Arbitrary-Oriented Ship Detection through Center-Head Point Extraction

Feng Zhang, Xueying Wang, Shilin Zhou, Yingqian Wang

***Abstract*—Ship detection in remote sensing images plays a crucial role in military and civil applications and has drawn increasing attention in recent years. However, existing multi-oriented ship detection methods are generally developed on a set of predefined rotated anchor boxes. These predefined boxes not only lead to inaccurate angle predictions but also introduce extra hyper-parameters and high computational cost. Moreover, the prior knowledge of ship size has not been fully exploited by existing methods, which hinders the improvement of their detection accuracy. Aiming at solving the above issues, in this paper, we propose a *center-head point extraction based detector* (named CHPDet) to achieve arbitrary-oriented ship detection in remote sensing images. Our CHPDet formulates arbitrary-oriented ships as rotated boxes with head points which are used to determine the direction. Key-point estimation is performed to find the center of ships. Then the size and head points of the ship is regressed. Finally, we use the target size as prior to finetune the results. Moreover, we introduce a new dataset for multi-class arbitrary-oriented ship detection in remote sensing Images at fixed ground sample distance (GSD) which is named FGSD2021. Experimental results on two ship detection datasets (i.e., FGSD2021 and HRSC2016) demonstrate that our CHPDet achieves state-of-the-art performance and can well distinguish bow and stern. The code and dataset will be made publicly available.**

***Index Terms*—Arbitrary-oriented ship detection, Remote sensing images, Keypoint estimation, Deep convolution neural networks**

I. Introduction

SHIP detection from high-resolution optical remote sensing images is widely applied in both civilian and military tasks such as illegal smuggling, port management, and military target reconnaissance. Recently, ship detection has received increasing attention and was widely investigated in the past decades [1] [2] [3]. However, ship detection in remote sensing images is a highly challenging task due to the arbitrary ship orientations, densely-parking scenarios, and complex backgrounds. To handle the multi-orientation issue, existing methods generally use a series of predefined anchors [4], which has the following shortcomings:

Inaccurate angle regression. Figures 1(a)-(d) illustrate four different representations of an arbitrary-oriented ship. Since ships in remote sensing images are generally in strips, the intersection over union (IoU) score is very sensitive to the angle of bounding boxes. As shown in Fig. 1(e), the ground truth box is the bounding box of a ship with an aspect ratio of 10:1. The red rotated box is generated by rotating the ground truth box with a small angle of 5° . It can be observed that such a small angle variation reduces the IoU between these two boxes to 0.63. Therefore, the anchor-based detectors which define the positive and negative anchors by IoU score usually suffer from an imbalance issue between different anchors, and thus result in detection performance degeneration [5]. Moreover, the angle of the ship is a periodic function, and it is discontinuous at the boundary (0° or 180°), as shown in Fig. 1(f). This discontinuity will also cause performance degeneration [6].

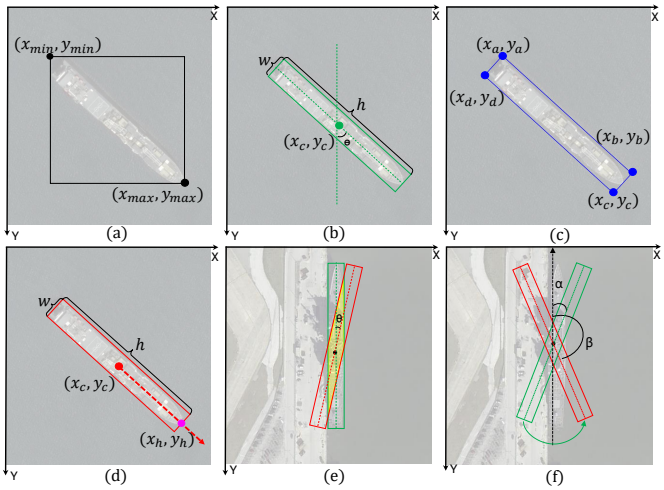


Fig. 1: Four different representations of the arbitrary-oriented ship and the disadvantage of the angle regression scheme. (a) Horizontal boxes parameterized by 4 tuples $(x_{min}, y_{min}, x_{max}, y_{max})$. (b) Rotated box with the angle parameterized by 5 tuples (x_c, y_c, w, h, θ). (c) Rotated box with vertices (a, b, c, d), parametrized by 8 tuples $(x_a, y_a, x_b, y_b, x_c, y_c, x_d, y_d)$. (d) Rotated box with head point which is parameterized by 6 tuples $(x_c, y_c, w, h, x_h, y_h)$. (e) A small angle disturbance will cause a large IoU decrease. (f) The angle is discontinous when reaches its range boundary.

Under-exploitation of prior information of ships. Most previous ship detectors directly used the same rotation detection algorithm as those in the area of remote sensing and scene text detection. However, ships in remote sensing images have its unique characteristics. Generally, the outline of the ship is a pentagon with two parallel long sides, and the position of the bow is relatively obvious and a certain category of the ship

This work was partially supported in part by the National Natural Science Foundation of China (Nos. 61972435, 61401474, 61921001).

Feng Zhang, Xueying Wang, Shilin Zhou, Yingqian Wang are with the College of Electronic Science and Technology, National University of Defense Technology (NUDT), P. R. China. Emails: {zhangfeng01, wangxueying, slzhou, wangyingqian16}@nudt.edu.cn. (Corresponding author: Xueying Wang)

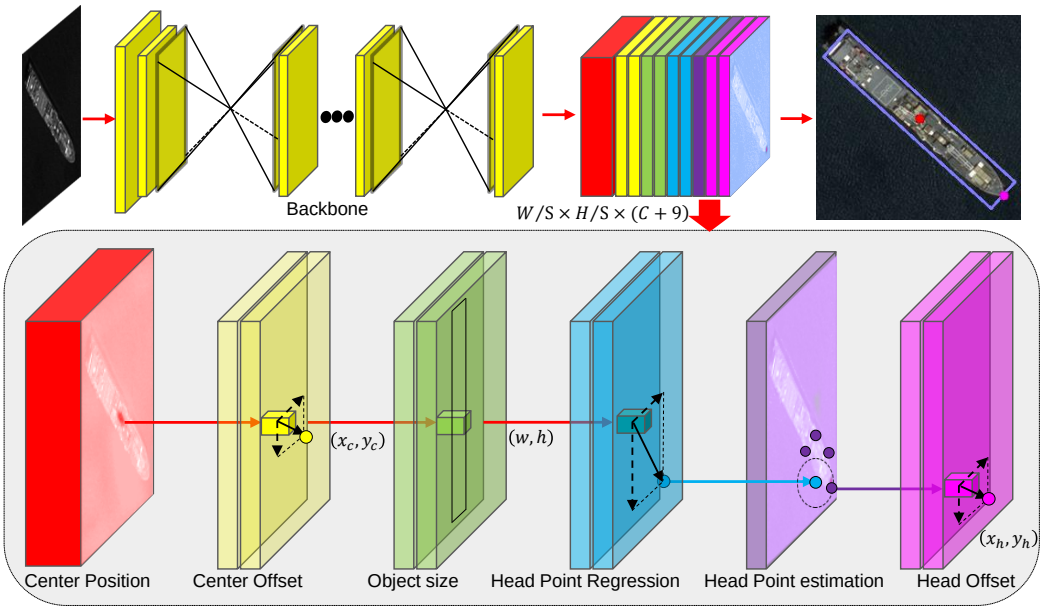


Fig. 2: The overall framework of our arbitrary-oriented ship detection method. Feature maps are first generated by using a fully convolutional network. Afterward, the peaks of the center feature map are selected as center points. Then, the center points offsets, object sizes and head regression locations are regressed on the corresponding feature maps on the position of each center point. The potential head points are collected by extracting peaks with confidence scores larger than 0.1 on the head feature map. The final head location is obtained by assigning each regressed location to its nearest potential head points.

in remote sensing images has a relatively fixed size range by normalizing the ground sample distance (GSD) of images. The size of the ship and position of the ship head and are important clues for detection. However, This prior information has been under-exploited.

Excessive hyper-parameters and high computational cost. Existing methods generally use oriented bounding boxes as anchors to handle rotated objects and thus introduce excessive hyper-parameters such as box sizes, aspect ratios, and orientation angles. Note that, these hyper-parameters have to be manually tuned for novel scenarios, which limits the generalization capability of these methods. Predefined anchor-based methods usually require a large number of anchor boxes. For example, in R^2PN [7], six different orientations were used in rotated anchor boxes, and there are a total of 24 anchors at each pixel on its feature maps. A large number of anchor boxes introduce excessive computational cost when calculating intersection-over-union (IoU) scores and executing non-maximum suppression (NMS) algorithm.

Motivated by the anchor-free detectors in natural scenes, in this paper, we propose a one-stage, anchor-free and NMS-free method for arbitrary-oriented ship detection in remote sensing images and formulates ships as rotated boxes with head points representing the direction. Specifically, feature maps are first generated by using a full convolution network. Afterward, the peaks of the feature map are selected as potential center points. Then, the offset, object sizes, and head positions are regressed on the corresponding feature maps at each center point position. Finally, target size information is used to adjust the classification score. The architecture of our CHPDet is shown in Fig. 2,

The major contributions of this paper are summarized as follows.

- We propose new representations for the arbitrary-oriented boxes, which can transform angle regression to a keypoint estimation and address the problem of the angle periodicity.
- We design a method to refine the detection results based on prior information to improve the detection accuracy.
- We introduce a new dataset named FGSD2021 for multi-class arbitrary-oriented ship detection in remote sensing images at fixed GSD. This dataset can make use of the prior knowledge of ship size and adapt to the actual application for remote sensing ship detection.
- We develop a new ship detection baseline, which significantly reduces the computation cost and hyper-parameters. Our method can predict angles in a large range(0°-360°), which can distinguish between bow and stern, and more accurately. Extensive experimental results in the ship detection dataset show that our CHPDet achieving state-of-the-art performance on both speed and accuracy, as shown in Fig. 3.

The rest of this paper is organized as follows. In Section II, we briefly review the related work. In Section III, we

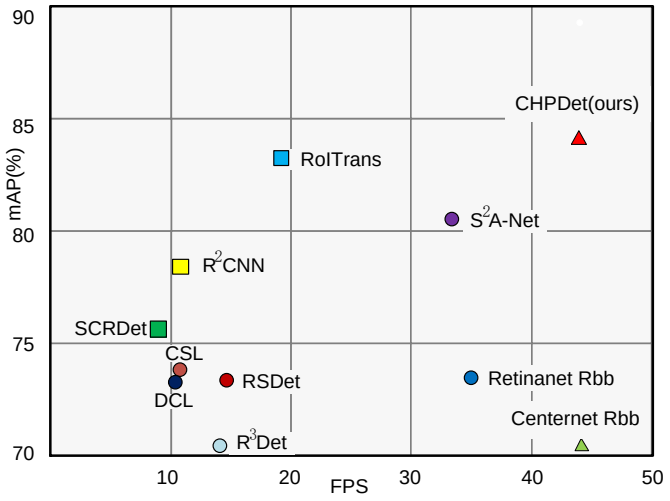


Fig. 3: Speed vs. accuracy on our proposed FGSD2021 dataset.

introduce the proposed method in detail. Experimental results and analyses are presented in Section IV. Finally, we conclude this paper in Section V.

II. Related Work

In this section, we briefly review the major works in horizontal object detection, rotated object detection, and remote sensing ship detection.

A. Horizontal Object Detection

In recent years, deep convolutional neural networks (DCNN) have been developed as a powerful tool for feature representation learning [8], and have achieved significant improvements in horizontal object detection [9]. Existing object detection methods generally represent objects as horizontal boxes, as shown in Fig. 1(a). According to different detection paradigms, deep learning-based object detection methods can be roughly divided into two-stage detectors, single-stage detectors, and multi-stage detectors. Two-stage detectors (e.g., RCNN [10], Fast-RCNN [11], Faster-RCNN [12], Mask-RCNN [13], R-FCN [14]) used a pre-processing approach to generate object proposals, and extract features from the generated proposals to predict the category. In contrast, one-stage detectors (e.g., YOLO [15], [16], SSD [17], RetinaNet [18]) did not have the pre-processing step and directly perform categorical prediction on the feature maps. Multi-stage detectors (e,g, cascade RCNN [19], HTC [20]) performed multiple classifications and regressions in the second stage, resulting in notable accuracy improvements. In summary, two-stage and multi-stage detectors generally achieve better performance, but one-stage detectors are usually more time-efficient. Compared to the above-mentioned anchor-based methods, anchor-free methods [21], [22] can avoid the requirement of anchors and have become a new research focus in recent years. For example, CornerNet [21] detected objects on each position of the feature map using the top-left and bottom-right corner points. CenterNet [22] modeled an object as a center point and performed keypoint estimation to find center points and regressed the object size. FCOS [23] predicts four distances, a center score, and classification score at each position of the feature map to detect objects. The above-mentioned approaches achieved significant improvement in general object detection tasks. However, these detectors only generate horizontal bounding boxes, which limits their applicability.

B. Arbitrary-oriented object detection

Arbitrary-oriented detectors are widely used in remote sensing and scene text images. Most of these detectors use rotated bounding boxes or quadrangles to represent multi-oriented objects. as shown in 1(b) (c). In $RRPN$ [24], rotated region proposal networks was proposed to improve the quality of the region proposals. In R^2CNN [6], a horizontal region of interest (RoI) was generated to simultaneously predict the horizontal and rotated boxes. RoI Transformer [25] transformed a horizontal RoI into a rotated RoI (RRoI). In SCRDet [26] and RSDet [27], novel losses were employed to address the boundary problem for oriented bounding boxes. In R^3Det [28], a refined single-stage rotated detector was proposed for the feature misalignment problem. In CSL [29] and DCL [30], angle regression was converted into a classification task to handle the boundary problem. In S^2A-Net [31], a fully convolutional layer was proposed to align features to achieve better performance. The afore-mentioned methods need a set of anchor boxes for classification and regression. These anchors introduce excessive hyper-parameters which limit the generalization capability and introduce an excessive computational cost. At present, several anchor-free arbitrary-oriented detectors, (e.g., O^2D-Net [32] and X-LineNet [33]) are proposed to detect oriented objects by predicting a pair of intersecting lines. However, their performance still lags behind that of the anchor-base detectors.

C. Ship detection in remote sensing images

Different from other objects in remote sensing images, ships are in strips with a large aspect ratio. Generally, the outline of the ships is a pentagon with two parallel long sides, and the position of the bow is relatively obvious. Consequently, a certain category of the ship in remote sensing images has a relatively fixed size range by normalizing the GSD of images.

Traditional ship detectors generally used a coarse-to-fine framework with two stages including ship candidate generation and false alarm elimination. For example, Shi et al. [34] first generated ship candidates by considering ships as anomalies and then discriminated these candidates using the AdaBoost approach [35]. Yang et al. [36] proposed a saliency-based method to generate candidate regions, and used a support vector machine (SVM) to further classify these candidates. Liu et al [37], [38] introduced an RRoI pooling layer to extract features of rotated regions. In R2PN [7], a rotated region proposal network was proposed to generate arbitrary-proposals with ship orientation angle information. The above detectors are also based on a set of anchors and cannot fully exploit the prior information of ships.

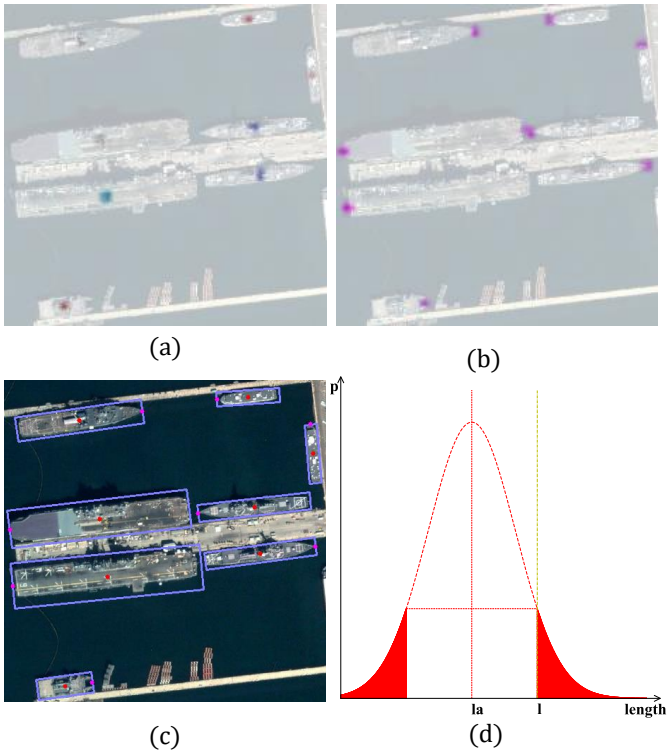


Fig. 4: A visualization of (a) center heatmap, (b) head heatmap, (c) detection results, and (d) ship probability density map. In center and head heatmaps, different colors represent different categories. In the ship probability density map, l_a represents the mean length of category a, l represents the length of detected ship. The red area is the probability that the target belongs to category a.

III. Proposed Method

In this section, the architecture of our CHPDet is introduced in detail. As shown in Fig. 2, our proposed method consists of six modules including an arbitrary-oriented ship representation module, feature extraction module, center point detection module, size regression module, head point estimation module, and Refine probability module.

A. Arbitrary-oriented ship representation

As shown in Fig. 1, the widely-used horizontal bounding boxes cannot be directly applied to the arbitrary-oriented ship detection task since excessive redundant background area is included. Moreover, since the arbitrary-oriented ships generally have a large aspect ratio and park densely, the NMS algorithm using a horizontal bounding box tends to produce miss detection. To this end, many methods represent ships as rotated bounding boxes, and these boxes are parameterized by five tuples (c_x, c_y, w, h, θ), where (x, y) is the coordinate of the center of the rotated bounding box, w and h are the width and length of the ship, respectively. The angle $\theta \in [0°, 180°)$ is the orientation of the long side with respect to the y-axis. This representation can result in the regression inconsistency issue near the boundary case. Recently, some detectors represent objects by four clockwise vertices, which is parameterized by 8 tuples $(x_a, y_a, x_b, y_b, x_c, y_c, x_d, y_d)$. This representation can also introduce regression inconsistency due to the order of the four corner points. To avoid the aforementioned inconsistency problem, we present ships as two points and the corresponding size, which is parameterized by six tuples $(x_c, y_c, w, h, x_h, y_h)$. (x_c, y_c) is the coordinate of the center of the rotated bounding box, w and h are the width and length of the ship, (x_h, y_h) is the coordinate of the head point of the ship. The direction of the ship is determined by connecting the bow and the center. This representation of ships converts discontinuous angle regression to continuous keypoint estimation, and can handle the discontinuity problem. This representation also extends the range of angle representation to $[0°, 360°)$ and makes the network have the ability to distinguish bow and stern.

B. Feature extraction

Let $\mathbf{I} \in \mathbb{R}^{W \times H \times 3}$ be an input image with width W and height H, the final feature map is $\mathbf{F} \in \mathbb{R}^{\frac{W}{s} \times \frac{H}{s} \times (C+9)}$, where S is the output stride and C is the number of classes. In this paper, we set the default stride value to 4. Several different backbone (e.g., deep layer aggregation (DLA) [39] and hourglass network (Hourglass) [40]) can be used to extract features from images. We followed CenterNet [22] to enhance DLA by replacing ordinary convolutions with deformable convolutions and add a 256 channel 3×3 convolutional layer before the output head. The hourglass network consists of two sequential hourglass modules. Each hourglass module includes 5 pairs of down and up convolutional networks with skip connections. This network generally yields better keypoint estimation performance [21].

C. Center point detection

As in [21], suppose $\text{center} = (x_i, y_i)$ be the ground truth center point of ship. For each center point of class c $\text{center} \in \mathbb{R}^2$, we compute a low-resolution equation $\tilde{\text{center}} = \left\lfloor \frac{\text{center}}{\text{s}} \right\rfloor$. Target map $\mathbf{C} \in \mathbb{R}^{\frac{W}{s} \times \frac{H}{s} \times C}$ is computed by placing a 2D Gaussian distribution $\exp\left(-\frac{(x-\tilde{center}_x)^2+(y-\tilde{center}_y)^2}{2\sigma_p^2}\right)$ around each $\tilde{\text{center}}$, where s is a downsampling stride and σ_p is a size-adaptive standard deviation. If two Gaussian kernels belong to the same class with overlap region, we take the maximum value at each pixel of the feature map. $\hat{\mathbf{C}} \in \mathbb{R}^{\frac{W}{s} \times \frac{H}{s} \times C}$ is a prediction on feature maps produced by the backbones. When training the heatmaps, only center points are positive, and all the other points are negative, which may cause a huge imbalance between positive and negative samples. To handle the imbalance issue, we use the variant focal loss:

$$\mathcal{L}_c = \frac{-1}{N} \begin{cases} \sum_{xyc} \left(1 - \hat{\mathbf{C}}_{xyc}\right)^{\gamma} \log\left(\hat{\mathbf{C}}_{xyc}\right) & \text{if } \mathbf{C}(xyc) = 1 \\ \sum_{xyc} \left(1 - \mathbf{C}_{xyc}\right)^{\beta} \left(\hat{\mathbf{C}}_{xyc}\right)^{\gamma} & \\ \log\left(1 - \hat{\mathbf{C}}_{xyc}\right) & \text{otherwise} \end{cases} \tag{1}$$

where γ and β are the hyper-parameters of the focal loss, N is the number of objects in image I which is used to normalize

all positive focal loss instances to 1. We set $\gamma = 2$ and $\beta = 4$ in our experiments empirically as in [41]. We extract locations with values larger or equal to their 8-connected neighbors as detected center points. The value of the peak point is set as a measure of its confidence, and the coordinates in the feature map are used as an index to get other attributes. Figure 4(a) shows a visualization of the center and heatmaps. Suppose that $c = \{(\hat{x}_k, \hat{y}_k)\}_{k=1}^{n}$ is the set of detected center points. Each center point location is given by an integer coordinates $c_k = (\hat{x}_i, \hat{y}_i)$ on feature map $\mathbf{C}$. In order to reduce the quantization error caused by the output stride, we produce local offset feature maps $\mathbf{O} \in \mathbb{R}^{\frac{W}{S} \times \frac{H}{S} \times 2}$. For each predicted center point c_k, let the value on the offset feature maps $off_k = (\delta\hat{x}_k, \delta\hat{y}_k)$ be the offset of center point c_k. The final center point location of class c is $\hat{center}_c = \{(\hat{x_k} + \delta\hat{x}_k, \hat{y_k} + \delta\hat{y}_k)\}_{k=1}^{n}$. Note that, all the classes share the same offset predictions to reduce the computational complexity. The offset is optimized with an L1 loss. This supervision is performed on all center point.

$$\mathcal{L}_{\text{co}} = \frac{1}{N}\sum_{k=1}^{N}\left|\mathbf{O}c_k - \left(\frac{\text{center}_\text{k}}{S} - c_k\right)\right|. \tag{2}$$

D. Size regression

Let $s_k = (w, h)$ be the size of ships, we obtain the ship size s_k on $\mathbf{S} \in \mathbb{R}^{\frac{W}{S} \times \frac{H}{S} \times 2}$ at each predicted center point $c_k \in \hat{center}$, and L1 loss function is also used.

$$\mathcal{L}_{\text{size}} = \frac{1}{N}\sum_{k=1}^{N}\left|\mathbf{S}_{c_k} - s_k\right|. \tag{3}$$

E. Head Point estimation

We perform two steps for better head points estimation.

1) Regression-based head point estimation: Let $\text{head}_\text{k} = (\text{h}_\text{x}, \text{h}_\text{y})$ be the k^th head point,we directly regress to the offsets $(\Delta\hat{x}_k, \Delta\hat{y}_k)$ on feature map $\mathbf{R} \in \mathbb{R}^{\frac{W}{S} \times \frac{H}{S} \times 2}$ at each predicted center point $c_k \in \hat{center}$. The regression-based head point is $\{(\hat{x_k} + \Delta\hat{x}_k, \hat{y_k} + \Delta\hat{y}_k)\}_{k=1}^{n}$, where $(\Delta\hat{x}_i, \Delta\hat{y}_i)$ is the head point regression, and an L1 loss is used to optimized head regression feature maps.

$$\mathcal{L}_{hr} = \frac{1}{N}\sum_{k=1}^{N}\left|\mathbf{R}_{c_k} - h_k\right|. \tag{4}$$

2) Bottom-up head point estimation: We use standard bottom-up multi-human pose estimation [42] to refine the head points. A target map $\mathbf{H} \in \mathbb{R}^{\frac{W}{s} \times \frac{H}{s} \times 1}$ is computed like in III-C. A low-resolution equation is $\tilde{\text{head}} = \lfloor \frac{\text{head}}{\text{s}} \rfloor$. Head point heatmap $\mathbf{E} \in \mathbb{R}^{\frac{W}{S} \times \frac{H}{S} \times 1}$ and local offset heatmap $\mathbf{HO} \in \mathbb{R}^{\frac{W}{S} \times \frac{H}{S} \times 2}$ are head maps produced by the backbones. These two head maps are trained with variant focal loss and an L1 loss.

$$\mathcal{L}_{he} = \frac{-1}{N}\sum_{xy}\begin{cases}(1-\mathbf{E}_{xy})^{\gamma}\log(\mathbf{E}_{xy}) & \text{if } \mathbf{H}_{xy}=1\\ (1-\mathbf{H}_{xy})^{\beta}(\mathbf{E}_{xy})^{\gamma} & \\ \log(1-\mathbf{E}_{xy}) & \text{otherwise}\end{cases} \tag{5}$$

$$\mathcal{L}_{ho} = \frac{1}{N}\sum_{k=1}^{N}\left|\mathbf{HO}_{c_k} - \left(\frac{\text{head}_\text{k}}{S} - \tilde{head}\right)\right|. \tag{6}$$

The bottom-up head point estimation is the same as the center point detection, the only difference is that in the center point detection, each category has a center point heat map, while in the head point estimation, all categories share one head points heatmap. We extract all peak point locations $\hat{\text{head}} = \left\{\tilde{l}_i\right\}_{i=1}$ with a confidence $\mathbf{HO}_{x,y} > 0.1$ as a potential head points set, and refine the potential head point locations by adding the offset. Figure 4(b) visualizes the head points heatmap. We then assign each regressed location $\hat{head}_r$ to its closest detected head point $\arg\min_{l \in \text{head}_\text{r}}\left(l - \hat{\text{head}}\right)^2$. Finally, we use the line connecting the head point and the center point to determine the orientation of detection. The final detection results are show in Fig. 4(d). We introduce weighted factor to balance the contribution of these parts, and set $\lambda_o = 1$, $\lambda_s = 0.1$, $\lambda_{\text{hr}} = 1$, $\lambda_{\text{he}} = 1$, and $\lambda_{\text{ho}} = 1$ in all our experiments. We set $\lambda_s = 0.1$ since the scale of the loss is ranged from 0 to the output size h/S. The overall training loss is

$$\mathcal{L} = \mathcal{L}_c + \lambda_o\mathcal{L}_o + \lambda_s\mathcal{L}_s + \lambda_{\text{hr}}\mathcal{L}_{\text{hr}} + \lambda_{\text{he}}\mathcal{L}_{\text{he}} + \lambda_{\text{ho}}\mathcal{L}_{\text{ho}}. \tag{7}$$

F. Refine probability according to size

By normalizing the GSD of remote sensing images, objects of the same size on the ground have the same size in all images. The size of the target is an important clue to identify the target because a certain type of targets in remote sensing images usually have a relatively fixed size range. We propose a method to adjust the confidence score. As shown in Fig. 4(d), suppose that the category of the detected box is a, the original confidence score is s_a, assume that the length of the detected ship obeys a normal distribution, the mean and standard deviation of the length of category a are L_a, δ_a. Then the probability of the target belonging to a is p_a,

$$p_a = \frac{2}{\delta_a\sqrt{2\pi}}\int_{-\infty}^{-|l-la|}\exp\left(-\frac{(x-la)^2}{2\delta_a^2}\right)dx. \tag{8}$$

In order to reduce hyper-parameters, we assume that the standard deviation is proportional to the mean $\delta_a = L_a * \lambda$ for all category of ships. We multiply the two probabilities to obtain the final detection confidence, $\hat{p_a} = p_a * s_a$.

G. From center and head point to rotated boxes

In the testing phase, We first extracted the center points on the output center heatmaps $\mathbf{C}$ for each category. We using a 3×3 max-pooling layer to get the peak points and selected the top 100 peaks as potential center points. Each center point location is represented as an integer coordinates $\hat{c} = (\hat{x}, \hat{y})$. Take out the offsets $(\delta_{\hat{x}}, \delta_{\hat{y}})$, size (w, h), and head points regression $(\Delta\hat{x}, \Delta\hat{y})$ on the corresponding feature map at the location of center points. We also picked all head peak point $\hat{h_u}$ on the output center heatmaps $\mathbf{E}$ with a scores $\mathbf{E}_{x,y} > 0.1$, and then assigned each regressed location $(\hat{x} + \Delta\hat{x}, \hat{y} + \Delta\hat{y})$ to its closest detected keypoint $\hat{h_u}$ as the final head point $(\hat{hx}, \hat{hy})$. Then we get the rotated boxes $(\hat{x} + \delta\hat{x}, \hat{y} + \delta\hat{y}, w, h, \hat{hx}, \hat{hy})$

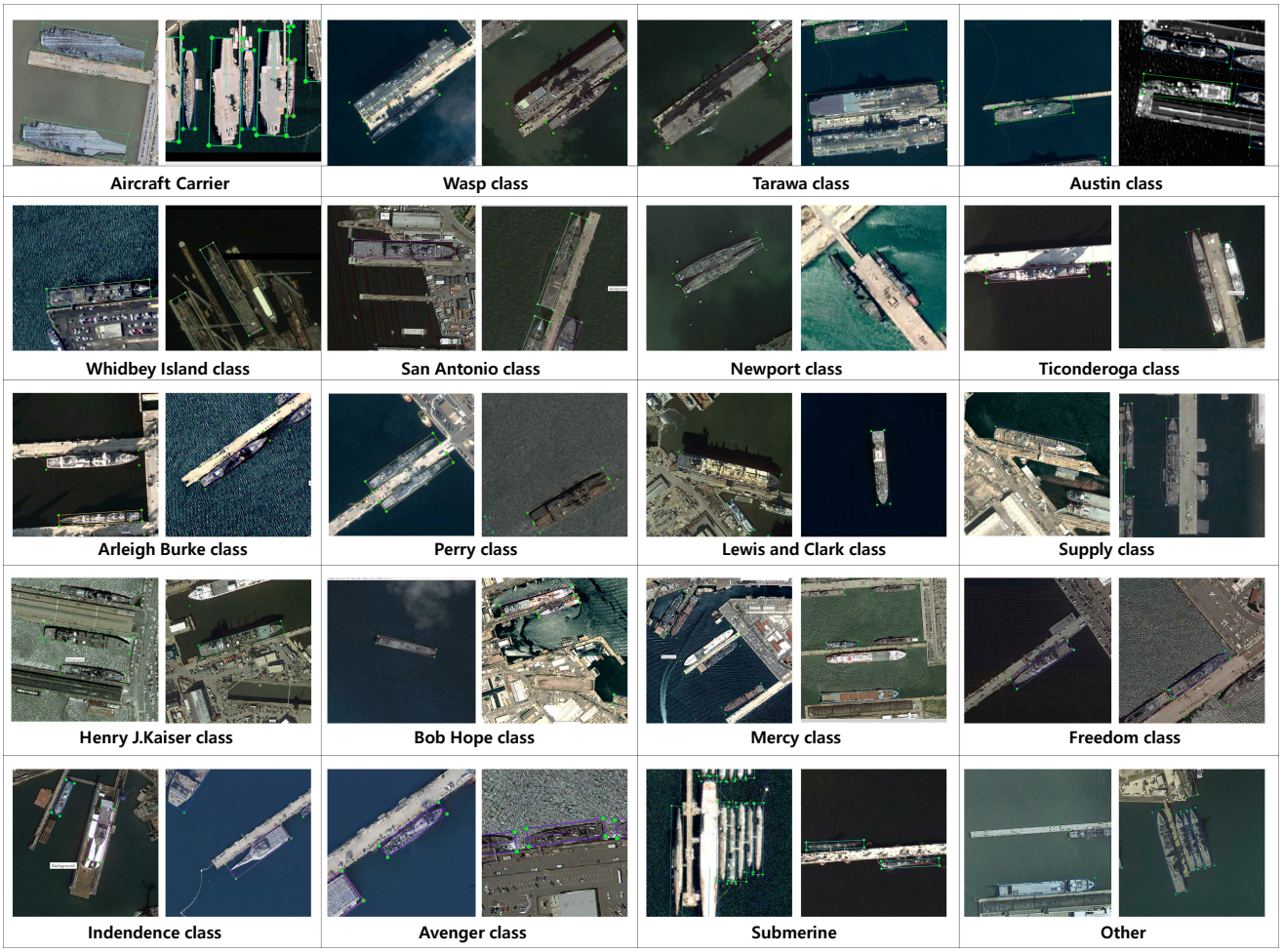


Fig. 5: Example images from the proposed FGSD2021 dataset.

IV. Experiments

We evaluate our method on our FGSD2021 dataset and the public HRSC2016 [43] dataset. In this section, we first introduce the datasets and implementation details, and then perform ablation studies and compare our network to several state-of-the-art methods.

A. Datasets

1) FGSD2021: Existing datasets ((e.g., DOTA [44], DIOR [45] NWUP [46], and HRSC2016) for arbitrary-orientation object detection in remote sensing images have the following shortcomings: First, the GSD is unknown, so we cannot get the size of objects in the image by the actual size on the ground. Second, these datasets usually cut the image into small patches, which is inconsistent with the actual remote sensing image detection task. To solve these problems, we propose a new ship detection dataset at fixed GSD remote sensing images named FGSD2021. Our dataset is developed by collecting high-resolution satellite images from publicly available Google Earth, which covers some famous Ports such as DanDiego, Kitsap-Bremerton, Norfolk, PearlHarbor and Yokosuka. Images in our dataset are of very large size, and we get multiple images of the same port on different days. We collected 636 images, including 5274 labeled targets. The GSD of all images is normalized to 1 meter per pixel. The image's width is ranged from 157 to 7789 pixels, and the average width is 1202 pixels. The image height is ranged from 224 to 6506 pixels, and the average height is 1205 pixels. Our FGSD2021 dataset is divided into 424 training images and 212 test images. The training set is used in the training phase. The detection performance of the proposed method is evaluated on the test set. 20 categories are chosen and annotated in our dataset, including Aircraft carriers, Wasp class, Tarawa class, Austin class, Whidbey Island class, San Antonio class, Newport class, Ticonderoga class, Arleigh Burke class, Perry class, Lewis and Clark class, Supply class, Henry J. Kaiser class, Bob Hope Class, Mercy class, Freedom class, Independence class, Avenger class, Submarine, and Others. We use the labelimg2[1] tools to label the ship, the angle range is 0-360 degrees, and the main direction is the direction of the bow.

2) HRSC2016: The HRSC2016 dataset is a challenging dataset for ship detection in remote sensing images, which collected six famous harbors on Google Earth. The training,

[1]https://github.com/chinakook/labelImg2

TABLE I: Results achieved on FGSD2021 with different ablation versions. '*Baseline*' represents adding a branch to predict the angle based on CenterNet. '*Head point extraction*' represents replacing the angle prediction branch to head point estimation module. '*Refine probability*' represents using the prior size information to adjust the confidence score of the detected boxes.

	baseline	Different Settings of CHPDet	
Head point extraction		✓	✓
Refine probability			✓
mAP	70.52	81.71	84.75

validation, and test sets include 436 images with 1207 samples, 181 with 541 samples, and 444 images with 1228 samples, respectively. The image size of this dataset ranges from 300×300 to 1500×900. This dataset includes three levels of tasks (i.e., L1, L2, and L3), and these three tasks contain 1 class, 4 classes, and 19 classes, respectively. Besides, The head point of ships is given in this dataset. Following [47] [31] [28], we evaluate our method on task L1. We used the training and validation set in the training phase and evaluated the detection performance on the test set.

B. Implementation Details

Our network was implemented in PyTorch on a PC with Intel Core i7-8700K CPU, NVIDIA RTX 2080Ti GPU. We used the Adam method [48] as the optimizer, and the initial learning rate is set to 2.5×10^{-4}. We trained our network for 140 epochs with a learning rate being dropped at 90 epochs. During the training phase, We used random rotation, random flipping, and color jittering for data augmentation. To maintain the GSD of the image, we cropped all images into 1024×1024 slices with a stride of 820, resized them to 512×512. We merged the detection results of all the slices to restore the detecting results on the original image. Finally, we apply Rotated-Non-maximum-suppression (RNMS) with an IoU threshold of 0.15 to discard repetitive detections. The speed of the proposed network is measured on a single NVIDIA RTX 2080Ti GPU.

C. Evaluation Metrics

The Intersection over Union (IoU) between oriented boxes is used to distinguish detection results. The mean average precision (mAP) and head direction accuracy are used to evaluate the performance of arbitrary-Oriented detectors.

1) IoU: The IoU is the result of dividing the overlapping area of two boxes by the union area of two boxes. We adopted the evaluation approach in DOTA [49] to get the IoU. If the IoU between a detection box and a ground-truth is higher than a threshold, the detection box is marked as true-positive (TP), otherwise false-positive (FP). And if a ground-truth box has no matching detections, it is marked as false negative (FN).

2) mAP: PASCAL VOC2007 metrics is used to compute the mAP in all of our experiments. The precision and recall are calculate by $\text{precision} = \frac{\text{TP}}{\text{TP+FP}}$ $\text{recall} = \frac{\text{TP}}{\text{TP+FN}}$. We first set a set of thresholds, and then we get a corresponding maximum precision for each recall threshold. AP is the average of these precision. The mean average precision (mAP) is the mean of APs over all classes. The $\text{mAP}_{0.5}$-$\text{mAP}_{0.8}$ is computed under the IoU threshold of 0.5-0.8 respectively.

3) Head direction accuracy.: The prediction angle range of the previous algorithm is 0-180 degrees, which can not distinguish between the bow and the stern of the ship. The mAP base on the IOU between two rotated boxes is taken as the only evaluation criterion, which can not reflect the detection accuracy of the bow direction. To solve this problem, we define bow direction accuracy as an additional evaluation. That is the proportion of the ships whose angle difference from the ground-truth less than 10 degrees in all TP.

D. Ablation Study

In this subsection, we present ablation experiments to investigate our models.

1) CenterNet as baseline: As an anchor-free detector, CenterNet performs keypoint estimation to find the center point and regresses the object size at each center point position. To carry out arbitrary-oriented ship detection, we add a extra branch to predict the angle as baseline which is named CenterNet Rbb. CenterNet Rbb use DLA34 as backbone, present ships as rotated boxes with angle and use L1 loss function to optimized angle regression feature maps. We set weighted factor $\lambda_{angle} = 0.1$ to balance the contribution of these parts, since the scale of the loss is ranged from 0 to 180. As shown in Table I, CenterNet achieves an mAP of 70.52 which demonstrates that our baseline achieves competitive performance.

2) Effectiveness of head point estimation.: When we replace the angle prediction branch to head point estimation module, the overall performance has been improved from 70.52 to 81.71. It is a great improvement in performance, which fully demonstrates the effectiveness of the head point estimation approach. To further verify the promoting effect of head point estimation for center point detection and size detection, we set all angle of ground-truth and the detected box to 0. Map has risen from 84.4 to 88.0

3) Refine probability according to size.: In our designed network, the size information of the ship is used to refine the confidence of the detected boxes. Table I shows the mAP values of different ablation versions on the test set. It can be observed that the baseline model achieves the lowest mAP. When the image resolution is increased or a better backbone is used, the accuracy is improved. When combining the prior size information, the performance has been improved. The effect on low-resolution images is more obvious, e.g., from 81.71 to 84.75, almost an increase of 3 percentages in mAP. It demonstrates that the prior size information can improve the classification accuracy.

To adjust the influence of size on probability, we set a variance coefficient. In the FGSD2021 dataset, the actual length of each category is determined. For example, the length of the Ticonderoga-class cruiser is 172.8 meters. Consequently, we use the length of this type of ship l_a multiplied by a coefficient r as the mean square error of this type δ_a, as can

TABLE II: Performance of CHEDet achieved on FGSD2021 with different variance coefficient. '*without refine*' represents using the original confidence without refinement. '*Ground truth class*' represents using ground truth class label to eliminate the misclassification.

Backbone	Image Size	coefficient λ								without refine	Ground truth class
		0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8		
DLA34	512×512	81.90	83.05	84.47	84.59	84.70	**84.75**	84.32	84.31	81.70	89.33
Hourglass104	512×512	81.47	84.55	**85.18**	85.13	85.08	85.11	85.09	85.08	84.45	89.52
DLA34	1024×1024	83.90	86.50	88.58	88.72	**88.95**	88.95	88.95	88.52	88.48	89.74

TABLE III: Detection accuracy on different types of ships and overall performance with the state-of-the-art methods on FGSD. The short names for categories are defined as (abbreviation-full name): Air - Aircraft carriers, Was - Wasp class, Tar - Tarawa class, Aus - Austin class, Whi - Whidbey Island class, San -San Antonio class, New - Newport class, Tic - Ticonderoga class, Bur- Arleigh Burke class, Per - Perry class, Lew -Lewis and Clark class, Sup - Supply class, Kai - Henry J. Kaiser class, Hop - Bob Hope Class, Mer - Mercy class, Fre - Freedom class, Ind - Independence class,Ave - Avenger class, Sub - Submarine and Oth - Other. CHPDet† means CHPDet with hourglass104 backbone, CHPDet$^{\urcorner}$ means CHPDet trained and detected with 1024×1024 image size.

Method	Air	Was	Tar	Oth	Aus	Whi	San	New	Tic	Bur	Per	Lew	Sup	Kai	Hop	Mer	Fre	Ind	Ave	Sub	mAP
R^{2}CNN [6]	89.9	80.9	80.5	57.2	79.4	87.0	87.8	44.2	89.0	89.6	79.5	80.4	47.7	81.5	87.4	100	82.4	100	66.4	50.9	78.09
Retinanet Rbb [50]	89.7	89.2	78.2	9.1	87.3	77.0	86.9	62.7	81.5	83.3	70.6	46.8	69.9	80.2	83.1	100	80.6	89.7	61.5	42.5	73.49
ROI Trans [25]	90.9	88.6	87.2	66.9	89.5	78.5	88.8	81.8	89.6	89.8	90.4	71.7	74.7	73.7	81.6	78.6	100	75.6	78.4	68.0	83.48
SCRDet [26]	77.3	90.4	87.4	57.1	89.8	78.8	90.9	54.5	88.3	89.6	74.9	68.4	59.2	90.4	77.2	81.8	73.9	100	43.9	43.8	75.90
CSL [29]	89.7	81.3	77.2	40.7	80.2	71.4	77.2	52.7	87.7	87.7	74.2	57.1	97.2	77.6	80.5	100	72.7	100	32.6	37.0	73.73
DCL [30]	89.9	81.4	78.6	45.6	80.7	78.0	87.9	49.8	78.7	87.2	76.1	60.6	76.9	90.4	80.0	78.8	77.9	100	37.1	31.2	73.34
R^{3}Det [28]	90.9	80.9	81.5	40.0	90.1	79.3	87.5	29.5	77.4	89.4	69.7	59.9	67.3	80.7	76.8	72.7	83.3	90.9	38.4	23.1	70.47
RSDet [27]	89.8	80.4	75.8	50.6	77.3	78.6	88.8	26.1	84.7	87.6	75.2	55.1	74.4	89.7	89.3	100	86.4	100	27.6	37.6	73.74
S^{2}A-Net [31]	90.9	81.4	73.3	64.7	89.1	80.9	89.9	81.2	89.2	90.7	88.9	60.5	75.9	81.6	89.2	100	68.6	90.9	61.3	55.7	80.19
CenterNet Rbb [22]	67.2	77.9	79.2	6.8	75.5	66.8	79.8	76.8	83.1	89.0	77.7	54.5	72.6	77.4	100	100	60.8	74.8	46.5	44.1	70.52
CHPDet	90.9	90.7	89.1	50.8	87.9	84.7	81.4	98.3	89.7	90.1	90.2	76.4	68.3	89.6	89.4	100	79.9	90.9	87.7	68.9	84.75
CHPDet†	90.6	90.1	89.6	57.7	81.2	87.9	90.7	97.7	89.1	90.4	90.5	70.1	70.1	88.0	87.9	100	93.9	90.9	75.2	71.2	85.18
CHPDet$^{\urcorner}$	90.9	90.9	90.6	57.0	90.5	90.3	90.3	90.3	89.9	90.2	90.2	78.2	92.4	88.9	89.2	100	99.4	99.4	81.9	88.6	**88.95**

been seen in Eq. 8 $\delta_a = l_a \times r$. The variance coefficient will affect classification accuracy. When the coefficient is large, the probability difference between different categories will be smaller, and the influence of the size on the confidence of the category will be smaller, and vice versa. As can be observed in Table II, when the coefficient is small, it is equivalent to using size as the main information to classify objects. However, it reduces accuracy. Accuracy increases gradually as the coefficient increases, and when the coefficient is larger than 0.5, the coefficient has little impact on the accuracy. When we treat all categories as one category and remove the categorys influence on the detection results, the mAP is 89.33, 89.52, and 89.74, respectively. At the same time, by adding prior information to adjust the classification confidence, 20 types of target detection under the 1024 resolution image obtained 88.95 mAP, which shows that after adding the prior information, almost all categories are classified correctly.

4) Bow direction accuracy: It can be seen from Table III that the bow direction accuracy of our CHPDet is up to 97.84, 98.14, and 98.39 respectively.

E. Comparison with other methods

In this section, we compare our proposed method with other representative ship detectors including RetinaNet Rbb [50] ROI trans[2], R2CNN, CSL, DCL, RSDet, SCRDet[3], and S^2A-Net[4]. For a fair comparison, we use the default settings of the original codes on the DOTA dataset, the same data augmentation strategy, and train roughly the same epochs in all experiments.

1) Results on FGSD2021.: Figure 6 shows several detection results using different methods. As shown in the first row, all the other methods have misclassification or false alarms, S^{2}A-Net has an inaccurate angle prediction, while our method precisely detects them. In the second row of Fig. 6, all other methods miss a ship or make inaccurate detections, while our method generates correct bounding boxes. For the densely parking scene in the second row of Fig. 6, all other detectors lost at least two submarines, and our method is not influenced by the densely parking scene. The third row of Fig. 6 is a harbor with a complex background, the two ships are not on the water but in the dry dock. ROI trans and S^{2}A-Net miss the targets, SCRDet has an inaccurate bounding box. Compared to these four methods, our method can better detect the ships in the complex background. Therefore, our method is more robust for challenging situations. It can be seen from Table III that the accuracy of ROI trans is 83.48 at the speed of 19.2 FPS, while the accuracy of our algorithm is 1.02 percentage higher than that of ROI trans at the speed of 43.5 FPS. Our algorithm achieves the highest accuracy at twice the speed of ROI trans. When higher resolution images are used, the accuracy can be improved by a large margin, up to 89.63. Angle prediction has a key impact on IoU. To further verify the accuracy of the prediction, we gradually increase the IoU threshold. As can be seen from Table IV, when the IOU threshold is gradually increased, the performance of other detectors have dropped

[2]https://github.com/dingjiansw101/AerialDetection/
[3]https://github.com/yangxue0827/RotationDetection
[4]https://github.com/csuhan/s2anet

TABLE IV: Detection performance on the FGSD2021 at different IoU thresholds and the accuracy of bow direction.

Method	Backbone	Image Size	FPS	$mAP_{0.5}$	$mAP_{0.6}$	$mAP_{0.7}$	$mAP_{0.8}$	Bow direction accuracy
R^2CNN [6]	Resnet50	512 × 512	10.3	78.09	75.03	64.83	36.41	_
Retinanet Rbb [50]	Resnet50	512 × 512	35.6	73.49	69.17	62.82	45.00	_
RoI Trans [25]	Resnet50	512 × 512	19.2	83.48	82.63	80.35	65.18	_
SCRDet [26]	Resnet50	512 × 512	9.2	75.90	70.98	61.82	35.12	_
CSL [29]	Resnet50	512 × 512	10.4	73.73	69.71	60.25	34.93	_
DCL [30]	Resnet50	512 × 512	10.0	73.34	69.19	57.80	28.54	_
R^3Det [28]	Resnet50	512 × 512	14.0	70.47	68.32	57.17	27.44	_
RSDet [27]	Resnet50	512 × 512	15.4	73.74	69.55	61.52	35.83	_
S^2A-Net [26]	Resnet50	512 × 512	33.1	80.19	79.58	75.65	58.82	_
CenterNet Rbb [22]	DLA34	512 × 512	**43.5**	70.52	69.34	65.52	45.33	_
CHPDet(ours)	DLA34	512 × 512	**43.5**	84.75	83.71	80.96	66.20	97.84
CHPDet(ours)	Hourglass104	512 × 512	13.7	85.18	84.82	82.34	65.90	98.14
CHPDet(ours)	DLA34	1024 × 1024	15.4	**88.95**	**88.20**	**86.05**	**72.28**	**98.39**

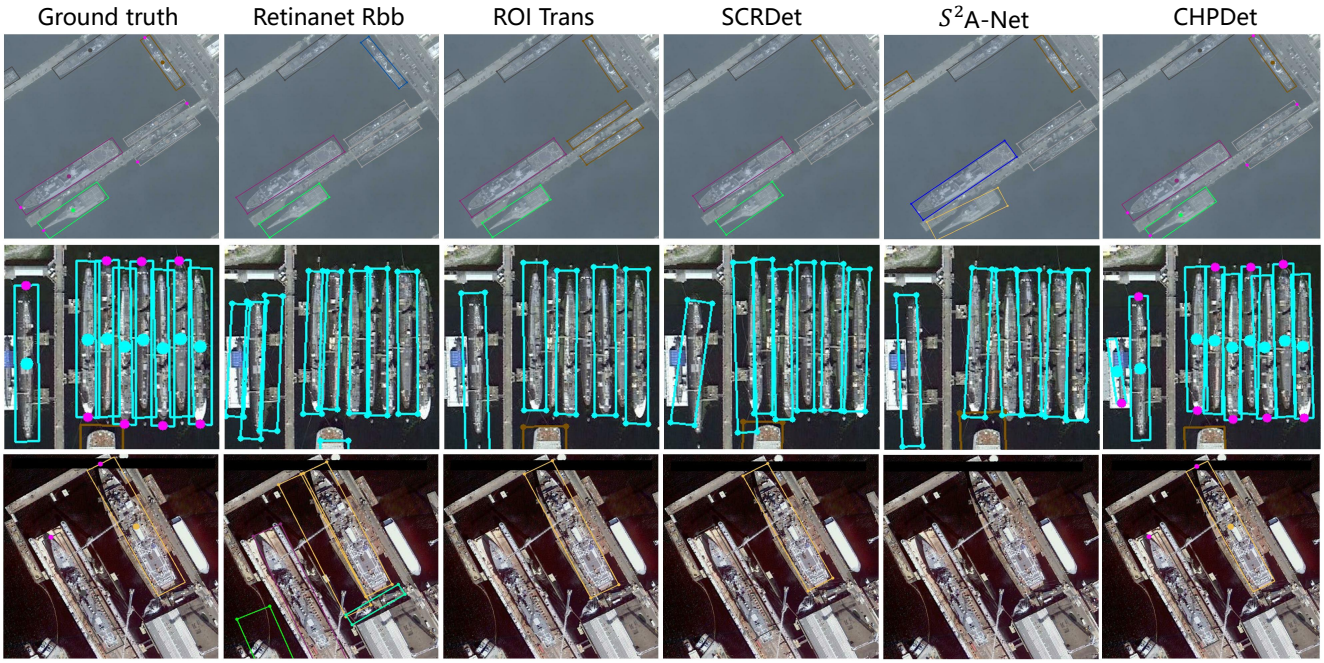


Fig. 6: Several detection results using different methods. The first column is the ground truth, and the second to the last columns are the results of Retinanet Rbb, ROI Trans, SCRDet, S^2A-Net, and CHPDet(ours), respectively. Different color of rotated boxes represents a different type of ships. The pink point represents the head point.

TABLE V: Detection accuracy on the HRSC2016 dataset, 07 means using the 2007 evaluation metric.

Method	Backbone	Image Size	mAP(07)
R^2CNN [6]	Resnet101	800 × 800	73.1
RRPN [47]	Resnet101	800 × 800	79.1
R^2PN [7]	VGG16	_	79.6
ROI trans [25]	Resnet101	512 × 800	86.2
Gliding Vertex [51]	Resnet101	_	88.2
R^3Det [28]	Resnet101	800 × 800	89.3
RSDet [28]	ResNet152	800 × 800	86.5
FR-Est [52]	Resnet101	_	89.7
S^2A-Net [31]	Resnet101	800 × 512	90.2
CHPDet	DLA34	512 × 512	88.8
CHPDet	Hourglass104	512 × 512	**90.6**

significantly, and the decline of our detector is relatively small. When the IOU threshold was increased to 0.8, The mAP of our CHPDet remained at 72.28.

2) Results on HRSC2016.: The performance comparison results between our proposed method and some state-of-the-art methods on the HRSC2016 dataset are shown in Table V. The R^2CNN [6] predicts inclined minimum area box based on concatenated ROI pooling features with VGG16 backbone, achieving an AP score of 73.07. RRPN [53] present the Rotation Region Proposal Networks and the Rotation Region-of-Interest (RRoI) pooling layer to efficiently adapt to rotating target detection and improves the accuracy to 79.08. ROI Transformer [25] learns a transformer to transform Horizontal Region of Interest (HRoI) into a Rotated Region of Interest (RRoI), and introduces a significant performance improvement with an accuracy of 86.20. R^3Det [28] uses a progressive regression approach from coarse to fine granularity, and promotes the performance from 86.20 to 89.26. S^2A-Net align features to achieve better performance at an accuracy of 90.17. Our proposed method achieves the best performance overall

the compared methods, at an accuracy of 90.55.

V. Conclusion

In this paper, we proposed an anchor-free detection framework to detect arbitrary-oriented ships from remote sensing images by making full use of the prior information of ships. Our method detects ships by extracting the ship's center and head keypoints and regresses the ship's size at each center point. CHPDet avoids complex anchor design and computing relative to the anchor-based methods and can accurately predict angles in a large range (0°-360°). To improve the classification accuracy of the framework, we proposed to refine the detected probability according to the size of the ship. Experimental results demonstrate that our method achieves better accuracy and efficiency as compared with other state-of-the-art ship detectors, especially in complex situations.

References

[1] G. Cheng and J. Han, "A survey on object detection in optical remote sensing images," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 117, pp. 11–28, 2016.

[2] Z. Deng, H. Sun, S. Zhou, and J. Zhao, "Learning deep ship detector in sar images from scratch," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 57, no. 6, pp. 4021–4039, 2019.

[3] Z. Deng, H. Sun, S. Zhou, J. Zhao, L. Lei, and H. Zou, "Multi-scale object detection in remote sensing imagery with convolutional neural networks," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 145, pp. 3–22, 2018.

[4] M. Li, W. Guo, Z. Zhang, W. Yu, and T. Zhang, "Rotated region based fully convolutional network for ship detection," in *IGARSS 2018-2018 IEEE International Geoscience and Remote Sensing Symposium*. IEEE, 2018, pp. 673–676.

[5] W. Qian, X. Yang, S. Peng, Y. Guo, and C. Yan, "Learning modulated loss for rotated object detection," *arXiv preprint arXiv:1911.08299*, 2019.

[6] Y. Jiang, X. Zhu, X. Wang, S. Yang, W. Li, H. Wang, P. Fu, and Z. Luo, "R2cnn: Rotational region cnn for orientation robust scene text detection," *arXiv: Computer Vision and Pattern Recognition*, 2017.

[7] Z. Zhang, W. Guo, S. Zhu, and W. Yu, "Toward arbitrary-oriented ship detection with rotated region proposal and discrimination networks," *IEEE Geoscience and Remote Sensing Letters*, vol. 15, no. 11, pp. 1745–1749, 2018.

[8] S. Liu, Q. Du, X. Tong, A. Samat, and L. Bruzzone, "Unsupervised change detection in multispectral remote sensing images via spectral-spatial band expansion," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 12, no. 9, pp. 3578–3587, 2019.

[9] L. Liu, W. Ouyang, X. Wang, P. Fieguth, J. Chen, X. Liu, and M. Pietikäinen, "Deep learning for generic object detection: A survey," *International journal of computer vision*, vol. 128, no. 2, pp. 261–318, 2020.

[10] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 580–587.

[11] R. Girshick, "Fast r-cnn," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 1440–1448.

[12] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," in *Advances in Neural Information Processing Systems*, 2015, pp. 91–99.

[13] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask r-cnn," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 2961–2969.

[14] J. Dai, Y. Li, K. He, and J. Sun, "R-fcn: Object detection via region-based fully convolutional networks," in *Advances in Neural Information Processing Systems*, 2016, pp. 379–387.

[15] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 779–788.

[16] J. Redmon and A. Farhadi, "Yolo9000: better, faster, stronger," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 7263–7271.

[17] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "Ssd: Single shot multibox detector," in *European Conference on Computer Vision*. Springer, 2016, pp. 21–37.

[18] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 2980–2988.

[19] Z. Cai and N. Vasconcelos, "Cascade r-cnn: Delving into high quality object detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 6154–6162.

[20] K. Chen, J. Pang, J. Wang, Y. Xiong, X. Li, S. Sun, W. Feng, Z. Liu, J. Shi, W. Ouyang *et al.*, "Hybrid task cascade for instance segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 4974–4983.

[21] H. Law and J. Deng, "Cornernet: Detecting objects as paired keypoints," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 734–750.

[22] X. Zhou, D. Wang, and P. Krähenbühl, "Objects as points," *arXiv preprint arXiv:1904.07850*, 2019.

[23] Z. Tian, C. Shen, H. Chen, and T. He, "Fcos: Fully convolutional one-stage object detection," in *Proceedings of the IEEE International Conference on Computer Vision*, 2019, pp. 9627–9636.

[24] J. Ma, W. Shao, Y. Hao, W. Li, W. Hong, Y. Zheng, and X. Xue, "Arbitrary-oriented scene text detection via rotation proposals," *IEEE Transactions on Multimedia*, vol. PP, no. 99, p. 1, 2017.

[25] J. Ding, N. Xue, Y. Long, G. Xia, and Q. Lu, "Learning roi transformer for detecting oriented objects in aerial images," *arXiv: Computer Vision and Pattern Recognition*, 2018.

[26] X. Yang, J. Yang, J. Yan, Y. Zhang, T. Zhang, Z. Guo, S. Xian, and K. Fu, "Scrdet: Towards more robust detection for small, cluttered and rotated objects," 2018.

[27] W. Qian, X. Yang, S. Peng, Y. Guo, and C. Yan, "Learning modulated loss for rotated object detection," *arXiv preprint arXiv:1911.08299*, 2019.

[28] X. Yang, Q. Liu, J. Yan, A. Li, Z. Zhang, and G. Yu, "R3det: Refined single-stage detector with feature refinement for rotating object," *arXiv preprint arXiv:1908.05612*, 2019.

[29] X. Yang and J. Yan, "Arbitrary-oriented object detection with circular smooth label," *arXiv preprint arXiv:2003.05597*, 2020.

[30] X. Yang, L. Hou, Y. Zhou, W. Wang, and J. Yan, "Dense label encoding for boundary discontinuity free rotation detection," *arXiv preprint arXiv:2011.09670*, 2020.

[31] J. Han, J. Ding, J. Li, and G.-S. Xia, "Align deep features for oriented object detection," *arXiv preprint arXiv:2008.09397*, 2020.

[32] H. Wei, Y. Zhang, Z. Chang, H. Li, H. Wang, and X. Sun, "Oriented objects as pairs of middle lines," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 169, pp. 268–279, 2020.

[33] H. Wei, Y. Zhang, B. Wang, Y. Yang, H. Li, and H. Wang, "X-linenet: Detecting aircraft in remote sensing images by a pair of intersecting line segments," *IEEE Transactions on Geoscience and Remote Sensing*, 2020.

[34] Z. Shi, X. Yu, Z. Jiang, and B. Li, "Ship detection in high-resolution optical imagery based on anomaly detector and local shape feature," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 52, no. 8, pp. 4511–4523, 2013.

[35] Y. Freund and R. E. Schapire, "A decision-theoretic generalization of on-line learning and an application to boosting," *Journal of Computer and System Sciences*, vol. 55, no. 1, pp. 119–139, 1997.

[36] F. Yang, Q. Xu, and B. Li, "Ship detection from optical satellite images based on saliency segmentation and structure-lbp feature," *IEEE Geoscience and Remote Sensing Letters*, vol. 14, no. 5, pp. 602–606, 2017.

[37] Z. Liu, H. Wang, L. Weng, and Y. Yang, "Ship rotated bounding box space for ship extraction from high-resolution optical satellite images with complex backgrounds," *IEEE Geoscience and Remote Sensing Letters*, vol. 13, no. 8, pp. 1074–1078, 2017.

[38] Z. Liu, J. Hu, L. Weng, and Y. Yang, "Rotated region based cnn for ship detection," in *IEEE International Conference on Image Processing*, 2018.

[39] F. Yu, D. Wang, E. Shelhamer, and T. Darrell, "Deep layer aggregation," *arXiv*, 2017.

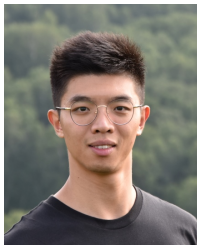
[40] A. Newell, K. Yang, and J. Deng, "Stacked hourglass networks for human pose estimation," *arXiv e-prints*, 2016.

[41] H. Law and J. Deng, "Cornernet: Detecting objects as paired keypoints," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 734–750.
[42] Z. Cao, T. Simon, S.-E. Wei, and Y. Sheikh, "Realtime multi-person 2d pose estimation using part affinity fields," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 7291–7299.
[43] Z. Liu, L. Yuan, L. Weng, and Y. Yang, "A high resolution optical satellite image dataset for ship recognition and some new baselines," in *International Conference on Pattern Recognition Applications and Methods*, vol. 2. SCITEPRESS, 2017, pp. 324–331.
[44] G.-S. Xia, X. Bai, J. Ding, Z. Zhu, S. Belongie, J. Luo, M. Datcu, M. Pelillo, and L. Zhang, "Dota: A large-scale dataset for object detection in aerial images," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 3974–3983.
[45] K. Li, G. Wan, G. Cheng, L. Meng, and J. Han, "Object detection in optical remote sensing images: A survey and a new benchmark," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 159, pp. 296–307, 2020.
[46] G. Cheng and J. Han, "A survey on object detection in optical remote sensing images," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 117, pp. 11–28, 2016.
[47] J. Ma, W. Shao, H. Ye, L. Wang, H. Wang, Y. Zheng, and X. Xue, "Arbitrary-oriented scene text detection via rotation proposals," *IEEE Transactions on Multimedia*, vol. 20, no. 11, pp. 3111–3122, 2018.
[48] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.
[49] G. S. Xia, B. Xiang, D. Jian, Z. Zhen, and L. Zhang, "Dota: A large-scale dataset for object detection in aerial images," 2017.
[50] T. Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PP, no. 99, pp. 2999–3007, 2017.
[51] Y. Xu, M. Fu, Q. Wang, Y. Wang, K. Chen, G. Xia, and X. Bai, "Gliding vertex on the horizontal bounding box for multi-oriented object detection," *arXiv: Computer Vision and Pattern Recognition*, 2019.
[52] K. Fu, Z. Chang, Y. Zhang, and X. Sun, "Point-based estimator for arbitrary-oriented object detection in aerial images," *IEEE Transactions on Geoscience and Remote Sensing*, 2020.
[53] J. Ma, W. Shao, H. Ye, L. Wang, H. Wang, Y. Zheng, and X. Xue, "Arbitrary-oriented scene text detection via rotation proposals," *IEEE Transactions on Multimedia*, 2018.

Feng Zhang received the B.E. degree in electronic information engineering from Harbin Institute of Technology(HIT), Harbin, China, in 2009, and the M.E. degree in information and communication engineering from National University of Defense Technology (NUDT), Changsha, China, in 2011. He is currently pursuing a Ph.D. degree from the College of Electronic Science and Technology, NUDT. His research interests focus on include remote sensing image processing, pattern recognition,and computer vision.

Xueying Wang received the B.S. degree in electronic information engineering from Beihang University, Beijing, China, in 2009, the M.S. and Ph.D. degrees in electronic science and technology from the National University of Defense Technology, Changsha, China, in 2011 and 2016. He is currently an Assistant Professor with the College of Electrical Science, National University of Defense Technology. His research interests include remote sensing image processing, pattern recognition.

Shilin Zhou received the B.S., M.S., and Ph.D. degrees in electrical engineering from Hunan University, Changsha, China, in 1994, 1996, and 2000, respectively. He is currently a Full Professor with the College of Electrical Science, National University of Defense Technology, Changsha. He has authored or co-authored over 100 referred papers. His research interests include image processing and pattern recognition.



Yingqian Wang received the B.E. degree in electrical engineering from Shandong University (SDU), Jinan, China, in 2016, and the M.E. degree in information and communication engineering from National University of Defense Technology (NUDT), Changsha, China, in 2018. He is currently pursuing a Ph.D. degree from the College of Electronic Science and Technology, NUDT. He has authored several papers in journals and conferences such as TPAMI, TIP, CVPR, and ECCV. His research interests focus on low-level vision, particularly on light field imaging and image super-resolution.