An Efficient Center-Based Method With Multilevel Auxiliary Supervision for Multiscale SAR Ship Detection

Yu Zhang, Xueqian Wang, Member, IEEE, Zhizhuo Jiang, Gang Li, Senior Member, IEEE, and You He

Abstract—The problem of multiscale ship detection in synthetic aperture radar (SAR) images has received much attention with the development of deep convolutional neural networks (DCNNs). However, existing DCNN-based multiscale SAR ship detection methods often lead to time-consuming detection process due to the massive parameters therein. To address this issue, a lightweight center-based detector with the multilevel auxiliary supervision (MLAS) structure is proposed in this article. First, an extremely lightweight backbone network is designed to improve the computation efficiency and extract SAR image features in a bottom-up manner. Then, a feature fusion network containing three multiscale feature fusion modules is introduced to combine semantic features with different levels. Finally, a novel MLAS-based framework is proposed to train our DCNN with multilevel auxiliary detection subnets. MLAS improves the performance of multiscale ship detection benefiting from the guidance of multilevel attention. Experimental results on the open SAR image dataset SSDD show that our proposed detector achieves a similar average precision with the state-of-the-art methods. The required number of floating points of operations in our method is only 0.68 million that is 5.63%, 1.10%, and 2.98% of those of CenterNet, YOLOv3, and RetinaNet, respectively, and the number of learnable weights in our method is only 0.68 million that is 5.63%, 1.10%, and 2.98% of those of the aforementioned three existing methods, respectively.

Index Terms—Convolutional neural network, multiscale ship detection, synthetic aperture radar (SAR), detection, synthetic aperture radar (SAR).

I. INTRODUCTION

S HIP detection has attracted increasing attention due to its essential role in marine safety and management tasks. Synthetic aperture radar (SAR) has been widely used for ship detection since it can work well in all-weather and all-day conditions and generate high-resolution images [1], [2]. The pipelines of traditional SAR ship detection methods often contain two or more stages, e.g., sea-land masking [3], [4], prescreening [5]–[10], and discrimination [11]–[14]. In [15], superpixel-level segmentation stage [16], [17] is exploited to improve the detection performance of traditional constant false alarm rate (CFAR)-based methods [6], [18]. Note that the handcraft features used in these stages often suffer from weak generalization performance and reduce the robustness of the final detection performance [9].

In recent years, deep convolutional neural networks (DCNNs)-based end-to-end methods provide the capability of automatic feature extraction and lead to the rapid development of target detection mechanisms in the field of computer vision [18]–[27]. Thanks to the success of DCNNs in the context of computer vision, several DCNN-based SAR ship detectors have been elaborately designed to obtain improved detection performance compared with traditional detectors based on handcraft features [28]–[30]. Note that SAR images often contain multiscale ship targets due to the diversity of ship sizes and imaging modes (resolutions). For example, the area occupied by a ship target ranges from several clustered pixels to the whole window of a 256 × 256 SAR image chip. Two-stage DCNN-based detectors are first considered to solve the problem of multiscale ship detection in SAR images [31]–[37]. Li et al. [31] use transfer learning strategies and feature fusion modules to achieve the task of multiscale ship detection with the ZF-Net backbone network [38]. Jiao et al. [32] utilize densely connected multiscale features to enhance the accuracy of multiscale ship detection with the ResNet101 backbone network [39]. Lin et al. [33] introduce the ranked squeeze and excitation attention modules following the VGG [45] backbone to enhance the performance of ship detection. Based on the feature pyramid network (FPN) [40], Cui et al. [34] and Zhao et al. [35] apply convolutional block attention module [41] and receptive field block, respectively, to effectively detect ship targets in large SAR images. Cheng et al. [36] combine the saliency map into DCNN to enhance the performance of inshore ship detection. Tang et al. [37] modify the cascade region-based CNN (R-CNN) [42] in accordance with the revised Bhattacharyya distance to detect the large-difference-scale ships. Despite that a breakthrough has been made in terms of detection accuracy, existing DCNN-based detectors [31]–[35] for multiscale SAR ship targets still suffer...
from low computation efficiency because of their large model sizes.

Another line of research works makes an effort to design efficient DCNN-based detection methods for multiscale ship targets in SAR images. Li et al. [43] propose a lightweight R-CNN detector, where the model size (number of weights in network) is only 19 million. Based on the framework of single shot multibox detector, Zhang et al. [44] reduce half of channels in the VGG network [45] and use a bidirectional feature fusion module to guarantee the high detection accuracy of multiscale ship targets. Chang et al. [46] propose a You Only Look Once (YOLO)-based ship detector, where the depth of DCNN is significantly reduced to save the computation cost of target detection. In [47], Han et al. design a new DCNN structure with parallel convolutional blocks to improve the robustness of detection performance of ship targets versus various background conditions. Recently, Mao et al. [48] use two simplified U-nets as the backbone network and adopt the center-based framework [25], [27]. It is worth mentioning that backbone networks with a small number of weights may weaken the generalization capability of CNNs in terms of feature extraction and lead to deteriorated detection performance of multiscale ship targets [49].

In this article, we propose an efficient DCNN-based detector with a novel multilevel auxiliary supervision (MLAS) strategy for the problem of multiscale ship detection in SAR images. Our detector adopts the popular center-based framework, similar to [26], [48]. First, we introduce an extremely lightweight CNN with the residual structure as the backbone network. Then, the multiscale features are exploited by the feature fusion network (FFN) that contains two FPN-based modules [40] and one adaptively spatial feature fusion [50] (ASFF) module. These two FPN-based modules are applied at both ends of the FFN, which is beneficial to boost the propagation of high-level semantic features. ASFF automatically learns the in-network fusion strategy to alleviate the inconsistency in multiscale features fusion. Then, the MLAS module is innovatively developed to help our method focus on the target areas from a supervised perspective with the multiscale ground truth (GT) information. Benefitting from the combination of superiorities of the tiny backbone network, FFN, and MLAS, the proposed method requires only 0.68 million weights (fewer than existing state-of-the-art detectors [20], [22]–[24]) but still maintains good detection performance for multiscale SAR ship targets. Experimental results based on the SSDD dataset [31] show that our proposed method achieves more efficient ship detection with similar detection performance compared with existing competitors [31], [32], [35], [44], [48]. In Fig. 1, we provide a quick look of our method and existing methods for multiscale SAR ship detection in terms of model sizes and detection accuracies for the convenience of readers.

The remainder of this article is organized as follows. Section II introduces the network structure of our proposed method and the details of key components. Section III provides experimental results of our method and existing state-of-the-art methods in depth. In Section IV, we conclude this article with several remarks and hint at plausible research lines in the future.

![Fig. 1. Size of weights versus average precision (AP) of existing methods and our method on the SSDD dataset. The three red points denote the AP values of the proposed method with three different input resolutions, respectively. The number in parentheses denotes the input size of detectors, e.g., “(256)” represents that the input images have 256 × 256 pixels.](image)
features with different levels. Then, the second FPN is applied to enhance the top–down propagation of the multilevel semantic information fused by ASFF, ensuring that the lowest level (level 1) of the feature map (used to conduct the prediction) can obtain both the fine spatial feature and rich semantic feature.

Our MLAS-based strategy is inspired by CenterNet [26], which adopts center-based one-stage detection structure and uses two parallel subnets as the detection heads. One head is used to predict the probability that each pixel in the feature map is the center of the ship target, and the other head is used to predict the size of the bounding boxes of ship targets. Note that CenterNet [26] only exploits the single level in the training stage. This may cause excessive attention over the single-level features and degrade the detection performance of the multiscale ship targets. In comparison with the original CenterNet [26], a novel MLAS-based strategy is used to train our DCNN with the supervision of multilevel GT. As shown in Fig. 2, the detection subnets with different resolutions are attached to the feature maps with corresponding levels from the FPN2 module. MLAS-based strategy effectively supervises the propagation of gradients at different levels in the training stage, thereby guiding the network to pay more attention on multiscale ship targets at each level. The first level of MLAS (with the highest resolution) is used for the prediction. Benefiting from the MLAS-based strategy, the decoding algorithm can be simplified to the regression of three parameters, i.e., center position, width, and height, without any time-consuming auxiliary operations (such as anchor-based regression [19] or nonmaximum suppression). lightweight backbone structure is designed in this subsection. Our lightweight backbone network inherits the idea of the residual structure, where the successive Conv blocks with shortcut connections [39] (of different spatial resolutions) are applied in the forward pathway. As shown on the left side of Fig. 2, the solid rectangles named “B” with numerical subscripts represent the Conv blocks with corresponding spatial resolutions.

In our backbone network, the Conv block $B_{512}$ contains only one $3 \times 3$ Conv with one stride and eight filters and each Conv block in $B_{16}$-$B_{256}$ can be formulated as

$$
\begin{align*}
    X_1 &= \text{Conv}_{3\times3,s=2}^{N_{\text{filters}},c}(X) \\
    X_2 &= \text{Conv}_{1\times1,s=1}^{N_{\text{filters}},c/2}(X_1) \\
    X_3 &= \text{Conv}_{3\times3,s=1}^{N_{\text{filters}},c}(X_2) \\
    Y &= X_3 \odot X_1
\end{align*}
$$

(1)

where $X$ is the input feature, $X_1$, $X_2$, and $X_3$ are the results after the first Conv to the third Conv, respectively. $\odot$ denotes the concatenation. $Y$ is the output features of the block, which is obtained by concatenating $X_3$ and $X_1$ via a shortcut connection. $N_{\text{filters}}$ and $s$ denote the number of filters and the stride number in Conv, respectively. The different values of $N_{\text{filters}}$, i.e., $c = 16, 24, 32, 48, \text{ and } 64$, are assigned to $B_{256}, B_{128}, B_{64}, B_{32}$, and $B_{16}$, respectively, which are smaller than those of popular DCNNs [39], [52]. To exploit the abundant features of SAR images, our backbone network is expanded deeper by stacking the Conv blocks in levels 3 and 4 three times, while those in levels 2 and 5 twice. The settings of $c$ are derived from the following extensive experiments to balance the model size and detection accuracy of our method. Note that in our backbone network (and FFN, see Section II-B), batch normalization [57] layers are added after each Conv to obtain normalized feature maps, and then the leaky rectified linear unit (Leaky ReLU) [58] is applied as the activation function.

High-resolution features with low levels in Fig. 2 mean higher computational complexity. The finest spatial resolution with level 0 will significantly increase computational cost, while this article mainly focuses on efficient detection. In addition, we empirically find that the features with higher levels than 5 show limited accuracy improvement in the detection stage. Thus, in
Fig. 3. Details of the FPN1 block, where FS represents the feature smooth function and the superscript \( k \) of \( X \) denotes the feature level.

Fig. 4. Details of the ASFF block in level 3. \( \odot \) and \( \oplus \) denote element-wise multiplication and element-wise addition, respectively. Each rectangle box inside denotes the Conv with the 1 \( \times \) 1 kernel and 4 filters.

order to obtain the compromise between detection speed and performance, we use features with levels 1–5 in MLAS-based detection heads.

C. FFN

In our detector, three feature fusion modules (FPN1 + ASFF + FPN2) are applied in a cascaded way to obtain the multi-level SAR image features with rich semantics. The FPN1 module is associated with the backbone network to extract the spatial and semantic features with different scales. The ASFF module is used to handle the incompatibility of features produced by FPN1 with different scales. The FPN2 module applies a shorter Convs path than FPN1 to enhance the top–down propagation of the high-level semantic features. The function of each module is based on the features processed by the previous module, and they play their roles in sequence to achieve the final fusion effect.

Two FPN modules are used in FFN, in which the FPN1 module is applied at the front end of FFN to facilitate the downward propagation of high-level features. As shown in Figs. 2 and 4, FPN1 blocks are presented via a top–down pathway in FFN. All the FPN1 blocks share the same structure, see Fig. 3, where \( X^{[k]} \) and \( F^{[k]} \) are the input and output features of the \( k \)th block and FS denotes the feature smooth (FS) function. FS contains three 1 \( \times \) 1 Convs and two 3 \( \times \) 3 Convs. 1 \( \times \) 1 Convs in FS is beneficial to reduce the computational burden (due to the small number of parameters therein) and enhance the capability to acquire nonlinear features. The 3 \( \times \) 3 Convs are used to alleviate the aliasing effect of features between different spatial levels [59].

It is worth mentioning that features extracted by FPN1 may suffer from inconsistency among different levels [50]. ASFF [50] utilizes learnable fusion weights of multilevel features to alleviate their inconsistency. In our detector, the original ASFF [50] is pruned by reducing the number of filters in Convs in the process of weighted fusion. All the ASFF blocks have the same structures. As an example, ASFF with level 3 is illustrated in Fig. 4. SAR image features obtained from the FPN1 modules are resized using up-sample or down-sample operation to the same spatial size as the ASFF block with level 3. Then, each feature map with the same spatial size is processed by a Conv that is pruned to 4 filters (significantly smaller than 16 filters in [50]). The outputs of Convs in ASFF are multiplied by four learnable weights \( \{\alpha, \beta, \gamma, \eta\} \) corresponding to the four levels. All the weighted feature maps are added together to generate the final feature map \( A^{[3]} \) of ASFF with level 3.

In order to enhance the feature map with the lowest level generated by ASFF, another FPN module is applied at the back end of FFN, which is denoted as FPN2. FPN2 encourages the feature map with the lowest level to obtain both rich semantic feature and fine spatial feature. The structure of the FPN2 is similar to FPN1, but only uses a single 1 \( \times \) 1 Conv without the FS function in FPN1. The shorter pathway in FPN2 compared with FPN1 is beneficial to simplify the downward propagation of high-level features to low levels [60].

D. MLAS-Based Detection Heads

A novel MLAS-based detection structure is proposed in our detector, which attaches four detection heads to the corresponding input feature levels during the training stage to encourage our network focusing on target regions from a multilevel perspective.

We elaborate the MLAS-based training structure in Fig. 5, where four detection heads are added to each feature level from \( F'_{32} \) to \( F'_{256} \) (see Fig. 2) independently. Four GT templates with
different levels are utilized in the training stage. Each level of the
GT template (see red and solid boxes on the right side of Fig. 5) is
obtained by resizing the initial GT to the corresponding spatial
resolution. The GT template with a high level lets the network pay
attention to the expansive neighboring areas of ship targets and
obtain rich semantic information of SAR images. The low level of
GT template leads to finely spatial features to outline ship targets. The multilevel GT templates in our MLAS-based
training structure combine both semantic and spatial information
for multiscale ship targets.

As illustrated in Fig. 5, the detection heads in MLAS with
different feature levels contain identical structures with two
parallel subnets inside. The first subnet is the classification head
that consists of a $3 \times 3$ Conv with 8 filters and a $1 \times 1$ Conv
with 1 filter, then use sigmoid activation to generate the probability
heatmap of center points of ship targets. The second subnet is
similar to the first one but uses $1 \times 1$ kernel with two filters
in the second Conv to predict the bounding boxes of ships. The
encoder in the training stage is defined as the translation from the
annotation of GT templates to the output form of the network.

For each pixel in the current feature map three output values
of our network are generated, i.e., $(p_c, t_w, t_h)$, where $p_c$ is the
output of the first detection subnet, i.e., the probability of
the current pixel contains a target center, $t_w$ and $t_h$ are the outputs
of the second subnet, and defined as

$$\begin{cases}
  t_w = \frac{W_b}{W} \\
  t_h = \frac{H_b}{H}
\end{cases}$$

(2)

where $(W_b, H_b)$ denote the width and height of the target
bounding boxes in GT annotations, respectively, and $(W, H)$ denote
the width and height of feature maps, respectively.

In MLAS, we use the feature map in level 1 with the highest
spatial resolution to conduct the prediction during the test stage.
Instead of considering the offset of center points of ship targets
like in [26], we simply assume that the center of a pixel region
directly overlaps with the ship center inside the pixel region
when the current pixel region is labeled as a ship center. As the
resolution of GT becomes finer, the discretization errors caused
by our simplification will gradually decrease and are negligible,
as shown in Fig. 6.

The predicted values of the detection subnets are translated
to the final results via a decoder. In the decoder, a $3 \times 3$ max-
pooling layer following the first subnet is used to capture the peak
values in the probability heatmap. Pixels with probabilities larger
than a threshold will be selected as center points of ship targets.

Then, the bounding box regression is conducted according to
each center-point:

$$\begin{align*}
  b_{\text{left}} & = \max (0, |x + 0.5 - W \times t_w / 2|) \\
  b_{\text{top}} & = \max (0, |y + 0.5 - H \times t_h / 2|) \\
  b_{\text{right}} & = \min ((x + 0.5 + W \times t_w / 2), W) \\
  b_{\text{bottom}} & = \min ((y + 0.5 + H \times t_h / 2), H)
\end{align*}$$

(3)

where $(b_{\text{left}}, b_{\text{top}})$ and $(b_{\text{right}}, b_{\text{bottom}})$ represent the left-top and
right-bottom corners of the bounding box of a ship target,
respectively, $\lfloor \cdot \rfloor$ denotes the rounding down operator, and $(x, y)$
denotes the position of the center point on the heatmap.

It is worth noting that since there are often sparse ships in an
input small SAR image chip, most of the pixels in the heatmap
are labeled as background pixels. Thus, a two-dimensional Gaussian
kernel is used to weight the probabilities in the heatmap, the same
as [27].

E. Loss Function

Here, the focal loss in [24] is modified to calculate the loss
discrimination between ship targets and background in our
method (4) shown at the bottom of the next page, where $y_{i,j} \in
[0, 1]$ is the GT value at the pixel location $(i, j)$ in the heatmap,
$\tilde{y}_{i,j}$ is the predicted value at pixel location $(i, j)$ of the heatmap,
$N_{\text{pos}}$ is the total number of pixels with label $y = 1$, $\nu$ is the tunable
parameter that is set to 2 in our method, and $(W^k, H^k)$ denote
the width and height of the feature map with the $k$th level from
FPN2.
The loss regarding the size of bounding boxes of ship targets is computed by the L1 loss

\[
L_{\text{size}} = \frac{1}{N_{\text{pos}}} \sum_{k=1}^{4} \sum_{i,j} \sum_{m \in \{t_u, t_h}\} |\tilde{y}_{i,j,m} - \hat{y}_{i,j,m}| \tag{5}
\]

where \(\tilde{y}_{i,j,m}\) and \(\hat{y}_{i,j,m}\) represent the GT values and the predicted values of the second detection subnet at the pixel location \((i, j)\) of the feature map from FPN2, respectively.

The total loss is calculated as

\[
L_{\text{total}} = L_{\text{cls}} + \lambda L_{\text{size}} \tag{6}
\]

where \(\lambda\) is a balance hyper-parameter. We set \(\lambda = 10\) in our experiments due to its good performance in experiments. The effect of the balance hyper-parameter \(\lambda\) on the detection performance will be discussed in Section III-C.

III. EXPERIMENTAL RESULTS

In this section, experiments are conducted to verify the superiority of the proposed SAR ship detector. The SAR ship detection dataset (SSDD) [31] is introduced at first, as well as the evaluation criteria and implementation details. Then, a series of experimental results are provided to evaluate the effectiveness of the lightweight backbone, FFN, and the MLAS-based strategy in our method. Next, the performance of the proposed method is compared with existing state-of-the-art methods. Finally, the effect of hyper-parameter settings on our method is analyzed for completeness.

A. SSDD Dataset and Implementation Details

SSDD [31] is a widely used benchmark dataset for ship detection in SAR images. SSDD contains 1160 image patches with 2456 ship targets in total. Ship targets in SSDD exhibit diversity in size, as shown in Fig. 7. We divided the SSDD into training dataset, validation dataset, and test dataset according to the ratio of 7:1:2, the same as [48]. We use model size, the floating points of operations (FLOPs) [61], and frame per second (FPS) to evaluate the efficiency of SAR ship detectors, where the model size is the total number of weight parameters in DCNNs. Small model sizes and FLOPs and large values of FPS mean rapid detection of ship targets. In order to compare the detection performance of the different methods, the two commonly used indicators, i.e., precision (P) and recall ratio (R) are used to evaluate the detection performance of the different methods

\[
P = \frac{N_{td}}{N_d} \tag{7}
\]

\[
R = \frac{N_{td}}{N_{GT}} \tag{8}
\]

where \(N_{td}\) is the number of correctly detected ship targets, \(N_d\) is the total number of detected ship targets, and \(N_{GT}\) is the GT number of ship targets in the test dataset. All of them are obtained under the same decision threshold of probability values in heatmap. In this article, the decision threshold is set to 0.5 in all the experiments. The evaluation metrics F1 and AP are also used to evaluate the detection performance [35]

\[
F_1 = \frac{2 \times P \times R}{P + R} \tag{9}
\]

\[
AP = \int_0^1 P(R)\text{d}R. \tag{10}
\]

Note that AP can be calculated with different intersection-over-union values [62], and we only use AP\(_{0.5}\) in this article without loss of generality.

The batch size is 8 and the maximum value of the training iteration number is 300 for all the competitors and our method. The setting of the initial learning rate needs to be determined according to the network structures of the ship detectors, where the initial learning rate \(10^{-3}\)can be applied to most methods, whereas \(10^{-4}\) is adopted to enable DCNN in [40]. When the number of iterations increases by 100, the learning rate decreases by 10 times.

The proposed method is implemented by Tensorflow Platform [63] (version 2.7.0) and CUDA [64] (version 11.0) in the Window10 Operation System. The type of CPU is i7-8850h with 32 GB RAM. Our GPU is NVIDIA P3200 with a 6 GB video memory.

B. Effectiveness of Main Components in Our Detector

First, we show the effectiveness of our lightweight backbone network. We select the widely used backbone networks, such as VGG [45], ResNet [39], MobilenetV2 [65], tiny Darknet [23], and Efficientnet [51] as competitors. All the networks are pruned to a tiny size according to [48] and approximately contain 0.2 million (M) weights. The comparison experiments are conducted based on the identical components in our detector except for the backbone network. In Table I, it can be seen that the model size and the number of FLOPs of the proposed backbone network are 0.19 M (ranking third) and 0.70 G (ranking third), respectively, achieving comparable AP value (ranking 2nd) in comparison with the state-of-the-art backbone networks.

Although the light ResNet requires the fewest FLOPs, it provides a significantly smaller AP value than state-of-the-art backbone networks. From Table I, we can conclude that our backbone network provides the basic ability to achieve efficient ship detection in SAR images without noticeable loss of detection performance.

Note that as the increasing of the number of filters in Convs, DCNN often shows better feature extraction ability but with a large model size [49]. In Fig. 8, we show the F1 and AP scores of our method with a different number of filters in

\[
L_{\text{cls}} = \frac{1}{N_{\text{pos}}} \sum_{k=1}^{4} \sum_{i,j} \sum_{\nu \in \{t_u, t_h\}} \begin{cases} 
(1 - \tilde{y}_{i,j})^{\nu} \log(\tilde{y}_{i,j}), \text{if } \tilde{y}_{i,j} = 1 \\
(\tilde{y}_{i,j})^{\nu} \log(1 - \tilde{y}_{i,j}), \text{otherwise}
\end{cases} \tag{4}
\]
TABLE I

COMPARISON WITH DIFFERENT LIGHTWEIGHT BACKBONE NETWORKS WITH THE SAME FPN AND MLAS MODULES

| Backbone network | Model size of backbone network (M) | FLOPs (G) | AP   |
|------------------|-----------------------------------|----------|------|
| Light VGG        | 0.47                              | 2.30     | 0.8682 |
| Light Resnet     | 0.18                              | 0.44     | 0.8251 |
| Light MobileNetV2| 0.19                              | 1.45     | 0.8501 |
| Light Darknet    | 0.16                              | 1.66     | 0.8510 |
| Light EfficientNet| 0.19                             | 0.59     | 0.8136 |
| Proposed method  | 0.19                              | 0.70     | 0.8529 |

Fig. 8. F1 and AP scores versus the model size of our backbone network, where the number of iterations is 100. The number of filters in Convs of the baseline backbone network (1) is $N_{\text{filters}} = [8, 8, 16, 32, 48, 64]$, as presented in Fig. 2. The values of 1/4, 1/2, 2, and 3 are multiplied on $N_{\text{filters}}$ to generate four backbone networks.

TABLE II

COMPARISON OF DIFFERENT COMBINATIONS OF FEATURE FUSION MODULES WITH 200 ITERS IN TRAINING PROCESS

| Modules       | Model size (M) | P        | R        | AP    |
|---------------|----------------|----------|----------|-------|
| FPN1          | 0.55           | 0.9461   | 0.7579   | 0.8551 |
| FPN1+ASFF     | 0.62           | 0.9287   | 0.7335   | 0.8468 |
| FPN1+FPN2     | 0.62           | 0.9292   | 0.7879   | 0.8731 |
| FPN1+ASFF+FPN2| 0.68           | 0.9448   | 0.8030   | 0.8798 |

TABLE III

PERFORMANCE METRICS OF DIFFERENT MLAS-BASED STRATEGIES AFTER 300 ITERS IN TRAINING PROCESS

| Strategy        | Model size | FLOPs | P   | R   | F1   | AP   |
|-----------------|------------|-------|-----|-----|------|------|
| MLAS-{1}        | 0.64M      | 4.10G | 0.9388 | 0.7410 | 0.823 | 0.8825 |
| MLAS-{2,1}      | 0.65M      | 4.18G | 0.9508 | 0.7760 | 0.8546 | 0.9030 |
| MLAS-{3,2,1}    | 0.66M      | 4.23G | 0.9477 | 0.7786 | 0.8549 | 0.9060 |
| MLAS-{4,3,2,1}  | 0.68M      | 4.87G | 0.9506 | 0.7921 | 0.8641 | 0.9112 |

The combination of FPN1, ASFF, and FPN2 achieves the best precision, recall and AP values (as the bold values in Table II) with a limited increase in model size. Fig. 9 also illustrates the representative visualization results to demonstrate the effectiveness of FPN containing FPN1, ASFF, and FPN2.

In Table III, the list after “MLAS” stands for the number of levels of GT target templates used in our MLAS-based strategy. As more levels of auxiliary supervision are used, the performance of detection gradually improves (as bold values in Table III) and the AP value reaches the maximum with MLAS-{4, 3, 2, 1}. This demonstrates the effectiveness of our proposed MLAS-based strategy.

C. Performance Comparisons With Other State-of-the-Art Detectors

In this subsection, experiments are conducted to compare the proposed detector with other state-of-the-art detectors. Three versions of our method with input sizes of $512 \times 512$, $320 \times 320$, and $256 \times 256$ are implemented considering the variety of the input sizes of existing detectors. Since the source codes of the SAR ship detectors are often not public, we use the performance index values of the original papers under the same setting of datasets to make fair comparisons. The detection results of existing detectors and our detector are listed in Table IV. Some of the items in Table IV are illustrated in Fig. 1 for a quick look. As shown in Table IV, the proposed method using a tiny structure with a model size less than 1 million achieves competitive performance in terms of AP, while other methods use weights at least dozens of times than our detector. It should be noted that Mao et al. [48] also propose an efficient detector that reaches 94% in precision with 0.93 M parameters. By contrast, our method uses 0.68 M parameters to achieve competitive 95.35% precision. The FPN Faster R-CNN obtains top values in terms of R and F1, while its computational complexity is higher than other methods. As the bold values shown in Table IV, the proposed method only averagely uses 18 ms (56 FPS) to conduct the detection of each image in SSDD with the $256 \times 256$ input size using the smallest model size and FLOPs, which is the faster one among existing commonly used methods. In order to verify the robustness of the proposed method, Fig. 10 shows detection results of ship targets in different imaging conditions of marine
Fig. 9. Selected detection results on SSDD. The bounding boxes with green lines are GT annotations and the prediction results of our methods are drawn with yellow boxes. The original GT annotations [31] of the last image have been revised following [68]. (a) Ground-Truth. (b) FPN1. (c) FPN1+FPN2. (d) FPN1+ASFF. (e) FPN1+ASFF+FPN2. The combination of FPN1+ASFF+FPN2 provides better detection performance than other combinations.

| Detector | Backbone | Input scale | Model size(M) | FLOPs (G) | FPS | Inferring time (ms) | P   | R   | F1   | AP   |
|----------|----------|-------------|---------------|-----------|-----|---------------------|-----|-----|------|------|
| FPN Faster R-CNN [40] | VGG16 | 512×512 | 33.14 | 204.42 | 11 | 91 | 0.8938 | 0.9250 | 0.9095 | 0.8872 |
| SSD300 [22] | VGG16 | 300×300 | 23.75 | 62.93 | 29 | 34 | 0.8409 | 0.6184 | 0.7463 | --- |
| SSD512 [22] | VGG16 | 512×512 | 24.39 | 178.77 | 19 | 53 | 0.9099 | 0.7580 | 0.8270 | 0.8339 |
| YOLOv3 [59] | Darknet | 256×256 | 61.58 | 24.76 | 28 | 36 | 0.8384 | 0.8180 | 0.8281 | 0.8052 |
| CenterNet (1) [26] | ResNet18 | 512×512 | 12.07 | 22.02 | 30 | 33 | 0.9698 | 0.7279 | 0.8316 | 0.9043 |
| CenterNet (2) [26] | DLA34 | 512×512 | 29.18 | 18.45 | 18 | 56 | 0.9548 | 0.8508 | 0.8998 | 0.9192 |
| RetinaNet [24] | VGG16 | 256×256 | 22.81 | 99.30 | 17 | 59 | 0.9545 | 0.6697 | 0.7871 | 0.7657 |
| Li [31] | VGG16 | 256×256 | --- | --- | --- | --- | --- | --- | --- | 0.7880 |
| Jiao [52] | ResNet101 | 512×512 | 75.00 | --- | --- | --- | 0.9670 | 0.8340 | 0.8960 | --- |
| Zhang [44] | SSD | 300×300 | --- | --- | --- | --- | --- | --- | --- | --- |
| Cui [34] | ResNet101 | --- | --- | --- | --- | --- | --- | --- | --- | 0.8012 |
| Han [47] | PCBMSK | 320×320 | 18.19 | --- | --- | --- | --- | --- | --- | 0.9057 |
| Mao [48] | Simplifed U-Net | 512×512 | 0.93 | --- | --- | --- | 0.9400 | --- | --- | --- |
| Li [43] | Hyper-Net [67] | --- | 19.11 | --- | --- | --- | --- | --- | --- | 0.8980 |
| Proposed method | Lightweight backbone | 256×256 | 0.68 | 1.21 | 56 | 18 | 0.9391 | 0.7985 | 0.8633 | 0.8941 |
| Proposed method | Lightweight backbone | 320×320 | 0.68 | 1.61 | 50 | 20 | 0.9535 | 0.8062 | 0.8718 | 0.9054 |
| Proposed method | Lightweight backbone | 512×512 | 0.68 | 4.78 | 40 | 25 | 0.9477 | 0.7785 | 0.8529 | 0.9112 |

*---* denotes default values that are not reported in existing literatures.

SAR images. In Fig. 10, our proposed method performs better or similar than existing competitors [59], [26].

In Fig. 11, we calculate the AP values with different balance parameters $\lambda$ in (6). The first peak of AP is obtained when $\lambda = 10$. When $5 \leq \lambda \leq 90$, the standard deviation of AP is only 0.017, that is a small value to show the robustness of our method.

IV. CONCLUSION

This article proposed an efficient center-based ship detector for multiscale ship detection in SAR images. First, an extremely lightweight backbone network is designed to reduce the model size of DCNN and improve the detection speed. Second, feature fusion modules are applied in a cascade way to aggregate feature maps from different levels and exploit rich semantic information in marine SAR images. Then, the proposed MLAS framework is used to perform the multilevel training and guide our network to focus on the target areas in the different spatial levels. Experiments show that our new detector is competitive with the state-of-the-art methods in terms of detection performance but requires significantly fewer parameters and FLOPs. In the future, the fast detection of ship targets in SAR images with fewer omissions of extremely small ships will be investigated.
Fig. 10. Detection results in different imaging conditions of SAR ships. (a) Ground-truth. (b) YOLOv3 [59]. (c) CenterNet with resNet18 [26]. (d) Our method. Rows 1–4: multiple targets, inshore targets, densely distributed targets, and strong sea clutter.

Fig. 11. AP curve with different values of balance parameter $\lambda$ in (6), where the input size of images is $256 \times 256$.

REFERENCES

[1] X. Wang, G. Li, X.-P. Zhang, and Y. He, “Ship detection in SAR images via local contrast of fisher vectors,” IEEE Trans. Geosci. Remote Sens., vol. 58, no. 9, pp. 6467–6479, Sep. 2020, doi: 10.1109/TGRS.2020.2976880.

[2] X. Wang, G. Li, A. Plaza, and Y. He, “Ship detection in SAR images via enhanced nonnegative sparse locality-representation of fisher vectors,” IEEE Trans. Geosci. Remote Sens., vol. 59, no. 11, pp. 9424–9438, Nov. 2021.

[3] M. N. Ferrara and A. Torre, “Automatic moving targets detection using a rule-based system: Comparison between different study cases,” in Proc. IEEE Int. Geosci. Remote Sens. Symp. Proc., 1998, pp. 1593–1595.

[4] C. C. Wackerman, K. S. Friedman, W. G. Pichel, P. Clemente-Colón, and X. Li, “Automatic detection of ships in RADSARAT-1 SAR imagery,” Can. J. Remote Sens., vol. 27, no. 5, pp. 568–577, 2001.

[5] Y. He, J. Guan, Y. Peng, and D. Lu, “A new CFAR detector based on ordered statistics and cell averaging,” in Proc. Int. Radar Conf., 1996, pp. 106–108, doi: 10.1109/ICR.1996.573783.

[6] G. Rongbing and W. Jianguo, “Distribution-based CFAR detectors in SAR images,” J. Syst. Eng. Electron., vol. 17, no. 4, pp. 717–721, 2006, doi: 10.1016/S1004-4132(07)60004-8.

[7] H. M. Finn and R. S. Johnson, “Adaptive detection mode with threshold control as a function of spatial sampled clutter level estimates,” RCA Rev., vol. 29, no. 1, pp. 414–464, 1968.

[8] M. di Bisceglie and C. Galdi, “CFAR detection of extended objects in high-resolution SAR images,” IEEE Trans. Geosci. Remote Sens., vol. 43, no. 4, pp. 833–843, Apr. 2005.

[9] X. Wang, G. Li, X.-P. Zhang, and Y. He, “A fast CFAR algorithm based on density-censoring operation for ship detection in SAR images,” IEEE Signal Process. Lett., vol. 28, pp. 1085–1089, 2021, doi: 10.1109/LSP.2021.3082034.

[10] X. Wang, G. Li, A. Plaza, and Y. He, “Revisiting SLIC: Fast superpixelsegmentation of marine SAR images using density features,” IEEE Trans. Geosci. Remote Sens., vol. 60, 2022, Art. no. 5221818.

[11] K. Eldhuset, “An automatic ship and ship wake detection system for spaceborne SAR images in coastal regions,” IEEE Trans. Geosci. Remote Sens., vol. 34, no. 4, pp. 1010–1019, Jul. 1996.

[12] Q. Jiang et al., “Ship detection in RADSARAT SAR imagery,” in Proc. IEEE Int. Conf. Syst., Man, Cybern., 1998, vol. 5, pp. 4562–4566.

[13] L. Du, Y. Wang, X. Xie, Z. Wang, and J. Chen, “A semisupervised infinite latent dirichlet allocation model for target discrimination in SAR images with complex scenes,” IEEE Trans. Geosci. Remote Sens., vol. 58, no. 1, pp. 666–679, Jan. 2020, doi: 10.1109/TGRS.2019.2939001.

[14] L. Du, H. Dai, Y. Wang, X. Xie, and Z. Wang. “Target discrimination based on weakly supervised learning for high-resolution SAR images in complex scenes,” IEEE Trans. Geosci. Remote Sens., vol. 58, no. 1, pp. 461–472, Jan. 2020.

[15] M. D. Li, X. C. Cui, and S. W. Chen. “Adaptive superpixel-level CFAR detector for SAR inshore dense ship detection,” IEEE Geosci. Remote Sens. Lett., vol. 19, 2022, Art. no. 4010405, doi: 10.1109/LGRS.2021.3059253.

[16] J. Liu, Q. Wang, J. Cheng, D. Xiang, and W. Jing, “Multitask learning-based for SAR image superpixel generation,” Remote Sens., vol. 14, no. 4, pp. 899, 2022.

[17] L. Zhang, S. Lu, C. Hu, D. Xiang, T. Liu, and Y. Su, “Superpixel generation for SAR imagery based on fast DBSCAN clustering with edge penalty,” IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens., vol. 15, pp. 804–819, 2022.

[18] R. Girshick, J. Donahue, T. Darrell, and J. Malik, “Rich feature hierarchies for accurate object detection and semantic segmentation,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2014, pp. 580–587, doi: 10.1109/CVPR.2014.81.

[19] R. Girshick, “Fast R-CNN,” in Proc. IEEE Int. Conf. Comput. Vis., 2015, pp. 1440–1448, doi: 10.1109/ICCV.2015.169.

[20] K. He, G. Gkioxari, P. Dollár, and R. Girshick, “Mask R-CNN,” in Proc. IEEE Int. Conf. Comput. Vis., 2017, pp. 2980–2988, doi: 10.1109/ICCV.2017.322.

[21] 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, 2017, doi: 10.1109/TPAMI.2016.2577031.
Xueqian Wang (Member, IEEE) received the B.S. from the University of Electronic Science and Technology of China, Chengdu, China, in 2015, and the Ph.D. degrees in information and communication engineering from the Tsinghua University, Beijing, China, in 2020.

From 2018 to 2019, he visited Syracuse University, Syracuse, NY, USA. From 2020 to 2022, he was a Post-Doctoral Fellow with the Department of Electronic Engineering, Tsinghua University, Beijing, China. He is currently an Assistant Professor with the Department of Electronic Engineering, Tsinghua University. His main research interests include remote sensing, target detection, information fusion, radar imaging, and compressed sensing.

Zhizhuo Jiang received the B.S. degree in electronics and information engineering and the M.S. degree in information and communication engineering from the Harbin Institute of Technology, Harbin, China, in 2014 and 2016, respectively, and the Ph.D. degree in information and communication engineering from Tsinghua University, Beijing, China.

He is currently a Post-Doctoral Fellow with the Tsinghua Shenzhen International Graduate School, Shenzhen, China. His main research interests include target detection, multimodal data fusion, deep learning, and signal processing.

Gang Li (Senior Member, IEEE) received the B.S. and Ph.D. degrees in electronic engineering from the Tsinghua University, Beijing, China, in 2002 and 2007, respectively.

Since July 2007, he has been a Faculty of Tsinghua University, where he is currently a Professor with the Department of Electronic Engineering. From 2012 to 2014, he visited Ohio State University, Columbus, OH, USA, and Syracuse University, Syracuse, NY, USA. He has authored or coauthored more than 200 journal and conference papers. He has authored the book Advanced Sparsity-Driven Models and Methods for Radar Applications (London, U.K.: SciTech/IET Publishing, 2020) and edited the book Radar Countermeasures for Unmanned Aerial Vehicles (London, U.K.: SciTech/IET Publishing, 2021). His research interests include radar signal processing, distributed signal processing, remote sensing, and information fusion.

Prof. Li serves as a Senior Area Editor for IEEE SIGNAL PROCESSING LETTERS and an Associate Editor for IEEE JOURNAL OF SELECTED TOPICS IN APPLIED EARTH OBSERVATIONS AND REMOTE SENSING. He served as an Associate Editor for the IEEE TRANSACTIONS ON SIGNAL PROCESSING, the Special Sessions Chair, for The International Conference on Acoustics, Speech, & Signal Processing (ICASSP) 2021, and the Tutorial Speaker for The IEEE Radar Conference 2017.

You He received the Ph.D. degree in electronic engineering from the Tsinghua University, Beijing, China, in 1997.

He is currently a Professor with the Naval Aeronautical University. He is also cross-appointed to the Department of Electronic Engineering, Tsinghua University. He has published over 300 academic articles. He is also the author of Radar Target Detection and CFAR Processing (Tsinghua University Press), Multi-Sensor Information Fusion With Applications and Radar Data Processing With Applications (Publishing House of Electronics Industry). His current research interests include detection and estimation theory, CFAR processing, distributed detection theory, and multi-sensor information fusion.

Dr. He is also a Fellow Member of the Chinese Academy of Engineering.