detection network to achieve fast and accurate ship detection. The architecture of P2RNet is shown in Fig. 2.

A. Key Point Guided Region Proposal Generation

For large-scale remote sensing images, if the ship detection is carried out after reducing the size, the detection performance will be reduced. If the sliding window method is used for ship detection, the detection efficiency will be reduced. To solve the above-mentioned problems, this article uses key points to guide the generation of region proposals in the first phase to effectively improve the detection performance and efficiency. Specifically, all possible key points in the heatmap are obtained

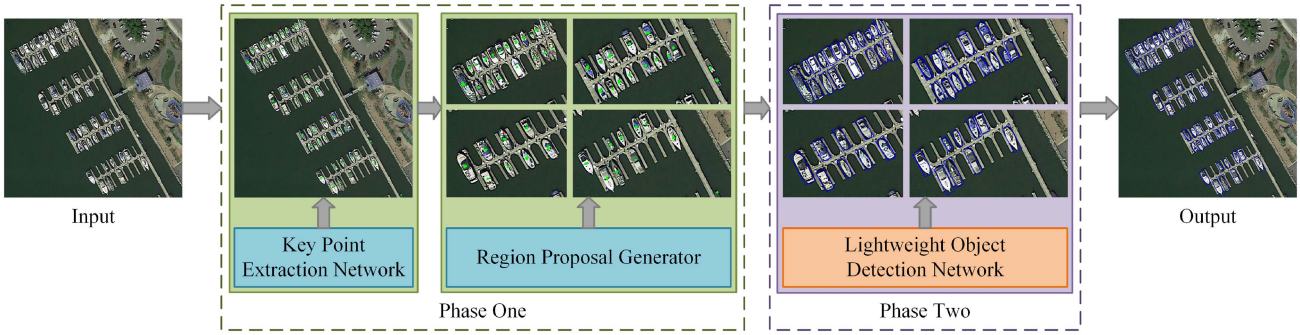


Fig. 2. Architecture of P2RNet. In the first phase, the key points of all suspected objects are obtained through the key point extraction network, and multiple region proposals are delineated using the region proposal generator. In the second phase, these regions are fed into the lightweight object detection network for high-performance object detection.

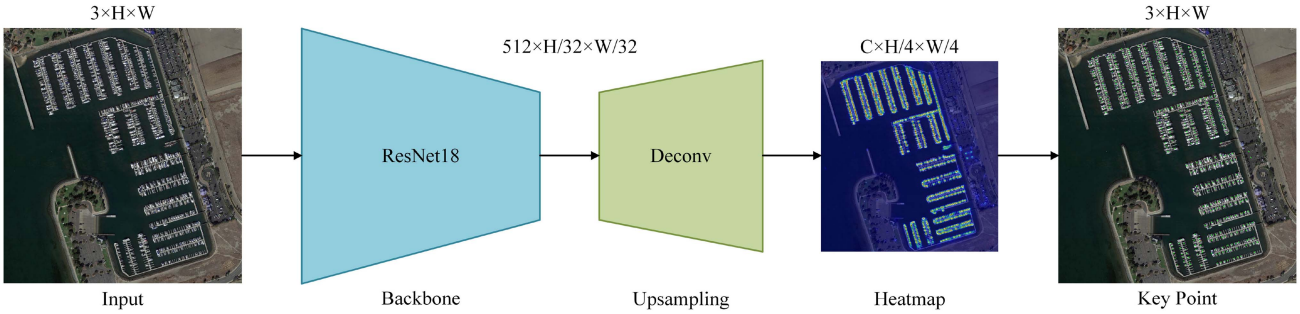


Fig. 3. Architecture of key point extraction network. Its backbone adopts ResNet18, and the upsampling method is deconvolution. The obtained heatmap is used to predict the key points of the object.

through the key point extraction network, and then these key points are divided into multiple region proposals using the region proposal generator. Although the key point extraction network has lightweight characteristics, the feature extraction capability is sufficient. Furthermore, the computational complexity of the region proposal generator is low. The above-mentioned two components together ensure the performance and efficiency advantages of the first phase.

1) Key Point Extraction Network: Inspired by CenterNet [33], this article proposes a key point extraction network, whose architecture is shown in Fig. 3. Considering the relatively small size of ships in large-scale remote sensing images, the low dependence on deep semantic features, and the need for lightweight in practical applications, this article selects the shallow residual network ResNet18 [34] as the backbone. For the input image $I \in \mathbb{R}^{3\times H\times W}$, where H and W represent the height and width of the image, respectively. After feature extraction and deconvolution upsampling of I, the prediction heatmap $\hat{Y} \in [0,1]^{C\times H/4\times W/4}$ is obtained, where C denotes the number of classes. There is only one ship class in this article, so $C = 1$ is set. $\hat{Y}$ is used to predict the key point of the object, namely the center point, where the value indicates the confidence that the point is the center point of the object. To ensure that all ships are recalled, the confidence threshold is set to a lower value of 0.25. By looking for local peak points in $\hat{Y}$, all peak points larger than the threshold are selected as the prediction key points.

For each ground truth key point $p \in \mathbb{R}^2$ in I, its corresponding position after $4\times$ downsampling is $\hat{p} = [\frac{p}{4}]$. The Gaussian kernel $Y_{xyc} = \exp(-\frac{(x-\hat{p}_x)^2+(y-\hat{p}_y)^2}{2\sigma_p^2})$ is used to map all the ground truth key points to the ground truth heatmap $Y \in [0,1]^{C\times H/4\times W/4}$, where σ_p indicates the size-adaptive standard deviation of the object. As the only loss function of the key point extraction network, the key point loss L_k is used to locate the key point to the center of the object, which is a pixelwise logistic regression with focal loss

$$L_k = \frac{-1}{N}\sum_{xyc} \begin{cases} \left(1-\hat{Y}_{xyc}\right)^{\alpha}\log\left(\hat{Y}_{xyc}\right) & \text{if } Y_{xyc}=1 \\ \left(1-Y_{xyc}\right)^{\beta}\left(\hat{Y}_{xyc}\right)^{\alpha}\log\left(1-\hat{Y}_{xyc}\right) & \text{otherwise} \end{cases} \tag{1}$$

where α and β are the hyperparameters of focal loss. In this article, $\alpha = 2$ and $\beta = 4$ are set by default. N is the number of key points in I, which is used to normalize all positive focal loss instances to 1.

2) Region Proposal Generator: After obtaining the key points based on heatmap from the key point extraction network, the region proposal generator is used to divide all the key points into regions for accurate ship detection in the second phase. This article uses K-means algorithm based on Euclidean distance to

divide n key points into k regions. The specific division process is as follows.

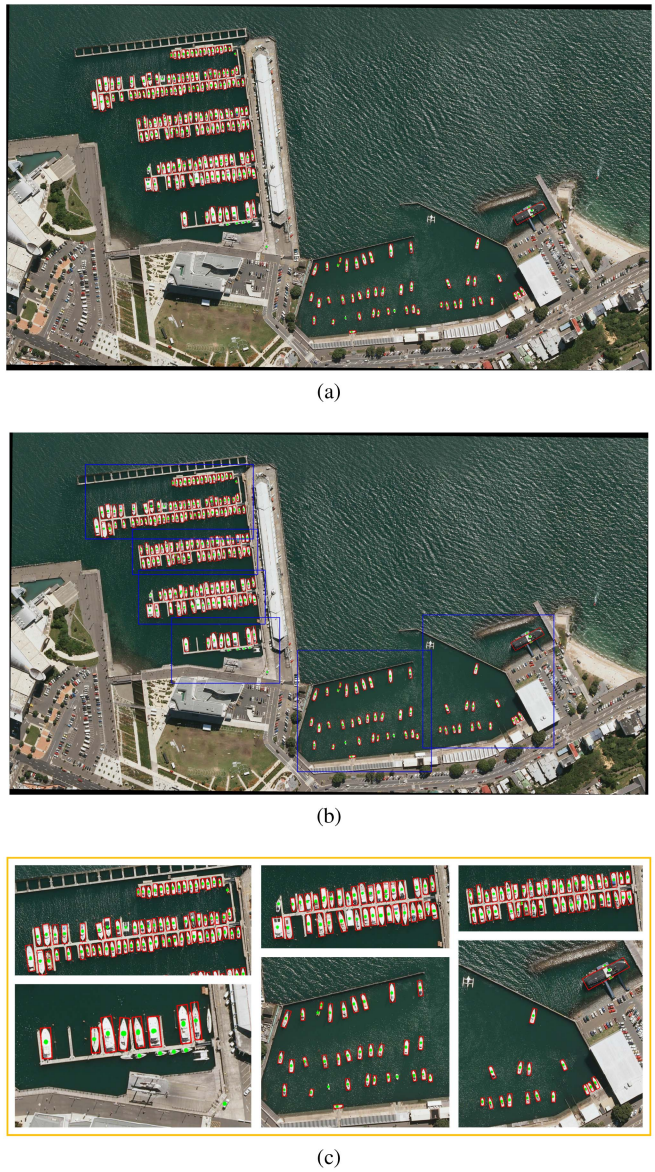
1) Randomly select k out of n key points as the clustering centers for the initial regions.
2) Calculate the Euclidean distance from the other $n-k$ key points to each clustering center, and divide them into corresponding regions according to the nearest distance criterion.
3) Calculate the average position of all key points in each region to obtain a new clustering center.
4) If the clustering centers of all regions remain unchanged, output the clustering results. Otherwise, repeat step 2.

The time and space complexity of K-means algorithm are both $O(n)$, and the consumption of computing resources is low. To determine the clustering number k, we calculate the contour coefficient of each key point

$$s_i = \frac{b_i - a_i}{\max(a_i, b_i)} \tag{2}$$

where a_i is the average Euclidean distance from key point i to other key points in the same cluster. b_i is the minimum average Euclidean distance from key point i to key points in different clusters. $s_i \in [-1, 1]$, and the closer it is to 1, the more reasonable the clustering of key point i is. The average of contour coefficients of all key points is calculated to obtain the contour coefficients of clustering results. Considering the distribution characteristics of overall dispersion and local aggregation of ships, we reasonably set $k \in [2, 8]$. The value of k corresponding to the maximum contour coefficient is taken as the optimal clustering number.

By using the above-mentioned method, k region proposals containing suspected ships are obtained. Fig. 4 shows an example of region proposal generation guided by key points. The red boxes in Fig. 4(a) are the ground truth bounding boxes of ships, and the green dots are the key points predicted by the key point extraction network. The blue boxes in Fig. 4(b) are the region proposals output by the region proposal generator. Fig. 4(c) shows the region proposals generated in the first phase, which will be used for accurate ship detection in the second phase.



(a) (b) (c)

Fig. 4. Example of region proposal generation guided by key points. (a) Prediction results of the key point extraction network. (b) Output process of the region proposal generator. (c) Region proposals obtained in the first phase.

B. Lightweight Object Detection Network

Common object detection networks use conventional convolution to extract deep semantic features, which leads to complex network structure and high resource consumption. Since multiple image blocks generated in the first phase will greatly increase the amount of data, it is necessary to carry out lightweight design for the object detection network. However, common lightweight object detection networks have weak feature extraction ability, low detection accuracy, and poor robustness in complex scenes. To solve the above-mentioned problems, this article proposes a lightweight object detection network with both speed and precision in the second phase. The region proposals generated in the first phase are input into the lightweight object detection network to achieve high-precision detection of ships.

YOLOv5 [35] is a fast single-stage object detection network, which adopts adaptive anchor box, pixel aggregation networks (PAN) [36], and multiscale detection to obtain high detection accuracy. Thanks to the effective feature fusion method and powerful multiscale detection ability, YOLOv5 can still achieve fast and accurate ship detection even in remote sensing images that are susceptible to noise interference, low resolution, and complex scenes. Therefore, the lightweight object detection network is improved on the basis of YOLOv5, and its structure is shown in Fig. 5. In order to achieve the lightweight design of object detection network, the backbone uses the improved MobileNetv2 built by G-SB to greatly reduce the number of parameters and computation of the network. It should be noted that the ConvBNSiLU module is obtained by connecting Conv,

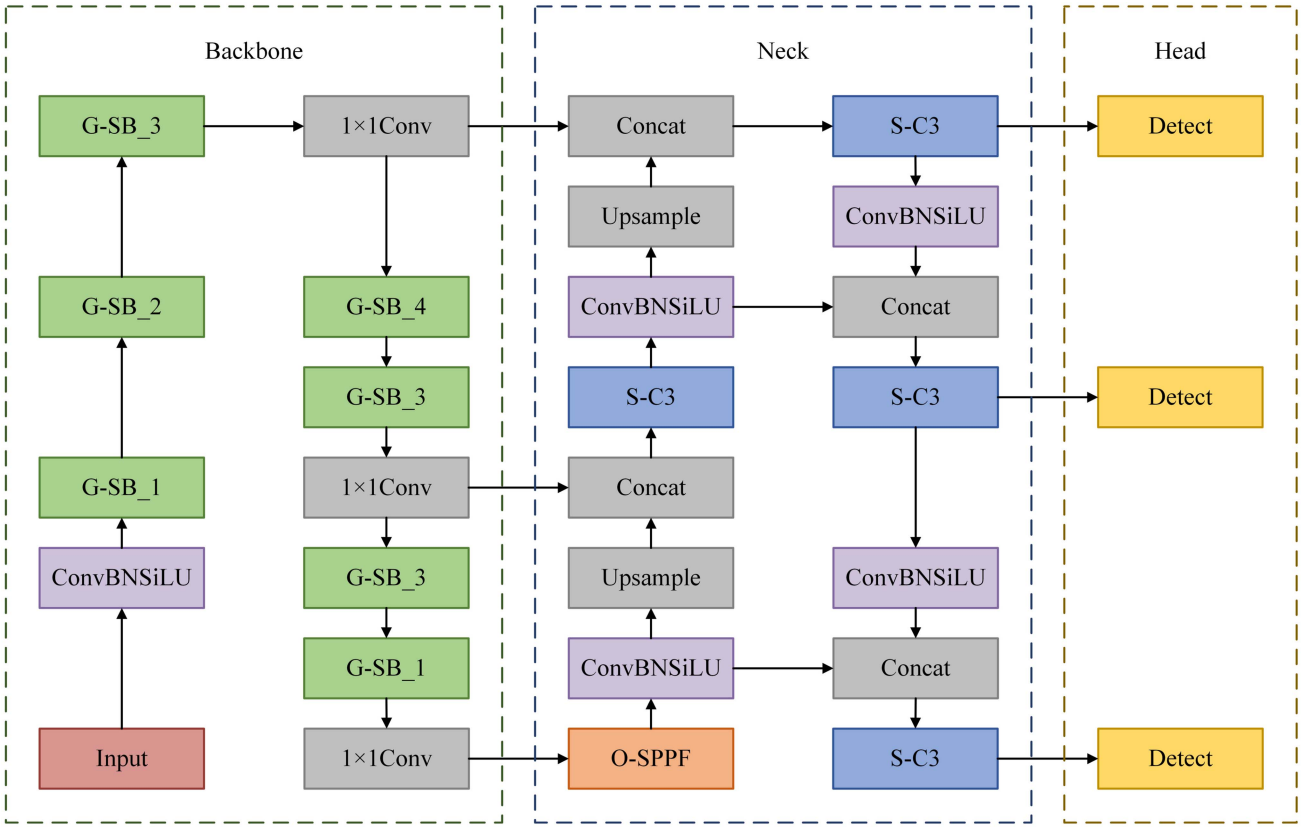


Fig. 5. Architecture of lightweight object detection network. It is designed based on YOLOv5 and consists of a backbone, neck and head. The basic module of the backbone is G-SB, the neck is a PAN constructed by ConvBNSiLU, S-C3 and O-SPPF, and the head contains three detection layers of different scales.

batch normalization (BN), and SiLU in turn. Among them, BN has the function of accelerating convergence speed and preventing overfitting. SiLU is a differentiable activation function that can stabilize the gradient calculation during backpropagation. To ensure the detection accuracy of the lightweight object detection network, SimAM is embedded in the C3 module of PAN to obtain the S-C3 module to enhance the feature representation of ships. Moreover, spatial pyramid pooling-fast (SPPF) and oriented pooling are combined to obtain O-SPPF to capture long-distance dependencies.

1) Grouped Sandglass Block: The basic idea of MobileNet is to use depthwise separable convolution instead of conventional convolution. It uses depthwise convolution for feature extraction and pointwise convolution for feature combination, which reduces the number of parameters and computation. As an improved version of MobileNet, the core structure of MobileNetv2 is the inverted residual block, which adopts a channel structure with narrow side and wide middle. The structure of the inverted residual block is shown in Fig. 6(a). First, the input features are extended using 1×1 convolution to map the low-dimensional features to the high-dimensional space. Then, 3×3 depthwise convolution is used for the high-dimensional features. Finally, the high-dimensional features are restored to the low-dimensional space using 1×1 convolution. Specifically, ReLU6 [37] connected after depthwise convolution is a function with sparse activation characteristics, and its activation range is limited, which can effectively reduce the computational complexity of the network.

For the inverted residual block, the dimension of the input features is low (fewer channels), which results in insufficient feature information being provided. To solve this problem and further reduce the number of parameters and computation of the network, this article uses G-SB to replace the inverted residual block to obtain an improved MobileNetv2. When the number of channels of input and output features are the same, the structure of G-SB is shown in Fig. 6(b). Specifically, the input features are divided into two branches after channel splitting. The secondary branch is the residual connection used to mitigate the gradient disappearance, and the primary branch is the bottleneck structure used to reduce the number of parameters. In the primary branch, two 3×3 depthwise convolutional layers are used to preserve the spatial dimension of the features, and two consecutive 1×1 convolutional layers are used to compress and expand the number of channels of the features. The primary and secondary branches are concatenated, and the output features are obtained after channel shuffling. When the number of channels of input and output features are different, the structure of G-SB is shown in Fig. 6(c). In this case, channel splitting and shuffling are not required.

Unlike the inverted residual block, the channel structure of G-SB is hourglass-shaped, which ensures sufficient feature extraction. In addition, G-SB has a wider network structure (more

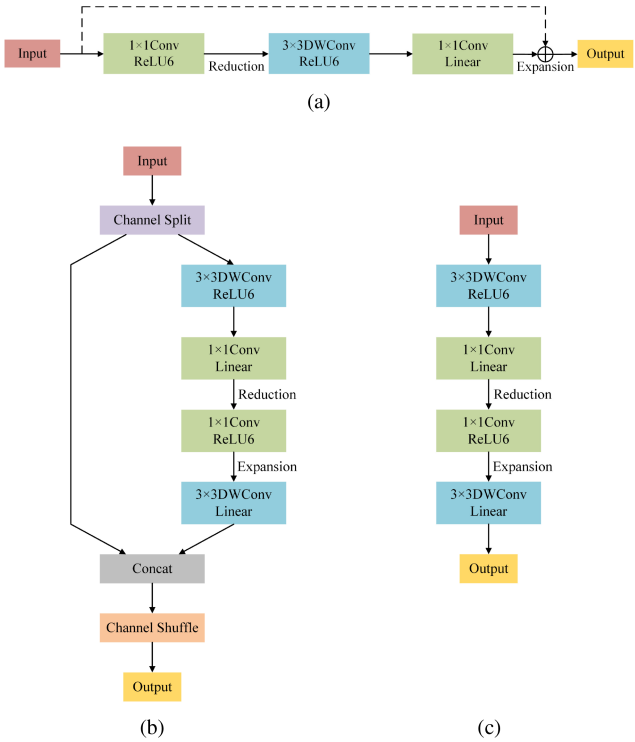


Fig. 6. Comparison between the inverted residual block and G-SB. (a) Structure of the inverted residual block. (b) Structure of G-SB when the number of input and output channels is the same. (c) Structure of G-SB when the number of input and output channels is different.

channels for input and output features), which helps to alleviate gradient confusion. The two major advantages of G-SB are the key to effectively improving network performance.

2) Simple Attention Module: Unlike common attention modules that simply connect spatial and channel attention in series or parallel, SimAM uses an energy function to explore the importance of each neuron, that is, to calculate the attention weights. Compared to other attention modules, SimAM can better focus on object features without introducing additional parameters to meet the lightweight requirements of the network.

Activated neurons in the visual nerve produce spatial inhibition of peripheral neurons, and neurons with this effect have a higher priority in visual processing. Therefore, linear separability is used to define the energy function

$$e_t\left(w_t, b_t, \mathbf{y}, x_i\right) = \frac{1}{M-1}\sum_{i=1}^{M-1}\left(-1-\left(w_t x_i + b_t\right)\right)^2 + \left(1-\left(w_t t + b_t\right)\right)^2 + \lambda w_t^2 \tag{3}$$

where t and x_i are the object neuron and other neurons in a single channel of the input feature $\mathbf{X} \in \mathbb{R}^{C\times H\times W}$, respectively. $w_t x_i + b_t$ and $w_t t + b_t$ are the linear transformations of t and x_i, respectively. M is the number of neurons in a single channel. According to the mean and variance of all neurons in a single channel, the closed-form solutions of weight w_t and bias b_t are

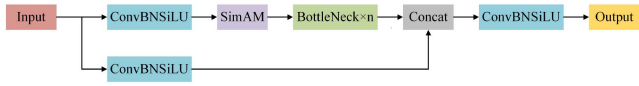


Fig. 7. Structure of S-C3 module. It contains three ConvBNSiLUs, one SimAM, and n BottleNecks, with SimAM located before BottleNeck.

calculated to obtain the minimum energy function

$$e_t^* = \frac{4\left(\hat{\sigma}^2 + \lambda\right)}{\left(t-\hat{\mu}\right)^2 + 2\hat{\sigma}^2 + 2\lambda} \tag{4}$$

where $\hat{\mu} = \frac{1}{M}\sum_{i=1}^{M} x_i$ and $\hat{\sigma}^2 = \frac{1}{M}\sum_{i=1}^{M}\left(x_i - \hat{\mu}\right)^2$. The above-mentioned equation shows that the lower the energy e_t^*, the object neuron t is more distinctive from surround neurons, and more important for visual processing. Inspired by the gain effect on neuron responses, we use the scaling operator for feature refinement. By applying SimAM, the output feature $\tilde{\mathbf{X}} \in \mathbb{R}^{C\times H\times W}$ is given as

$$\tilde{\mathbf{X}} = \mathrm{Sigmoid}\left(\frac{1}{\mathbf{E}}\right) \odot \mathbf{X} \tag{5}$$

where $\mathbf{E}$ groups all e_t^* across channel and spatial dimensions. Sigmoid means a nonlinear activation function, which is used to restrict too large value in $\mathbf{E}$ to avoid affecting the relative importance of each neuron.

To strengthen the multiscale feature fusion process and enhance the network's attention to ships, SimAM is embedded in the C3 module to effectively improve detection accuracy. As shown in Fig. 7, SimAM is added before BottleNeck to obtain the S-C3 module.

3) Oriented Spatial Pyramid Pooling-Fast: Spatial pooling can effectively capture long-distance contextual information in object detection tasks. Spatial pyramid pooling (SPP) [38] expands the receptive field of the network by combining pooling layers with different kernel sizes to provide global information for images. Constrained by the square pooling kernel, SPP lacks the ability to capture directional contextual information. Considering the importance of directional information for ship detection, Oriented Pooling is proposed in this article to capture long-distance dependencies more effectively. Compared with global pooling, oriented pooling has the following advantages.

1) It deploys striped pooling kernels along a spatial dimension to capture long-distance relationships in isolated regions.
2) It has narrow pooling kernels in other spatial dimensions to capture local context information and prevent interference from irrelevant regions.

The structure of oriented pooling is shown in Fig. 8. For the input feature map $X \in \mathbb{R}^{C\times H\times W}$, where C denotes the number of channels of the feature map. H and W represent the height and width of the feature map, respectively. The spatial range of the striped pooling kernel is $(H, 1)$ or $(1, W)$, and it averages all feature values in rows or columns. First, we input X into two parallel paths and perform horizontal and vertical pooling,

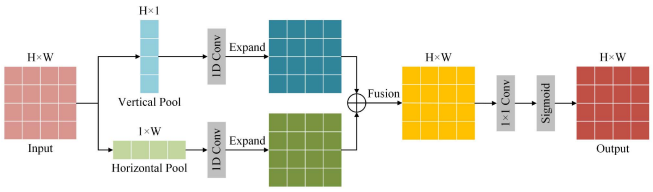


Fig. 8. Structure of oriented pooling. It utilizes striped pooling kernels in the spatial dimension to capture long-distance dependencies.

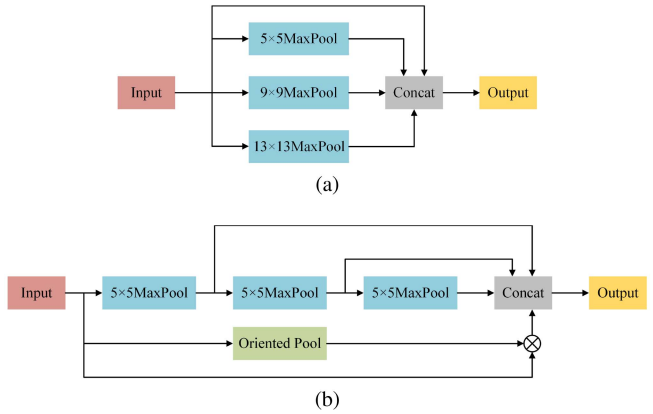


Fig. 9. Comparison between SPP and O-SPPF. (a) Structure of SPP. (b) Structure of O-SPPF.

respectively

$$Y_{c,i}^{h} = \frac{1}{W} \sum_{0 \le j < W} X_{c,i,j} \tag{6}$$

$$Y_{c,j}^{v} = \frac{1}{H} \sum_{0 \le i < H} X_{c,i,j} \tag{7}$$

where $Y_c^h \in \mathbb{R}^{1\times H}$ and $Y_c^v \in \mathbb{R}^{1\times W}$. Then, Y^h and Y^v are modulated by 3×3 1-D convolution to the current position and its adjacent features. Furthermore, the output feature maps of the two paths are added to obtain a feature map containing more useful global priors. The above-mentioned operations can be expressed as

$$Y_c = \mathrm{Conv}_{3\times3}^{1\mathrm{D}}\left(Y_c^h\right) + \mathrm{Conv}_{3\times3}^{1\mathrm{D}}\left(Y_c^v\right) \tag{8}$$

where $Y_c \in \mathbb{R}^{H\times W}$, and $\mathrm{Conv}_{3\times3}^{1\mathrm{D}}$ denotes 3×3 1-D convolution. Finally, we successively perform 1×1 convolution and $\mathrm{Sigmoid}$ normalization on Y_c to obtain the output feature map

$$Z = \mathrm{Sigmoid}\left(\mathrm{Conv}_{1\times1}(Y)\right) \tag{9}$$

where $Z \in \mathbb{R}^{C\times H\times W}$, and $\mathrm{Conv}_{1\times1}$ denotes 1×1 convolution.

The structure of SPP is shown in Fig. 9(a), which connects three pooling layers with kernel sizes of 5×5, 9×9, and 13×13 in parallel. Unlike SPP, SPPF connects three pooling layers with kernel size 5×5 in series to improve detection speed while maintaining the same detection accuracy. It is worth noting that all 5×5 max pooling stride is 1 and padding is 2, which guarantees that the input and output feature maps are the same size. For regions where semantic information is densely distributed, SPPF is a necessary condition for capturing local context information. Oriented pooling makes it possible to connect dispersed regions and encode striped regions, which are used to capture long-distance dependencies between different locations. For object detection tasks, long-distance dependencies [39] are intended to establish associations between isolated regions in an image, helping to capture relationships between objects. To effectively improve the detection accuracy, this article combines SPPF and oriented pooling to propose O-SPPF, whose structure is shown in Fig. 9(b).

IV. Experiments

A. Datasets and Evaluation Metrics

In our experiments, we evaluate the performance of P2RNet on two optical remote sensing datasets, the public DOTA Ship and the self-built GE-JL1.

1) DOTA Ship: DOTA [40] is a large-scale dataset for remote sensing object detection, which includes 2806 images with 15 object classes. A total of 434 images containing 37 028 ship objects are selected from DOTA to construct a new remote sensing ship dataset DOTA Ship. These images are randomly divided to obtain a training set and a test set including 326 and 108 images, respectively. Considering that the original image in DOTA is too large, we crop it to the size of 1024×1024 and set the overlap to 25%. The cropped DOTA Ship contains a total of 80 222 ship objects, among which the training set and test set include 1983 and 641 images, respectively.

2) GE-JL1: Based on Google Earth[1] and Jilin-1 satellite,[2] a high-quality remote sensing ship dataset GE-JL1 was established. The remote sensing images of Google Earth integrate two data sources, satellite images and aerial images. The cumulative coverage area of the Jilin-1 satellite constellation has reached 133 million square kilometers, and 138 satellites are expected to be networked in the future, of which the spatial resolution of Gaofen 03D satellite image is better than 0.75 m. We collected a total of 2840 offshore and nearshore remote sensing images with a size of 1024×1024 containing ships, of which 1904 images are from Google Earth and 936 images are from Jilin-1 satellite. Moreover, 80% of these remote sensing images are used for training and 20% for testing. The sample images in GE-JL1 are shown in Fig. 10, where the first and second rows of images are from Google Earth and Jilin-1 satellite, respectively.

In our experiments, we adopt the authoritative evaluation metric average precision (AP) to evaluate the performance of different ship detection methods. Precision and recall are defined as follows:

$$\mathrm{Precision} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}} \tag{10}$$

$$\mathrm{Recall} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}} \tag{11}$$

[1][Online]. Available: https://earth.google.com/
[2][Online]. Available: https://www.jl1mall.com/

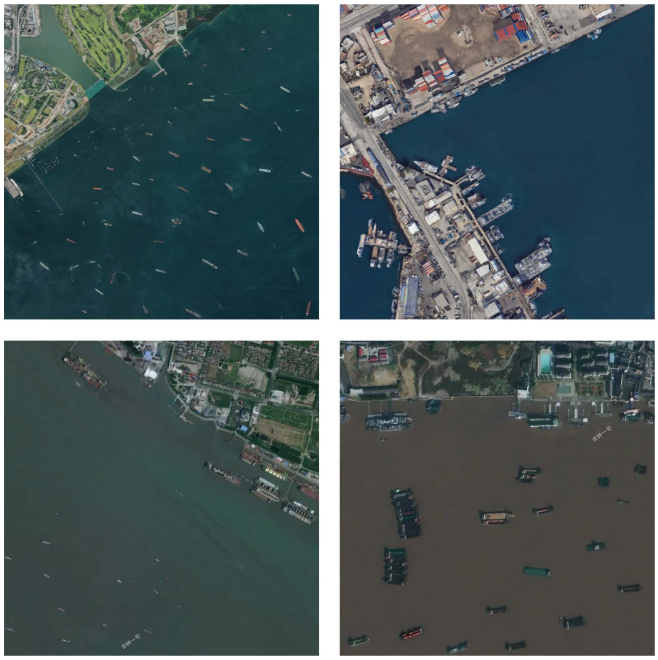


Fig. 10. Sample images in the GE-JL1 dataset.

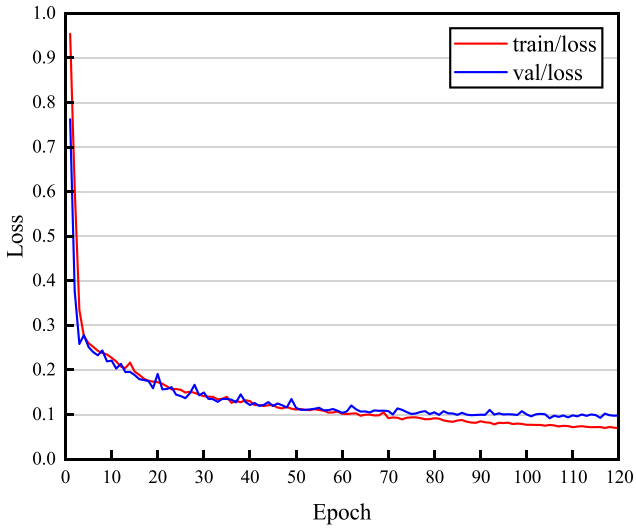


Fig. 11. Loss curves during training of key point extraction network. The validation loss tends to stable after 100 epochs.

where TP, FP, and FN represent the number of true positive, false positive, and false negative samples, respectively. For the discrimination standard of TP, when the IoU between the prediction box and the ground truth box exceeds 0.5, the object is considered to be correctly detected. AP can be calculated as

$$\mathrm{AP} = \int_0^1 P(R)dR \tag{12}$$

where $P(\cdot)$ and R represent precision and recall, respectively.

In addition, we use the number of parameters (Params) and floating-point operations per second (FLOPs) to evaluate the complexity of the network, and the frames per second as an evaluation metric of detection speed.

B. Implementation Details

We implement the proposed P2RNet in the Pytorch framework on Ubuntu v20.04 system. All experiments are evaluated on a high-performance computer with Intel Core i7 10700F CPU, NVIDIA GeForce RTX 3070 GPU, and 32-GB memory.

For the training of key point extraction network, we initialize the backbone network with ResNet18 pretrained on ImageNet. The Adam optimizer [41] is adopted for network training, and the momentum and weight decay are set to 0.9 and 0, respectively. We train the network 120 epochs in total, and the initial learning rate is set to 0.0005. The input image size is set to 1024 × 1024, and the batch size is set to 2.

For the training of lightweight object detection network, SGD [42] is selected as the optimizer, and the momentum and weight attenuation are set to 0.937 and 0.0005, respectively. We train the network 240 epochs in total using the warmup learning rate adjustment strategy, and the initial learning rate was set to 0.01. The input image size is set to 1024 × 1024, and the batch size is set to 8. Furthermore, random scaling, random flipping, and Mosaic are used for data augmentation.

C. Ablation Studies

The key point extraction network is trained on DOTA Ship, and the loss curves during training are shown in Fig. 11. The red and blue curves represent training and validation losses, respectively. The fitting process of validation loss is stable and the convergence effect is good. After 100 epochs, it gradually tends to be stable.

The lightweight object detection network is trained on DOTA Ship, and the loss curves during training are shown in Fig. 12. The red and blue curves represent training and validation losses, respectively. Fig. 12(a) shows the bounding box regression loss, where the validation loss gradually stabilizes after 160 epochs. Fig. 12(a) shows the confidence loss, where the validation loss gradually stabilizes after 180 epochs. Fig. 12(c) shows the angle classification loss, where the validation loss gradually stabilizes after 200 epochs.

In order to fully evaluate the contribution of each module in the lightweight object detection network, we performed ablation studies on DOTA Ship. It should be noted that all experiments adopt the same training and data augmentation strategies, and the research results are shown in Table I. The bold data in this table indicates the maximum value. In this article, the Baseline is YOLOv5-O with MobileNetv2 as the backbone and BCE Loss as the angle classification loss.

There is only 75.96% AP at Baseline, while FLOPs and Params reach 7.75 G and 6.3 M, respectively. G-SB uses channel splitting and shuffling operations, as well as a bottleneck structure to reduce the number of parameters and computation of the network. After replacing the inverted residual block in the backbone of Baseline with G-SB, FPS increases from 108 to 116, FLOPs and Params decrease to 5.01 G and 5.1 M, respectively, while AP decreases by only 0.15%.

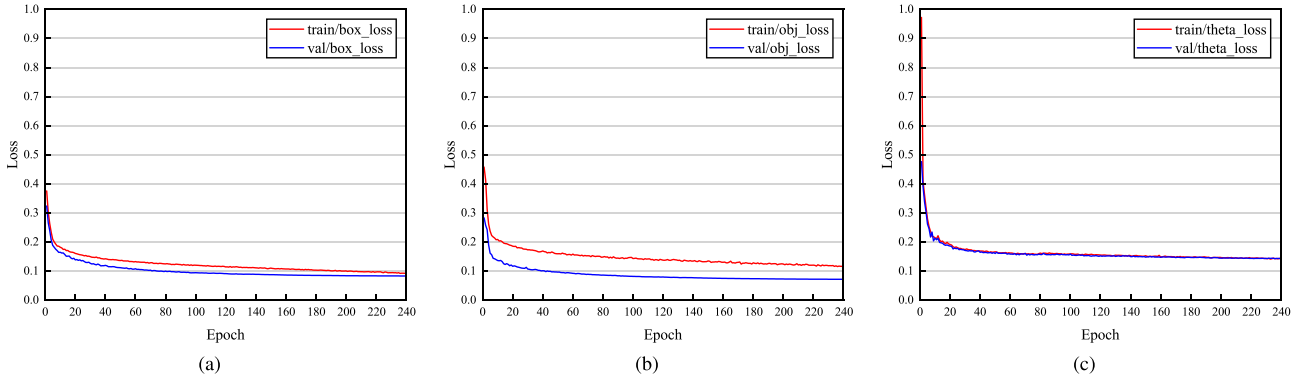


Fig. 12. Loss curves during training of lightweight object detection network. (a) Bounding box regression loss. (b) Confidence loss. (c) Angle classification loss. The validation losses of bounding box regression, confidence and angle classification tend to stabilize after 160, 180, and 200 epochs, respectively.

TABLE I
ABLATION STUDIES OF EACH MODULE ON THE DOTA SHIP DATASET

Baseline	G-SB	SimAM	O-SPPF	AP(%)	FPS	FLOPs(G)	Params(M)
✓				75.96	108	7.75	6.3
✓	✓			75.81	**116**	**5.01**	**5.1**
✓		✓		78.76	103	8.77	6.3
✓			✓	77.63	106	8.40	6.4
✓	✓	✓		78.38	113	5.73	**5.1**
✓	✓	✓	✓	**80.59**	112	6.17	5.2

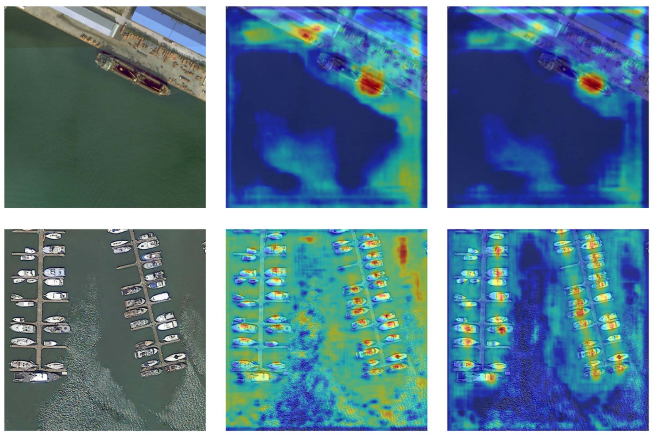


Fig. 13. Heatmap visualization after adding SimAM. (a) Original image. (b) Baseline. (c) Baseline+SimAM.

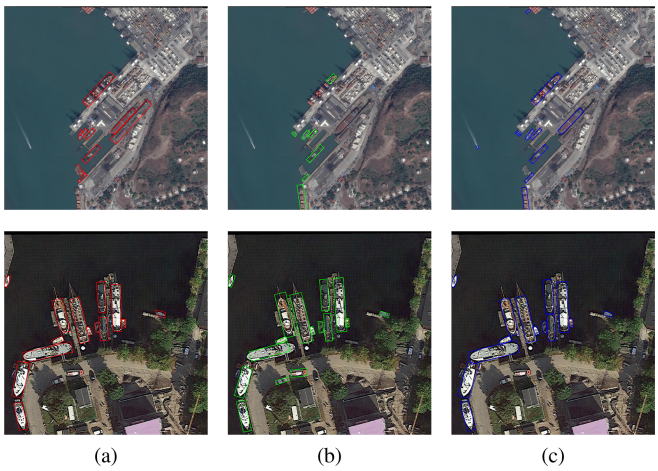
SimAM uses an energy function to calculate attention weights to enhance the network's attention to ships. After only embedding SimAM in the C3 module of Baseline, AP increases by 2.80%, while FPS decreases by only 5 and FLOPs increase by only 1.02 G, and Params remain unchanged. After adding G-SB and SimAM simultaneously in Baseline, AP and FPS reach 78.38% and 113, respectively, indicating good compatibility between the two. Furthermore, we visually verified the effectiveness of SimAM. Fig. 13 shows the heatmap visualization after adding SimAM. Fig. 13(a) shows the original image, Fig. 13(b) shows the heatmap visualization of Baseline, and Fig. 13(c) shows the heatmap visualization of Baseline+SimAM. Compared with Baseline, Baseline+SimAM can greatly suppress noise interference and better focus on ship features.

O-SPPF combines oriented pooling and SPPF to capture long-distance dependencies and local context information. After only replacing the SPPF in the backbone of neck with O-SPPF, AP increases by 1.67%, while FPS decreases by only 2, and FLOPs and Params remain almost unchanged. After adding G-SB, SimAM, and O-SPPF simultaneously in Baseline, AP and FPS reach 80.59% and 112, respectively, indicating that the three can work together effectively. Furthermore, we visually verified the effectiveness of O-SPPF. Fig. 14 shows the visualization of detection results after adding O-SPPF. Fig. 14(a) shows ground truth image, Fig. 14(b) shows the detection results of Baseline, and Fig. 14(c) shows the detection results of Baseline+O-SPPF. It is obvious that Baseline missed several inconspicuous small ships, and most of the detection boxes have a poor fit. Compared with Baseline, Baseline+O-SPPF can effectively capture the long-distance dependencies between densely distributed ships, as well as between the bow and stern of a ship. Therefore, it shows better detection performance in complex scenes.

In summary, by adding G-SB, SimAM, and O-SPPF, the lightweight object detection network achieves high AP and FPS while maintaining low FLOPs and Params. Ablation studies fully demonstrate the effectiveness of G-SB for reducing the number of parameters and computation, as well as the importance of SimAM and O-SPPF for improving the detection accuracy.

TABLE II
PERFORMANCE COMPARISON OF DIFFERENT DETECTION METHODS ON THE DOTA SHIP DATASET

Method	Label	Image size	AP(%)	FPS	FLOPs(G)	Params(M)
Faster R-CNN	HBB	512 × 512	59.76	10.4	49.37	40.2
YOLOv3	HBB	416 × 416	62.81	74.7	65.31	58.7
YOLOv4-Tiny	HBB	416 × 416	76.84	**113.5**	6.96	6.1
DETR	HBB	512 × 512	79.26	28.0	86.0	41.0
YOLOv7	HBB	640 × 640	80.38	84.2	104.7	36.9
RRPN	OBB	800 × 800	65.12	7.6	87.53	65.3
RRD	OBB	384 × 384	79.76	21.2	98.44	83.6
RoI-Trans	OBB	512 × 800	83.04	11.3	115.42	96.0
Oriented RepPoint	OBB	800 × 800	**83.90**	16.1	49.4	35.6
YOLOv5-O	OBB	1024 × 1024	75.96	108	7.75	6.3
Ours	OBB	1024 × 1024	80.59	112	**6.17**	**5.2**



(a) (b) (c)

Fig. 14. Visualization of detection results after adding O-SPPF. (a) Ground truth image. (b) Baseline. (c) Baseline+O-SPPF.

D. Comparison With Other Methods

To fully verify the validity of the lightweight object detection network, we compared it with ten representative object detection methods on DOTA Ship, including faster R-CNN, YOLOv3 [43], YOLOv4-Tiny [44], DETR [45], YOLOv7 [46], RRPN [47], RRD [48], RoI-Trans, oriented RepPoint [49], and YOLOv5-O. The performance comparison of different detection methods on DOTA Ship is shown in Table II. The bold data in this table indicates the maximum value, and the underlined data indicates the sub-maximum value.

Our proposed method achieves 80.59% AP and 112 FPS, respectively, while FLOPs are only 6.17 G and Params are only 5.2 M. Compared with faster R-CNN, YOLOv3, DETR, YOLOv7, RRPN, and RRD, our proposed method achieves the highest AP and the fastest FPS, and its FLOPs and Params are also the lowest. Compared with advanced RoI-Trans and oriented RepPoint, our proposed method has slightly lower AP, but FPS, FLOPs, and Params are substantially ahead. Our proposed method sacrifices a small amount of detection accuracy, but saves a lot of computational resources. Compared with YOLOv4-Tiny, the AP of our proposed method increases by 3.75%, FLOPs and Params decrease by 0.79 G and 0.9 M, respectively, while FPS decreases by only 1.5. Compared with the baseline model YOLOv5-O, the evaluation metrics of the proposed method are leading across the board. In conclusion, our proposed method achieves a good balance between detection accuracy and speed, meeting the lightweight requirements of the network.

To visually compare the performance of different object detection methods, we visualized the detection results. Fig. 15 shows the comparison of detection results of different methods on DOTA Ship. Fig. 15(a)–(e) shows the detection results of faster R-CNN, YOLOv3, RRPN, RRD, and our proposed method, respectively. Faster R-CNN and YOLOv3 use horizontal boxes for labeling, while RRPN, RRD, and our proposed method use rotating boxes for labeling. Because the horizontal box will contain a large amount of background information, the detection results of faster R-CNN and YOLOv3 are poor in dense scenes. Obviously, both of them have missed detections (labeled with a red box), and the fit between the detection box and the ship is poor. The detection results of RRPN and RRD are relatively good, but there are false detections in complex scenes. Compared with the above-mentioned three methods, our proposed method can identify and locate objects more accurately, no matter small or dense ships in complex scenes.

We further validate the robustness of the lightweight object detection network on GE-JL1. Specifically, it is compared with seven classic object detection methods, Libra R-CNN [50], SSD, YOLOv8, R^2CNN [51], SCRDet [52], R^3Det [53], and YOLOv5-O. The performance comparison of different detection methods on GE-JL1 is shown in Table III. The bold and underlined data in this table indicate the maximum and sub-maximum values, respectively.

Our proposed method achieves 82.62% AP and 112 FPS, respectively. Compared with Libra R-CNN, SSD, and R^2CNN, the AP of our proposed method increases by 15.08%, 12.9%, and 9.23%, respectively. Meanwhile, its FPS, FLOPs, and Params are several times better than the above-mentioned three methods. Compared with YOLOv8, the AP and FPS of our proposed method increase by 0.91% and 15.5, respectively, and FLOPs and Params are more advantageous. Compared with the latest

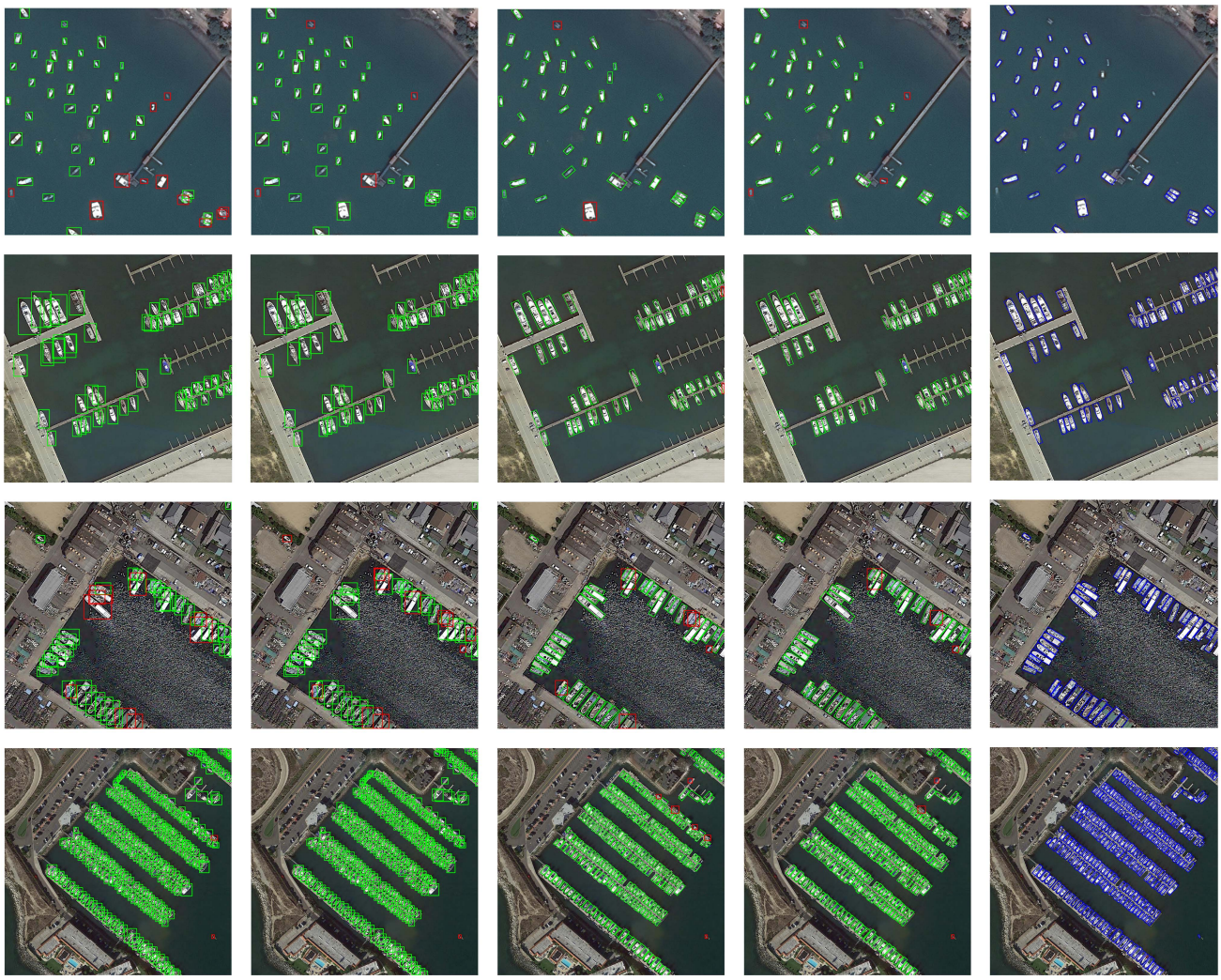


Fig. 15. Comparison of detection results of different methods on the DOTA Ship dataset. (a) Faster R-CNN. (b) YOLOv3. (c) RRPN. (d) RRD. (e) Ours.

TABLE III
PERFORMANCE COMPARISON OF DIFFERENT DETECTION METHODS ON THE GE-JL1 DATASET

Method	AP(%)	FPS	FLOPs(G)	Params(M)
Libra R-CNN	67.54	9.6	46.75	41.4
SSD	69.72	32.1	87.71	24.2
YOLOv8	81.71	96.5	28.60	11.2
R^2CNN	73.39	7.3	76.58	53.9
SCRDet	84.67	13.9	54.20	39.3
R^3Det	**85.03**	15.6	52.7	37.8
YOLOv5-O	77.63	108	7.75	6.3
Ours	82.62	**112**	**6.17**	**5.2**

SCRDet and R^3Det, the AP of our proposed method decreases by only 2.05% and 2.41%, respectively, while FPS, FLOPs and Params are completely superior. Compared with the lightweight YOLOv5-O, the AP of the proposed method is improved by 4.99%, and other evaluation metrics are also more advantageous. Obviously, our proposed method is robust on GE-JL1.

In order to verify the feasibility of P2RNet, we visualized its detection results on large-scale remote sensing images. The detection results of P2RNet on the uncropped original DOTA Ship are shown in Fig. 16. Thanks to the two-phase detection strategy from key points to region proposals, P2RNet achieves good detection results in different scenes while maintaining lightweight. Since the region proposals are generated by the guidance of key points, the recall can be greatly guaranteed to avoid missed detection. In addition, SimAM and O-SPPF in the lightweight object detection network can effectively improve the detection accuracy and avoid false detection. In conclusion, for small and dense ships in complex scenes in large-scale remote sensing images, P2RNet efficiently realize accurate ship detection while maintaining very low missed and false detection rates.

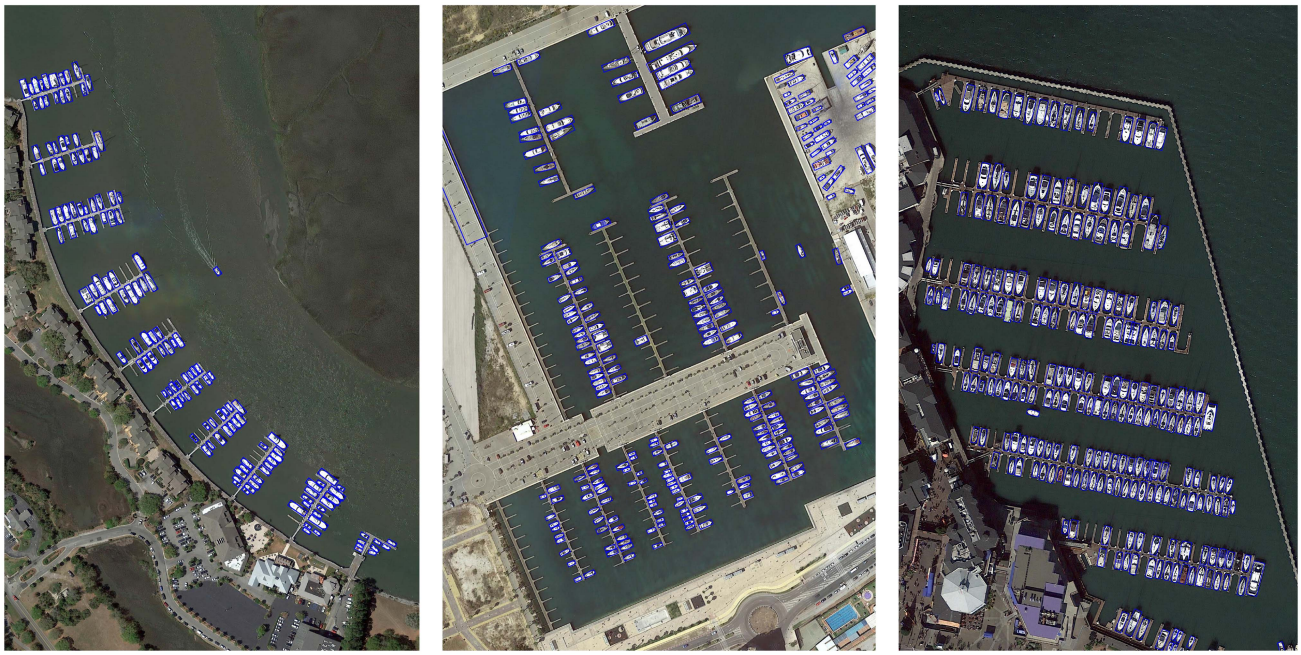


Fig. 16. Detection results of P2RNet on the original DOTA Ship dataset. It can quickly and accurately detect dense small ships in various scenes.

V. Conclusion

In this article, a two-phase object detection network P2RNet from key points to region proposals is proposed to achieve efficient maritime object detection on large-scale remote sensing images. In the first phase, the position information of suspected objects, namely key points, is obtained through the key point extraction network, and then region proposals are divided using the region proposal generator. In the second phase, these region proposals are input into the lightweight object detection network to achieve real-time and high-precision ship detection. The lightweight object detection network is based on YOLOv5 for multiscale ship detection. To reduce the number of parameters and computation, G-SB is used to construct a new lightweight backbone to extract sufficient feature information. To improve the detection accuracy, SimAM is embedded in the feature fusion network to strengthen the feature fusion process, and O-SPPF is proposed to capture long-distance dependencies.

In this article, the detection performance of P2RNet is verified on the DOTA Ship and GE-JL1 datasets. Ablation studies fully prove the effectiveness of G-SB, SimAM, and O-SPPF in the lightweight object detection network. Comparative experiments on the DOTA Ship dataset show that the lightweight object detection network reaches 80.59% AP and 112 FPS, respectively, while maintaining low FLOPs and Params, meeting the lightweight requirements of the network. In addition, the lightweight object detection network achieves 82.62% AP on the GE-JL1 dataset, indicating that its robustness has been fully demonstrated. Finally, the detection results on the original large-scale remote sensing images fully demonstrate the feasibility of P2RNet.

References

[1] X. You and W. Li, "A sea-land segmentation scheme based on statistical model of sea," in *Proc. 4th Int. Congr. Image Signal Process.*, 2011, vol. 3, pp. 1155–1159.

[2] J. Tang, C. Deng, G.-B. Huang, and B. Zhao, "Compressed-domain ship detection on spaceborne optical image using deep neural network and extreme learning machine," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 3, pp. 1174–1185, Mar. 2015.

[3] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, 2005, vol. 1, pp. 886–893.

[4] X. Nie, M. Duan, H. Ding, B. Hu, and E. K. Wong, "Attention mask R-CNN for ship detection and segmentation from remote sensing images," *IEEE Access*, vol. 8, pp. 9325–9334, 2020.

[5] Z. Ren, Y. Tang, Z. He, L. Tian, Y. Yang, and W. Zhang, "Ship detection in high-resolution optical remote sensing images aided by saliency information," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5623616.

[6] C. Qin, X. Wang, G. Li, and Y. He, "An improved attention-guided network for arbitrary-oriented ship detection in optical remote sensing images," *IEEE Geosci. Remote Sens. Lett.*, vol. 19, 2022, Art. no. 6514805.

[7] Y. Han, X. Yang, T. Pu, and Z. Peng, "Fine-grained recognition for oriented ship against complex scenes in optical remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2021, Art. no. 5612318.

[8] Y. Li, B. He, F. Melgani, and T. Long, "Point-based weakly supervised learning for object detection in high spatial resolution remote sensing images," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 14, pp. 5361–5371, 2021.

[9] N. Su, Z. Huang, Y. Yan, C. Zhao, and S. Zhou, "Detect larger at once: Large-area remote-sensing image arbitrary-oriented ship detection," *IEEE Geosci. Remote Sens. Lett.*, vol. 19, 2022, Art. no. 6505605.

[10] P. Wang, X. Sun, W. Diao, and K. Fu, "FMSSD: Feature-merged single-shot detection for multiscale objects in large-scale remote sensing imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 5, pp. 3377–3390, May 2020.

[11] Y. Yu, X. Yang, J. Li, and X. Gao, "A cascade rotated anchor-aided detector for ship detection in remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5600514.

[12] Y. Shen, D. Liu, F. Zhang, and Q. Zhang, "Fast and accurate multi-class geospatial object detection with large-size remote sensing imagery using CNN and truncated NMS," *ISPRS J. Photogrammetry Remote Sens.*, vol. 191, pp. 235–249, 2022.

[13] 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, 2016.
[14] 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.*, 2016, pp. 779–788.
[15] W. Liu et al., "SSD: Single shot multibox detector," in *Proc. 14th Eur. Conf. Comput. Vis.*, 2016, pp. 21–37.
[16] Y. Wu, K. Zhang, J. Wang, Y. Wang, Q. Wang, and X. Li, "GCWNeT: A global context-weaving network for object detection in remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5619912.
[17] G. Shi, J. Zhang, J. Liu, C. Zhang, C. Zhou, and S. Yang, "Global context-augmented objection detection in VHR optical remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 12, pp. 10604–10617, Dec. 2021.
[18] L.-g. Zhang, L. Wang, M. Jin, X.-s. Geng, and Q. Shen, "Small object detection in remote sensing images based on attention mechanism and multi-scale feature fusion," *Int. J. Remote Sens.*, vol. 43, no. 9, pp. 3280–3297, 2022.
[19] M. Zhu et al., "Arbitrary-oriented ship detection based on retinanet for remote sensing images," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 14, pp. 6694–6706, 2021.
[20] J. Ding, N. Xue, Y. Long, G.-S. Xia, and Q. Lu, "Learning ROI transformer for oriented object detection in aerial images," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 2844–2853.
[21] Y. Li, C. Kong, L. Dai, and X. Chen, "Single-stage detector with dual feature alignment for remote sensing object detection," *IEEE Geosci. Remote Sens. Lett.*, vol. 19, 2022, Art. no. 6503605.
[22] Y. Liu, Q. Li, Y. Yuan, Q. Du, and Q. Wang, "ABNeT: Adaptive balanced network for multiscale object detection in remote sensing imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5614914.
[23] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 1800–1807.
[24] F. N. Iandola, S. Han, M. W. Moskewicz, K. Ashraf, W. J. Dally, and K. Keutzer, "Squeezenet: Alexnet-level accuracy with 50x fewer parameters and $<$ 0.5 mb model size," 2016, *arXiv:1602.07360*.
[25] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "MobileNetV2: Inverted residuals and linear bottlenecks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 4510–4520.
[26] X. Zhang, X. Zhou, M. Lin, and J. Sun, "ShuffleNet: An extremely efficient convolutional neural network for mobile devices," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 6848–6856.
[27] K. Han, Y. Wang, Q. Tian, J. Guo, C. Xu, and C. Xu, "GhostNet: More features from cheap operations," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 1577–1586.
[28] Y. Li, J. Li, W. Lin, and J. Li, "Tiny-DSOD: Lightweight object detection for resource-restricted usages," 2018, *arXiv:1807.11013*.
[29] Y.-M. Zhang, C.-C. Lee, J.-W. Hsieh, and K.-C. Fan, "CSL-YOLO: A cross-stage lightweight object detector with low FLOPs," in *Proc. IEEE Int. Symp. Circuits Syst.*, 2021, pp. 2730–2734.
[30] Z. Qin et al., "ThunderNet: Towards real-time generic object detection on mobile devices," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2019, pp. 6717–6726.
[31] Q. Tang, J. Li, Z. Shi, and Y. Hu, "Lightdet: A lightweight and accurate object detection network," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process.*, 2020, pp. 2243–2247.
[32] P. Ding, H. Qian, and S. Chu, "Slimyolov4: Lightweight object detector based on yolov4," *J. Real-Time Image Process.*, vol. 19, no. 3, pp. 487–498, 2022.
[33] X. Zhou, D. Wang, and P. Krähenbühl, "Objects as points," 2019, *arXiv:1904.07850*.
[34] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 770–778.
[35] G. Jocher et al., "ultralytics/yolov5: V6. 1-tensorrt, tensorflow edge TPU and openvino export and inference," *Zenodo*, 2022.
[36] W. Wang et al., "Efficient and accurate arbitrary-shaped text detection with pixel aggregation network," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2019, pp. 8439–8448.
[37] A. G. Howard et al., "Mobilenets: Efficient convolutional neural networks for mobile vision applications," 2017, *arXiv:1704.04861*.
[38] K. He, X. Zhang, S. Ren, and J. Sun, "Spatial pyramid pooling in deep convolutional networks for visual recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 9, pp. 1904–1916, Sep. 2015.
[39] Z. Fu, J. Li, L. Ren, and Z. Chen, "SLDDNet: Stage-wise short and long distance dependency network for remote sensing change detection," *IEEE Trans. Geosci. Remote Sens.*, vol. 61, 2023, Art. no. 3000319.
[40] G.-S. Xia et al., "DOTA: A large-scale dataset for object detection in aerial images," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 3974–3983.
[41] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*.
[42] J. Han, J. Ding, J. Li, and G.-S. Xia, "Align deep features for oriented object detection," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5602511.
[43] J. Redmon and A. Farhadi, "Yolov3: An incremental improvement," 2018, *arXiv:1804.02767*.
[44] C.-Y. Wang, A. Bochkovskiy, and H.-Y. M. Liao, "Scaled-YOLOv4: Scaling cross stage partial network," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 13024–13033.
[45] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, "End-to-end object detection with transformers," in *Proc. Eur. Conf. Comput. Vis.*, Springer, 2020, pp. 213–229.
[46] C.-Y. Wang, A. Bochkovskiy, and H.-Y. M. Liao, "YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2023, pp. 7464–7475.
[47] J. Ma et al., "Arbitrary-oriented scene text detection via rotation proposals," *IEEE Trans. Multimedia*, vol. 20, no. 11, pp. 3111–3122, Nov. 2018.
[48] M. Liao, Z. Zhu, B. Shi, G.-s. Xia, and X. Bai, "Rotation-sensitive regression for oriented scene text detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 5909–5918.
[49] W. Li, Y. Chen, K. Hu, and J. Zhu, "Oriented reppoints for aerial object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 1819–1828.
[50] J. Pang, K. Chen, J. Shi, H. Feng, W. Ouyang, and D. Lin, "Libra R-CNN: Towards balanced learning for object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 821–830.
[51] Y. Jiang et al., "R2CNN: Rotational region CNN for orientation robust scene text detection," in *Proc. Int. Conf. Pattern Recognit.*, 2018, pp. 3610–3615.
[52] X. Yang et al., "SCRDet: Towards more robust detection for small, cluttered and rotated objects," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2019, pp. 8231–8240.
[53] X. Yang, J. Yan, Z. Feng, and T. He, "R3det: Refined single-stage detector with feature refinement for rotating object," in *Proc. AAAI Conf. Artif. Intell.* 2021, vol. 35, no. 4, pp. 3163–3171.

Yantong Chen received the Ph.D. degree in mechanical and electronic engineering from the Changchun Institute of Optics, Fine Mechanics and Physics, Chinese Academy of Sciences, Changchun, China, in 2017.

He is currently with Dalian Maritime University, Dalian, China. His research interests include ocean remote sensing image processing, computer vision, and deep learning.

Jialiang Wang received the B.S. degree from Dalian Maritime University, Dalian, China, in 2021. He is currently working toward the master's degree in information and communication engineering with the School of Information Science and Technology, Dalian Maritime University.

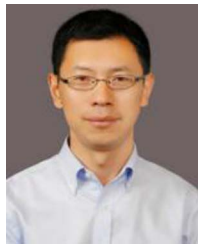
His research focuses on object detection.

Yanyan Zhang received the B.S. degree from Dalian Maritime University, Dalian, China, in 2021. He is currently working toward the master's degree in information and communication engineering with the School of Information Science and Technology, Dalian Maritime University.

His research focuses on object detection.

Yang Liu received the B.S. degree from Xinxiang University, Xinxiang, China, in 2020. He is currently working toward the master's degree in electronic information with the School of Information Science and Technology, Dalian Maritime University, Dalian, China.

His research focuses on object detection.



Junsheng Wang received the Ph.D. degree in mechanical and electronic engineering from the Harbin Institute of Technology, Harbin, China, in 2007.

He is currently with Dalian Maritime University, Dalian, China. His research interests include optoelectronic sensing and detection, micronano optofluidic chip, and signal and image processing.