SAR Ship Instance Segmentation With Dynamic Key Points Information Enhancement

Fei Gao, Xu Han, Jun Wang, Jinping Sun, *Member, IEEE*, Amir Hussain, and Huiyu Zhou

***Abstract*—There are several unresolved issues in the field of ship instance segmentation in synthetic aperture radar (SAR) images. First, in inshore dense ship area, the problems of missed detections and mask overlap frequently occur. Second, in inshore scenes, false alarms occur due to strong clutter interference. In order to address these issues, we propose a novel ship instance segmentation network based on dynamic key points information enhancement. In the detection branch of the network, a dynamic key points module is designed to incorporate the target's geometric information into the parameters of the dynamic mask head using an implicit encoding technique. In addition, we introduce a dynamic key points encoding branch, which encodes the target's strong scattering regions as dynamic key points. It strengthens the network's ability to learn the correspondence between local regions with strong scattering and overall ship targets, effectively mitigating mask overlap issues. Moreover, it enhances the discriminative ability of network between ship targets and clutter interference, leading to a reduction in false alarm rates. To further enhance the dynamic key points information, an instancewise attention map module is designed, which decodes the key points during the mask prediction period, generating instancewise attention maps based on 2-D Gaussian distribution. This module further enhances the sensibility of network to specific instances. Simulation experiments conducted on the Polygon Segmentation SAR Ship Detection Dataset and High-Resolution SAR Images Dataset demonstrate the superiority of our proposed method over other state-of-the-art methods in inshore and offshore scenes.**

***Index Terms*—Implicit encoding, key points detection, ship instance segmentation, synthetic aperture radar (SAR).**

I. Introduction

DUE to the excellent penetration capability, synthetic aperture radar (SAR) produces high-resolution images in all weather and all time. The SAR technology has witnessed rapid development and has been widely applied in marine observation tasks [1], including fisheries monitoring and maritime vessel management [2], [3]. The key to SAR image ship detection lies in obtaining the ship's location. Horizontal bounding box (HBB) detection represents the target's position using a horizontal rectangular box. However, for ship targets with a large aspect ratio, the HBB often covers a significant amount of background clutter. Moreover, in inshore dense ship area, the HBB of different ships overlap with each other, which is the disadvantage of subsequent interpretation. To overcome the limitations of HBB detection, researchers have proposed oriented bounding box (OBB) detection to obtain the OBB results that compactly enclose each object [4], [5]. The OBB not only reduces the background ratio within the bounding box but also provides target's heading information, which is crucial for advanced tasks, such as trajectory prediction [6]. Since the ship is still selected by rectangular boxes, the shape information of the target is missing. Instance segmentation, on the other hand, goes a step further by assigning category labels pixel by pixel, obtaining pixelwise masks that contain information about the target's category, location, and contour. However, as a more sophisticated detection method for ship targets, instance segmentation faces significant challenges in achieving high precision, especially in inshore scenes [7].

Currently, researchers have proposed various instance segmentation architectures that achieve outstanding performance in natural scenes, such as Mask R-CNN [8], Cascade R-CNN [9], Hybrid Task Cascade (HTC) [10], and InstaBoost [11]. However, the complex contours of targets in inshore scenes pose significant challenges to the performance of the network. Some researchers have conducted extensive research on contour-based detection methods, and proposed methods that encode the mask contour into a set of concrete encodings, called explicit encoding [12], [13], [14], [15], [16]. In these methods based on explicit encoding [12], [13], [14], [15], [16], the network structures are often complex and require careful design of encoding and decoding methods.

With the deepening research on contour-based instance segmentation methods, some researchers have discovered the tremendous potential of using implicit encoding [17], [18], [19], [20]. The earliest research can be traced back to the "You Only Look At CoefficienTs" (Yolact) method proposed by Bolya et al. [17]. Inspired by Yolact, researchers have done some research based on implicit encoding [18], [19], [20]. Among these methods, CondInst [19] and SOLOv2 [20] have achieved outstanding accuracy. Unlike previous instance segmentation

Manuscript received 12 January 2024; revised 3 March 2024; accepted 20 March 2024. Date of publication 16 April 2024; date of current version 19 June 2024. This work was supported in part by the National Natural Science Foundation of China under Grant 62371022 and in part by UK Engineering and Physical Sciences Research Council (EPSRC) under Grant EP/M026981/1, Grant EP/T021063/1, and Grant EP/T024917/1. *(Corresponding authors: Jun Wang; Jinping Sun.)*

Fei Gao is with the School of Electronic and Information Engineering, Beihang University, Beijing 100191, China, and also with the Beihang Hangzhou Innovation Institute Yuhang, Xixi Octagon City, Hangzhou 310023, China (e-mail: feigao2000@163.com).

Xu Han, Jun Wang, and Jinping Sun are with the School of Electronic and Information Engineering, Beihang University, Beijing 100191, China (e-mail: hanxu2017@buaa.edu.cn; wangj203@buaa.edu.cn; sunjinping@buaa.edu.cn).

Amir Hussain is with the Cyber and Big Data Research Laboratory, Edinburgh Napier University, EH11 4BN Edinburgh, U.K. (e-mail: a.hussain@napier.ac.uk).

Huiyu Zhou is with the Cognitive Big Data and Cybersecurity (CogBiD) Research Lab, Edinburgh Napier University, EH10 Edinburgh, UK (e-mail: hz143@leicester.ac.uk).

Digital Object Identifier 10.1109/JSTARS.2024.3383779

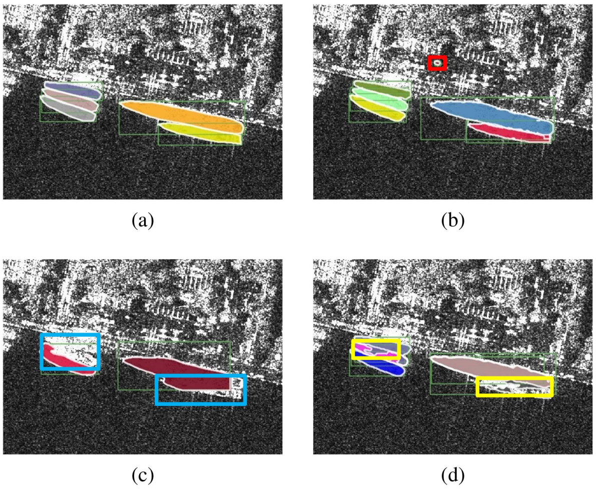


Fig. 1. Issues in the results of instance segmentation. (a) Ground truth. (b) False alarms (red rectangles). (c) Missed detections (blue rectangles). (d) Mask overlap (yellow rectangles). The green rectangles represent results of HBB detection.

methods, neither of them reduces the dimensionality of mask features, which means they use massive parameters to represent the masks. Considering that instance masks often have complex contours, this choice may be the key to their success. Nevertheless, CondInst and SOLOv2 are designed for optical images, and when it comes to SAR images, the presence of interference, such as speckle noise, can affect the features extracted by the network [21], [22], [23], [24], [25], [26], leading to a decrease in segmentation accuracy. It is particularly evident in inshore scenes, as shown in Fig. 1, where ship targets densely distribute, making it prone to mask overlap issues. Moreover, influenced by land clutter and interference between ships, false alarms and missed detections are likely to occur.

To address the limitations of the above methods, this article proposes a ship instance segmentation method for SAR images based on key points information enhancement. The method contains two main modules: the dynamic key points module (DKPM) and the instancewise attention map module (IAMM). The DKPM encodes the strong scattering region of each target, particularly the ship's bow, into a set of dynamic key points. Through implicit encoding, the key points information, including the target's size, shape, and strong scattering region, is embedded into the parameters of the dynamic mask head. This refinement of features improves the network's performance in inshore and offshore scenes. The IAMM module further enhances key points information. Specifically, it decodes dynamic key points information and generates an instancewise attention map based on a 2-D Gaussian distribution, thereby increasing the network's sensitivity to specific targets. Specifically, considering the prior knowledge that ships have large aspect ratios, we employ binary coded label to encode the angles of ship targets firstly [27]. Furthermore, by exploiting the prominent scattering characteristics exhibited by the bow region of ship targets due to its distinctive structural features, we adopt a dynamic sampling approach with nonuniform key points to accurately predict the bow contour. By combining angle prediction with nonuniform key points sampling, we encode the strong scattering region of ship targets, enhancing the correspondence between the target center and the strong scattering region. Harnessing the powerful feature representation potential of implicit encoding, we incorporate the prediction of angle and key points containing the information of strong scattering region into the parameter generation branch of the dynamic mask head. This enhances the sensitivity of the dynamic mask head to the target contour. In the mask branch, we augment the CondInst with a 2-DGaussian distribution heatmap [28]. We take this heatmap as an instancewise attention map, which includes target angle, width, key points information, etc. This further strengthens the information of the strong scattering region. The 2-D Gaussian distribution heatmap is obtained by decoding the prediction of angle and key points encoding in the detection branch. By enhancing the dynamic key points information twice in the detection branch and mask branch, the segmentation accuracy is improved in scenes with dense ship and strong clutter interference. A series of experiments are conducted on the Polygon Segmentation SAR Ship Detection Dataset (PSeg-SSDD) and High-Resolution SAR Images Dataset (HRSID) [29], [30], [31] to validate the effectiveness of the proposed DKPM and IAMM based on the 2-D Gaussian distribution heatmap. Comparative experiments with other typical instance segmentation methods demonstrate the superior segmentation performance of our proposed method.

The main contributions of this article are summarized as follows.

1) *DKPM:* It is proposed to encode the strong scattering region of the ship. Through incorporating fine-grained features into the implicit encoding process of the dynamic mask head parameters, the module enhances the performance of the network in both inshore and offshore scenes.
2) *IAMM:* It is proposed to generate attention map based on 2-D Gaussian distribution, which further enhances the perception of the network for specific instances. The module further alleviates the issues of mask overlap and false alarms.
3) Experiments are conducted on the PSeg-SSDD and HRSID. Comparative results with state-of-the-art (SOTA) instance segmentation methods show that our proposed method outperforms comparative methods.

The rest of this article is organized as follows. Section II provides a brief review of related works on instance segmentation and key points estimation. Section III presents a detailed description of our proposed method. Section IV provides a detailed description of the experimental setup and presents the results of our method on the PSeg SSDD and HRSID datasets. Section V discusses the ablation studies and parameter settings.

II. Related Work

A. Instance Segmentation

Mask R-CNN [8] is a representative method in deep learning-based instance segmentation. It extends the classical two-stage detection method, Faster R-CNN [32], by adding a parallel mask branch alongside the detection branch. It proposes the use of region of interest (RoI) pooling instead of RoI align, achieving good segmentation results with reduced computational cost. Cascade R-CNN [9], on the other hand, improves the overall network performance by designing a cascaded network and using different intersect over union (IoU) thresholds in different subnetworks to gradually enhance the performance. Subsequently, Chen et al. [10] proposed the HTC, which generates direct spatial context information through a fully convolutional branch, combining the refining operations for different subnetworks into a multistage processing task. InstaBoost [11] introduced a mechanism for local information fusion, which reinforces the network's attention to the target by performing enhancement on multiple regions of the image and fusing the enhanced local images. This approach ultimately improves the segmentation accuracy. Huang et al. [33] recognized the limitations of using classification confidence as the mask quality criterion and introduced Mask Scoring R-CNN. They proposed a module to learn the quality of instance mask predictions and designed a mask scoring strategy that allows the predicted masks to approach the ground truth in terms of quality scores. Consistent and significant improvements were achieved across different models. Cheng et al. [34] proposed a class-specific attention encoding (CAE) module to enforce the convolutional neural networks (CNNs) to explicitly encode class attentions. The CAE module can be conveniently embedded into current CNNs to build an end-to-end CANet to extract highly category-related feature representations.

Some researchers have noticed that adding attention mechanism to instance segmentation networks can improve the network performance [35], [36], [37], [38]. Yang et al. [35] noticed the limitations of the HBB, which often include redundant backgrounds and even docks or other ships with significant scattering interference. Therefore, they proposed an instance segmentation network based on the OBB detection, called SRNet. Similar to attention mechanism, SRNet can focus more on specific instances while reducing interference from surrounding sea, land, and other ships. This further improves the detection accuracy. Ke et al. [36] addressed the issue of limited bounding box localization capability affecting the instance segmentation accuracy of networks. They proposed a global context boundary-aware network that utilizes a global context information modeling block to increase the network's receptive field and a boundary-aware box prediction block for better cross-scale bounding box predictions. Zhang and Zhang [37] identified two limitations of the region of interest extractor (RoIE) in SAR ship instance segmentation methods, namely, single-level and noncontext extraction. They proposed a full-level context squeeze-and-excitation RoIE that extracts contextual information on feature maps at all scales of the feature pyramid network (FPN) [39]. By highlighting valuable features and suppressing irrelevant features, the segmentation accuracy of the network is improved. To further enhance the performance of SAR ship instance segmentation models, especially for small objects, Zhang and Zhang [38] subsequently proposed a mask attention interaction and scale enhancement (MAI-SE) network. MAI utilizes asymmetric spatial pyramid pooling to obtain multiresolution feature responses, while SE employs the content-aware reassembly of features block to generate additional pyramid levels at the bottom to improve performance on small ships. Some studies [36], [37], [38] demonstrate the positive impact of combining attention mechanism with bottom feature maps in improving network segmentation accuracy, particularly for small objects. When dealing with small objects, their small size often makes them easily occluded by the surrounding background, resulting in inaccurate results. However, the introduction of attention mechanism enables the network to automatically learn and focus on the crucial features for object segmentation. In addition, the high resolution of the bottom feature maps maximizes the effectiveness of the attention mechanism, leading to more precise and reliable segmentation results.

In the above studies, the attention mechanism is often introduced by processing the context information. For example, in the spatial dimension, the context information is extracted from output feature maps for each scale of FPN. Correspondingly in the channel dimension, the weight of each channel is learned adaptively. This approach introduces two problems. First, it introduces new convolutional layers, which require additional training time and computational resources. Second, due to the use of attention mechanism globally, it is still susceptible to interference from strong scattering areas on land and densely distributed ships.

B. Key Points Detection

In inshore scenes, the clutter from land region, such as ports and docks, introduces strong interference, and the inherent speckle noise in SAR images poses a challenge to the accuracy of instance segmentation methods. Therefore, in inshore scenes, SAR images detection has always been a challenging task. Some researchers noticed the particular scattering characteristics of ship targets in SAR images and proposed detection methods based on key points estimation. For example, Ma et al. [40] proposed a SAR ship detection method based on key points estimation and attention mechanism. They optimized the selection process of target centers in dense distributions using a DKPM and utilized attention mechanism to improve the network's ability to extract target information, suppressing noise and achieving high precision in horizontal box prediction. Sun et al. [41] introduced a SAR image OBB detection method based on strong scattering points. Specifically designed for large-scale SAR images, they extracted strong scattering points from SAR images and performed regression on the OBB, achieving SOTA performance on large-scale SAR image datasets. Yi et al. [42] proposed BBAVector, which used the midpoints of the four edges of the OBB as key points. By predicting vectors from the center of the OBB to the midpoints of the four edges, they decoded the OBB results. However, BBAVector neglected the scattering characteristics of ship targets in SAR images. Inspired by BBAVector and considering the scattering characteristics of SAR images, He et al. [43] introduced a polar coordinate encoding method. They increased the sampling number of key points and extended key points from edge midpoints to the entire OBB. This approach represented the OBB using a set of vectors pointing from the ship target's center to the boundary points, called polar coordinate encoding. By using polar coordinate encoding instead of traditional width and height regression in training and inferring processes, the network enhanced its perception of ship contours

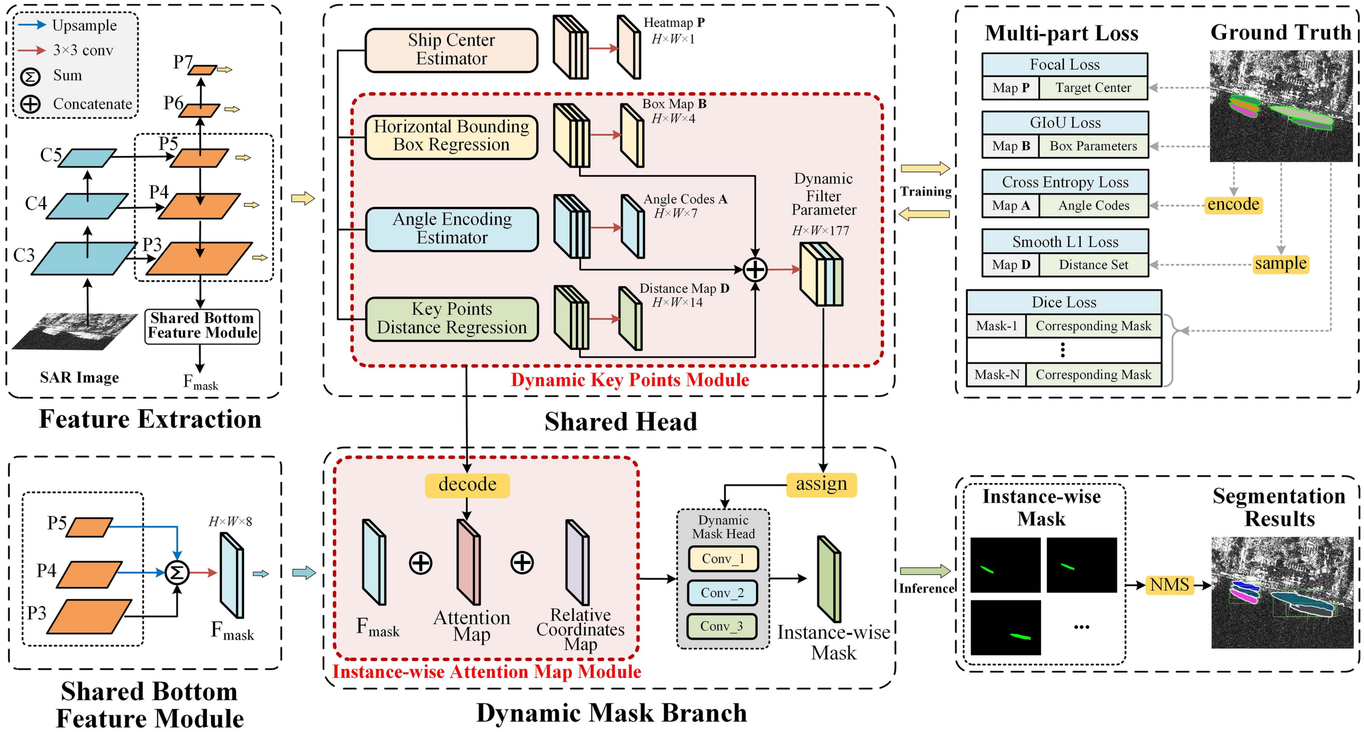


Fig. 2. Overall architecture of our method. The network consists of four main parts: feature extraction network, shared bottom feature module, shared detection head enhanced with key points information, and dynamic mask branch. The four branches of shared head output: the center heatmap **P**, the box map **B**, the angle codes **A**, and the distance map **D**.

and improved detection performance. On the basis of polar coordinate encoding, Gao et al. [44] noticed that the bow region of ship targets often exhibits strong scattering characteristics. They redesigned the key points sampling method based on this characteristic. Instead of sampling key points on the OBB, such as He et al.'s method [43], they obtained the inscribed ellipse through the OBB annotation and performed key points sampling on the ellipse. Compared with key points sampling on the OBB, ellipse sampling better conforms to the contour of ship targets, resulting in superior detection performance. Ge et al. [45] proposed a detection method called KeyShip based on anchor-free key points. They modeled ships as a combination of three kinds of key points: the center of the shorter sides, the center of the longer sides, and the target center of the OBB. By detecting these key points separately and clustering them based on predicted shape descriptors, they constructed the final OBB result. This method implicitly learns the shape information of the target through key points information and achieves high-performance of the OBB detection. Zhang et al. [46] conducted an in-depth analysis of the scattering characteristics of targets in SAR images and designed a method called scattering-point-guided oriented ship detection, which achieved higher precision in rotation box detection. To address the interference of background and noise in SAR images, they designed a scattering-point-guided region proposal network (SPG-RPN) based on the scattering characteristics of SAR. The SPG-RPN predicted potential key scattering points and improved its focus on the vicinity of these key scattering points during the regression and classification stages. In addition, they introduced contrastive learning to alleviate minor differences between target categories.

Compared with the methods of sampling key points on the four sides of the OBB, the method proposed by Gao et al. [44] has better performance in the OBB detection because it makes use of the prior knowledge that the ship's bow exhibits strong scattering characteristics. However, the bow region of ship targets is not strictly an ellipse, and encoding key points distributed on the inscribed ellipse as ground truth during key points sampling is equivalent to introducing lower quality annotations during network training. The mismatch between the encoding results and ground truth can ultimately lead to a decrease in accuracy. In addition, most methods based on key point detection are used for HBB detection and OBB detection, while there are few research works on the application of key points detection to instance segmentation.

III. Methodology

In this section, the DKPM and IAMM are developed, and details of the implementation procedures are also presented. First, we provide a brief overview of the network architecture. After that, every module of the proposed method is described in detail to show how it works. Finally, the loss function of the proposed method is given.

A. Overview

The overall architecture of our approach is shown in Fig. 2, which consists of four components: feature extraction, shared bottom feature module, shared head, and dynamic mask branch. Next, we will provide a detailed introduction to the four components.

1) *Feature Extraction:* In feature extraction, a ResNet50 [47] network and an FPN are used to extract features from an input SAR image.
2) *Shared Bottom Feature Module:* The shared bottom feature module follows the configuration in [19], which is used to generate high-quality and high-resolution basic mask feature map F_{mask}.
3) *Shared Head:* The shared head processes the feature maps at different scales and consists of four subbranches: ship center estimator branch, HBB regression branch, angle encoding estimator branch, and key points distance regression branch. The inputs of the four branches are the multiscale feature maps from the FPN. Each branch consists of four cascaded 3 × 3 convolutional layers. The ship center estimator and HBB branch, respectively, produce the center heatmap $\mathbf{P} \in \mathbb{R}^{\tilde{H}\times\tilde{W}\times 1}$ and the HBB parameter map $\mathbf{B} \in \mathbb{R}^{\tilde{H}\times\tilde{W}\times 4}$, where HBB results $\mathbf{B}$ are solely used for the NMS process. The structures of these two branches are similar to the anchor-free detection method FCOS, which saves the number of parameters and the amount of computation through the anchor-free design [48]. The angle encoding estimator and key points distance regression branch output angle codes $\mathbf{A} \in \mathbb{R}^{\tilde{H}\times\tilde{W}\times M}$ and distance map $\mathbf{D} \in \mathbb{R}^{\tilde{H}\times\tilde{W}\times N}$ separately, where M denotes the number of bits for angle codes, and N denotes the number of sample points. For the ship center estimator branch, focal loss is employed for supervised training; cross-entropy loss is utilized in the angle encoding estimator to compute the loss between angle codes and ground truth; the other two branches employ smooth L1 loss for loss calculation. We combined these four parts of losses to form the overall loss function guiding the network regression. In the shared head, the HBB branch, angle encoding estimator, and key points distance regression jointly form the DKPM. In DKPM, we concatenate the intermediate feature maps of the three branches and generate dynamic filter parameters through an implicit encoding method. This implicit encoding method effectively incorporates the information of key points into the instance segmentation network, which will be introduced in detail in the following.
4) *Dynamic Mask Branch:* The dynamic mask branch mainly consists of the IAMM and the dynamic mask head. IAMM takes the basic mask feature map F_{mask}, the attention map, and the relative coordinate map as inputs. F_{mask} is an eight channels feature map obtained from the shared bottom feature module. For each positive sample point obtained from the shared head, the relative coordinate feature map is generated by calculating the difference between the coordinates of each point on the feature map and the coordinates of the positive sample point. The predictions from DKPM are then decoded to obtain a 2-D Gaussian distribution map that contains information about the target's angle, size, and key points distribution as an instancewise attention map [28]. The F_{mask}, relative coordinate feature map, and 2-D Gaussian heatmap corresponding to each positive point are concatenated to enhance the angle and key points information of the target, and then fed into the dynamic mask head to obtain the segmentation result.

B. Dynamic Key Points Module

The DKPM can be divided into two parts, dynamic key points encoding and implicit encoding. Both of them are explained in detail as follows.

1) Dynamic Key Points Encoding: The dynamic key points encoding method consists of two parts, namely, dense label encoding (DCL) [27] and symmetric nonuniform key points distances sampling.

In the field of OBB detection, predicting the angle of target has always been a challenge. There are two typical approaches to angle prediction. One is based on regression, which directly calculates the loss between the predicted angle and the ground truth angle to make the prediction converge to the ground truth. However, this approach suffers discontinuity at the boundaries due to the periodicity of angles. The loss function undergoes a sudden change at the angle definition boundaries and causing inconsistency in the regression form between the boundary and nonboundary cases, leading to instability during the training process. The other approach is based on classification, where the angle prediction is transformed into a classification problem to avoid the issue of discontinuity. The methods of angle prediction based on classification are one-hot encoding, circular smooth label (CSL) [49], and DCL. However, the prediction layer of one-hot encoding and CSL is too thick to require lager computing resources and longer training time than DCL [27]. So, we introduce DCL to encode and decode the angle information. For the target angle represented using the long edge representation, the angle can be converted into decimal encoding L_{Decimal} using the following equation:

$$L_{\text{Decimal}} = \left\lfloor \frac{\theta}{\Delta\delta} \right\rfloor, \Delta\delta = \frac{180^\circ}{2^n} \tag{1}$$

where $\Delta\delta$ represents the angular interval between adjacent categories and n is the number of encoding bits. Subsequently, converting the decimal encoding L_{Decimal} to binary encoding L_{Bin} yields the result of dense label angle encoding. In the decoding process, each bit of the n-dimensional angle category prediction result can be rounded to obtain standard binary encoding. Then, converting L_{Bin} to decimal encoding L_{Decimal} and multiplying it by the angular interval $\Delta\delta$ yields the angle of the target.

This method can represent a larger range of values with fewer encoding lengths, effectively mitigating the problem of long encoding lengths in CSL and one-hot encoding methods. It reduces the thickness of the prediction layer and keeps the angle error within an acceptable range. For example, when using a 7-bit dense binary encoding to represent angles, the angle error can be calculated as following:

$$n = \lceil \log_2(\mathrm{AR}/\Delta\delta) \rceil \tag{2}$$

where n represents the encoding length, AR represents the angle range, and in the long-edge representation, the angle label is the angle between the target's major axis direction and the x-axis, with an angle range of $[0, 180^\circ)$. Therefore, AR is set to 180°, and

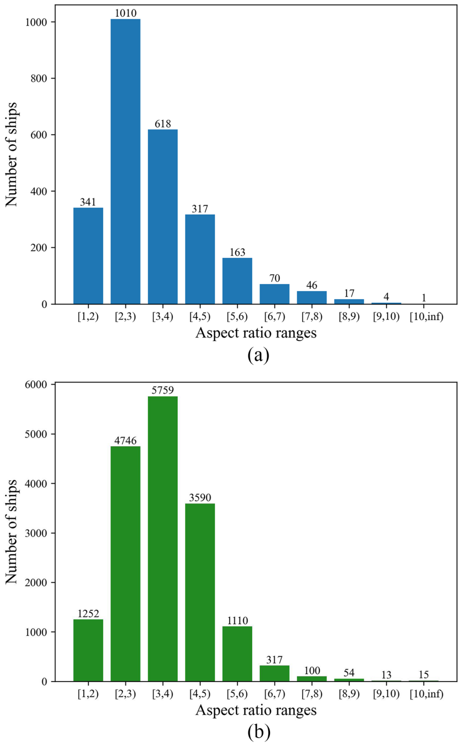


Fig. 3. Aspect ratio distribution of ship targets in different datasets. (a) PSeg-SSDD. (b) HRSID.

Algorithm 1: Dynamic Key Points Encoding Algorithm Based on Symmetric Nonuniform Sampling.

```
Input: The set of discrete points of the target contour
    $P_D = \{p_i' \mid p_i' = (x_i', y_i'), i = 1, 2, \ldots, M\}$, major axis
    direction $\theta_M$, target center $c = (x_c, y_c)$.
Output: Coding vector of the dynamic key points
    $\overrightarrow{\mathrm{V}} = (d_1, d_2, \ldots, d_{14})$.
Connect discrete sets of points $P_D$ in turn to obtain the
    continuous point set of the target contour
    $P = \{p_i \mid p_i = (x_i, y_i), i = 1, 2, \ldots, N\}$;
Calculates the angles of the points in $P$ with respect to
    the target center point $\theta$,
    $\theta = \left\{\theta_i \mid \theta_i = \arctan\left(\frac{y_i - y_c}{x_i - x_c}\right), i = 1, 2, \ldots, N\right\}$
From $\theta_M$ and angle template for symmetric
    non-uniform sampling, obtain the set of sampling
    angles $\theta_s = \{\theta_{s1}, \theta_{s2}, \ldots, \theta_{s14}\}$;
for $i = 1$ to $14$ do
(1) Find the angle $\theta_j$ closest to $\theta_{si}$ in angle set $\theta$;
(2) Calculate the distance code $d_i$ corresponding to $\theta_j$,
    $d_i = \sqrt{(x_j - x_c)^2 + (y_j - y_c)^2}$
return $\overrightarrow{\mathrm{V}} = (d_1, d_2, \ldots, d_{14})$.
```

$\Delta\delta$ represents the difference between adjacent angle categories. Hence, when using a 7-bit dense binary encoding, the angle error is $\Delta\delta/2$, which is 0.7°. The impact on accuracy can be neglected.

The symmetric nonuniform key points distances sampling strategy is inspired by the large aspect ratio of ships, which is the significant geometric feature and prior knowledge. In order to ensure that key points can sample the strong scattering region at the bow of the ship, we analyze the aspect ratio distribution of ship targets in the dataset to determine the sampling range. In Fig. 3, the aspect ratio distribution of ship targets in the PSeg-SSDD and HRSID is shown in Fig. 3(a) and (b), separately.

Based on the statistical results shown in Fig. 3, it can be observed that the aspect ratios of the majority of targets (94% in PSeg-SSDD and 97% in HRSID) are smaller than 6. In order to accommodate ship targets with various aspect ratios and enable network to effectively sample the strong scattering region of the bow, we design a strategy for symmetric nonuniform sampling within a 60° range centered around the ship's direction, as illustrated in Fig. 4.

Our sampling strategy can be summarized as follows: we sample at angle α, as well as angles that are 5°, 15°, and 30° away from the main axis direction. The reason for sampling within a range of 60° is that the aspect ratios of ship targets have both large numerical values and a wide range of variations. The nonuniform angle sampling is performed because our focus is on the strong scattering region at the bow of the ship, and we do not pay excessive attention to information of direction perpendicular to the main axis. In addition, considering that the ship dataset uses the long-side representation for the OBB, where the angle annotation represents the angle between the longer side of the rectangle and the x-axis, the angle annotation does not always correspond to the direction of the ship's bow but rather the angle of the ship's main axis within the range of [0, 180°). Similarly, the target angles predicted by the angle branch also correspond to the angle of the ship's main axis within the range of [0, 180°). Therefore, to ensure that the sampling angles for key points effectively fall within the strong scattering region at the bow, we not only need to perform nonuniform sampling within the range of [α-30°, α+30°], but also symmetrically perform nonuniform sampling within the angle range of [α+150°, α+210°]. Hence, the nonuniform angle sampling strategy can be summarized as follows: for the main axis angle α of the ship, we set the sampled angles obtained by combining it with the angle template as {α-30°, α-15°, α-5°, α, α+5°, α+15°, α+30°, α+150°, α+165°, α+175°, α+180°, α+185°, α+195°, α+210°}. The specific algorithmic flow is shown in Algorithm 1.

In instance segmentation, the target's polygon mask is often defined by a sequence of ordered discrete points. By sequentially connecting these discrete points, a set of points with continuity representing the target contour can be obtained, where continuity refers to that adjacent points in the set are within each other's 8-neighborhood. Based on the coordinates of the contour points and the center point, the angle of each point relative to the center point can be calculated. However, in cases where the target size is small, the points on the target contour may not precisely match specific sampling angles. Therefore, it is necessary to find the contour point that is closest to the sampling angle as an approximate representation of the dynamic key points. The

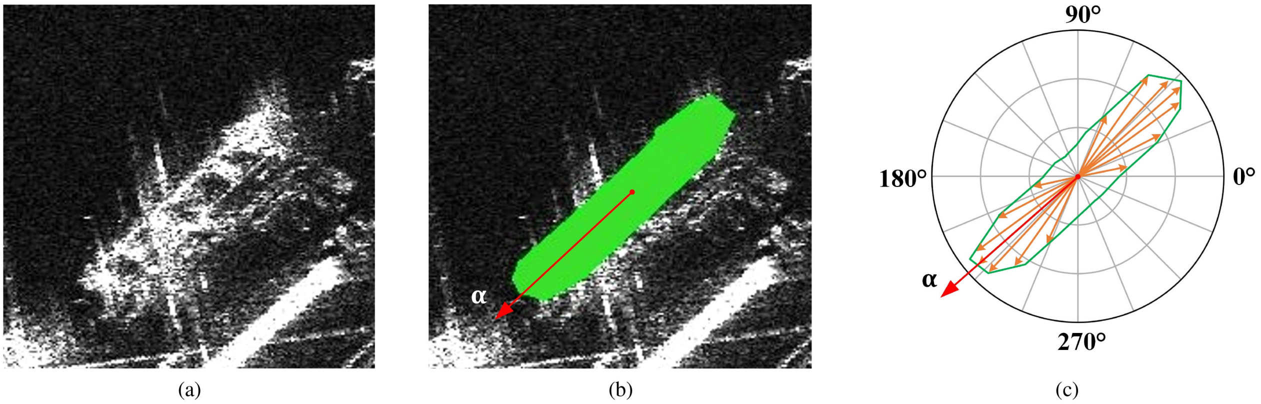


Fig. 4. Illustration of symmetric nonuniform sampling and key points distances encoding. (a) Original image. (b) Ground truth. (c) Encoded result.

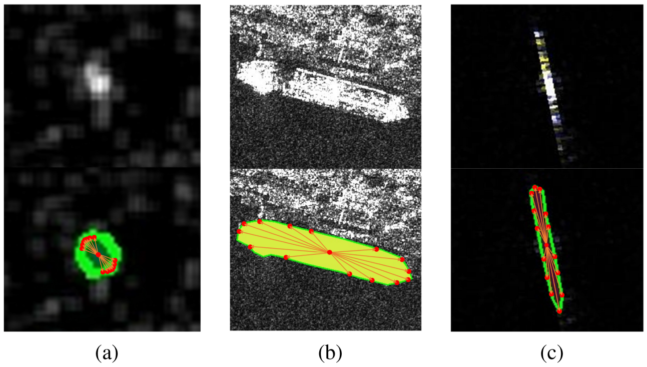


Fig. 5. Illustration of sampling results of targets with different aspect ratios. (a) Aspect ratio = 1.09. (b) Aspect ratio = 5. (c) Aspect ratio = 10.81.

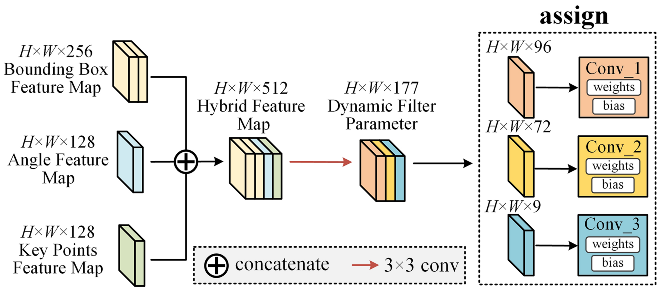


Fig. 6. Parameter generation process of the dynamic mask head using an implicit encoding method.

corresponding distance encoding result can be calculated based on the coordinates of the contour point (x_i, y_i) and the target's center point (x_c, y_c) using the following equation:

$$d_i = \sqrt{(x_i - x_c)^2 + (y_i - y_c)^2}. \tag{3}$$

In order to validate the effectiveness of the sampling strategy for ship targets with various aspect ratios, we conducted tests specifically for ship targets with an aspect ratio of 5, as well as targets with the maximum and minimum aspect ratios in the dataset. The results of these tests are shown in Fig. 5.

Fig. 5(a) and (c), respectively, shows the sampling results for targets with extreme aspect ratios of 1.09 and 10.81, while Fig. 5(b) presents the sampling results for a ship target with an aspect ratio of 5. Based on the sampling results in Fig. 5, it can be observed that when the target has an aspect ratio of 1.09, the network performs dense and detailed sampling in the bow. When the target has an aspect ratio of 10.81, the sampling points are dispersed along the overall contour of the ship, but at least three points are still sampled in the bow, indicating effective sampling of the bow region. The remaining points distributed perpendicular to the ship's main axis direction may not sample the bow region directly, but they contribute to generating more accurate instancewise attention maps. The results demonstrate that our nonuniform sampling strategy effectively samples the strong scattering region of the ship's head under representative and extreme conditions.

2) Implicit Encoding: The process of obtaining parameters for the dynamic mask head through the convolution operation on the feature map of the detection branch is referred to as implicit encoding [19]. In this process, the shape and position of the bounding box are encoded into the generated parameters. Usually, the feature map used for generating the mask head filtering parameters is the same as the intermediate feature map used for generating the four parameters of the HBB. Implicit encoding has been proven to be a powerful method for feature representation in [19]. Therefore, it is natural for us to consider whether using more refined features as the input to the dynamic filter parameter generation network can further enhance the sensibility of the dynamic filter for fine-grained features of the target. We design a dynamic mask head parameter generation module that strengthens the information of the target's key points, as illustrated in Fig. 6.

We first extract the intermediate feature maps from the HBB detection branch, angle encoding prediction branch, and key points distances regression branch. The channel numbers of these intermediate feature maps are 256, 128, and 128, respectively. Each of these feature maps undergoes a 3×3 convolutional decoding operation to obtain the corresponding parameters (e.g., the HBB detection branch produces horizontal box

TABLE I
CORRESPONDENCE BETWEEN THE PARAMETERS OF THE CONVOLUTIONAL LAYERS IN THE DYNAMIC MASK HEAD AND THEIR RESPECTIVE CHANNELS

Layer	C_{in}	C_{out}	Weights/Biases	Corresponding channels
Conv_1	11	8	88 / 8	[1, 88] / [89, 96]
Conv_2	8	8	64 / 8	[97, 160] / [161, 168]
Conv_3	8	1	8 / 1	[169, 176] / 177

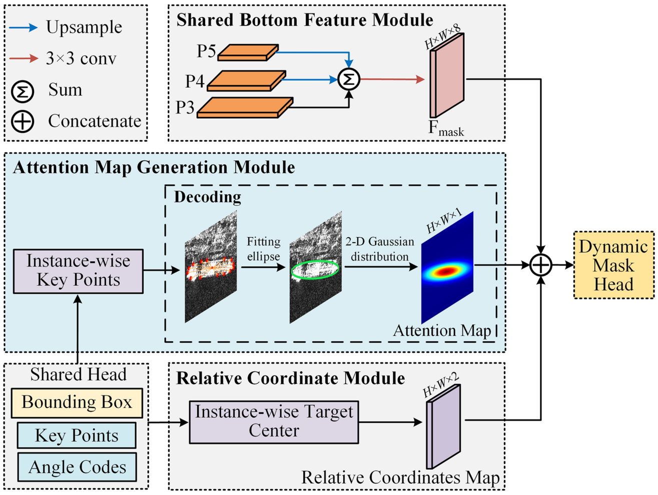


Fig. 7. Illustration of IAMM.

detection results with a shape of $H\times W\times 4$). To preserve the feature representation capability of the feature maps, we use the intermediate feature maps directly as inputs to the parameter generation branch of the dynamic mask head. Specifically, we concatenate the three intermediate feature maps and then apply a 3×3 convolution to obtain the 177 channels dynamic mask head convolutional parameters. The 177 channels correspond to the parameters of the three convolutional layers. Channels 1–88 and 89–96 contain the weights and biases of the first layer Conv_1, of the dynamic mask head, respectively. Similarly, channels 97–160 and 161–168 correspond to the weights and biases of the second layer Conv_2, respectively, while channels 169–176 and 177 correspond to the weights and biases of the third layer Conv_3, respectively. The mapping between parameters and channels is given in Table I, where C_{in} and C_{out} represent the input and output channel numbers of the convolutional layers, respectively.

C. Instancewise Attention Map Module

Motivated by the idea of combining high resolution feature maps and attention mechanism in [36], [37], [38], we designed the IAMM, as shown in Fig. 7. The IAMM consists of three components: the shared bottom feature module, attention map generation module (AMGM), and relative coordinate module. The resolution of the input feature map in the mask branch plays a crucial role in the accuracy of the segmentation results, especially in inshore scenes. Therefore, we adopt the shared bottom feature map module and relative coordinate module from [19] to generate high-resolution and high-quality bottom feature maps, which is constructed by concatenating the basic mask feature map F_{mask} and the relative coordinate map. In addition, we design the AMGM to generate the attention map, and we introduce the attention mechanism by concatenating it with the bottom feature maps.

Fig. 7 illustrates the process of generating the mask feature map and provides an overview of the generation process for the basic mask feature map F_{mask}, relative coordinate maps, and attention maps. It is worth noting that the shared bottom feature map module depicted in Fig. 7 generates imagewise feature maps. This means that for a single image, the basic mask feature map F_{mask} is computed only once. On the other hand, the feature maps generated by the relative coordinate module and the AMGM are instancewise feature maps. This means that different targets within the same image will have different feature maps, and the instancewise information is derived from the shared detection head. Using relative coordinate maps is a method to enhance the sensibility of the dynamic mask head for corresponding targets [19]. It involves the concatenation of a relative coordinate map onto the basic feature map F_{mask}. In the relative coordinate map, the value of each pixel represents the difference between the pixel's absolute coordinates and the center coordinates of the target. The relative coordinate map essentially acts as an attention map centered around the instance's center coordinates, decaying at the same rate along both the x-axis and y-axis. Since the values in the relative coordinate map are solely determined by the difference between the pixel's absolute coordinates and the target's center coordinates, it contains minimal additional information about the target, apart from the coordinates of the target center.

Therefore, to provide the dynamic mask head with more refined information of specific target, we design the AMGM. Building upon the basic mask feature map F_{mask} and the relative coordinate map, AMGM decodes the angle and key points predictions from the detection branch. This decoding process generates a 2-D Gaussian distribution heatmap [28] containing the target's angle, aspect ratio, and key points information. Together with F_{mask} and the relative coordinate map, this heatmap is inputted into the dynamic mask head, enabling the network to focus on regions with a higher likelihood of containing the target. The decoding algorithm for IAMM is outlined in Algorithm 2. The key points information of the strong scattering regions in the targets is encoded implicitly into the parameters of the dynamic mask head. In addition, the introduction of the 2-D Gaussian distribution heatmap serves as an instancewise attention mechanism, further enhancing the information related to the distribution region and strong scattering regions of specific instances.

During the decoding process, we first decode the angle code from the shared detection head to obtain the main axis angle θ_M of the target. Then, a set of sampling angles θ_s is obtained using predefined angle templates. For each sampling angle, the corresponding key points distance encoding is assigned, resulting in a set of dynamic key points relative coordinates P. The dynamic key points set is then fitted to elliptical parameters using the direct least-squares fitting approach. The equation of an ellipse in a Cartesian coordinate system can be represented as $x^2 + Axy + By^2 + Cx + Dy + E = 0$. To calculate

Algorithm 2: Dynamic Key Points Decoding Algorithm.

```
Input: DCL of the target angle $L$, coding vector of dynamic key point $\overrightarrow{\mathrm{V}} = (d_1, d_2, \ldots, d_{14})$, target center $c = (x_c, y_c)$.
Output: Parameters of 2-D Gaussian distribution $\{\mu, \Sigma\}$.
Decode the DCL $L$ to get the major axis direction of the target $\theta_M \in [0, 180^\circ)$;
From $\theta_M$ and angle template for symmetric non-uniform sampling, obtain the set of sampling angles $\theta_s = \{\theta_{s1}, \theta_{s2}, \ldots, \theta_{s14}\}$;
The relative coordinate set of dynamic key points $P$ is calculated according to the dynamic key point coding vector $\overrightarrow{\mathrm{V}}$ and sampling angle $\theta_s$,
    $P = \{p_i \mid p_i = (x_i, y_i), i = 1, 2, \ldots, 14\}$
    where $x_i = d_i \times \cos(\theta_i), y_i = d_i \times \sin(\theta_i)$;
Use direct least squares to fit a set of dynamic key points to a set of elliptical parameters $\{x_c, y_c, a, b, \theta\}$;
$\{x_c, y_c, a, b, \theta\}$ is converted to parameters of 2-D Gaussian distribution $\{\mu, \Sigma\}$;
return $\{\mu, \Sigma\}$.
```

the parameters $\{A, B, C, D, E\}$ that best fit the point set, we minimize the objective function $F(A, B, C, D, E)$ using a direct least-squares fitting approach. The expression of F is shown in the following equation:

$$\begin{aligned} &F(A, B, C, D, E) \\ &= \sum_{i=1}^{2N} \left(x_i^2 + A x_i y_i + B y_i^2 + C x_i + D y_i + E\right)^2. \end{aligned} \tag{4}$$

To obtain the minimum value of F, we need to solve the system of equations where the partial derivatives of F with respect to each parameter are set to zero. Based on the solved result $\{A, B, C, D, E\}$, the ellipse parameters can be calculated as follows:

$$\begin{aligned} x_c &= \frac{2BC - AD}{A^2 - 4B} \\ y_c &= \frac{2D - AD}{A^2 - 4B} \\ a &= \sqrt{\frac{2\left(ACD - BC^2 - D^2 + 4BE - A^2E\right)}{\left(A^2 - 4B\right)\left(B - \sqrt{A^2 + (1 - B)^2} + 1\right)}} \\ b &= \sqrt{\frac{2\left(ACD - BC^2 - D^2 + 4BE - A^2E\right)}{\left(A^2 - 4B\right)\left(B + \sqrt{A^2 + (1 - B)^2} + 1\right)}} \\ \theta &= \arctan\left(\mathrm{sqrt}\left(\frac{a^2 - b^2 B}{a^2 B - b^2}\right)\right) \end{aligned} \tag{5}$$

where (x_c, y_c) represents the center coordinates of the ellipse, a and b are the major and minor axes of the ellipse, respectively, and θ is the orientation angle of the ellipse, ranging from 0 to 180°. Based on the aforementioned ellipse parameters, the corresponding 2-D Gaussian distribution parameters $\{\mu, \Sigma\}$ can be calculated as follows:

$$\begin{aligned} \mu &= (x_c, y_c) \\ \Sigma &= R \cdot \Sigma_0 \cdot R^\top \end{aligned} \tag{6}$$

where μ denotes the mean and Σ denotes the covariance matrix. R denotes the rotation matrix, and Σ_0 for the covariance matrix at angle 0. The expressions for R and Σ_0 are as follows:

$$\begin{aligned} R &= \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \\ \Sigma_0 &= \frac{1}{12}\begin{bmatrix} a^2 & 0 \\ 0 & b^2 \end{bmatrix}. \end{aligned} \tag{7}$$

After calculating the parameters of the 2-D Gaussian distribution using (6) and (7), the probability density of the distribution can be computed following the method described in [28], as shown below:

$$f(X) = \frac{1}{2\pi|\Sigma|^{1/2}} \exp\left(-\frac{1}{2}(X - \mu)^T \Sigma^{-1} (X - \mu)\right) \tag{8}$$

where X represents the coordinates, which is a 2-D vector. According to (8), each pixel on the feature map can be assigned a value of the 2-D Gaussian distribution corresponding to a specific instance. The closer the pixel is to the target center, the larger the value of the pixel will be. The covariance matrix Σ includes the information about the target's angle and aspect ratio.

D. Loss Function

Multitask loss function for network training in this article can be divided into three parts: the HBB detection branch, the dynamic key points detection branch, and the mask prediction branch. The multitask loss function is as follows:

$$\mathcal{L}_{\text{overall}} = \mathcal{L}_{\text{FCOS}} + \lambda \mathcal{L}_{\text{DKPM}} + \mu \mathcal{L}_{\text{mask}} \tag{9}$$

where $\mathcal{L}_{\text{FCOS}}$ represents the loss function for the HBB detection branch, $\mathcal{L}_{\text{DKPM}}$ represents the loss function for the key points detection branch, and $\mathcal{L}_{\text{mask}}$ represents the loss function for the mask prediction branch. λ and μ both are equal to 1. The following is a detailed introduction to the three parts of the loss function.

Since we adopt the FCOS framework [48] as the HBB detection branch, the loss function for this part is designed following the same principles as in [48], including the classification branch, bounding box regression branch, and centerness branch, as follows:

$$\begin{aligned} \mathcal{L}_{\text{FCOS}} = \frac{1}{N_{\text{pos}}} \Bigg[&\sum_{x,y} \mathcal{L}_{\text{cls}}\left(c_{x,y}, c_{x,y}^*\right) + \sum_{x,y} \mathcal{L}_{reg}\left(\mathbf{t}_{x,y}, \mathbf{t}_{x,y}^*\right) \\ &+ \sum_{x,y} \mathcal{L}_{\text{centerness}}\left(m_{x,y}, m_{x,y}^*\right) \Bigg]. \end{aligned} \tag{10}$$

where $\mathcal{L}_{\text{cls}}$ denotes the focal loss, $\mathcal{L}_{\text{reg}}$ represents the IoU loss, and $\mathcal{L}_{\text{centerness}}$ represents the cross entropy loss. N_{pos} is the number of positive samples. $c_{x,y}$ and $c_{x,y}^*$ are the predicted and ground truth class labels at positive sample points, respectively. Similarly, $\mathbf{t}_{x,y}$ and $\mathbf{t}^*_{x,y}$ are the predicted and ground truth HBB parameters, respectively. In addition, $m_{x,y}$ and $m^*_{x,y}$ are the predicted and ground truth centerness values, respectively.

The loss function $\mathcal{L}_{\text{DKPM}}$ for the dynamic key points detection branch consists of angle encoding loss and dynamic key points distance encoding loss. The angle encoding is performed using DCL, where the angle information is encoded as a set of natural binary codes. The supervised training is done using cross entropy loss. In addition, considering that prediction errors in higher order bits may lead to larger angle errors, the class weights are set to linearly decay from higher order bits to lower order bits. The dynamic key points distance encoding is represented as a 14-D vector, and we use the smooth L1 loss function [50] for supervised training. The calculation formula is as follows:

$$\mathcal{L}_{\text{DKPM}} = \frac{1}{N_{\text{pos}}}\left[\sum_{x,y}\mathcal{L}_{\text{angle}}\left(\theta_{x,y}, \theta^*_{x,y}\right) + \sum_{x,y}\mathcal{L}_{\text{KPD}}\left(V_{x,y}, V^*_{x,y}\right)\right] \tag{11}$$

where $\mathcal{L}_{\text{angle}}$ represents the cross entropy loss function used in the angle encoding branch, and $\mathcal{L}_{\text{KPD}}$ represents the smooth L1 loss used in the dynamic key points distance encoding branch. $\theta_{x,y}$ and $\theta^*_{x,y}$ represent the predicted and ground truth values for angle encoding, respectively. Similarly, $V_{x,y}$ and $V^*_{x,y}$ represent the predicted and ground truth values for key points distance encoding, respectively.

The mask branch is supervised with the dice loss [51], which evaluates the quality of the predicted mask by calculating the overlap between the predicted mask and the ground truth mask. The loss function is defined as follows:

$$\mathcal{L}_{\text{mask}} = \frac{1}{N_{\text{pos}}}\sum_{x,y}\mathcal{L}_{\text{dice}}\left(M_{x,y}, M^*_{x,y}\right) \tag{12}$$

where $\mathcal{L}_{\text{dice}}$ denotes the dice loss. $M_{x,y}$ and $M^*_{x,y}$ are predicted mask and ground truth mask, respectively.

IV. Experimental Results and Evaluation

In this section, we conduct experiments to evaluate the detection performance of the proposed method. First, we introduce the dataset and the settings of related experiments. Next, the evaluation criteria are described. At last, we conduct comparative experiments between our proposed method and other SOTA methods on the PSeg-SSDD and HRSID to validate the effectiveness of our approach.

A. Dataset and Settings

The dataset SSDD is a publicly available SAR ship image dataset used for horizontal box detection [29]. Due to subsequent extensions by researchers, which include annotations for the OBB and instance segmentation, it is also referred to as the PSeg-SSDD or SSDD++. The dataset consists of 1160 marine SAR images with resolutions ranging from 1 to 15 m. These images are captured by different sensor models and polarization, and contain a total of 2587 ships. In our experiments, following the official release standards of PSeg-SSDD [30], we split the dataset into 928 training images and 232 testing images. The testing set comprises 46 images from inshore scenes and 186 images from offshore scenes. We conducted a series of experiments on the PSeg-SSDD to validate the effectiveness of our proposed method.

The dataset HRSID contains 138 panoramic SAR imageries with ranging resolution from 1 to 5 m, and 5605 slices with 800 × 800 resolution under the overlapped ratio of 25% representing various imaging modes, polarization techniques, and resolutions, along with 16 951 ship targets [31]. Compared with the size distribution of targets in the PSeg-SSDD (2009 small targets, 507 medium targets, and 71 large targets), the size distribution of targets in the HRSID is more balanced, which includes 9242 small objects, 7388 medium-sized objects, and 321 large objects. In our experiments, following the official release standards of HRSID [31], we split the HRSID into 3642 training images and 1962 testing images to assess how well each model performs in various scenes.

In the training process, we use ResNet50 as the backbone network and initialize it with pretrained weights from the ImageNet dataset. For the newly added layers, we initialize them using the Kaiming initialization method. The network is trained using stochastic gradient descent for 20 K iterations, with an initial learning rate of 0.001. The learning rate is reduced by a factor of 10 at 15 and 18 K iterations. The weight decay and momentum are set to 0.0001 and 0.9, respectively. The experiments are conducted using the MMDet framework [52]. The comparative methods, including Mask R-CNN [8], Yolact [17], PointRend [53], SOLOv2 [20], QueryInst [54], Mask2Former [55], SparseInst [56], and RT-MDet [57] are also implemented using the MMDet framework. The CondInst [19] is implemented using the AdelaiDet framework [58]. All experiments are performed on a platform with Ubuntu 20.04, 32 GB memory, and a GTX 3090 GPU.

B. Evaluation Metric

In the field of instance segmentation in SAR image interpretation, the accuracy of mask prediction is defined by the IoU between the predicted mask and the ground truth. The calculation formula for mask IoU is

$$\text{IoU}_{\text{Mask}} = \frac{\text{Mask}_{\text{pred}} \bigcap \text{Mask}_{\text{GT}}}{\text{Mask}_{\text{pred}} \cup \text{Mask}_{\text{GT}}} \tag{13}$$

where $\text{Mask}_{\text{pred}}$ and Mask_{GT} are predicted mask and ground truth mask, respectively.

To evaluate the segmentation accuracy of the methods under different IoU threshold conditions, we introduce three evaluation metrics from Microsoft Common Objects in Context [59]: average precision (AP), AP_{50}, and AP_{75}. Generally, AP is the most commonly used evaluation criterion as it represents the AP of the segmentation network across IoU thresholds in the range of [0.5, 0.95] with a step of 0.05, providing a comprehensive reflection of the network's performance. AP_{50} and AP_{75} represent the precision of the network at IoU thresholds of

0.5 and 0.75, respectively. Clearly, AP_{75} is more challenging and better represents the segmentation accuracy of the network. In addition, since the dataset contains ships of different sizes and varying segmentation difficulties, we introduced AP_S, AP_M, and AP_L to evaluate the network's segmentation capability for different-sized ships. Similar to AP, they represent the AP of the segmentation network across IoU thresholds in the range of [0.5, 0.95] with a step size of 0.05. AP_S corresponds to small targets with a mask area less than 32^2, AP_M corresponds to medium-sized targets with a mask area between 32^2 and 64^2, and AP_L corresponds to large targets with a mask area larger than 64^2. Furthermore, to visually demonstrate the performance of each network, we plot precision–recall (PR) curve for each network. The PR curve intuitively illustrates the variation of segmentation performance of the network across IoU thresholds in the range of [0, 1]. A larger area under the PR curve indicates better segmentation performance of the network.

In addition, some evaluation metrics for network complexity are introduced, such as model size, the number of parameters, and floating point operations (FLOPs). Model size and the number of parameter count represent the storage cost of the model, where a larger parameter count requires more training data. With limited training data, a larger number of parameters make the network more prone to overfitting. FLOPs, on the other hand, represent the computational complexity of the network.

C. Experimental Results

To validate the effectiveness of our method, we selected Mask R-CNN [8], Yolact [17], PointRend [53], SOLOv2 [20], CondInst [19], QueryInst [54], Mask2Former [55], SparseInst [56], and RTMDet [57] for comparison, as they have demonstrated excellent performance in instance segmentation. A brief introduction of these methods is provided below.

1) *Mask R-CNN:* It is a classic two-stage instance segmentation method. It extends Faster R-CNN by adding a parallel mask branch along with the detection branch. It performs mask prediction on RoI regions, resulting in better performance while incurring only a minimal increase in computational overhead compared with Faster R-CNN.
2) *Yolact:* It is one of the earliest attempts at real-time instance segmentation methods. It utilizes a fully convolutional model to construct a one-stage detector and decomposes the instance segmentation task into mask prototypes and corresponding mask coefficients. This method achieves fast and straightforward instance segmentation based on global masks.
3) *PointRend:* Taking a novel perspective, PointRend treats the instance segmentation problem as a rendering problem. It introduces a point-based rendering neural network module and utilizes an iterative refinement algorithm to perform segmentation at adaptively selected positions.
4) *SOLOv2:* Inspired by the ideas of Yolact, SOLOv2 further reduces dependency on bounding boxes and decouples mask generation into predictions of mask kernels and mask feature maps. It separately generates convolutional kernels and the input feature map for the kernels, and achieves great performance.
5) *CondInst:* CondInst not only frees instance segmentation from relying on bounding boxes but also eliminates the need for mask prototypes. Instead, it adopts the concept of dynamic filters to predict a corresponding mask head for each instance. The shape and size of the mask are implicitly encoded in the convolutional parameters of the mask head, reducing the parameter size and computational complexity of the mask head, which enables the network to predict the global mask directly.
6) *QueryInst:* QueryInst offers a new perspective for instance segmentation by employing a query-based multistage end-to-end network. It unifies the representation of instance attributes, such as class, bounding box, and mask into a single framework.
7) *Mask2Former:* The authors propose a concise and versatile model that addresses both semantic segmentation and instance segmentation tasks, called masked-attention mask transformer (Mask2Former). By incorporating masked attention within the transformer architecture, it achieves faster convergence and performance improvements.
8) *SparseInst:* SparseInst is a novel and efficient fully convolutional instance segmentation framework. Unlike many instance segmentation methods that rely on object detection, this network represents objects using a set of sparse instance activation maps. It aggregates information from highlighted regions of each instance to obtain instancewise segmentation results. In addition, based on a bipartite matching approach, it achieves one-to-one instance prediction, avoiding the need for postprocessing nonmaximum suppression operations and speeding up the inference process.
9) *RTMDet:* RTMDet is an efficient real-time object detector. The model is built on the architecture that has compatible capacities in the backbone and neck, which is constructed by a basic building block that consists of large-kernel depthwise convolutions. It achieves the great parameter accuracy tradeoff for various application scenarios.

Table II presents the instance segmentation performance of different methods on the PSeg-SSDD in both inshore and offshore scenes. From Table II, it can be seen that our method achieves the highest accuracy in both inshore and offshore scenes. Although there are comparative methods that show similar accuracy to our method when evaluated separately in inshore or offshore scenes, such as PointRend with an AP score only 0.2% lower than our method in inshore scenes, and Mask2Former with an AP score only 0.5% lower than our method in offshore scenes, these methods often fail to perform well in both scenes simultaneously. In contrast, our method not only balances the accuracy in both scenes, but also achieves the highest accuracy in both scenes. Particularly for small targets in inshore scenes, our method exhibits the best segmentation performance. Even the method with the best performance among the comparative methods in inshore scenes, PointRend, falls behind our method by 2.2%. This is due to the fact that small objects

TABLE II
INSTANCE SEGMENTATION PERFORMANCE OF DIFFERENT METHODS ON THE PSEG-SSDD

Methods	Scene	AP	AP_{50}	AP_{75}	AP_{S}	AP_{M}	AP_{L}
Mask R-CNN	offshore	0.685	0.983	0.848	0.675	0.747	0.674
	inshore	0.556	0.893	0.668	0.571	0.551	0.486
Yolact	offshore	0.649	0.971	0.825	0.631	0.744	0.636
	inshore	0.499	0.887	0.555	0.493	0.530	0.501
PointRend	offshore	0.686	0.979	0.897	0.648	0.757	0.755
	inshore	0.626	0.942	**0.764**	0.604	**0.706**	0.584
SOLOv2	offshore	0.716	0.989	0.914	0.703	0.775	0.711
	inshore	0.570	0.907	0.690	0.537	0.678	0.585
CondInst	offshore	0.691	0.988	0.881	0.662	0.752	0.786
	inshore	0.603	0.948	0.725	0.565	0.686	**0.674**
QueryInst	offshore	0.717	0.983	0.917	0.700	0.791	0.713
	inshore	0.570	0.888	0.656	0.559	0.663	0.473
Mask2Former	offshore	0.715	0.975	0.914	0.701	0.790	0.728
	inshore	0.605	0.903	0.757	0.581	0.704	0.670
SparseInst	offshore	0.702	0.978	0.883	0.705	0.708	0.600
	inshore	0.484	0.774	0.560	0.509	0.486	0.288
RTMDet	offshore	0.655	0.988	0.836	0.631	0.763	0.708
	inshore	0.509	0.896	0.485	0.448	0.696	0.614
Ours	offshore	0.720	0.989	0.935	0.708	0.787	0.728
	inshore	**0.628**	**0.960**	0.760	**0.626**	0.663	0.590

Bold items denote the optimal inshore values in the columns, the underlined items indicate the optimal offshore values in the columns.

in inshore scenes are often densely distributed and susceptible to strong scattering interference from the land, making them more challenging to detect. However, our method tackles this difficulty by predicting and assigning strong scattering points for each target, as well as utilizing instancewise attention maps generated based on the information of strong scattering points, thus optimizing the network's perception for each instance under the interference of adjacent ships and land region. Among the comparison methods, Mask2Former achieves the highest accuracy for medium targets in inshore scenes, indicating that its introduced attention mechanism effectively enhances the network's perception capability for larger objects. However, the gain for smaller targets is not as strong as for larger targets. On the other hand, while SparseInst performs well in offshore scenes, its performance significantly drops in complex inshore scenes. This may be attributed to the adoption of instance activation mapping for sparse prediction, where densely distributed small targets in SAR images and interference from land clutter greatly affect the process of aggregating features. In addition, the clutter introduces significant interference in feature extraction for small objects. Since targets with small size are the majority in inshore scenes, the performance of SparseInst experiences a rapid decline in inshore scenes. Analyzing the accuracy of these methods in both offshore and inshore scenes reveals the difficulty of achieving good performance in both scenes. This is due to the varying levels of clutter interference and the differences in number of targets between the two scenes. In contrast, our method optimizes accuracy in inshore scenes while also benefiting accuracy in offshore scenes. It achieves a balance in instance segmentation performance between inshore and offshore scenes. Particularly, for the challenging AP_{75} metric, which represents the network's fine-grained detection capability, our method achieves the highest AP_{75} in offshore scenes, surpassing QueryInst by 1.8%. In inshore scenes, our method ranks second with an AP_{75} score only 0.4% lower than the highest performing PointRend. Considering the overall detection metrics in both scenes, our method demonstrates the highest level of fine-grained detection.

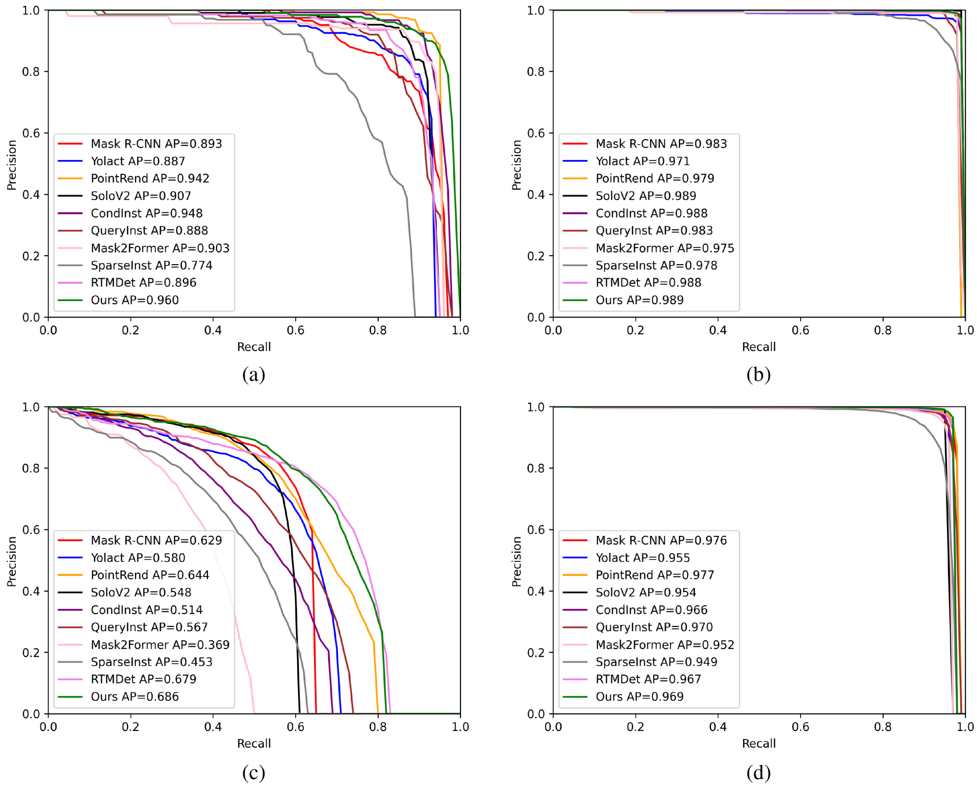
According to the results of comparative experiments on the HRSID in Table III, our method still maintains a leading position in most metrics, especially in inshore scenes. which is similar to the experimental results on the PSeg-SSDD. Although our method did not achieve the highest AP_{50} metric in offshore scenes, the PointRend with the best performance only outperforms our method by 0.8%. Moreover, in terms of the challenging AP_{75} metric, our method leads by 3.5% in offshore scenes and 3.7% in inshore scenes. Due to the more balanced distribution of targets of different sizes and the higher resolution of the original SAR images in the HRSID, the performance gap in detection accuracy between our method and others for large targets has been further narrowed. Our method even achieves the highest AP_{M} in offshore scenes.

Fig. 8 shows the PR curves, which comprehensively demonstrate the instance segmentation performance of various methods in inshore and offshore scenes. Overall, the area under PR curve of the same method is significantly smaller in the inshore scenes compared with the offshore scenes, indicating greater segmentation difficulty in inshore scenes. From Fig. 8(a), it can be seen that the CondInst and PointRend perform well in inshore scenes of PSeg-SSDD, even slightly outperforming our method at lower recall. However, as the recall increases, the precision of the CondInst and PointRend decreases noticeably faster than our proposed method. As for the results in offshore scenes of HRSID in Fig. 8(c), it is evident that both RTMDet and our

TABLE III
INSTANCE SEGMENTATION PERFORMANCE OF DIFFERENT METHODS ON THE HRSID

Methods	Scene	AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L
Mask R-CNN	offshore	0.623	0.976	0.764	0.620	0.683	0.581
	inshore	0.325	0.629	0.307	0.315	0.440	0.239
Yolact	offshore	0.579	0.955	0.688	0.570	0.688	0.681
	inshore	0.248	0.580	0.168	0.225	0.437	0.271
PointRend	offshore	0.633	0.977	0.795	0.626	0.730	0.736
	inshore	0.342	0.644	0.381	0.311	**0.552**	0.301
SOLOv2	offshore	0.628	0.954	0.804	0.619	0.736	0.719
	inshore	0.282	0.548	0.267	0.247	0.531	0.370
CondInst	offshore	0.658	0.966	0.822	0.653	0.729	0.680
	inshore	0.284	0.514	0.301	0.265	0.436	0.303
QueryInst	offshore	0.659	0.970	0.841	0.654	0.728	0.649
	inshore	0.342	0.567	0.347	0.300	0.481	**0.394**
Mask2Former	offshore	0.606	0.952	0.749	0.595	0.732	0.732
	inshore	0.179	0.369	0.162	0.147	0.418	0.311
SparseInst	offshore	0.627	0.949	0.797	0.618	0.723	0.645
	inshore	0.210	0.453	0.171	0.194	0.359	0.116
RTMDet	offshore	0.649	0.967	0.798	0.644	0.720	0.447
	inshore	0.341	0.679	0.300	0.335	0.455	0.169
Ours	offshore	0.695	0.969	0.876	0.688	0.754	0.731
	inshore	**0.361**	**0.686**	**0.384**	**0.337**	0.526	0.322

Bold items denote the optimal inshore values in the columns, the underlined items indicate the optimal offshore values in the columns.



(a)

(b)

(c)

(d)

Fig. 8. PR curves of different methods in different datasets. (a) PR curves in the inshore scenes of PSeg-SSDD. (b) PR curves in the offshore scenes of PSeg-SSDD. (c) PR curves in the inshore scenes of HRSID. (d) PR curves in the offshore scenes of HRSID.

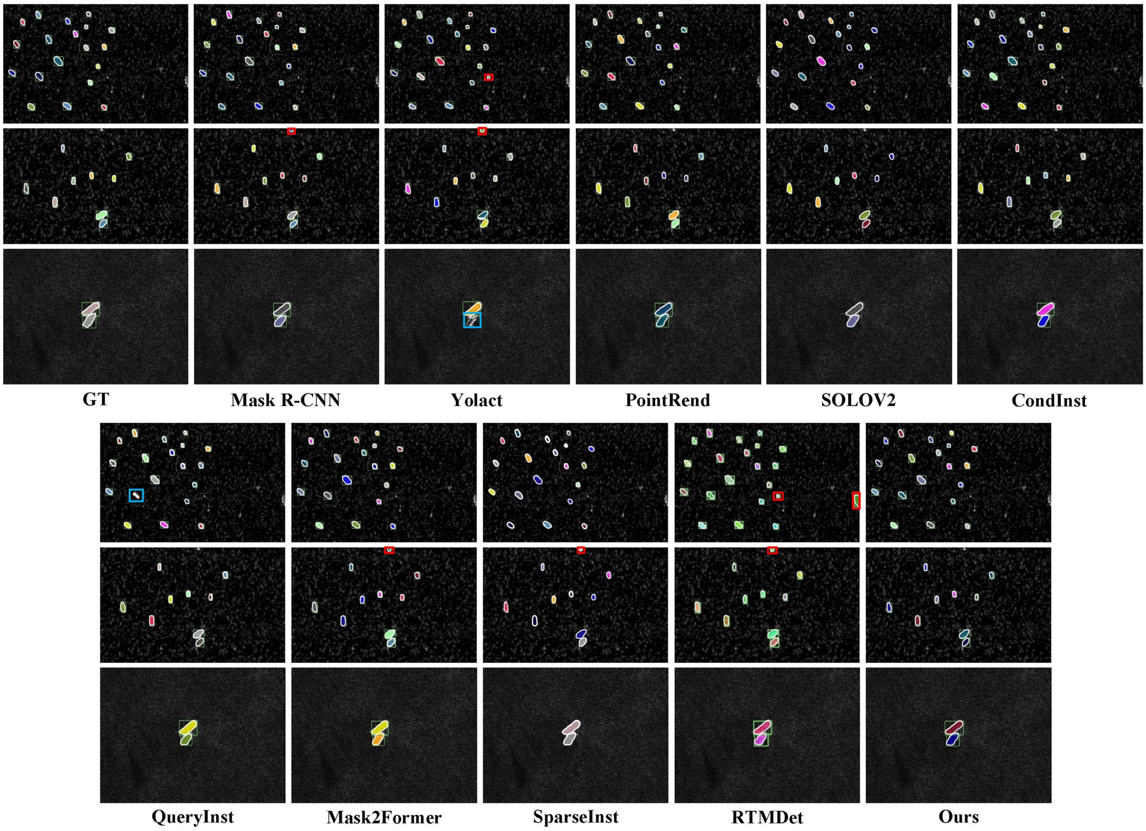


Fig. 9. Results of different methods in the offshore scenes of PSeg-SSDD. The red rectangles indicate false alarms, the blue rectangles represent missed detections, and green rectangles represent results of HBB detection (only some of these methods support detecting HBB).

method exhibit a significant lead compared with other methods. Our method achieves higher precision at low recall, whereas RTMDet demonstrates higher precision at high recall. Overall, our method holds a slight advantage in performance compared with RTMDet. In summary, the results depicted in the Fig. 8 validate the superior performance of our proposed method.

In order to compare our method with the other methods, the segmentation results of experiments conducted on PSeg-SSDD and HRSID are given in Figs. 9– 12. Figs. 9 and 10, respectively, illustrate the results in the offshore scenes of the PSeg-SSDD and HRSID. Similarly, Figs. 11 and 12, respectively, illustrate the results in the inshore scenes of the PSeg-SSDD and HRSID.

From Figs. 9 and 10, it can be observed that although small targets are abundant in the offshore scenes, they are sparsely distributed and not affected by strong scattering interference from land clutter. Therefore, the differences among the comparison methods are not obvious. Mask R-CNN, due to its low resolution of feature map, is not particularly favorable for small object detection and exhibits some false alarms in Fig. 9. Even in the high-resolution SAR images of HRSID depicted in Fig. 10, Mask R-CNN exhibits both missed detections and false alarms in the detection of small targets in scenes. Among the results depicted in Figs. 9 and 10, QueryInst shows more missed detections in both the PSeg-SSDD and HRSID. In SparseInst, the mask results of each target are aggregated from multiple activation regions, which may be affected by sea clutter and incorporate some noise as target features during the aggregation process, resulting in partial false alarms. However, overall, due to the relatively low detection difficulty in the offshore scenes, both the comparison methods and our proposed method perform well.

Figs. 11 and 12 illustrate the results of different methods in the inshore scenes. The red rectangles represent false alarms, the blue rectangles indicate missed detections, and the yellow rectangles represent mask overlap. The green rectangles represent HBB results as supplements to the segmentation results, but only some of these methods support detecting HBB. Compared with Figs. 9 and 10, it is evident that segmentation in the inshore scenes is more challenging.

Based on the segmentation results in Fig. 11, it can be observed that Mask R-CNN, Yolact, and SparseInst perform poorly. This is because the upsampling process of low-resolution feature maps in Mask R-CNN cannot accurately capture the mask contours of the targets, thus failing to improve the accuracy effectively. Yolact utilizes the prediction of mask prototypes and the corresponding mask coefficients, which are aggregated to obtain the final prediction. This places higher demands on the feature extraction capability of the network. When the network fails to adequately differentiate between target features and interference, its performance rapidly deteriorates. Similarly, SparseInst relies on sparse feature predictions and aggregation, which are susceptible to interference from land region with

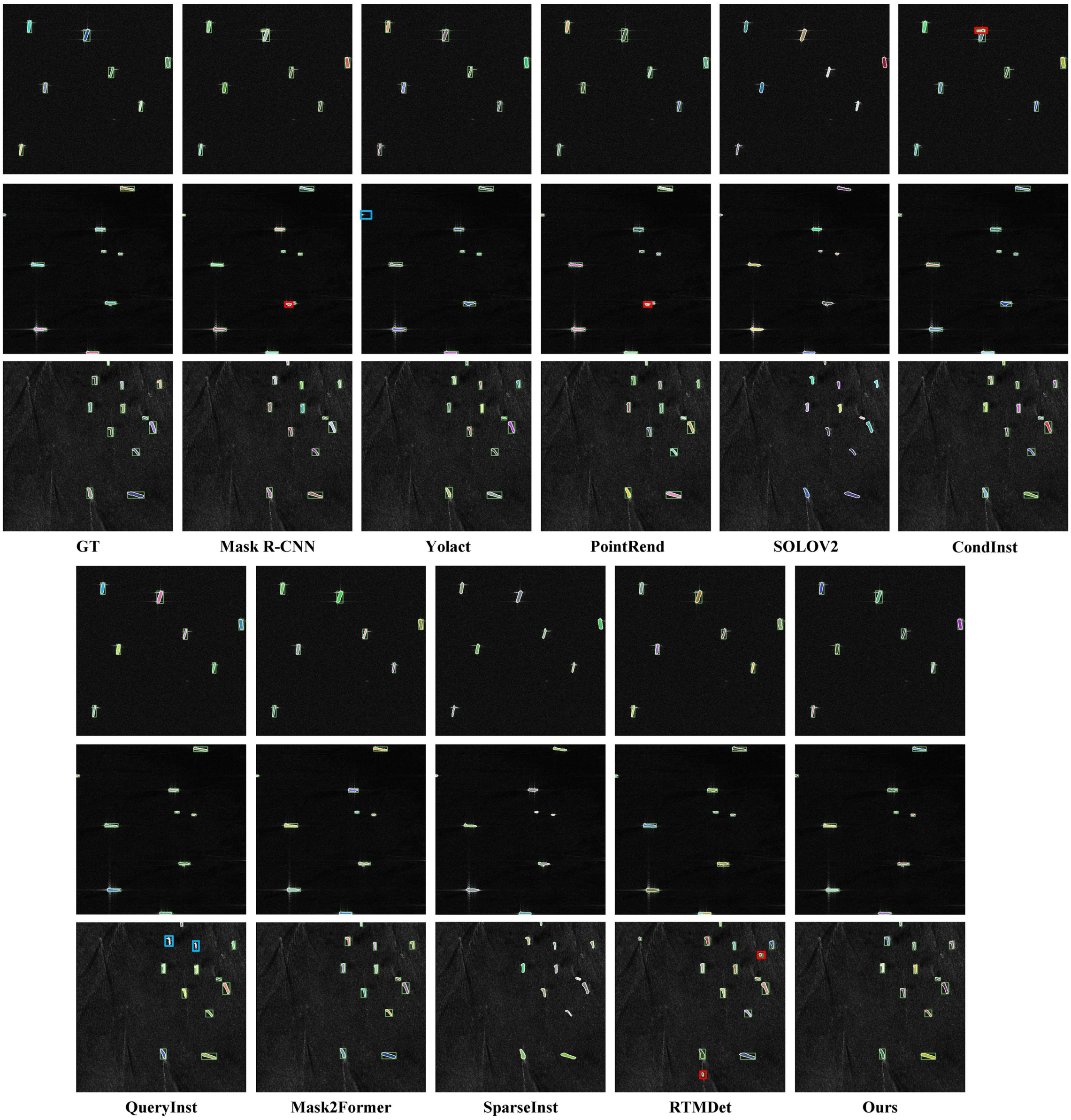


Fig. 10. Results of different methods in the offshore scenes of HRSID. The red rectangles indicate false alarms, the blue rectangles represent missed detections, and green rectangles represent results of HBB detection (only some of these methods support detecting HBB).

strong scattering in SAR images, leading to missed detections. In addition, the network lacks the ability to distinguish densely distributed objects, resulting in mask overlap and ultimately causing missed detections in the second row of the images. CondInst exhibits missed detections in the images of the first and second rows. The missed targets are docked at the pier that is surrounded by high scattering interference from land region and densely distributed target, respectively. The contours of most artificial structures in land are rectangles with a large aspect ratio, which make the network easy to misclassify the ship targets as land structures, and eventually lead to missed detections. Furthermore, Mask R-CNN, SparseInst, CondInst, and RTMDet exhibit evident mask overlap issues in the images in the second row. This is because the ships in the second row of Fig. 11 are densely parked together, resulting in blurred boundaries between individual targets, posing a challenge to the discriminative ability of network. Based on the segmentation results in the inshore scenes of HRSID, as shown in Fig. 12, it can be observed that the detection difficulty is higher in the HRSID compared with the PSeg-SSDD. Despite the higher resolution of images in the HRSID, the inshore scenes of images in HRSID are more complex, such as the port with tremendous cargo handling capacity or the crisscrossed busy canals throughout the trading cities. The interference from artificial structures, such as ports and docks, is intense in inshore images. Simultaneously, a great deal of large containers and small islands similar to ship targets lead to a significant number of false alarms in contrastive methods. As shown in the second row of results of each comparative method in Fig. 12, rich prismatic structure in artificial structures, such as large containers and docks, exhibit strong scattering

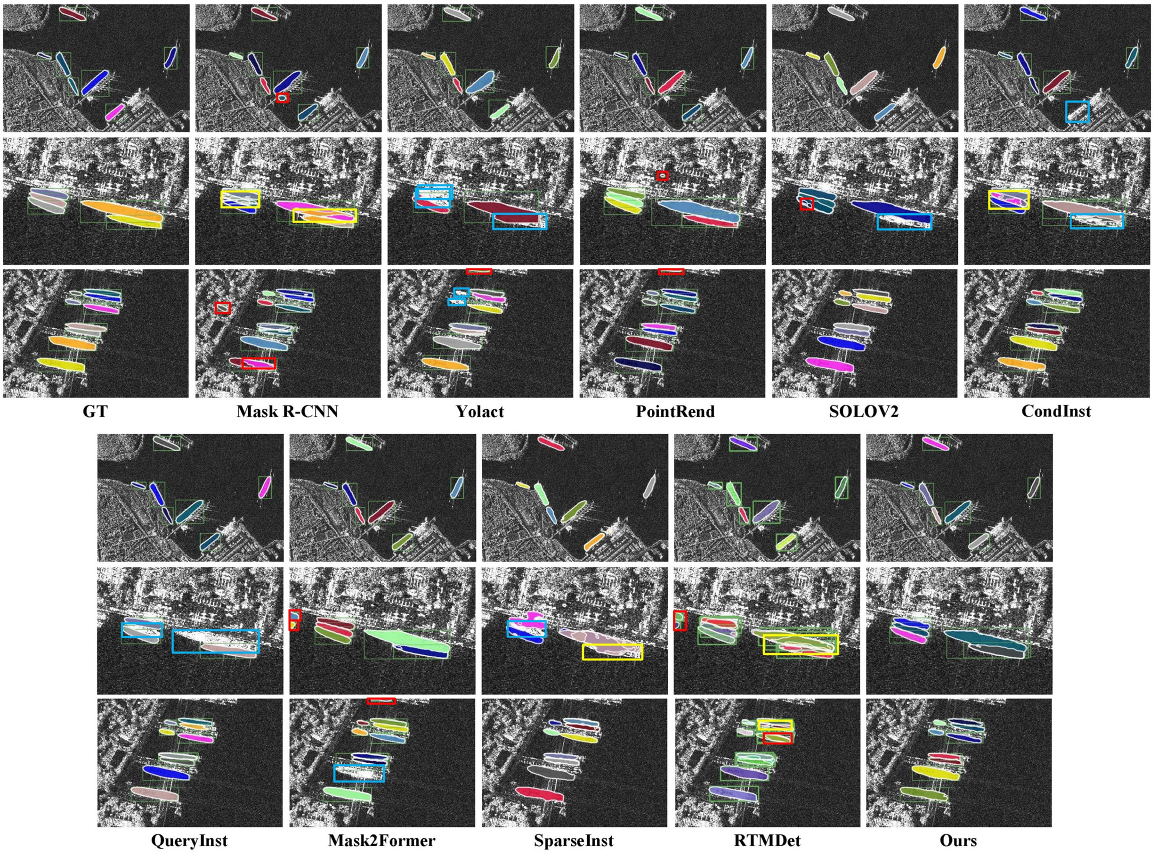


Fig. 11. Results of different methods in the inshore scenes of PSeg-SSDD. The red rectangles represent false alarms, the blue rectangles represent missed detections, the yellow rectangles represent mask overlap, and green rectangles represent results of HBB detection (only some of these methods support detecting HBB).

characteristics and possesses a large aspect ratio, making them prone to being detected as ship targets. Similarly, small islands near the shore are also prone to being detected as ship targets, as illustrated in the first row images of PointRend, CondInst, and Mask2Former in Fig. 12. It is worth noting that false alarms are particularly severe in all the contrastive methods, and most false alarms are associated with small targets. Therefore, improving the network's ability to distinguish small targets from background is crucial for enhancing detection performance. This is also the reason why PointRend achieves good AP metric by using an iterative subdivision algorithm to perform point-based segmentation predictions at adaptively selected locations. In contrast, our proposed method demonstrates consistent optimization for the small targets detection in HRSID, similar to PSeg-SSDD, significantly reducing the number of false alarms in the detection results, as shown in Fig. 12.

The occurrence of false alarms, missed detections, mask overlap issues in the inshore dense ship region and land interference conditions depicted in Figs. 11 and 12 can be attributed to the network's failure to learn the correspondence between local strong scattering regions and the overall ship targets. The false alarms observed in most comparative methods indicate a lack of discriminative capacity of the network in distinguishing interference from ship targets. To address these issues, we employ a set of key points to represent the strong scattering regions of the targets and encode them implicitly into the dynamic mask head. In addition, we use 2-D Gaussian distribution heatmaps decoded from the key points to generate instancewise attention maps, further enhancing the network's perception of specific targets. From Figs. 11 and 12, it can be concluded that our proposed method achieves better detection results in the inshore scenes, with significant improvements in false alarms and mask overlap issues. This demonstrates the positive effect of introducing key points information in achieving higher accuracy in instance segmentation, validating the effectiveness of our proposed method.

V. Discussion

In this section, we set up an ablation experiment to verify the effectiveness of the DKPM and IAMM, and we conducted a series of experiments to investigate the impact of sampling number on the performance of the network. At last, we compared the model size, FLOPs, and the number of parameters of the proposed method with other methods.

A. Ablation Experiment

In order to demonstrate the effectiveness of the DKPM and the IAMM based on 2-D Gaussian heatmaps and quantitatively assess their impact on the network's performance, we conduct ablation experiments on both modules. The DKPM includes the

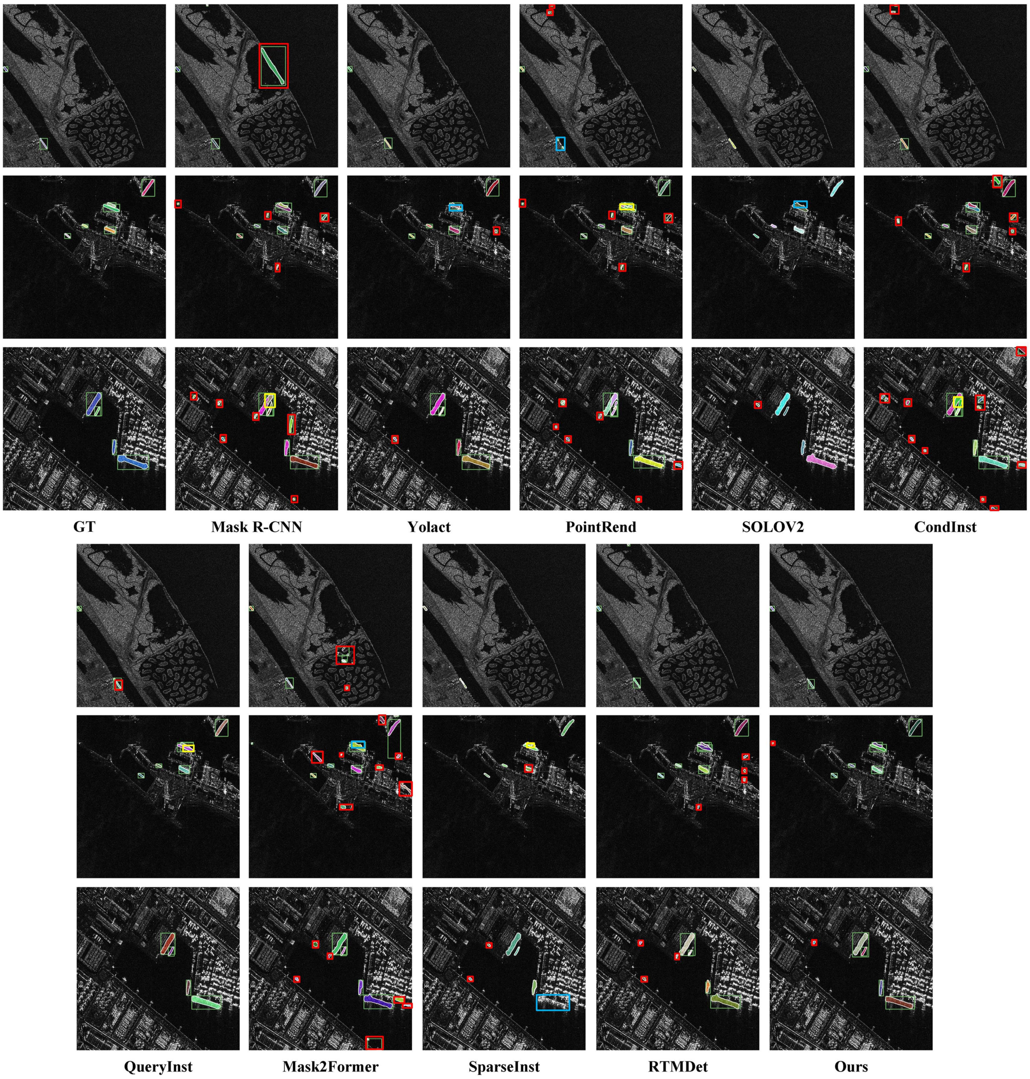


Fig. 12. Results of different methods in the inshore scenes of HRSID. The red rectangles represent false alarms, the blue rectangles represent missed detections, the yellow rectangles represent mask overlap, and green rectangles represent results of HBB detection (only some of these methods support detecting HBB).

TABLE IV
RESULTS OF ABLATION EXPERIMENTS ON THE PSEG-SSDD

DKPM	IAMM	Scene	AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L
–	–	offshore	0.691	0.988	0.881	0.662	0.752	0.786
		inshore	0.603	0.948	0.725	0.565	**0.686**	**0.674**
√	–	offshore	0.710	0.991	0.916	0.702	0.775	0.720
		inshore	0.621	0.956	0.748	0.624	0.650	0.571
–	√	offshore	0.703	0.986	0.905	0.691	0.775	0.732
		inshore	0.614	0.942	0.742	0.612	0.649	0.582
√	√	offshore	0.720	0.989	0.935	0.708	0.787	0.728
		inshore	**0.628**	**0.960**	**0.760**	**0.626**	0.663	0.590

Bold items denote the optimal inshore values in the columns, the underlined items indicate the optimal offshore values in the columns.

dynamic key points distance encoding as well as nonuniform sampling strategy modules. The results of the ablation experiments are given in Tables IV and V.

The results of the ablation experiments in Table IV demonstrate that the standalone application of IAMM leads to improvements of 1.2% and 1.1% in offshore and inshore scenes, respectively. Nevertheless, the standalone application of DKPM results in improvements of 1.9% and 1.6% in offshore and inshore scenes, respectively. It can be seen that the two modules mainly improve the detection accuracy for small and medium targets in the PSeg-SSDD. DKPM focuses on the strong scattering regions of the targets, and objects with different sizes often

TABLE V
RESULTS OF ABLATION EXPERIMENTS ON THE HRSID

DKPM	IAMM	Scene	AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L
–	–	offshore	0.658	0.966	0.822	0.653	0.729	0.680
		inshore	0.284	0.514	0.301	0.265	0.436	0.303
√	–	offshore	0.681	0.968	0.851	0.674	0.752	0.683
		inshore	0.332	0.582	0.360	0.313	0.468	**0.326**
–	√	offshore	0.670	0.969	0.841	0.665	0.732	0.709
		inshore	0.314	0.576	0.339	0.294	0.470	0.329
√	√	offshore	0.695	0.969	0.876	0.688	0.754	0.731
		inshore	**0.361**	**0.686**	**0.384**	**0.337**	**0.526**	0.322

Bold items denote the optimal inshore values in the columns, the underlined items indicate the optimal offshore values in the columns.

exhibit different scattering characteristics in imaging. On the other hand, IAMM guides the network to pay more attention to specific target. Therefore, both DKPM and IAMM are instance-wise feature enhancement modules that are affected by data imbalance problem. The number of targets with different sizes in the PSeg-SSDD is quite different. Small targets predominate in both inshore and offshore scenes, whereas large targets account for a small proportion. The number of large targets in inshore and offshore scenes is 19 and 52, accounting for 3.58% and 2.54% of the total targets in the two scenes, respectively. The ratio of small targets to medium targets is also large, nearly 4:1 (2009 small targets to 507 medium targets). The problem of imbalanced data in the PSeg-SSDD leads the two modules to primarily focus on the key points features of small targets, while not receiving equally sufficient training on large targets. From the table, it is evident that both modules significantly improve the detection performance of small targets, while the improvement for large targets is inapparent. In addition, due to the small amount of large targets in PSeg-SSDD, metric may suffer from larger fluctuations. However, considering the overall precision for small and medium targets, both modules contribute significantly to the accuracy of the network.

The results of ablation experiments on the HRSID are presented in Table V. In contrast to the PSeg-SSDD, the distribution of samples with various sizes in HRSID is more balanced. In inshore and offshore scenes, the quantities of small and medium targets are roughly equal, while there is also a certain number of large targets for the network to learn from. Therefore, we conducted additional ablation experiments on the HRSID dataset as a supplement to the experiments on the PSeg-SSDD dataset. From Table V, it can be observed that the standalone application of IAMM or DKPM leads to improvements across all metrics in inshore and offshore scenes. Specifically, using IAMM alone achieves a 3.0% and 1.2% improvement in AP metrics in inshore and offshore scenes, respectively, whereas using DKPM alone results in a 4.8% and 2.3% improvement in AP metrics in inshore and offshore scenes, respectively. In the PSeg-SSDD, the improvement brought by the use of both IAMM and DKPM is mainly reflected in the small and medium targets detection. However, in the HRSID, the use of both two modules not only improves the detection accuracy of small and medium targets, but also achieves the best performance of large targets in offshore scenes among the ablation experiments. For large target detection in inshore scenes, using DKPM alone achieves the best performance in ablation experiments, but the gap of AP_L metrics is within 0.5% in the experiments using single and both modules. It can be seen that due to the more balanced and diverse samples of different sizes in HRSID, the IAMM and DKPM exhibit positive effects on the detection accuracy of large target samples, which further validates the effectiveness of these two modules.

By comparing the improvement of the two modules, it can be observed that DKPM has a stronger effect than IAMM, especially in terms of accuracy at high IoU thresholds. For example, the AP_{75} metric is improved by 2.3% and 3.5% in inshore and offshore scenes of PSeg-SSDD, respectively, which is close to the performance achieved by using both modules simultaneously. This indicates that DKPM is the key to enhancing the network's fine-grained detection capability and demonstrates the feasibility of introducing dynamic key points information for improving fine-grained detection in instance segmentation tasks. Between the two modules, DKPM has the higher complexity, and IAMM requires the predicted results from DKPM as input, allowing the strong scattering region information obtained in the detection branch to be further enhanced in the mask branch. Thus, the two modules complement each other. Based on the results of the ablation experiments of both modules, it can be concluded that using both modules simultaneously can provide the network with more fine-grained segmentation performance.

B. Influence of Different Sampling Number

The number of sampling points N in the key points sampling process is an important hyperparameter in our method. We conducted experiments to investigate the impact of different values of N on the network's performance, and the results are given in Table VI.

Table VI demonstrates the impact of different numbers of sampling points on network performance. When $N = 5$, the network shows a smaller improvement compared with the experiment with $N = 7$. This is because a lower number of samples is insufficient to adequately represent the strong scattering regions of the bow, leading to ineffective guidance for network training. In addition, during the decoding process of key points detection results to obtain instancewise heatmaps, a smaller number of sampling points may result in increased distortion when fitting the ellipse, failing to accurately cover the distribution area of the

TABLE VI
INSTANCE SEGMENTATION PERFORMANCE OF DIFFERENT SAMPLING NUMBER ON PSEG-SSDD

N	Scene	AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L
–	offshore	0.691	0.988	0.881	0.662	0.752	0.786
	inshore	0.603	0.948	0.725	0.565	**0.686**	**0.674**
5	offshore	0.711	0.989	0.904	0.702	0.775	0.720
	inshore	0.617	0.942	0.752	0.616	0.650	0.561
7	offshore	0.720	0.989	0.935	0.708	0.787	0.728
	inshore	**0.628**	0.960	**0.760**	**0.626**	0.663	0.590
9	offshore	0.716	0.989	0.923	0.706	0.780	0.735
	inshore	0.623	0.961	0.756	0.624	0.651	0.571
11	offshore	0.718	0.990	0.928	0.705	0.785	0.755
	inshore	0.625	**0.963**	0.756	0.620	0.661	0.594

Bold items denote the optimal inshore values in the columns, the underlined items indicate the optimal offshore values in the columns.

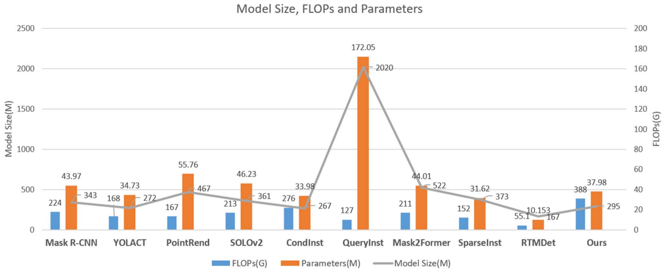


Fig. 13. Comparison of model size, FLOPs, and the number of parameters across different methods.

ship targets. As the number increases, the network's performance remains stable. Overall, selecting $N = 7$ as the number of sampling points maintains good performance in both scenes while achieving higher efficiency. Hence, in this article, we choose 7 as the final hyperparameter for the number of sampling points.

C. Computational Efficiency

To discuss the computational efficiency of different methods, we compute the size of the models, FLOPs, and the number of parameters used during the testing process, as shown in Fig. 13. Due to the network architecture of QueryInst, which includes one query-based detection branch and six parallel-supervised dynamic mask heads, it has significantly higher number of parameters and model size compared with other methods. However, QueryInst requires less computational resources during the inference stage compared with all other methods except RTMDet. The FLOPs and number of parameters of RTMDet are significantly smaller than other methods, so it is suitable for deployment on devices with limited computing resources. Since our network incorporates dynamic key points related detection branches, such as angle branch and key points distance encoding branch, there is a slight increase in the number of parameters and model size compared with CondInst. However, the number of parameters and model size of our method remain close to those of other comparative methods. In terms of FLOPs, our method has a 40% increase compared with CondInst. The additional computations are mainly for the dynamic key points prediction and the encoding and decoding processes, which is the cost of improving the segmentation precision of the network. Overall, the experimental results demonstrate that although the introduction of dynamic key points prediction branches and encoding and decoding processes increases the computational complexity, the increase in number of parameters and model size is relatively small, and FLOPs of our method remains close to that of other comparative methods. Despite the higher computational complexity, it achieves higher detection precision, making it suitable for applications that require fine-grained detection.

VI. CONCLUSION

In this article, we propose a SAR ship image instance segmentation method based on key points information enhancement. To overcome the issue of mask overlap in inshore dense ship areas, as well as the problem of excessive false alarms caused by land clutter interference in inshore scenes, we propose the DKPM that encodes the strong scattering region of bow as a set of dynamic key points and incorporate key points information into the implicit encoding process, leveraging the powerful feature representation capability of implicit encoding to encode the size, shape, and strong scattering region information of specific targets into the parameters of dynamic mask head. In addition, to further enhance the key points information, we propose an IAMM based on 2-D Gaussian distribution, improving the network's perception of specific targets. Experimental results demonstrate that our proposed method achieves better performance compared with other SOTA instance segmentation methods, particularly in terms of fine-grained metrics at high IoU thresholds, validating the effectiveness of our method.

REFERENCES

[1] Z. Wu, B. Hou, B. Ren, Z. Ren, S. Wang, and L. Jiao, "A deep detection network based on interaction of instance segmentation and object detection for SAR images," *Remote Sens.*, vol. 13, no. 13, 2021, Art. no. 2582.

[2] W. Ao, F. Xu, Y. Li, and H. Wang, "Detection and discrimination of ship targets in complex background from spaceborne ALOS-2 SAR images," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 11, no. 2, pp. 536–550, Feb. 2018.

[3] Y. Zhao, L. Zhao, B. Xiong, and G. Kuang, "Attention receptive pyramid network for ship detection in SAR images," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 13, pp. 2738–2756, 2020.

[4] Y. Yao et al., "On improving bounding box representations for oriented object detection," *IEEE Trans. Geosci. Remote Sens.*, vol. 61, 2023, Art. no. 5600111.

[5] K. Fu, Z. Chang, Y. Zhang, and X. Sun, "Point-based estimator for arbitrary-oriented object detection in aerial images," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 5, pp. 4370–4387, May 2021.

[6] Z. Zhang, G. Ni, and Y. Xu, "Ship trajectory prediction based on LSTM neural network," in *Proc. IEEE 5th Inf. Technol. Mechatronics Eng. Conf.*, 2020, pp. 1356–1364.

[7] D. Zhao, C. Zhu, J. Qi, X. Qi, Z. Su, and Z. Shi, "Synergistic attention for ship instance segmentation in SAR images," *Remote Sens.*, vol. 13, no. 21, 2021, Art. no. 4384.

[8] K. He, G. Gkioxari, P. Dollar, and R. Girshick, "Mask R-CNN," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 02, pp. 386–397, Feb. 2020.

[9] Z. Cai and N. Vasconcelos, "Cascade R-CNN: High quality object detection and instance segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 5, pp. 1483–1498, May 2021.

[10] K. Chen et al., "Hybrid Task Cascade for instance segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 4974–4983.

[11] H.-S. Fang, J. Sun, R. Wang, M. Gou, Y.-L. Li, and C. Lu, “InstaBoost: Boosting instance segmentation via probability map guided copy-pasting,” in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2019, pp. 682–691.

[12] W. Xu, H. Wang, F. Qi, and C. Lu, “Explicit shape encoding for real-time instance segmentation,” in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2019, pp. 5168–5177.

[13] E. Xie et al., “PolarMask: Single shot instance segmentation with polar representation,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 12193–12202.

[14] H. U. M. Riaz, N. Benbarka, and A. Zell, “FourierNet: Compact mask representation for instance segmentation using differentiable shape decoders,” in *Proc. IEEE 25th Int. Conf. Pattern Recognit.*, 2021, pp. 7833–7840.

[15] E. Xie, W. Wang, M. Ding, R. Zhang, and P. Luo, “PolarMask: Enhanced polar representation for single-shot instance segmentation and beyond,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 9, pp. 5385–5400, Sep. 2022.

[16] Z. Huang, S. Sun, and R. Li, “Fast single-shot ship instance segmentation based on polar template mask in remote sensing images,” in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, 2020, pp. 1236–1239.

[17] D. Bolya, C. Zhou, F. Xiao, and Y. J. Lee, “YOLACT: Real-time instance segmentation,” in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2019, pp. 9157–9166.

[18] H. Chen, K. Sun, Z. Tian, C. Shen, Y. Huang, and Y. Yan, “BlendMask: Top-down meets bottom-up for instance segmentation,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 8573–8581.

[19] Z. Tian, B. Zhang, H. Chen, and C. Shen, “Instance and panoptic segmentation using conditional convolutions,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 1, pp. 669–680, Jan. 2023.

[20] X. Wang, R. Zhang, T. Kong, L. Li, and C. Shen, “SOLOv2: Dynamic and fast instance segmentation,” *Adv. Neural Inf. Process. Syst.*, vol. 33, pp. 17721–17732, 2020.

[21] F. Ma, X. Sun, F. Zhang, Y. Zhou, and H.-C. Li, “What catch your attention in SAR images: Saliency detection based on soft-superpixel lacunarity cue,” *IEEE Trans. Geosci. Remote Sens.*, vol. 61, 2022, Art. no. 5200817.

[22] H. Huang, F. Gao, J. Wang, A. Hussain, and H. Zhou, “An incremental SAR target recognition framework via memory-augmented weight alignment and enhancement discrimination,” *IEEE Geosci. Remote Sens. Lett.*, vol. 20, 2023, Art. no. 4005205, doi: 10.1109/LGRS.2023.3269480.

[23] Z. Yue et al., “A novel semi-supervised convolutional neural network method for synthetic aperture radar image recognition,” *Cogn. Comput.*, vol. 13, pp. 795–806, 2021.

[24] Y. Zhou, F. Zhang, Q. Yin, F. Ma, and F. Zhang, “Inshore dense ship detection in SAR images based on edge semantic decoupling and transformer,” *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 16, pp. 4882–4890, 2023, doi: 10.1109/JSTARS.2023.3277013.

[25] F. Gao et al., “SAR target incremental recognition based on features with strong separability,” *IEEE Trans. Geosci. Remote Sens.*, vol. 62, 2024, Art. no. 5202813, doi: 10.1109/TGRS.2024.3351636.

[26] H. Huang, F. Gao, J. Sun, J. Wang, A. Hussain, and H. Zhou, “Novel category discovery without forgetting for automatic target recognition,” *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 17, pp. 4408–4420, 2024, doi: 10.1109/JSTARS.2024.3358449.

[27] X. Yang, L. Hou, Y. Zhou, W. Wang, and J. Yan, “Dense label encoding for boundary discontinuity free rotation detection,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 15819–15829.

[28] Z. Li, B. Hou, Z. Wu, L. Jiao, B. Ren, and C. Yang, “FCOSR: A simple anchor-free rotated detector for aerial object detection,” *Remote Sens.*, vol. 15, no. 23, 2023, Art. no. 5499.

[29] J. Li, C. Qu, and J. Shao, “Ship detection in SAR images based on an improved Faster R-CNN,” in *Proc. IEEE SAR Big Data Era: Models, Methods Appl.*, 2017, pp. 1–6.

[30] T. Zhang et al., “SAR ship detection dataset (SSDD): Official release and comprehensive data analysis,” *Remote Sens.*, vol. 13, no. 18, 2021, Art. no. 3690.

[31] S. Wei, X. Zeng, Q. Qu, M. Wang, H. Su, and J. Shi, “HRSID: A high-resolution SAR images dataset for ship detection and instance segmentation,” *IEEE Access*, vol. 8, pp. 120234–120254, 2020.

[32] R. Girshick, “Fast R-CNN,” in *Proc. IEEE Int. Conf. Comput. Vis.*, 2015, Art. no. 1440.

[33] Z. Huang, L. Huang, Y. Gong, C. Huang, and X. Wang, “Mask scoring R-CNN,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 6409–6418.

[34] G. Cheng, P. Lai, D. Gao, and J. Han, “Class attention network for image recognition,” *Sci. China Inf. Sci.s*, vol. 66, no. 3, 2023, Art. no. 132105.

[35] X. Yang, Q. Zhang, Q. Dong, Z. Han, X. Luo, and D. Wei, “Ship instance segmentation based on rotated bounding boxes for SAR images,” *Remote Sens.*, vol. 15, no. 5, 2023, Art. no. 1324.

[36] X. Ke, X. Zhang, and T. Zhang, “GCBANet: A global context boundary-aware network for SAR ship instance segmentation,” *Remote Sens.*, vol. 14, no. 9, 2022, Art. no. 2165.

[37] T. Zhang and X. Zhang, “A full-level context squeeze-and-excitation ROI extractor for SAR ship instance segmentation,” *IEEE Geosci. Remote Sens. Lett.*, vol. 19, 2022, Art. no. 4506705.

[38] T. Zhang and X. Zhang, “A mask attention interaction and scale enhancement network for SAR ship instance segmentation,” *IEEE Geosci. Remote Sens. Lett.*, vol. 19, 2022, Art. no. 4511005.

[39] T.-Y. Lin, P. Dollar, R. Girshick, K. He, B. Hariharan, and S. Belongie, “Feature pyramid networks for object detection,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 936–944.

[40] X. Ma, S. Hou, Y. Wang, J. Wang, and H. Wang, “Multiscale and dense ship detection in SAR images based on key-point estimation and attention mechanism,” *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5221111.

[41] Y. Sun, X. Sun, Z. Wang, and K. Fu, “Oriented ship detection based on strong scattering points network in large-scale SAR images,” *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2021, Art. no. 5221111.

[42] J. Yi, P. Wu, B. Liu, Q. Huang, H. Qu, and D. Metaxas, “Oriented object detection in aerial images with box boundary-aware vectors,” in *Proc. IEEE/CVF Winter Conf. Appl. Comput. Vis.*, 2021, pp. 2150–2159.

[43] Y. He, F. Gao, J. Wang, A. Hussain, E. Yang, and H. Zhou, “Learning polar encodings for arbitrary-oriented ship detection in SAR images,” *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 14, pp. 3846–3859, 2021.

[44] F. Gao, Y. Huo, J. Sun, T. Yu, A. Hussain, and H. Zhou, “Ellipse encoding for arbitrary-oriented SAR ship detection based on dynamic key points,” *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5240528.

[45] J. Ge, Y. Tang, K. Guo, Y. Zheng, H. Hu, and J. Liang, “KeyShip: Towards high-precision oriented SAR ship detection using key points,” *Remote Sens.*, vol. 15, no. 8, 2023, Art. no. 2035.

[46] Y. Zhang, D. Lu, X. Qiu, and F. Li, “Scattering-point-guided RPN for oriented ship detection in SAR images,” *Remote Sens.*, vol. 15, no. 5, 2023, Art. no. 1411.

[47] 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.

[48] Z. Tian, C. Shen, H. Chen, and T. He, “FCOS: Fully convolutional one-stage object detection,” in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2019, pp. 9627–9636.

[49] X. Yang and J. Yan, “Arbitrary-oriented object detection with circular smooth label,” in *Proc. Comput. Vis.–ECCV: 16th Eur. Conf.*, 2020, pp. 677–694.

[50] R. Girshick, “Fast R-CNN,” in *Proc. IEEE Int. Conf. Comput. Vis.*, 2015, pp. 1440–1448.

[51] F. Milletari, N. Navab, and S.-A. Ahmadi, “V-Net: Fully convolutional neural networks for volumetric medical image segmentation,” in *Proc. IEEE 4th Int. Conf. 3D Vis.*, 2016, pp. 565–571.

[52] K. Chen et al., “MMDetection: Open MMLAB detection toolbox and benchmark,” 2019, *arXiv:1906.07155*.

[53] A. Kirillov, Y. Wu, K. He, and R. Girshick, “PointRend: Image segmentation as rendering,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 9799–9808.

[54] Y. Fang et al., “Instances as queries,” in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2021, pp. 6910–6919.

[55] B. Cheng, I. Misra, A. G. Schwing, A. Kirillov, and R. Girdhar, “Masked-attention mask transformer for universal image segmentation,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 1290–1299.

[56] T. Cheng et al., “Sparse instance activation for real-time instance segmentation,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 4433–4442.

[57] C. Lyu et al., “RTMDET: An empirical study of designing real-time object detectors,” 2022, *arXiv:2212.07784*.

[58] Z. Tian, H. Chen, X. Wang, Y. Liu, and C. Shen, “AdelaiDet: A toolbox for instance-level recognition tasks,” 2019. [Online]. Available: https://git.io/adelaidet

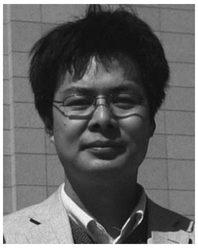
[59] T.-Y. Lin et al., “Microsoft COCO: Common objects in context,” in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 740–755.

Fei Gao received the B.S. degree in industrial electrical automation and the M.S. degree in electromagnetic measurement technology and instrument from Xi'an Petroleum Institute, Xi'an, China, in 1996 and 1999, respectively, and the Ph.D. degree in signal and information processing from Beihang University, Beijing, China, in 2005.

He is currently a Professor with the School of Electronic and Information Engineering, Beihang University. His research interests include target detection and recognition, image processing, and deep learning for applications in remote sensing.

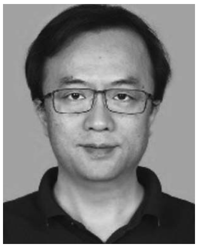
Xu Han received the B.S. degree in electronic and information engineering in 2021 from Beihang University, Beijing, China, where he is currently working toward the M.S. degree in information and communication engineering.

His current research interests include target detection and synthetic aperture radar image processing.



Jun Wang received the B.S. degree in communication engineering from North western Polytechnical University, Xi'an, China, in 1995, and the M.S. and Ph.D. degrees in signal and information processing from the Beijing University of Aeronautics and Astronautics (BUAA), Beijing, China, in 1998 and 2001, respectively.

He is currently a Professor with the School of Electronic and Information Engineering, BUAA. His research has resulted in more than 40 papers in journals, books, and conference proceedings. His research interests include signal processing, DSP/FPGA real-time architecture, target recognition and tracking, and so on.



Jinping Sun (Member, IEEE) received the M.Sc. and Ph.D. degrees in signal and information processing from Beihang University (BUAA), Beijing, China, in 1998 and 2001, respectively.

He is currently a Professor with the School of Electronic and Information Engineering, BUAA. His research interests include statistical signal processing, high-resolution radar signal processing, target tracking, image understanding, and robust beamforming.

Amir Hussain received the B.Eng. and Ph.D. degrees in electronic and electrical engineering from the University of Strathclyde, Scotland, U.K., in 1992 and 1997, respectively.

Following Postdoctoral and Senior Academic positions with the West of Scotland (1996–98), Dundee (1998–2000) and Stirling Universities (2000–2018), respectively, he joined Edinburgh Napier University as Founding Head of the Cognitive Big Data and Cybersecurity Research Lab and the Centre for AI and Data Science. His research interests include cognitive computation,machine learning and computer vision.

Huiyu Zhou received the B.Eng. degree in radio technology from the Huazhong University of Science and Technology, Wuhan, China, in 1990, the M.S. degree in biomedical engineering from the University of Dundee, Dundee, U.K., in 2002, and the Ph.D. degree in computer vision from Heriot-Watt University, Edinburgh, U.K., in 2006.

He is currently a Full Professor with the School of Computing and Mathematical Sciences, University of Leicester, Leicester, U.K. He has authored or coauthored over 380 peer reviewed papers in the field.