



Arbitrary-oriented ship detection based on Kullback-Leibler divergence regression in remote sensing images

Yantong Chen¹ · Jialiang Wang¹ · Yanyan Zhang¹ · Yang Liu¹

Received: 12 April 2023 / Accepted: 21 August 2023 / Published online: 31 August 2023
© The Author(s), under exclusive licence to Springer-Verlag GmbH Germany, part of Springer Nature 2023

Abstract

Ship detection is a meaningful and challenging task in processing of remote sensing image data. Specifically, the main challenges faced by ship detection in remote sensing images (RSIs) include the variable direction and dense arrangement of ships, as well as the complex nearshore scenes. To address the above challenges, this article proposes an arbitrary-oriented ship detection network based on Kullback-Leibler divergence (KLD) regression. Firstly, a coordinate attention module (CAM) is designed to capture direction-aware and position-sensitive features, which enhances the attention to ships in complex scenes. Meanwhile, a reinforced feature fusion network (RFF-Net) combined with CAM is constructed to obtain fusion features containing rich semantic and location information. Then, the orientation-invariant model (OIM) is introduced to generate depth rotation-invariant features, which improves the adaptability of the network to arbitrary-oriented ships. Finally, a regression loss function based on KLD is defined to solve the problem of boundary discontinuity and realize the accurate detection of densely arranged ships. The experimental results demonstrate that the proposed method reaches 89.87% and 83.62% average precision (AP) on the HRSC2016 and DOTA Ship data sets, respectively, achieving state-of-the-art ship detection performance.

Keywords Reinforced feature fusion network · Orientation-invariant model · Kullback-Leibler divergence · Ship detection · Remote sensing images (RSIs)

Introduction

With the vigorous development of deep learning and optical remote sensing technology, the study of optical remote sensing images (RSIs) has attracted extensive attention. Ship positioning and monitoring play a vital role in civil and military affairs such as ocean environment perception,

enemy reconnaissance, and maritime management. It is a key research subject in the field of RSI intelligent processing (Deng et al. 2019). Due to the complexity of scenes and the difference of ships, it is undoubtedly a great challenge to accurately and quickly detect ships from optical RSIs.

At present, ship detection in optical RSIs is faced with the following three difficulties.

- (1) Dense arrangement: As shown in Fig. 1(a), port ships are usually densely arranged, and the intersection over union (IoU) between ship bounding boxes is more sensitive to angle changes due to the large aspect ratio.
- (2) Variable direction: As shown in Fig. 1(b), ships in RSIs may appear in any direction, which requires the detector to have accurate angle prediction ability.
- (3) Complex background: The background is complex. The detection of nearshore ships is easily disturbed by the complex background. As shown in Fig. 1(c), the container area on shore is easily confused with the cargo ship on shore.

Communicated by: H. Babaie

✉ Yantong Chen
chenyantong@dmlu.edu.cn

Jialiang Wang
wangjialiang@dmlu.edu.cn

Yanyan Zhang
zhangyanyan999@dmlu.edu.cn

Yang Liu
ly1120211369@dmlu.edu.cn

¹ Department of Information Science and Technology, Dalian Maritime University, Dalian 116026, China

Fig. 1 Difficulties in remote sensing ship detection. (a) Dense arrangement. (b) Variable direction. (c) Complex background



Traditional ship detection methods mostly rely on hand-designed features, and require a lot of prior knowledge to set many parameters in the algorithm, which has high complexity. Shuai et al. (2016) first selected the threshold through the binary segmentation algorithm to maximize the differences between objects and backgrounds. Then ships are extracted accurately through morphological operation, feature calculation and object identification. Song et al. (2014) presented an approach based on biological visual heuristic features, which combines local binary pattern (LBP) and visual saliency mechanism to focus on ship detection in complex background. Li-Bing et al. (2011) described the outline of offshore ships by using the curve of variable angle chain, which is invariant to rotation, scaling, and translation. This method can promote the detection accuracy of offshore ships to a certain degree. Zhu et al. (2010) proposed a hierarchical and operable ship detection method based on shape and texture features. It uses simple shape analysis to eliminate false region proposals, so as to extract ship region proposals with as few missed detections as possible. Finally, the ship is detected by the semi-supervised classification method combining multiple features. Although these traditional methods claim to achieve fine detection results, the detection performance in complex scenes is unsatisfactory.

In the past decade, deep learning has been deeply applied in remote sensing object detection, and various new and reliable ship detection methods emerge in endlessly. Nie et al. (2020) improved the Mask R-CNN and obtained a ship detection and segmentation network with better detection effect, which uses spatial and channel attention mechanisms to adjust the weights of each pixel and channel, respectively. Therefore, the object features can obtain better response in the feature map. For the specific aspect ratio and arrangement of ships in RSIs, Zhao et al. (2020) reset the proportion and number of anchors, which effectively improves the speed and accuracy of ship detection. Shi et al. (2020) added deconvolution and pooling feature modules on the basis of SSD to fuse deep and shallow features, which strengthened the correlation between object features and promoted the detection accuracy of the network. Chen et al. (2021) improved

YOLOv3 by using the attentional mechanism. By using a lightweight expansive attentional module, significant features for ship detection tasks are extracted to achieve the optimal balance between detection accuracy and speed. All of these methods directly transfer the general object detection method to ship detection. However, in the case of dense arrangement of ships, labeling the ships with horizontal boxes will result in partial overlap of objects. Moreover, ships usually have a large aspect ratio, and horizontal box labeling will lead to limited detection accuracy.

To solve the problems when using horizontal boxes to label ships, Zhong et al. (2019) adopted the rotated boxes for ship detection. By introducing the feature pyramid pooling module into the rotation region of interest (RRoI), the precise positioning of ships is achieved. Yang et al. (2018) designed a rotating dense feature pyramid network (FPN) to improve the efficiency of feature fusion, which can accurately detect ships in various scenes. Liu et al. (2018) proposed a network for detecting arbitrary-oriented ships, which improves the detection effect of small ships by extracting fine-grained features. Moreover, the network adds angle information to the bounding box regression, so that the detection box can accurately locate the entire ship area. Li et al. (2020) presented a multi-level adaptive pooling based on spatial variables, which can enable the network to obtain more appropriate ship features. Furthermore, the application of double branch regression network makes the angle and other variables can be predicted independently, which undoubtedly improves the accuracy of ship positioning. Han et al. (2021) designed a two-way dense feature fusion network, which can maximize the use of multi-layer features. The dual mask attention module can refine the fused features to improve the detection performance in dense scenes. Most of these methods are two-stage rotating object detection algorithms with high detection accuracy. Since a great quantity of anchors with different scales and angles are set, these detection networks have the problems of many parameters, large amount of calculation and slow detection speed. Moreover, these methods all adopt regression method based on five-parameters, in which the angle regression will face the problem of boundary discontinuity. Therefore, it

is difficult to achieve high-precision arbitrary-oriented ship detection.

Given the difficulties in the field of remote sensing ship detection, and the limitations of current ship detection methods. In this article, we propose a single-stage Kullback-Leibler divergence regression network (KRNet) based on RetinaNet (Lin et al. 2017) for arbitrary-oriented ship detection in RSIs. Specifically, the main work and contributions of this article are concluded as follows.

- (1) A reinforced feature fusion network (RFF-Net) combining coordinate attention module (CAM) is constructed to extract fusion features containing rich semantic and location information. The CAM can enhance the feature representation of ships, so as to detect ships in complex backgrounds more accurately.
- (2) The orientation-invariant model (OIM) is introduced to generate depth rotation-invariant features, which can improve the adaptability of the network to arbitrary-oriented ships.
- (3) A regression loss function based on Kullback-Leibler divergence (KLD) is proposed, which can solve the problem of boundary discontinuity and promote the scale invariance of the network. By using KLD loss, high-precision detection of arbitrary-oriented ships can be realized, especially for densely arranged ships.

The remaining of this article is structured as follows. Section 2 introduces the related works of object detection, including general object detection and rotating object detection. Section 3 describes the details of the proposed ship detection network KRNet. The experimental results are reported and analyzed in Section 4. Finally, the conclusion of this article is given in Section 5.

Related work

General object detection

At present, there are mainly two kinds of general object detection algorithms, namely single-stage and two-stage object detection algorithms. The two-stage object detection algorithm products region proposals using corresponding visual features and classifies each region of interest (RoI). As the foundation work of the two-stage object detection algorithm, R-CNN (Girshick et al. 2014) takes the lead in using selective search to product bounding boxes and extracts RoI features through convolutional neural network (CNN). SPP-Net (He et al. 2015) proposes spatial pyramid pooling (SPP), which directly extracts fixed-size features from the feature map for the classification and regression of the candidates. Faster R-CNN (Ren et al. 2015) innovatively replaces selective search

with region proposal network (RPN), so that the detection process shares the convolution features of all images. RPN predicts both object bounding box and category confidence at each location, which speeds up the inference speed. R-FCN (Dai et al. 2016) proposes a location sensitive score map to weigh the translation sensitivity of the network. Besides, full convolution network (FCN) is used to enhance feature extraction and improve classification effect.

The single-stage object detection algorithm can predict the location and category of the object simultaneously, which is the essential difference between it and two-stage object detection algorithm. YOLO (Redmon et al. 2016) divides the input image into multiple grid cells and directly locates and classifies the objects in each grid cell, which significantly improves the detection speed. SSD (Liu et al. 2016) detects objects independently on different level feature maps. Specifically, low-level and high-level feature maps are used to detect small and large objects, respectively. This simple optimization strategy is actually very useful for small object detection. RefineDet (Zhang et al. 2018) proposes an anchor refinement module, which performs quadratic regression on the bounding box to further promote the detection accuracy. DSSD (Fu et al. 2017) uses the deconvolution module to strengthen the feature extraction of the network and introduces residual units to optimize the detection effect.

Rotating object detection

For the directionality of text objects and remote sensing objects, a variety of rotating object detection algorithms are developed, which can simultaneously predict the location, size and angle of objects. RRPN (Ma et al. 2018) first borrows the structure of RPN and adds the setting of anchor angle in RPN. It also presents RRoI-Pooling to obtain the object features of region proposals, which ensures the efficiency of text detection. RRCNN (Liu et al. 2017) extracts rotating object features through RRoI-Pooling and uses multi-task non-maximum suppression (NMS) for multi category objects. To reduce computational complexity, RoI-Transformer (Ding et al. 2019) designs an RRoI learning mechanism that uses horizontal anchors to explore rotation information. CAD-Net (Zhang et al. 2019) proposes global context network and pyramid local context network, which are used to capture global scene and local object context information, respectively. Moreover, a spatial attention mechanism is introduced to focus on regions with more important information.

The single-stage object detection algorithm also performs well in rotating object detection. EAST (Zhou et al. 2017) uses FCN and NMS to greatly improve detection efficiency. To accommodate text regions in all directions, both horizontal and rotated anchors are adopted. TextBoxes++ (Liao et al. 2018) uses horizontal and rotated rectangular boxes to label

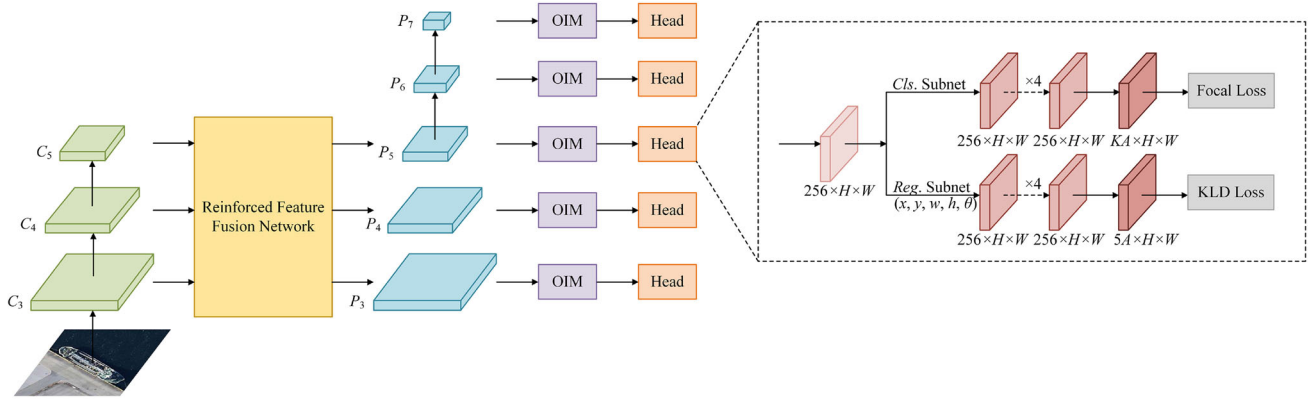


Fig. 2 Architecture of KRNet

multi-oriented text and introduces a long convolution kernel to accommodate slender text lines. R³Det (Yang et al. 2021) adopts rotated anchors during the feature refinement stage to accommodate dense scenes. Meanwhile, a feature thinning module is designed to reconstruct and align features through feature interpolation, which greatly promotes the detection accuracy. To accommodate the periodicity of angles, CSL (Yang and Yan 2020) transforms the regression task of angles into a classification task. Although good detection results have been obtained, the method still has problems such as unbalanced angle classes, many parameters and large loss.

Proposed method

In this section, we exhaustively describe the architecture of KRNet. The architecture of KRNet is shown in Fig. 2, which is composed of backbone, RFF-Net, OIM, and prediction network. The backbone is ResNet50 (He et al. 2016), which uses the residual structure to solve the gradient disappearance problem during deep network training. Based on FPN (Lin et al. 2017), RFF-Net embedded in CAM was constructed to enhance the process of feature fusion and the representation of the objects. OIM (Zhou et al. 2017) can generate

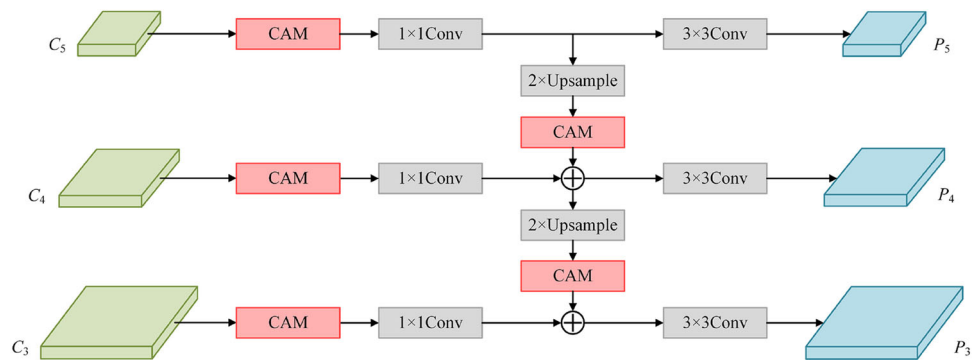
depth rotation-invariant features to effectively promote the detection ability of rotating objects. The prediction network classifies and regresses the objects, with KLD Loss as the regression loss function to achieve high-precision detection of rotating objects.

Reinforced feature fusion network

CNNs usually needs to carry out multiple downsampling to obtain different levels of feature maps, and then complete the task of multi-scale object detection. The high-level feature map has low resolution, large receptive field, and strong semantic information. But it contains weak location information, which may lead to the loss of small objects. On the contrary, the low-level feature map has high resolution, small receptive field, and weak semantic information. However, thanks to its strong location information, it is beneficial to small object detection. Therefore, in order to obtain feature maps containing rich semantic and location information, it is necessary to strengthen the process of multi-level feature fusion.

RFF-Net is built by skillfully embedding CAM in FPN, and its structure is shown in Fig. 3. After an input image is processed by the feature extraction network ResNet50, three-stage feature maps $\{C_3, C_4, C_5\}$ with the number of channels

Fig. 3 Structure of RFF-Net



{512, 1024, 2048} are obtained. For the above feature maps, a top-down feature fusion strategy is adopted. Firstly, feature map $C_5/C_4/C_3$ is input into CAM to capture spatial and channel information, and then feature map $\bar{P}_5/\bar{P}_4/\bar{P}_3$ with 256 channels is obtained after a 1×1 convolution. After a 3×3 convolution, feature map \bar{P}_5 is converted to feature map P_5 . Feature map \bar{P}_5/\bar{P}_4 is added to feature map \bar{P}_4/\bar{P}_3 after up-sampling and CAM, and feature map P_4/P_3 is obtained after a 3×3 convolution. Finally, fusion feature maps $\{P_3, P_4, P_5\}$ with 256 channels are obtained. Furthermore, to further strengthen the network’s ability to detect multi-scale objects, feature map P_5 is downsampled twice successively to obtain feature maps P_6 and P_7 . The above feature fusion process is described as follows:

$$\bar{P}_l = \text{Conv}_{1 \times 1}(\text{CAM}(C_l)), l = 3, 4, 5 \tag{1}$$

$$P_l = \begin{cases} \text{Conv}_{3 \times 3}(\text{CAM}(\text{Upsample}(\bar{P}_{l+1})) + \bar{P}_l), & l = 3, 4 \\ \text{Conv}_{3 \times 3}(\bar{P}_l), & l = 5 \\ \text{Downsample}(P_{l-1}), & l = 6, 7 \end{cases} \tag{2}$$

where $\text{Conv}_{1 \times 1}$ denotes 1×1 convolution, and $\text{Conv}_{3 \times 3}$ denotes 3×3 convolution. CAM represents coordinate attention module.

Traditional attention mechanisms usually only consider spatial or channel attention. Even if both spatial and channel attention are considered, they cannot be effectively combined. CAM can capture cross-channel information as well as orientation and location sensitive information. Considering that the ship has long and narrow shape features and obvious direction features, CAM is introduced into FPN. This method can enhance the feature representation of ships, so as to locate

and identify ships more accurately. The application of CAM can greatly improve the detection effect of ships under complex background. The structure of CAM is shown in Fig. 4. The calculation process of CAM can be roughly described as follows. Firstly, the feature map is orthogonally pooled in the spatial dimension to obtain orientation and location sensitive feature vectors. Secondly, the two orthogonal feature vectors are concatenated and encoded to integrate cross-channel information. Finally, the encoded feature vector is split and decoded to apply attention weight.

Specifically, for an input map $F_{in} \in \mathbb{R}^{C \times H \times W}$, where C denotes the number of channels. H and W denote height and width, respectively. By averaging pooling along orthogonal directions in the spatial dimension, we obtain two orthogonal feature vectors containing orientation and location sensitive information, which can be calculated as follows:

$$v_x(k, i) = \frac{1}{W} \sum_{i=1}^W F_{in}(k, i, j) \tag{3}$$

$$v_y(k, j) = \frac{1}{H} \sum_{j=1}^H F_{in}(k, i, j) \tag{4}$$

where $v_x(k, i) \in \mathbb{R}^{C \times H \times 1}$ and $v_y(k, j) \in \mathbb{R}^{C \times 1 \times W}$ represent the horizontal and vertical feature vectors, respectively.

To embed the spatial information into the channel dimension, we concatenate the two orthogonal feature vectors together to obtain the aggregated feature vector $v_{x,y} \in \mathbb{R}^{C \times 1 \times (H+W)}$. Next, we encode the aggregated feature vector, which means squeezing its channel by r times. Through the above squeezing operation, we integrate cross-channel information into the encoded feature vector, which can be

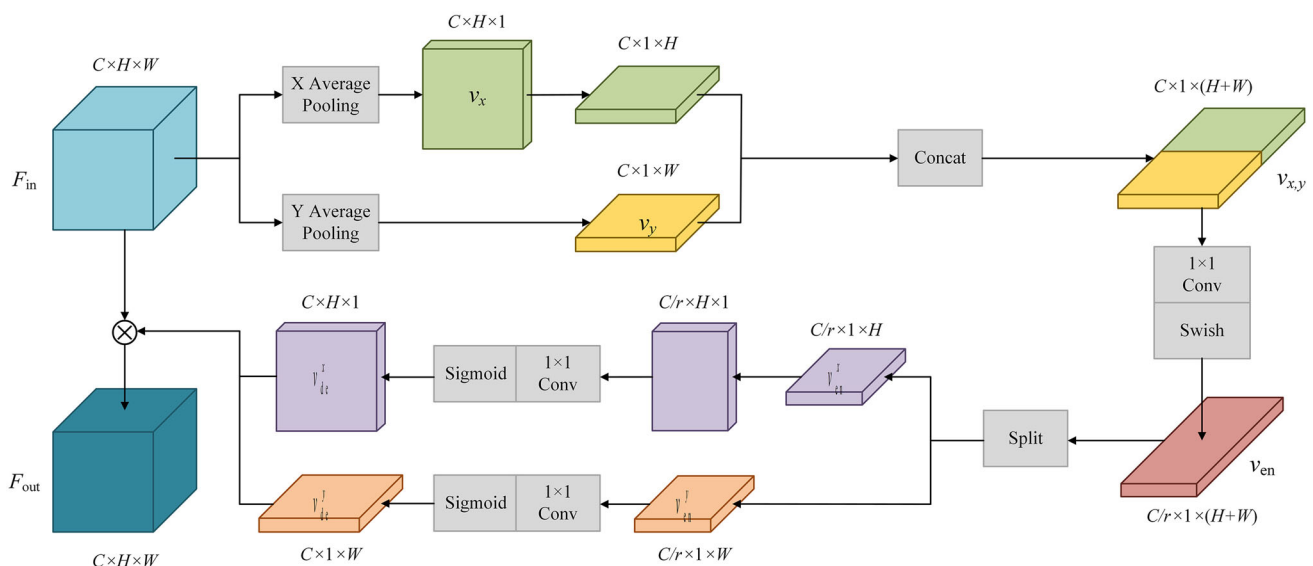


Fig. 4 Structure of CAM

expressed as

$$v_{en} = \text{Swish}(\text{Conv}_{1 \times 1}(v_{x,y})) \tag{5}$$

where Swish (Ramachandran et al. 2017) means a smooth and non-monotonic activation function. The expression for the Swish function is $\text{Swish}(v) = v / (1 + e^{-\beta v})$, where β is a constant or trainable parameter, and $\beta = 1$ is set by default. The vector $v_{en} \in \mathbb{R}^{C/r \times 1 \times (H+W)}$.

After obtaining the encoded feature vector containing accurate location information, a decoding stage is required to apply the attention weight to the input feature map. We first split the feature vector v_{en} along the horizontal and vertical directions, and the two split feature vectors are shown in

$$v_{en}^x, v_{en}^y = \text{Split}(v_{en}) \tag{6}$$

where Split indicates the dimension splitting operation. The vector $v_{en}^x \in \mathbb{R}^{C/r \times 1 \times H}$, $v_{en}^y \in \mathbb{R}^{C/r \times 1 \times W}$.

For the split feature vectors, a 1×1 convolution is adopted to restore the number of channels before the encoding stage. Finally, we obtain the feature vectors with the same number of channels as the input feature map. The decoded attention weights along two orthogonal directions can be expressed as follows:

$$v_{de}^x = \text{Sigmoid}(\text{Conv}_{1 \times 1}(v_{en}^x)) \tag{7}$$

$$v_{de}^y = \text{Sigmoid}(\text{Conv}_{1 \times 1}(v_{en}^y)) \tag{8}$$

where Sigmoid means a non-linear activation function. $v_{de}^x \in \mathbb{R}^{C \times H \times 1}$ and $v_{de}^y \in \mathbb{R}^{C \times 1 \times W}$ are the attention weights embedded in the horizontal and vertical spatial directions, respectively.

By applying the decoded attention weights, the final output feature map $F_{out} \in \mathbb{R}^{C \times H \times W}$ is given as

$$F_{out}(k, i, j) = v_{de}^x(k, i) \times v_{de}^y(k, j) \times F_{in}(k, i, j). \tag{9}$$

Orientation-invariant model

The classical linear convolution is not rotation-invariant at all, and the rotation invariance of the network only comes from data augmentation and multiple pooling operations. In this case, the network is weak in detecting rotating objects due to lack of rotation invariance. To improve the adaptability of the backbone network to rotating objects, OIM is introduced to enhance the consistency of features. Specifically, we embed OIM into the prediction network to generate depth rotation-invariant features to promote the detection ability of arbitrary-oriented ships. OIM is composed of active rotating filter (ARF) and oriented response pooling (ORPooling).

We define ARF as a $K \times K \times N$ filter, where K denotes the size of the filter kernel, and N denotes the number of orientation channels of the filter. ARF constructs the arrangement of oriented structures in an extra dimension, and it serves to explicitly encode the location and orientation of the input feature map. Specifically, ARF actively rotates $N - 1$ times during convolution to product a feature map with N orientation channels, which contains explicitly encoded location and orientation information. The oriented response convolution between ARF \mathcal{F} and feature map M is described as

$$I^{(i)} = \sum_{n=0}^{N-1} \mathcal{F}_{\theta_i}^{(n)} * M^{(n)}, \theta_i = i \frac{2\pi}{N}, i = 0, \dots, N-1 \tag{10}$$

where $I^{(i)}$ represents the i -th orientation channel of the output feature map I , and \mathcal{F}_{θ_i} represents a new filter obtained by rotating \mathcal{F} clockwise by θ_i . $\mathcal{F}_{\theta_i}^{(n)}$ and $M^{(n)}$ are the n -th orientation channel of \mathcal{F}_{θ_i} and M , respectively.

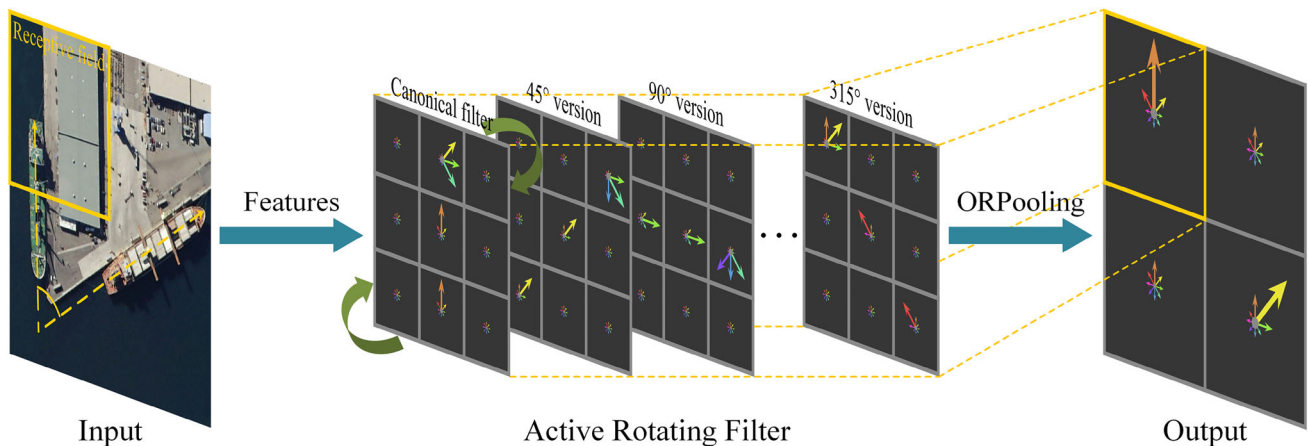


Fig. 5 Structure of OIM

Since ARF only encodes the captured multi-directional image responses, the current output feature map is not rotation-invariant. Next, we introduce ORPooling to extract within-class rotation-invariant features. ORPooling is devoted to select the orientation channel with the strongest response in the feature map I as the final output feature map \hat{I} , which is shown in

$$\hat{I} = \max(I^{(n)}), 0 < n < N - 1. \tag{11}$$

The structure of OIM is shown in Fig. 5. ARF contains the canonical filter itself and its multiple non-materialized rotated versions, and it can be visualized as N directional points on a $K \times K$ grid. ORPooling is essentially a pooling operation. For the feature map extracted by CNN, it first convolves with ARF to capture the location and orientation information of different ships, and then ORPooling is performed to obtain the feature map with rotation invariance. The rotation-invariant features of arbitrary-oriented objects with the same center point are identical, which is a very useful and noteworthy information for arbitrary-oriented ship detection. In addition, oriented response convolution only introduces a few parameters, and ORPooling does not introduce any parameters at all. Therefore, embedding OIM hardly affects the inference speed of the network.

Kullback-Leibler divergence

Among the detection methods for arbitrary-oriented objects, the five-parameter regression method represents arbitrary objects by adding an additional angle parameter θ . This regression method introduces boundary discontinuity, including angle periodicity and boundary commutativity. The former is mainly due to the bounded periodicity of the angle parameter, and the latter is related to the definition of the bounding box. The dramatic increase in loss at the boundary makes the regression form of the model inconsistent at the boundary and non-boundary, which will lead to the problem

of boundary discontinuity. In addition, the eight parameter regression method uses the coordinates of four vertices or the vectors from the center point to the four sides to represent arbitrary-oriented objects. This regression method is conducive to learning the geometric features of the object, but it will inevitably introduce too many parameters. To solve the problems brought by traditional methods and further promote the detection ability of the network for arbitrary-oriented ships, we define a regression loss function based on KLD. The bounding box of the arbitrary-oriented ship and its 2-D Gaussian distribution are shown in Fig. 6.

We convert the rotated bounding box $\mathcal{B}(x, y, w, h, \theta)$ to a 2-D Gaussian distribution $\mathcal{N}(\mu, \Sigma)$. The mean value μ and covariance matrix Σ can be calculated as follows:

$$\mu = (x, y)^\top \tag{12}$$

$$\begin{aligned} \Sigma^{\frac{1}{2}} &= \mathbf{R} \Lambda \mathbf{R}^\top \\ &= \begin{pmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{pmatrix} \begin{pmatrix} \frac{w}{2} & 0 \\ 0 & \frac{h}{2} \end{pmatrix} \begin{pmatrix} \cos \theta & \sin \theta \\ -\sin \theta & \cos \theta \end{pmatrix} \\ &= \begin{pmatrix} \frac{w}{2} \cos^2 \theta + \frac{h}{2} \sin^2 \theta & \frac{w-h}{2} \cos \theta \sin \theta \\ \frac{w-h}{2} \cos \theta \sin \theta & \frac{w}{2} \sin^2 \theta + \frac{h}{2} \cos^2 \theta \end{pmatrix} \end{aligned} \tag{13}$$

where x, y, w, h and θ represents the center point, length, width, and angle of the rotated bounding box, respectively. \mathbf{R} indicates the rotation matrix, and Λ indicates the diagonal matrix of eigenvalues.

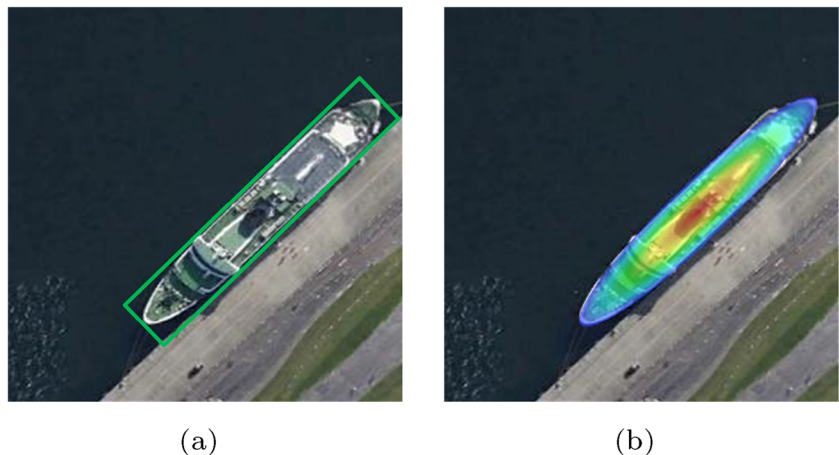
The KLD between two 2-D Gaussian distributions is given as

$$D_{kl}(\mathcal{N}_p \parallel \mathcal{N}_t) = \frac{1}{2}(\mu_p - \mu_t)^\top \Sigma_t^{-1} (\mu_p - \mu_t) + \frac{1}{2} \text{Tr} \left(\Sigma_t^{-1} \Sigma_p \right) + \frac{1}{2} \ln \frac{|\Sigma_t|}{|\Sigma_p|} - 1 \tag{14}$$

or

$$D_{kl}(\mathcal{N}_t \parallel \mathcal{N}_p) = \frac{1}{2}(\mu_p - \mu_t)^\top \Sigma_p^{-1} (\mu_p - \mu_t) + \frac{1}{2} \text{Tr} \left(\Sigma_p^{-1} \Sigma_t \right) + \frac{1}{2} \ln \frac{|\Sigma_p|}{|\Sigma_t|} - 1 \tag{15}$$

Fig. 6 Rotated bounding box and its 2-D Gaussian distribution. (a) Rotated bounding box of ship. (b) 2-D Gaussian distribution of rotated bounding box



where $\mathcal{N}_p(\mu_p, \Sigma_p)$ and $\mathcal{N}_t(\mu_t, \Sigma_t)$ represent the 2-D Gaussian distributions of the predicted box and ground-truth box, respectively. Combining Eqs. (12) and (13), it can be seen that each term in $D_{kl}(\mathcal{N}_t \parallel \mathcal{N}_p)$ is composed of partial parameter coupling. Specifically, the first term of $D_{kl}(\mathcal{N}_t \parallel \mathcal{N}_p)$ is a coupling term about x_p, y_p, w_p, h_p and θ_p , the second term is a coupling term about w_p, h_p and θ_p , and the third term is a coupling term about w_p and h_p . It is obvious that all parameters in $D_{kl}(\mathcal{N}_t \parallel \mathcal{N}_p)$ form a chain coupling relationship, called full coupling. In contrast to $D_{kl}(\mathcal{N}_t \parallel \mathcal{N}_p)$, although the second and third terms in $D_{kl}(\mathcal{N}_p \parallel \mathcal{N}_t)$ are both coupling terms, the first term is semi-coupled, which is caused by Σ_t^{-1} . Therefore, $D_{kl}(\mathcal{N}_p \parallel \mathcal{N}_t)$ is semi-coupled. Since the parameters in the fully-coupled $D_{kl}(\mathcal{N}_t \parallel \mathcal{N}_p)$ influence each other and optimize together, we design the regression loss function based on it, so that the optimization mechanism of the network is self-modulated.

The parameter gradient is dynamically updated according to the characteristics of the object, which is the most prominent feature of KLD and the best embodiment of its advantages. In particular, the gradient weight of the angle parameter can be updated according to the aspect ratio of the object, which is crucial for high-precision detection. For objects with large aspect ratio, a slight angle deviation will result in a severe accuracy degradation. Furthermore, KLD has been shown to be scale invariant. It can be concluded that using KLD Loss as the regression loss function of arbitrary-oriented ship detection can not only solve the problem of boundary discontinuity, but also further promote the scale invariance of the network. In this way, we can carry out high-precision detection of arbitrary-oriented ships, especially for densely arranged ships.

Prediction network

As for the design of anchors, we match five scale anchors {8, 16, 32, 64, 128} with five feature layers $\{P_3, P_4, P_5, P_6, P_7\}$. Each anchor has three scales $\{2^0, 2^{\frac{1}{3}}, 2^{\frac{2}{3}}\}$ and three ratios $\{1 : 2, 1 : 1, 2 : 1\}$, so there are nine anchors at each position of the feature layer. As shown in Fig. 2, multi-scale ship detection was carried out on five feature layers $\{P_3, P_4, P_5, P_6, P_7\}$.

The prediction network is composed of five scale detection heads, and each detection head contains two sub-networks, namely classification and regression subnets. The classification subnet predicts the probability of the detection object appearing in the anchor. The location, size and angle of the detection box are predicted by the regression subnet. The input of the prediction network is the output of the OIM, and both sub-networks consist of five 3×3 convolution layers. The size of the prediction feature layer of the classification and regression subnets is $KA \times H \times W$ and

$5A \times H \times W$, respectively. Where A indicates the number of anchors at each position of the feature layer, and $A = 9$ in this article. K indicates the number of object categories to be detected. There is only one object category of ships in this article, so $K = 1$. The regression of rotated bounding box $\mathcal{B}(x, y, w, h, \theta)$ is shown in

$$\begin{aligned} t_x &= (x_p - x_a)/w_a, t_y = (y_p - y_a)/h_a \\ t_w &= \log(w_p/w_a), t_h = \log(h_p/h_a) \\ t_\theta &= \theta_p - \theta_a \end{aligned} \quad (16)$$

where $\mathcal{B}_p(x_p, y_p, w_p, h_p, \theta_p)$ and $\mathcal{B}_a(x_a, y_a, w_a, h_a, \theta_a)$ represent the predicted box and anchor, respectively. The output of the prediction network is $\mathcal{B}_t(t_x, t_y, t_w, t_h, t_\theta)$, which includes the normalized coordinates of the prediction box relative to the anchor. Since we use the horizontal anchors, $\theta_a = -\pi/2$ is set by default.

In addition, MaxIoUAssigner is adopted to distinguish positive and negative samples, and the IoU thresholds of positive and negative samples are set to 0.5 and 0.4, respectively. Finally, we adopt the Rotate-NMS post-processing strategy to remove redundant predicted boxes.

Loss function

For the end-to-end detection network KRNet, the multi-task loss function for training can be expressed as

$$L = \frac{\lambda_1}{N} \sum_{n=1}^N L_{cls}(p_n, t_n) + \frac{\lambda_2}{N_{pos}} \sum_{n=1}^{N_{pos}} L_{reg}(b_n, g_{t_n}) \quad (17)$$

where N and N_{pos} indicate the number of total and positive anchors, respectively. L_{cls} indicates the classification loss function. t_n means the binary label of n -th anchor, and the labels of the positive and negative anchors correspond to 1 and 0, respectively. p_n denotes the n -th prediction probability of the corresponding class. L_{reg} indicates the regression loss function. b_n means the n -th bounding box, and g_{t_n} denotes the ground-truth of n -th object. The hyper-parameters λ_1 and λ_2 are used to control the trade-off. In this article, $\lambda_1 = 1$ and $\lambda_2 = 2$ are set by default.

Focal Loss is used as the classification loss function, which is given as

$$L_{cls} = \begin{cases} -\alpha(1-p)^\gamma \log p, & t = 1 \\ -(1-\alpha)p^\gamma \log(1-p), & t = 0 \end{cases} \quad (18)$$

where t denotes the label of the sample, and p means the prediction probability of a positive sample. The hyper-parameter γ is used to reduce the loss weight of easily classified samples, which makes the network focus more on difficultly

classified samples. The hyper-parameter α is the balance factor that balances the number of positive and negative samples. In this article, $\gamma = 2$ and $\alpha = 0.25$ are set by default.

KLD Loss is used as the regression loss function, which is given as

$$L_{reg} = 1 - \frac{1}{\tau + f(D)}, \tau \geq 1 \tag{19}$$

where D indicates the KLD, and $D = D_{kl}(\mathcal{N}_t \parallel \mathcal{N}_p)$ is set by default. $f(\cdot)$ is a function that can perform nonlinear transformation on D , which can make the loss smoother and more expressive. In this article, we adopt the non-linear function $f(D) = \log(D + 1)$. The hyper-parameter τ is used to adjust the loss, and $\tau = 1$ is set by default. KLD Loss can not only solve the problem of boundary discontinuity, but also has the characteristics of scale invariance and high-precision detection.

Experiments

Data sets and evaluation metrics

We evaluate the proposed KRNet in two public optical RSI data sets, HRSC2016 and DOTA Ship.

(1) HRSC2016: HRSC2016 (Liu et al. 2017) is a popular remote sensing ship data set, in which all images are obtained from six well-known ports on Google Earth. The data set includes 1061 images with 2976 ship objects, and the training, validation and test sets are composed of 436, 181 and 444 images, respectively. In HRSC2016, the image size ranges from 300×300 to 1500×900 , and the image resolution is between 2 and 0.4 meters.

(2) DOTA Ship: DOTA (Xia et al. 2018) is a large-scale data set for remote sensing object detection, which includes 2806 images with 15 common object categories. In DOTA, the image size is between 800×800 and 4000×4000 . In our experiments, 420 labeled images with 36258 ship objects are selected to construct a new remote sensing ship dataset DOTA Ship. Specifically, we obtain the training set and test set with 315 and 105 images, respectively, by randomly assigning these images. Considering the size of the original image in DOTA is too large, we crop it to the size of 1024×1024 and retain as many ship objects as possible.

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

$$\text{Precision} = \frac{TP}{TP + FP} \tag{20}$$

$$\text{Recall} = \frac{TP}{TP + FN} \tag{21}$$

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

AP can be calculated as

$$AP = \int_0^1 P(R) dR \tag{22}$$

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

Implementation details

We implement the proposed KRNet in the Pytorch framework on Ubuntu v20.04 system. All experiments are evaluated on a high-performance computer with Intel Core i7 10700F CPU, NVIDIA GeForce RTX 3070 GPU, and 32-GB memory. For the two public data sets, we uniformly resize the input image size to 800×800 . Besides, We initialize the backbone network with ResNet50 pre-trained on ImageNet. The SGD optimizer is adopted for network training, and the momentum and weight decay are set to 0.9 and 0.0001, respectively. The initial learning rate is set to 0.0025, and the batch size is set to 2. We train the network 50k iterations in total, and the learning rate decays by 10 and 100 times after 40k and 45k iterations, respectively. Moreover, we adopt random flipping, random rotation, and random scaling for data augmentation to improve the robustness of the network.

Comparison with other methods

The loss curves of KRNet during training on HRSC2016 are shown in Fig. 7. loss_cls means the classification loss, and

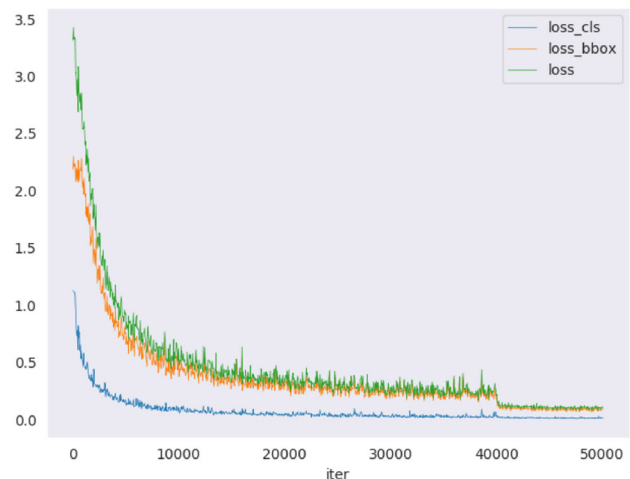


Fig. 7 Loss curves during training on the HRSC2016 data set

Table 1 Performance comparison of different detection methods on the HRSC2016 data set

Method	Backbone	Image size	AP(%)	Speed(FPS)
R ² CNN	ResNet101	800 × 800	73.07	2
IENet	ResNet101	1024 × 1024	75.01	9
RRPN	ResNet101	800 × 800	79.08	3.5
R ² PN	VGG16	800 × 800	79.6	< 1
RRD	VGG16	384 × 384	84.3	-
RoI-Trans	ResNet101	512 × 800	86.2	6
R ³ Det	ResNet101	800 × 800	89.26	12
KRNet	ResNet50	800 × 800	89.87	19.7

loss_bbox means the regression loss. In this experiment, the network is trained by 50k iterations in total, and the learning rate decays after 40k and 45k iterations, respectively. As shown in Fig. 7, the fitting process of the total loss curve is stable and the convergence effect is good. After 40k iterations, the loss gradually tends to be stable.

To verify the validity and feasibility of the proposed KRNet, we compared KRNet with seven popular rotation detection methods on HRSC2016, including R²CNN (Jiang et al. 2017), IENet (Lin et al. 2019), RRPN, R²PN (Zhang et al. 2018), RRD (Liao et al. 2018), RoI-Trans and R³Det. The performance comparison of different detection methods on HRSC2016 is shown in Table 1.

As shown in Table 1, KRNet reaches 89.87% AP with a detection speed of 19.7 frames per second (FPS) on HRSC2016, which surpasses the above seven comparison methods. Compared with the classical two-stage rotation detection method R²CNN, our proposed method achieves significant improvements by 16.80% in AP and 17.7 FPS in detection speed. Furthermore, compared with advanced single-stage rotation detection method R³Det, our proposed method has faster detection speed and AP is further improved by 0.61%. The backbone of all detection methods except KRNet is ResNet101 or VGG16, which have far more parameters than ResNet50. Nevertheless, KRNet still achieves the best detection performance, which fully highlights its superiority.

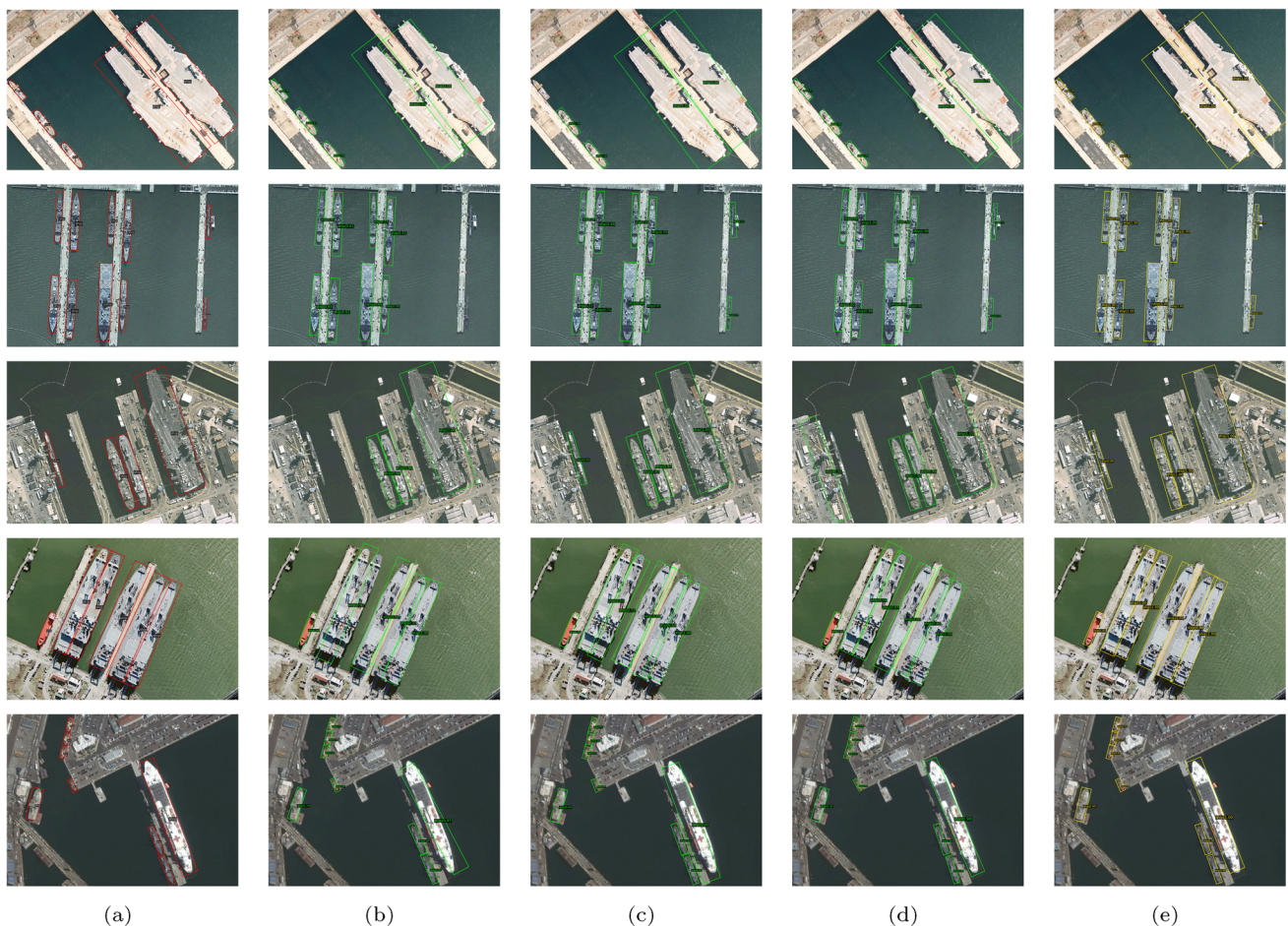
**Fig. 8** Comparison of the detection results by different methods. (a) Ground Truth. (b) RRPN. (c) RoI-Trans. (d) R³Det. (e) KRNet

Figure 8 shows the detection results of different methods in various scenes on the HRSC2016 data set. Figure 8(a) is Ground Truth, Fig. 8(b) is RRPN, Fig. 8(c) is RoI-Trans, Fig. 8(d) is R³Det, and Fig. 8(e) is the proposed KRNet. RRPN has poor detection results for small ships and densely arranged ships, which is mainly manifested as missed detection. Besides, its detection box does not fit well with the ship. RoI-Trans also missed a few small ships in specific scenes. The detection result of R³Det is relatively good, but it has false detection in complex background areas. Compared with the above three methods, our proposed method can locate and detect ships more accurately. In particular, KRNet has good detection results for small and densely arranged ships, and its false detection rate is lower in complex background areas.

To further verify the robustness of the proposed KRNet, we evaluated KRNet on DOTA Ship with larger scenes and more ship objects. Correspondingly, KRNet is compared with other rotation detection methods, including R²CNN, RRPN, SCRDet (Yang et al. 2019), R³Det and RoI-Trans. The performance comparison of different detection methods on DOTA Ship is shown in Table 2.

The experimental results in Table 2 show that KRNet reaches 83.62% AP on DOTA Ship, which is superior to the other five comparison methods. Compared with the common rotation detection methods R²CNN, RRPN and SCRDet, our proposed method achieves great improvements by 27.37%, 24.50%, and 8.93% in AP, respectively. In addition, compared with advanced rotation detection methods R³Det and RoI-Trans, our proposed method achieves improvements by 3.87% and 0.58% in AP. The experimental results on DOTA Ship fully verify the robustness and effectiveness of KRNet.

Ablation studies

Our proposed method introduces RFF-Net to increase object attention, OIM to generate depth rotation-invariant features, and high-precision KLD Loss to promote the detection accuracy of ships. In order to fully evaluate the contribution of

Table 2 Performance comparison of different detection methods on the DOTA Ship data set

Method	Backbone	Image size	AP(%)
R ² CNN	ResNet50	800 × 800	56.25
RRPN	ResNet50	800 × 800	59.12
SCRDet	ResNet101	800 × 800	74.69
R ³ Det	ResNet101	800 × 800	79.75
RoI-Trans	ResNet101	512 × 800	83.04
KRNet	ResNet50	800 × 800	83.62

Table 3 Ablation studies of each module on the HRSC2016 data set

Baseline	RFF-Net	OIM	KLD	AP(%)	Speed(FPS)
✓				79.95	21.4
✓	✓			83.91	20.6
✓	✓	✓		85.93	19.7
✓	✓	✓	✓	89.87	19.7

each module in KRNet, we performed ablation studies on HRSC2016. It should be noted that all experiments adopt the same training and data augmentation strategies, and the research results are shown in Table 3. In this article, the baseline is RetinaNet-H (Yang et al. 2021) with Smooth L1 Loss as the regression loss.

As shown in Table 3, there is only 79.95% AP at baseline, and the detection speed is 21.4 FPS. When FPN in baseline is replaced by RFF-Net, AP increases by 3.96%, and the detection speed reduces by 0.8 FPS. With the addition of OIM, AP increases by 2.02% again, and the detection speed reduces by 0.9 FPS. After replacing Smooth L1 loss with KLD Loss, the detection speed remained unchanged, and AP further increases by 3.94% to 89.87%. Compared with baseline, the detection speed of KRNet is only reduced by 1.7 FPS, still maintaining a fast detection speed of 19.7 FPS, and AP unexpectedly increases by 9.92%. Obviously, the ablation studies fully demonstrate the effectiveness and importance of RFF-Net, OIM and KLD Loss in KRNet.

Conclusion

In this article, an arbitrary-oriented ship detection network KRNet is proposed to solve the problem of low detection accuracy caused by dense arrangement, variable direction, and complex background of nearshore ships in RSIs. Firstly, an RFF-Net combined with CAM is constructed to obtain rich multi-scale fusion features and enhance the attention to ships. Secondly, the OIM is embedded before the prediction network to generate depth rotation-invariant features and improves the ability of the network to detect rotating objects. Finally, a regression loss function based on KLD is defined to further promote the detection accuracy of arbitrary-oriented ships while solving the problem of boundary discontinuity.

Experiments on the HRSC2016 and DOTA Ship data sets show the performance comparison between the proposed KRNet and other popular rotation detection methods. On the HRSC2016 data set, KRNet achieves 89.87% AP at the detection speed of 19.7 FPS. Meanwhile, on the more challenging DOTA Ship data set, KRNet also reaches 83.62% AP. The ablation studies fully verify the contribution of the modules

proposed in KRNet. In summary, the experimental studies prove that the proposed method reaches state-of-the-art ship detection performance.

Author Contributions Yantong Chen and Jialiang Wang wrote the main manuscript text and designed the ship detection model. Yanyan Zhang and Yang Liu prepared Figs. 1–7 and performed the data processing and analysis. All authors reviewed the manuscript.

Funding This work was supported by the National Natural Science Foundation of China [Grant No.: 61901081]; China Postdoctoral Science Foundation [Grant No.: 2020M680927]; Fundamental Research Funds for the Central Universities [Grant No.: 3132022237].

Availability of data and materials Data sharing not applicable to this article as no datasets were generated or analysed during the current study.

Declarations

Conflict of interest The authors have no relevant financial or non-financial interests to disclose.

References

- Deng Z, Sun H, Zhou S, Zhao J (2019) Learning deep ship detector in sar images from scratch. *IEEE Trans Geosci Remote Sens* 57(6):4021–4039
- Shuai T, Sun K, Wu X, Zhang X, Shi B (2016) A ship target automatic detection method for high-resolution remote sensing. In: 2016 IEEE international geoscience and remote sensing symposium (IGARSS), pp 1258–1261. IEEE
- Song Z, Sui H, Wang Y (2014) Automatic ship detection for optical satellite images based on visual attention model and lbp. In: 2014 IEEE workshop on electronics, computer and applications, pp 722–725. IEEE
- Li-Bing J, Zhuang W, Wei-Dong H (2011) An aiac-based inshore ship target detection approach. *Remote Sens Technol Appl* 22(1):88–94
- Zhu C, Zhou H, Wang R, Guo J (2010) A novel hierarchical method of ship detection from space-borne optical image based on shape and texture features. *IEEE Trans Geosci Remote Sens* 48(9):3446–3456
- Nie X, Duan M, Ding H, Hu B, Wong EK (2020) Attention mask r-cnn for ship detection and segmentation from remote sensing images. *IEEE Access* 8:9325–9334
- Zhao J, Zhang X, Yang L, Ma S, Wang Y, Dong Y, Sun M, Chen C (2020) Ship detection in remote sensing images based on deep learning. *Sci Surv Mapp* 45(3):110–116
- Shi W-X, Jiang J-H, Bao S-L (2020) Ship detection method in remote sensing image based on feature fusion. *Acta Photo Sin* 49(7):0710004
- Chen L, Shi W, Deng D (2021) Improved yolov3 based on attention mechanism for fast and accurate ship detection in optical remote sensing images. *Remote Sens* 13(4):660
- Zhong W, Guo F, Xiang S, Pan C (2019) Ship detection in remote sensing based with rotated rectangular region. *J Comput-Aided Des Comput Graph* 31(11):1935–1945
- Yang X, Sun H, Fu K, Yang J, Sun X, Yan M, Guo Z (2018) Automatic ship detection in remote sensing images from google earth of complex scenes based on multiscale rotation dense feature pyramid networks. *Remote Sens* 10(1):132
- Liu W, Ma L, Chen H (2018) Arbitrary-oriented ship detection framework in optical remote-sensing images. *IEEE Geosci Remote Sens Lett* 15(6):937–941
- Li L, Zhou Z, Wang B, Miao L, Zong H (2020) A novel cnn-based method for accurate ship detection in hr optical remote sensing images via rotated bounding box. *IEEE Trans Geosci Remote Sens* 59(1):686–699
- Han Y, Yang X, Pu T, Peng Z (2021) Fine-grained recognition for oriented ship against complex scenes in optical remote sensing images. *IEEE Trans Geosci Remote Sens* 60:1–18
- Lin TY, Goyal P, Girshick R, He K, Dollár P (2017) Focal loss for dense object detection. In: Proceedings of the IEEE international conference on computer vision, pp 2980–2988
- Girshick R, Donahue J, Darrell T, Malik J (2014) Rich feature hierarchies for accurate object detection and semantic segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 580–587
- He K, Zhang X, Ren S, Sun J (2015) Spatial pyramid pooling in deep convolutional networks for visual recognition. *IEEE Trans Pattern Anal Mach Intell* 37(9):1904–1916
- Ren S, He K, Girshick R, Sun J (2015) Faster r-cnn: towards real-time object detection with region proposal networks. *Adv Neural Inf Process Syst* 28
- Dai J, Li Y, He K, Sun J (2016) R-fcn: object detection via region-based fully convolutional networks. *Adv Neural Inf Process Syst* 29
- Redmon J, Divvala S, Girshick R, Farhadi A (2016) You only look once: Unified, real-time object detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 779–788
- Liu W, Anguelov D, Erhan D, Szegedy C, Reed S, Fu CY, Berg AC (2016) Ssd: single shot multibox detector. In: Computer vision—ECCV 2016: 14th European conference, Amsterdam, The Netherlands, October 11–14, 2016, proceedings, Part I 14, pp 21–37. Springer
- Zhang S, Wen L, Bian X, Lei Z, Li SZ (2018) Single-shot refinement neural network for object detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 4203–4212
- Fu CY, Liu W, Ranga A, Tyagi A, Berg AC (2017) Dssd: deconvolutional single shot detector. arXiv preprint [arXiv:1701.06659](https://arxiv.org/abs/1701.06659)
- Ma J, Shao W, Ye H, Wang L, Wang H, Zheng Y, Xue X (2018) Arbitrary-oriented scene text detection via rotation proposals. *IEEE Trans Multimedia* 20(11):3111–3122
- Liu Z, Hu J, Weng L, Yang Y (2017) Rotated region based cnn for ship detection. In: 2017 IEEE international conference on image processing (ICIP), pp 900–904. IEEE
- Ding J, Xue N, Long Y, Xia GS, Lu Q (2019) Learning roi transformer for oriented object detection in aerial images. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 2849–2858
- Zhang G, Lu S, Zhang W (2019) Cad-net: a context-aware detection network for objects in remote sensing imagery. *IEEE Trans Geosci Remote Sens* 57(12):10015–10024
- Zhou X, Yao C, Wen H, Wang Y, Zhou S, He W, Liang J (2017) East: an efficient and accurate scene text detector. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 5551–5560
- Liao M, Shi B, Bai X (2018) Textboxes++: a single-shot oriented scene text detector. *IEEE Trans Image Process* 27(8):3676–3690
- Yang X, Yan J, Feng Z, He T (2021) R3det: refined single-stage detector with feature refinement for rotating object. *Proc AAAI Conf Artif Intell* 35:3163–3171
- Yang X, Yan J (2020) Arbitrary-oriented object detection with circular smooth label. In: Computer vision—ECCV 2020: 16th European conference, Glasgow, UK, August 23–28, 2020, proceedings, Part VIII 16, pp 677–694. Springer

- He K, Zhang X, Ren S, Sun J (2016) Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 770–778
- Lin TY, Dollár P, Girshick R, He K, Hariharan B, Belongie S (2017) Feature pyramid networks for object detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 2117–2125
- Zhou Y, Ye Q, Qiu Q, Jiao J (2017) Oriented response networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 519–528
- Ramachandran P, Zoph B, Le QV (2017) Searching for activation functions. arXiv preprint [arXiv:1710.05941](https://arxiv.org/abs/1710.05941)
- Liu Z, Yuan L, Weng L, Yang Y (2017) A high resolution optical satellite image dataset for ship recognition and some new baselines. In: ICPRAM, pp 324–331
- Xia GS, Bai X, Ding J, Zhu Z, Belongie S, Luo J, Datcu M, Pelillo M, Zhang L (2018) DOTA: a large-scale dataset for object detection in aerial images. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 3974–3983
- Jiang Y, Zhu X, Wang X, Yang S, Li W, Wang H, Fu P, Luo Z R2cnn: rotational region cnn for orientation robust scene text detection. [arXiv:1706.09579](https://arxiv.org/abs/1706.09579)
- Lin Y, Feng P, Guan J, Wang W, Chambers J (2019) Ienet: interacting embranchment one stage anchor free detector for orientation aerial object detection. arXiv preprint [arXiv:1912.00969](https://arxiv.org/abs/1912.00969)
- Zhang Z, Guo W, Zhu S, Yu W (2018) Toward arbitrary-oriented ship detection with rotated region proposal and discrimination networks. *IEEE Geosci Remote Sens Lett* 15(11):1745–1749
- Liao M, Zhu Z, Shi B, Xia GS, Bai X (2018) Rotation-sensitive regression for oriented scene text detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 5909–5918
- Yang X, Yang J, Yan J, Zhang Y, Zhang T, Guo Z, Sun X, Fu K (2019) Srdet: towards more robust detection for small, cluttered and rotated objects. In: Proceedings of the IEEE/CVF international conference on computer vision, pp 8232–8241

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Springer Nature or its licensor (e.g. a society or other partner) holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.