Fine-Grained Object Detection in Remote Sensing Images via Adaptive Label Assignment and Refined-Balanced Feature Pyramid Network

Junjie Song, Lingjuan Miao, Qi Ming, Zhiqiang Zhou, *Member, IEEE*, and Yunpeng Dong

***Abstract*—Object detection in high-resolution remote sensing images remains a challenging task due to the uniqueness of its viewing perspective, complex background, arbitrary orientation, etc. For fine-grained object detection in high-resolution remote sensing images, the high intra-class similarity is even more severe, which makes it difficult for the object detector to recognize the correct classes. In this article, we propose the refined and balanced feature pyramid network (RB-FPN) and center-scale aware (CSA) label assignment strategy to address the problems of fine-grained object detection in remote sensing images. RB-FPN fuses features from different layers and suppresses background information when focusing on regions that may contain objects, providing high-quality semantic information for fine-grained object detection. Intersection over Union (IoU) is usually applied to select the positive candidate samples for training. However, IoU is sensitive to the angle variation of oriented objects with large aspect ratios, and a fixed IoU threshold will cause the narrow oriented objects without enough positive samples to participate in the training. In order to solve the problem, we propose the CSA label assignment strategy that adaptively adjusts the IoU threshold according to statistical characteristics of oriented objects. Experiments on FAIR1M dataset demonstrate that the proposed approach is superior. Moreover, the proposed method was applied to the fine-grained object detection in high-resolution optical images of 2021 Gaofen challenge. Our team ranked sixth and was awarded as the winning team in the final.**

***Index Terms*—Feature pyramid network, fine-grained object detection, label assignment, remote sensing images.**

I. Introduction

OBJECT detection in high-resolution remote sensing image is to accurately locate and identify the object of interest. Automated analysis and understanding for remote sensing images have become critically important in many real-world applications, such as town planning, strategic deployment in the military field, and Earth observation [1], [2], [3], [4]. Thus, object detection in remote sensing images has a very broad application prospect.

Manuscript received 22 June 2022; revised 2 September 2022; accepted 16 November 2022. Date of publication 24 November 2022; date of current version 7 December 2022. This work was supported by the National Natural Science Foundation of China under Grant 62173040. *(Corresponding author: Zhiqiang Zhou.)*

The authors are with the School of Automation, Beijing Institute of Technology, Beijing 100081, China (e-mail: bitsongjj@gmail.com; miaolingjuan@bit.edu.cn; chaser.ming@gmail.com; zhzhzhou@bit.edu.cn; bitdyp@gmail.com).

Digital Object Identifier 10.1109/JSTARS.2022.3224558

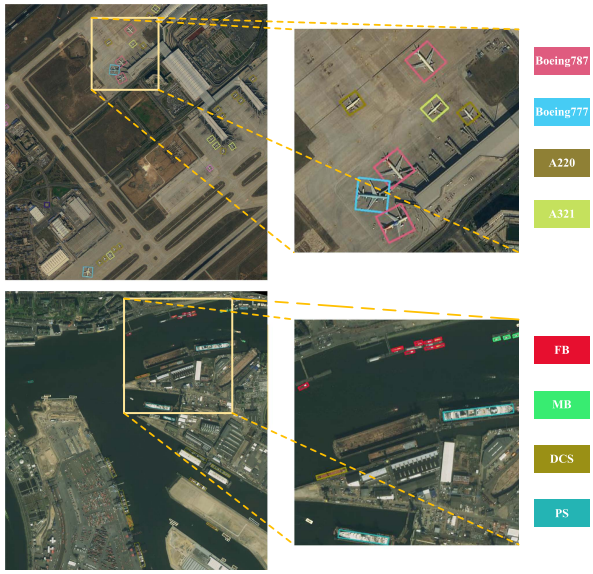

Fig. 1. Illustration of fine-grained objects detection, objects have high inter-class variation and low inter-class variation, which make object detection an even more challenging task.

In recent years, with the development of convolutional neural networks (CNN), the field of computer vision has grown considerably due to the powerful feature extraction capability of CNN. Various vision-based tasks including classification, object detection, and semantic segmentation have been able to achieve superior performance. A number of CNN-based object detectors [5], [6], [7], [8] have made significant progress and achieved excellent performance on MS COCO dataset [9] and PASCAL VOC dataset [10]. However, most of the existing techniques tend to suffer from dramatic performance degradation when applied to remote sensing images, mainly due to the difference between remote sensing images and natural scene images. Objects in remote sensing images are usually densely distributed, appear in arbitrary orientations, and have large scale variations, which make object detection an even more challenging task. As shown in Fig. 1, for fine-grained object detection, the high intra-class variation and low inter-class variation lead to limitations in the performance of various detectors.

To solve these issues, a number of approaches [11], [12], [13], [14], [15], [16], [17], [18], [19] have been developed. Feature pyramid network (FPN) [20] provides an effective solution to

the problem of large-scale variation in images. The hierarchical structure of FPN makes the feature maps at different levels contain feature information at different scales. In the FPN, information can be interacted between different layers, which effectively improve the accuracy of multiscale object detection. However, the FPN makes the semantic information between the feature maps of nonadjacent levels sparse when fusing the information of feature maps. Furthermore, objects in remote sensing images usually suffer from a large amount of noises, which greatly affects the performance of the detector. In fine-grained object detection, we need to feed high quality feature maps with richer semantic information into the detector.

Meanwhile, most of existing object detectors employ the Intersection of Union (IoU) as a matching metric to select the high-quality samples for classification and localization. The performance of the detector will be greatly affected if label assignment strategy is not appropriate. The fixed IoU threshold is used in RetinaNet [21], while the ATSS [22] set the IoU threshold dynamically by automatically selecting positive and negative training samples according to the statistical characteristics of the data. But the label assignment strategies applied directly to remote sensing images has some drawbacks, which does not make full use of the statistical characteristics of oriented objects. Moreover, there are a large number of narrow objects with arbitrary orientation in the remote sensing images. IoU is extremely sensitive to angle changes for narrow oriented objects, a small angular deviation leads to a dramatic drop in IoU. Fixed IoU threshold will lead to narrow oriented objects without sufficient positive samples, which limits the performance of the detector.

To tackle the above issues, we propose the refined and balanced feature pyramid network (RB-FPN) and center-scale aware (CSA) label assignment strategy. RB-FPN closes the semantic information gap between different layers of the FPN, and forces each layer of the network to learn the features of the objects at different resolutions. Moreover, the RB-FPN eliminates the complex background information and enhances the semantic representation of feature maps, and enhances the variance between different features. The obtained high-quality feature maps are more effective for the recognition of fine-grained objects. Then we propose a CSA label assignment strategy to automatically select positive samples according to statistical characteristics of oriented objects. The CSA label assignment strategy selects more high-quality positive samples during training. On the other hand, dynamically adjusting the IoU threshold according to the statistical characteristics of the ground truth (GT) boxes enhance the robustness of the detector. To summarize, the main contributions of this article are as follows.

1) A refined and balanced feature pyramid network is proposed to reduce the semantic information gap of FPN and suppress background information while focusing on regions that potentially contain objects. The obtained high-quality feature maps enable efficient fine-grained object detection.
2) A novel center-scale aware label assignment strategy is proposed to dynamically adjust the IoU threshold based on the IoU distribution around the GT and its aspect ratios.
3) Comprehensive experiments are conducted on the FAIR1M dataset of Gaofen Challenge to demonstrate the efficacy as well as the superiority of the proposed methods.

II. Related Works

A. Generic Object Detection

With the advancement of the deep learning techniques, object detection has achieved great progress owing to the powerful representative ability of deep convolutional neural networks. Most of the existing detectors can be divided into two types: 1) two-stage methods and 2) one-stage methods. The two-stage detector is a coarse-to-fine structure. In the first stage, a region proposal network (RPN) is used to generate region of interest (RoI) that potentially contains objects. In the second stage, category prediction and location regression are performed on the selected RoIs. The representative two-stage detectors are the pioneering RCNN family [5], [23], [24].

The simple architecture of one-stage detectors allows for tradeoffs between accuracy and speed and is more suitable for real-time detection tasks. One-stage detectors get rid of the complex regional proposal stage and predict the object instance categories and their locations directly from densely predesigned candidate boxes. One-stage detectors are popularized by SSD [7], YOLO family [6], [25], [26], [27], and RetinaNet [21].

FPN and other similar top–down structures [28], [29], [30], [31] are proposed to solve the problem of scale variations of objects. FPN takes advantage of the pyramid shape of convolution features and combines them in various resolutions to construct a feature pyramid with rich semantic information to recognize objects at different scales. PAFPN [32] adds a bottom–up fusion path to the FPN, fully exploiting the shallow features of the network. Liu et al. [33] proposed a data-driven strategy for pyramidal feature fusion method, which learns the way to spatially filter conflictive information to suppress the inconsistency.

Many recent works have refined the process of label assignment to further improve detection performance. ATSS [22] automatically selects positive and negative training samples based on the statistical characteristics of the objects. Kim et al. [34] assume that the distribution of joint loss for positive and negative samples follows the Gaussian distribution. Hence, it uses Gaussian mixture model to fit the distribution of training samples, and then uses the center of positive sample distribution as the positives/negatives division boundary. Autoassign [35] tackles label assignment in a fully data-driven manner by automatically determining the positives/negatives in both spatial and scale dimensions. OTA [36] views the label assignment process as an optimal transportation problem, and the number of anchor boxes assigned to each GT is dynamically calculated according to a global anchor box regression state.

B. Oriented Object Detection in Remote Sensing Images

Oriented object detection has attracted plenty of interest, especially in remote sensing images. Oriented object detectors locate and classify objects with oriented bounding

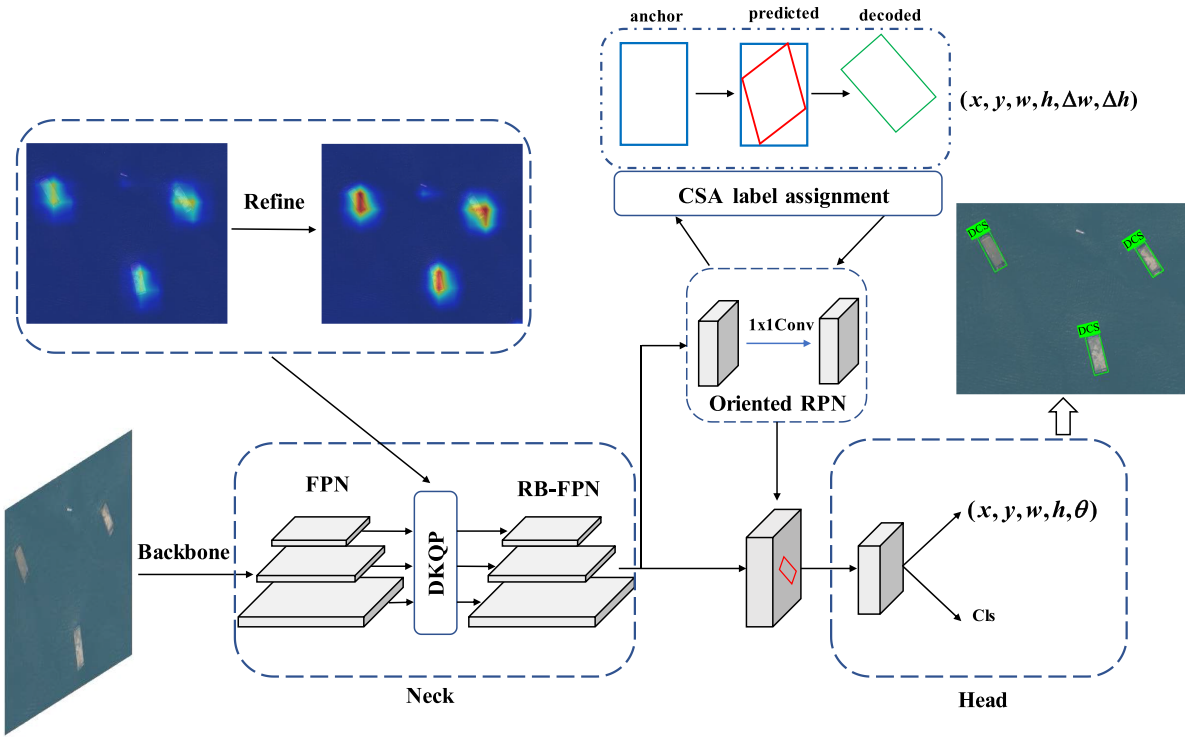

Fig. 2. Architecture of the proposed method. The output feature maps of FPN are balanced and refined through RB-FPN. CSA label assignment is used to select more promising samples and prevent filtering out high quality samples. The DKQP attention module is designed to suppress complex background information while focusing on regions that may contain objects.

boxes, which provide more accurate orientation information of objects. Yang et al. [14] built an oriented object detection method on the generic object detection framework of faster R-CNN. Xu et al. [37] proposed the Gliding Vertex, which learns the four vertex gliding offsets on the regression branch to achieve oriented object detection. Wei et al. [38] proposed a one-stage, anchor-free, and nms-free model (O^2-DNet) to detect oriented objects by predicting a pair of midlines inside each object. ReDet [39] encodes rotation equivariance and rotation invariance in image features to increase the accuracy of oriented object detection. Ming et al. [40] designed a new label assignment strategy for one-stage oriented object detection based on RetinaNet [21]. It assigns the positive or negative anchors dynamically through a new matching strategy. Zhang et al. [41] proposed aspect ratio-guided label assignment to adjust the IoU threshold, and aspect ratio guided IoU loss is designed to automatically adjust the weights of the angle regression.

In recent years, an increasing number of works have focused on fine-grained object detection in remote sensing images. Sun et al. [42] proposed a cascaded hierarchical object detection network (CHODNet). CHODNet consists of four stages: 1) feature refinement network, 2) region proposal network, 3) proposals refinement network, and 4) fine-grained detection network. CHDONet learns external and internal representations independently from the dataset using a cascaded hierarchical structure. Zhang et al. [43] proposed a multiscale semantic segmentation feature fusion module, which merges the semantic features with the original features layer by layer to distinguish the foreground from the cluttered background. R^2 IPoints [44] employs a set of category-aware points to encode spatial and semantic information oriented to arbitrary objects.

III. Proposed Method

Oriented R-CNN (ORCNN) [45] is a superior two-stage oriented object detector. Our method is based on ORCNN and consists of the backbone network, RB-FPN, CSA label assignment strategy, oriented RPN, and R-CNN detection head. The proposed framework is illustrated in Fig. 2. RB-FPN provides higher quality feature maps for fine-grained object detection by eliminating background information and balancing the feature maps in the FPN. CSA label assignment strategy is designed to select potentially high quality samples based on the statistical characteristics of GT box. Overall, the model predicts the location and fine-grained category information of objects in remote sensing images more efficiently. More details are to be discussed in the following subsections.

A. Refined and Balanced Feature Pyramid Network

Deep features in backbones are with more semantic information, while the shallow low-level features are more descriptive in terms of detailed information. The top–bottom hierarchical network structure of FPN allows the feature maps of different layers to deliver information. But the sequential manner in this methods will make fused features focus more on adjacent resolution. The semantic information contained in nonadjacent levels is diluted in each fusion during the information flow.

Thus, it is crucial to utilize the features at different levels. Besides, the complex background information in remote sensing images usually affects the performance of the detector. It is important to effectively eliminate the background information to provide a higher quality feature map for the subsequent tasks.

The proposed refined and balanced feature pyramid network (RB-FPN) makes the information of different levels of feature maps more balanced, enriching the semantic information in the feature maps. First, the feature maps of the different layers are resized to the same resolution, and the resized feature maps are then summed in pixelwise. Second, the integrated feature map is refined by the proposed deformable key–query–position (DKQP) attention module. Finally, the refined feature map is added to the original feature maps in the FPN by upsampling and downsampling, respectively.

Self-attention mechanism has been widely used in the field of computer vision and performed very well. In determining the attention weight assigned to a key for a given query, several properties of the input are usually considered. One is the content of the query. For self-attention, the query content can be a feature at a query pixel in an image. The second is the content of the key, where the key may be a pixel within the local neighborhood of the query. The third is the relative position of the query and key. Based on these input properties, Dai et al. [46] argue that the attention weights are expressed as a sum of four terms (ϵ_1, ϵ_2, ϵ_3, ϵ_4). Specifically, these factors are the query and key content (ϵ_1), the query content and relative position (ϵ_2), the key content only (ϵ_3), and the relative position only (ϵ_4).

In our DKQP attention module, we only focus on ϵ_2 and ϵ_3 of the attention factors, since the performance gain provided by other factors is insignificant when the computational overhead is taken into account [47]. In addition, deformable convolution [48] efficiently exploits sparse local locations and captures high-quality features, and is designed for capturing regions of interest. Inspired by the properties of deformable convolution, we use deformable convolutions and learnable vectors in the self-attention module to focus on regions that may contain objects, thereby obtaining high-quality feature information. The proposed DKQP attention focus more on potential object regions to enhance the semantic information of the feature map.

In deformable convolution, for each position p_i in the output feature map, the output $y(p_i)$ is defined as

$$y(p_i) = \sum_{p_n \in R} w(p_n) x(p_i + p_n + \Delta p_n) \tag{1}$$

where $w(p_n)$ is the weight for position p_n, $x(p)$ is the feature at position p, p_n enumerates all the positions in grid R, and Δp_n is the offset of the convolution sampling location. As illustrated in Fig. 3, key content attention is ϵ_3. The deformable convolution and another learnable vector are calculated to obtain ϵ_2^{kqp}, it can be formulated as follows:

$$\epsilon_2^{\text{kqp}} = l_m^T x_q, \tag{2}$$

where l_m is a learnable vector and x_q is the reshaped output of the deformable convolution. Generalized attention formulation

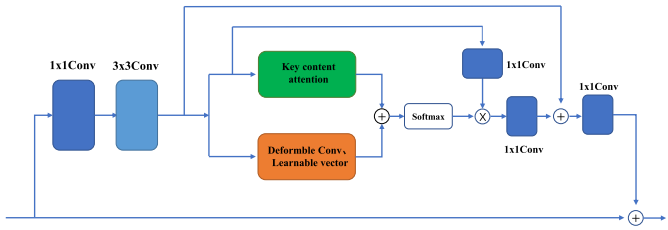

Fig. 3. Structure of the DKQP attention.

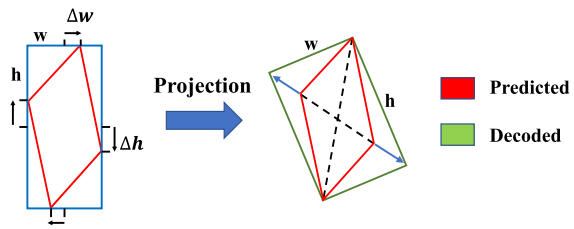

Fig. 4. Illustration of the process of projection. Red box is a parallelogram proposal generated by oriented RPN, the green box is the projected proposal.

is as follows:

$$y(q) = \sum_{m=1}^{M} W_m \cdot \left[\sum_{k \in \Omega_q} A_m^{\text{deform}}(q, k, z_q, x_k) \cdot \overline{W}_m x_k \right]. \tag{3}$$

Here, $A_m^{\text{deform}}(q, k, z_q, x_k)$ denotes the attention weights in the mth attention head, which is calculated from ϵ_2^{kqp} and ϵ_3, z_q is query content at index q, and x_k is key content at index k. W_m and $\overline{W}_m$ are learnable weights. Ω_q specifies the supporting key region for the query. In DKQP attention, we use deformable convolution and learnable vector instead of query content and relative position. The feature capture capability of deformable convolution allows the model to focus more on the RoI Learnable vector captures global positional bias between the key and deformable convolution elements. And DKQP attention brings a lower computational overhead by sampling a sparse set of key element for each query making the complexity linear to the query element number.

RB-FPN reduces the semantic gap between different scale feature layers of the traditional FPN while forcing each layer of the network to learn the features of the objects at different resolutions. In refine stage, our DKQP attention module suppresses complex background information while focusing on regions that potentially contain objects.

B. Center-Scale Aware Label Assignment

In our model, the oriented RPN uses six parameters $(x, y, w, h, \Delta w, \Delta h)$ to denote an oriented proposal. The oriented proposal needs to be projected into a oriented bounding box, as shown in Fig. 4. During the projection, there will be misalignment in the region represented by the box. But the center position does not change during the projection, so the center position of the prediction box is particularly significant. If the center distance between the anchor and GT box are relatively far when selecting positive samples, the quality of the learned

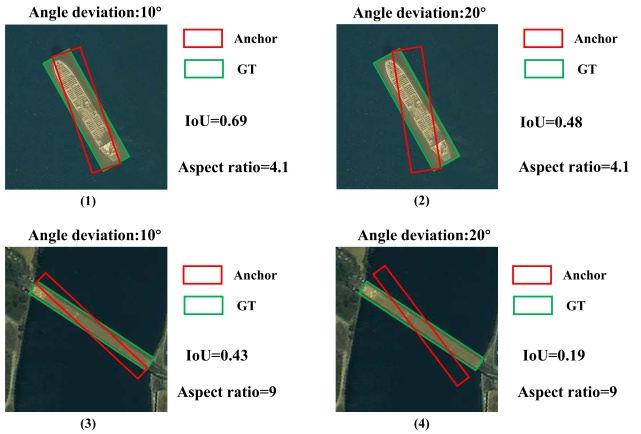

Fig. 5. Example of IoU for angle deviation of oriented bounding boxes with different aspect ratios.

samples will be inferior. In sample selection, the samples located near the center of the GT box are more representative. Moreover, there are a large number of narrow oriented objects in the remote sensing images. As shown in Fig. 5, IoU is very sensitive to the narrow oriented object, at the same aspect ratio, a small angular deviation causes a sharp drop in IoU. The anchor may still be a potentially high quality sample at this position, we expect to select this potentially high quality sample. But it will be filtered out because the IoU between the anchor and the GT box is less than a fixed threshold.

To solve the above problems, we propose the CSA label assignment strategy, which adaptively adjusts the threshold of IoU according to statistical characteristics of oriented objects. The CSA label assignment algorithm is shown in Algorithm 1. For each GT box g on the image, we first find out its candidate positive samples. On each pyramid level, we select k ($k = 1, 2, 3, 4. . .15$, default $k = 9$) anchor boxes whose centers are closest to the center of GT box based on Euclidean distance. After that, we compute the IoU between these candidates and the GT boxes as D_g, whose mean is computed as IoU_m. Then, calculate the aspect ratio of each GT box as r. The aspect ratio of GT box is mapped to a constant value greater than or equal to 1 by the function $g(r)$, The specific function $g(r)$ is as follows:

$$g(r) = \begin{cases} r & \text{if } r > 1 \\ 1/r & \text{otherwise} \end{cases} \tag{4}$$

where $r = \frac{w}{h}$, w and h are the width and height of the GT box, respectively. Then the function $f(r)$ is used as a mapping function for the aspect ratio of GT box. The function is to allow larger aspect ratios to have lower value and mine enough higher potential samples. The function is defined as follows:

$$f(r) = \frac{1}{1.5 + ln(g(r))} \tag{5}$$

where $g(r)$ is obtained from (4), with mapping function $g(r)$ and the computed average IoU, the final adaptive IoU threshold is available via CSA label assignment strategy. The specific IoU

Algorithm 1: Center-Scale Aware Label Assignment.

Input: $\mathcal{P}$ is the number of feature pyramid levels;
$\mathcal{G}$ is a set of ground truth boxes on the imag;
$\mathcal{A}$ is a set of all anchor boxes;
$\mathcal{A}_i$ is a set of anchor boxes from the ith pyramid levels;
$f(r)$ is a function that maps the aspect ratio
k is a hyperparameter with a default value of 9;
Output: $\mathcal{S}_p$ is a set of positive samples;
$\mathcal{S}_n$ is a set of negative samples;

for each ground truth $g \in \mathcal{G}$ **do**
  build an empty set for candidate positive samples of the ground truth g: $C_g \leftarrow \emptyset$;
  for each level $i \in \mathcal{P}$ **do**
    $S_i \leftarrow$ select k anchors from $\mathcal{A}_i$ whose center are closest to the center of ground truth g based on Euclidean distance;
    $C_g = C_g \cup S_i$;
  end for;
  compute IoU between C_g and g : $D_g = IoU(C_g, g)$;
  compute mean of D_g : $\text{IoU}_m = \text{Mean}(D_g)$;
  r $\leftarrow$ compute the aspect ratio of ground truth g
  compute IoU threshold $T_{\text{IoU}} := \text{IoU}_m * f(r)$;
  for each candidate c $\in$ C **do**
    if IoU $> T_{\text{IoU}}$ and center of c in g **then**
      $\mathcal{S}_p = \mathcal{S}_p \cup c$;
    end if
  end for;
end for; $\mathcal{S}_n = \mathcal{A} - \mathcal{S}_p$;
return $\mathcal{S}_p, \mathcal{S}_n$;

threshold is as follows:

$$T_{\text{IoU}} = f(r) * \text{IoU}_m \tag{6}$$

where $f(r)$ is obtained from (5), IoU_m is the mean value of IoU of candidate proposals around GT box. Finally, we select these candidates whose IoU are greater than or equal to the threshold T_{IoU} as positive samples.

The proposed CSA label assignment dynamically adjusts the threshold of IoU according to statistical characteristics of the GT boxes. Using the CSA label assignment strategy, oriented objects with large aspect ratios have smaller thresholds, thus ensuring the potential samples are selected. On the other hand, CSA label assignment ensures the number of positive samples changes dynamically according to statistical characteristics of GT boxes, which help to avoid the training loss being dominated by massive negatives. It is worthy of mentioning that the proposed label assignment strategy is only used for training, which does not incur the computational load at the inference stage.

C. Training

Our method consists of an oriented RPN and an R-CNN detection head. It is a two-stage detector, where the first stage generates high-quality oriented proposals in a nearly cost-free manner and the second stage is R-CNN detection head for proposal classification and regression. Next, we describe the

Fig. 6. Detection results of our method on the FAIR1M dataset.

loss function and representation of oriented RPN and R-CNN detection head in detail. The oriented RPN uses six parameters $(x, y, w, h, \Delta w, \Delta h)$ to denote an oriented proposal. For bounding box regression, we adopt the affine transformation, which is formulated as follows:

$$
\begin{aligned}
u_x &= \frac{(x - x_a)}{w_a},\ u_y = \frac{(y - y_a)}{h_a} \\
u_w &= \log\left(\frac{w}{w_a}\right),\ u_h = log\left(\frac{h}{h_a}\right) \\
u_{\Delta w} &= \frac{\Delta w}{w},\ u_{\Delta h} = \frac{\Delta h}{h}
\end{aligned} \tag{7}
$$

$$
\begin{aligned}
u_x^* &= \frac{(x^* - x_a)}{w_a},\ u_y^* = \frac{(y^* - y_a)}{h_a} \\
u_w^* &= log\left(\frac{w^*}{w_a}\right),\ u_h^* = log\left(\frac{h^*}{h_a}\right) \\
u_{\Delta w}^* &= \frac{\Delta w^*}{w^*},\ u_{\Delta h}^* = \frac{\Delta h^*}{h^*}
\end{aligned} \tag{8}
$$

where (x, y), w, and h are the center coordinate, width, and height of external rectangle, respectively. Specifically, x_a, x, x^* represent values related to anchors, the predicted boxes, and the GT boxes, the same for y_a, y, y^*. Δw and Δh are the offsets of the top and right vertices of the prediction box and anchor relative to the top and left midpoints. Δw^* and Δh^* are the

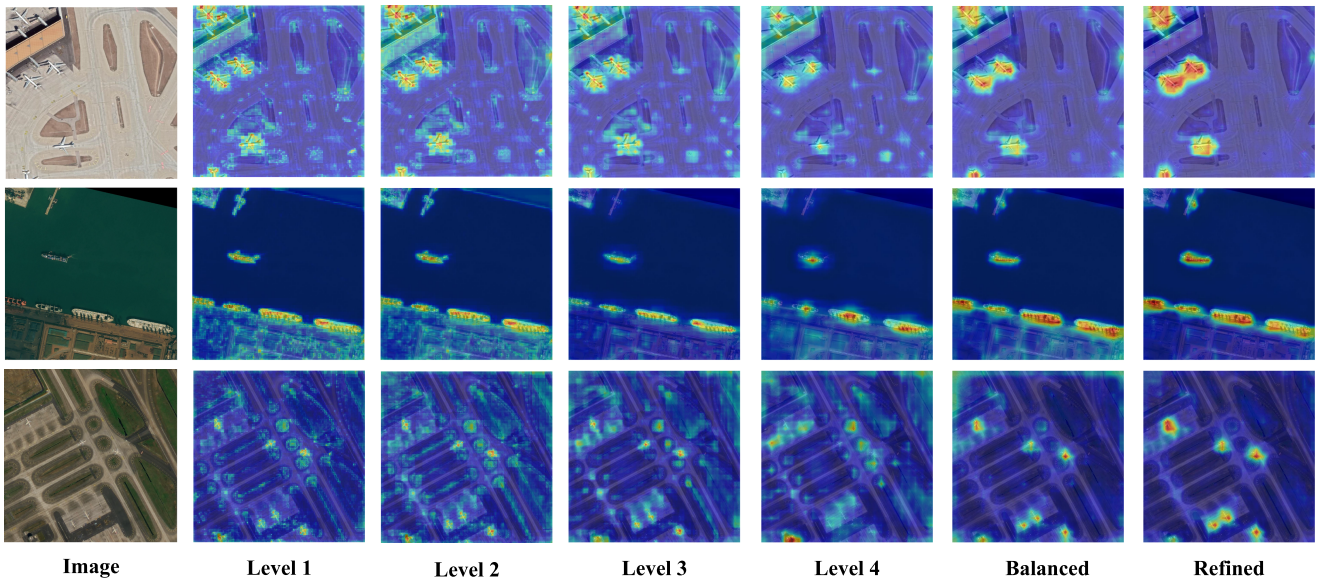

Fig. 7. Visualization of different layer feature maps in FPN. The balanced feature map is obtained by fusing different layers of feature maps. The refined feature map represents the balanced feature map optimized by the DKQP attention module. Brighter regions indicate higher attention responses. The proposed DKQP attention module suppresses background information while focusing on regions that potentially contain objects.

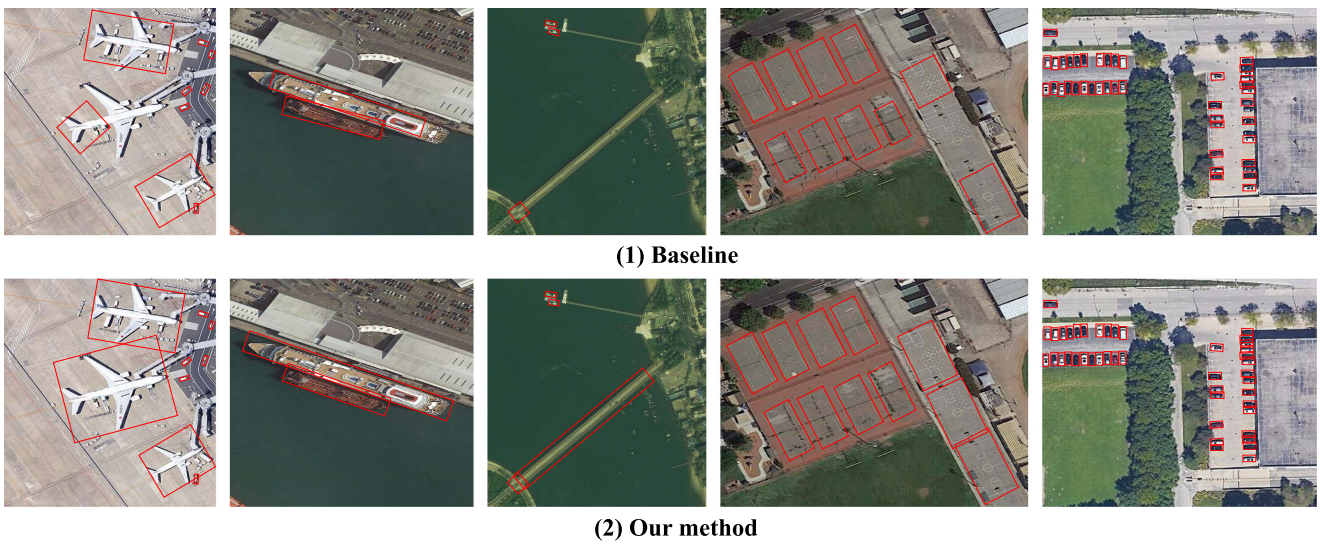

Fig. 8. Comparison of the detection results of the objects with large aspect ratios between the ORCNN (baseline) and our method on the FAIR1M dataset. The first row is the detection results of the baseline, and the second row is our CSA method.

offsets of the top and right vertices of GT and anchor relative to the top and left midpoints.

Following is the loss function to train oriented RPN:

$$L_{\text{rpn}} = \frac{1}{N_{\text{cls}}} \sum_{i} L_{\text{cls}}(p_i, p_i^*) + \lambda_1 \frac{1}{N_{\text{reg}}} \sum_{i=1}^{N} L_{\text{reg}}(u_i, u^*). \quad (9)$$

Here, i is the index for anchors, p_i^* is the GT label of the ith anchor, p_i is the output of the classification branch of oriented RPN. u_i^* is the supervision offset of the GT box relative to ith anchor, u_i indicate outputs of the regression branch of R-CNN detection head. L_{cls} is the cross entropy loss, L_{reg} is the Smooth L1 loss. λ_1 is the balance parameters (default by 1).

The R-CNN detection head uses the five parameters (x, y, w, h, θ) to represent an oriented bounding box. The bounding box regression can be described by the following formulas:

$$\begin{aligned} v_x &= \frac{(x - x_p)}{w_p}, \; v_y = \frac{(y - y_p)}{h_p} \\ v_w &= log\left(\frac{w}{w_p}\right), \; v_h = log\left(\frac{h}{h_p}\right), \; v_\theta = \theta - \theta_p \quad (10) \\ v_x^* &= \frac{(x^* - x_a)}{w_p}, \; v_y^* = \frac{(y^* - y_p)}{h_p} \\ v_w^* &= log\left(\frac{w^*}{w_p}\right), \; v_h^* = log\left(\frac{h^*}{h_p}\right), \; v_\theta^* = \theta^* - \theta_p. \quad (11) \end{aligned}$$

Here (x, y), w, and h are the center coordinate, width, and height of external rectangle, respectively. Specifically, $x_p,\ x,\ x^*$, represent values related to proposal samples, the predicted box, and the GT box, the same for $y_p,\ y,\ y^*$, respectively, the θ and θ^* denote the angle of the GT box and the angle of the proposal box.

Following is the loss function to R-CNN detection head:

$$L_{\text{head}} = \frac{1}{N_{\text{cls}}} \sum_i L_{\text{cls}}(p_i, p_i^*) + \lambda_2 \frac{1}{N_{\text{reg}}} \sum_{i=1}^{N} L_{\text{reg}}(v_i, v_i^*) \quad (12)$$

where L_{cls} is the cross entropy loss. L_{reg} is the Smooth L1 loss. λ_2 is the balance parameters (default by 1). i is the index for proposal, v_i^* is the supervision offset of the GT box relative to ith proposal, v_i indicate outputs of the regression branch of R-CNN detection head. p_i^* is the GT box label of the ith anchor, p_i is the output of the classification branch of R-CNN detection head.

IV. Experiments

A. Dataset

Based on the 2021 Gaofen Challenge, we conducted experiments on the FAIR1M dataset [42]. FAIR1M dataset is a large-scale dataset for fine-grained object detection in remote sensing images. Images in the FAIR1M dataset are with a spatial resolution ranging from 0.3 to 0.8 m. In FAIR1M dataset, there are more than 40 000 remote sensing images with 1 million instances from Gaofen satellites and Google Earth platform. Each image is of the size in the range from 1000 × 1000 to 10000 × 10000 pixels and contains objects exhibiting a wide variety of scales, orientations, and shapes. All images are annotated with oriented bounding boxes and with respect to 5 categories and 37 subcategories. The types of airplanes include Boeing 737 (737), Boeing 777 (777), Boeing 747 (747), Boeing 787 (787), Airbus A320 (A320), Airbus A220 (A220), Airbus A330 (A330), Airbus A350 (A350), COMAC C919 (C919), COMAC ARJ21(ARJ21), and other-airplane (OA). There are eight specific categories for ships, including liquid cargo ships (LCS), dry cargo ships (DCS), motorboat (MB), fishing boat (FB), passenger ship (PS), tugboat (TB), engineering ship (ES), warship (WS), and other-ship (OS). There are nine specific categories of vehicles, including small car (SC), bus, cargo truck (CT), dump truck (DT), van, trailer (TL), tractor (TR), excavator (EX), truck tractor (TT), and other-vehicle (OV). Courts includes basketball court (BC), tennis court (TC), football field (FF), baseball field (BF), and roads includes intersection (IN), roundabout (RA), bridge (BR).

B. Implement Details

We choose ResNet-50 [49] with FPN as the backbone network for ablation experiments and hyperparameters of these models are set to default values if not specified. We conduct the experiments on a server with four RTX 3090 GPUs using a total batch size of eight (two images per GPU) for training. We use a single RTX 3090 GPU for inference. The experimental results are produced on the mmdetection platform. The stochastic gradient descent (SGD) optimizer is used in training. The initial learning rate is set to 0.01 with the warming up for 500 iterations, and the learning rate is decreased by a factor of 0.1 at each decay step. The momentum and weight decay are set to 0.9 and 0.0001, respectively. We train the models with 12 epochs for FAIR1M dataset. The experimental environment was ubuntu 18.04, torch 1.7.0, and cuda 11.0 for the model training.

For FAIR1M dataset, we select 16 488 images as the training set and 8137 images as the testing set. The test results are submitted to the ISPRS Benchmark online validation platform. We first convert the annotations to DOTA dataset [51] format. Then, we crop the original images into patches with 800 × 800, the pixel overlap between two adjacent patches is 200. With regard to multiscale training and testing, we first resize the original images into three scales (0.5, 1.0, and 1.5) and then crop them into 800 × 800 patches with the stride of 200. We also apply random flipping and random rotation argument method during training. At the testing stage, we conduct the same data augmentation on the images to ensure consistency between training and testing.

C. Evaluation Metric

In the task of object detection, each image may contain objects of multiple categories. Therefore, a measure of detector performance is needed to validate the localization and classification capabilities. The average precision (AP) and mean average precision (mAP) are the most commonly used evaluation metrics. AP is determined by recall and precision, where recall refers to the ability of the model to find all objects, and precision refers to the ability of the model to correctly identify the detected objects. Each category uses a PR curve (P refers to precision and R refers to recall) to calculate the AP. There are currently two versions of the evaluation metric, PASCAL VOC2007 metrics and PASCAL VOC2012 metrics. We evaluate the models at the testing set in terms of PASCAL VOC07 metrics.

D. Comparisons With State of the Art

We compare the proposed approach with other state-of-the-art methods on FAIR1M dataset. To ensure the independence of the training, the results of these models are submitted to the online validation site of ISPRS benchmark to evaluate performance. Note that all methods adopt ResNet-50 as the backbone network, our state-of-the-art experiments which adopt Swin-Transformer (Swin-T) backbone network. As shown in Table I, our method obtains 42.62% mAP, which outperforms the baseline model by 2.4% mAP. With limited data augmentation (i.e., multiscale data and random rotation), our approach reaches 45.18% mAP. The backbone of the model is replaced with Swin-T [52], our method achieves 47.58% mAP, surpassing almost all recent state-of-the-art detectors. Swin-T has a powerful feature extraction capability and focuses on the global information in the feature map, which effectively distinguishes the features between different categories, so it has a higher performance in some categories. We visualize some detection results in Fig. 6. It can also be observed

TABLE I
COMPARISON OF THE PROPOSED APPROACH WITH THE STATE-OF-THE-ART APPROACH IN THE FAIR1M DATASET ON THE ISPRS BENCHMARK ONLINE VALIDATION DATASET

Coarse Category	Fine-Grained Category	RetinaNet [21]	S^2ANet [50]	Faster R-CNN [5]	Gliding Vertex [37]	RoI Trans. [18]	Baseline	Ours	Ours*	Ours
Backbone		ResNet-50	R50	ResNet-50	ResNet-50	ResNet-50	ResNet-50	ResNet-50	ResNet-50	Swin-T
mAP		27.6736	36.1215	33.6988	35.8624	38.2674	40.2257	42.6228	45.181	47.5753
FPS		20.8	20.1	19.2	18.4	18.2	18.3	19.0	19.0	9.5
Airplane	Boeing737	35.0100	36.0389	36.0537	36.3221	35.8354	35.0958	39.1589	45.4473	43.9079
	Boeing747	83.7229	84.4201	85.1864	82.6137	82.7387	84.4681	86.7031	86.7808	86.9736
	Boeing777	12.6413	12.0861	12.4484	11.2893	12.8081	14.8808	16.1724	14.3606	20.0669
	Boeing787	36.6836	52.3170	45.3512	48.6875	43.9033	48.8179	49.1561	50.4617	59.3205
	C919	1.4367	5.8145	15.4492	24.4795	15.7698	17.7265	13.5979	25.3663	27.9089
	A220	45.4363	46.3362	49.4960	50.0075	48.6819	46.2137	49.1840	48.7603	55.0686
	A321	64.9481	67.7965	63.1642	65.2655	67.3503	67.4341	66.9068	69.3679	70.5426
	A330	58.5229	66.3129	65.8944	69.9844	65.5620	69.0000	71.4291	70.8829	72.8709
	A350	71.4517	72.3102	62.6917	65.1758	62.9163	68.6785	74.3388	74.2677	76.0312
	ARJ21	3.5982	2.3962	31.2533	33.2417	33.6010	35.0287	28.9988	28.5255	36.8108
Ship	Passenger Ship	3.8265	6.9292	6.2424	8.9216	15.2002	16.1047	18.3813	21.9315	22.8528
	Motorboat	22.0295	52.5785	44.3727	52.0388	58.0448	60.8346	68.4102	70.1976	73.1209
	Fishing Boat	2.1218	7.3588	3.7101	5.1114	9.3686	9.4215	10.5547	12.3105	12.0027
	Tugboat	13.3417	16.0891	26.0481	28.4913	30.1663	35.4589	38.2720	34.1890	40.2088
	Engineering Ship	9.1074	10.0499	6.8826	9.7272	10.8670	12.2476	11.8316	13.9355	14.0499
	Liquid Cargo Ship	4.3720	22.8776	9.5006	15.6694	19.2780	20.0353	25.1245	27.5677	26.1598
	Dry Cargo Ship	14.4898	37.8156	17.7820	26.7470	33.0237	35.4302	38.4079	39.9888	41.3548
	Warship	3.8067	27.8149	6.3662	13.6678	24.9047	26.3109	34.717	35.9528	40.7024
Vehicle	Small Car	41.9096	65.1249	51.4416	49.5293	57.7326	58.1764	70.7494	75.2777	75.1231
	Bus	5.5547	13.8452	21.0015	22.0365	31.2258	34.4643	36.1439	52.9540	51.3328
	Cargo Truck	20.6871	38.9427	32.8857	36.6875	42.4586	44.7412	44.9402	52.3273	54.2142
	Dump Truck	16.5438	41.7978	40.0400	39.5190	45.2607	47.8985	50.1990	58.9680	59.4030
	Van	34.0930	60.8419	45.9618	43.6492	54.4889	54.8711	70.7682	75.3280	75.8658
	Trailer	0.3293	5.0758	7.8162	11.6542	15.5435	15.7067	16.7536	19.4164	21.4032
	Tractor	0.3593	1.3878	3.7741	2.8967	3.5453	4.7180	1.6773	5.5233	2.7527
	Excavator	0.5198	8.5986	9.2804	12.4875	12.7826	15.2655	17.2358	20.0231	23.4754
	Truck Tractor	0.0112	0.5838	1.7096	3.6607	2.5933	1.5491	0.4875	1.7287	4.1933
Court	Basketball Court	22.2838	35.4687	39.9219	39.8457	42.8666	48.909	54.588	53.7384	58.6447
	Tennis Court	78.6226	78.5472	76.9730	76.9804	78.4010	80.3171	80.4906	80.7676	87.9879
	Football Field	59.4606	62.0729	52.3577	50.7895	59.2985	58.2778	65.6882	62.3604	70.0206
	Baseball Field	86.4601	88.8023	87.5563	86.8527	86.6019	88.7349	88.9209	89.0139	90.1002
Road	Intersection	57.3259	55.6658	57.1100	58.5871	58.1779	60.1240	56.6370	62.7909	61.2788
	Roundabout	20.3018	23.1784	22.2822	20.4891	19.3371	25.2627	20.3565	21.6325	27.2243
	Bridge	9.8908	20.8564	7.7526	16.2131	20.7577	25.4700	32.1948	34.2490	34.5861

Ours denotes ORCNN with the proposed CSAsa and RB-FPN. * means multiscale training and testing.

that the proposed method can accurately detect densely arranged objects. The proposed RB-FPN provides high-quality feature maps that can effectively identify categories. The proposed CSA provides more high-quality samples that accurately learn the bounding box of the object.

E. Ablation Studies

Effectiveness of RB-FPN: We conduct ablation experiments on the online verification of ISPRS Benchmark to evaluate the effectiveness of the proposed RB-FPN. We use the RetinaNet with an orientation prediction in the regression branch

TABLE II
PERFORMANCE OF RB-FPN WITH DIFFERENT BACKBONES AND MODELS

Method	Backbone	RB-FPN	mAP
RetinaNet-obb	ResNet-50		27.67
	ResNet-50	✓	28.92
	ResNet-101		27.87
	ResNet-101	✓	29.60
ORCNN	ResNet-50		40.22
	ResNet-50	✓	42.23
	ResNet-101		40.76
	ResNet-101	✓	42.57

TABLE III
EXPERIMENT OF DIFFERENT METHODS, IE., CSA AND RB-FPN IN FAIR1M DATASET. WE CHOOSE ORCNN AS THE BASELINE

	Baseline	Different settings of model		
RB-FPN		✓		✓
CSA			✓	✓
mAP	40.22	42.24	42.23	42.62
GFLOPs	134.51	134.81	134.51	134.85
Parameters	41.16 M	41.99 M	41.16 M	41.99 M

TABLE IV
PERFORMANCE OF LABEL ASSIGNMENT STRATEGY WITH DIFFERENT METHODS

Modules	Baseline	Different methods	
CA	ORCNN+ResNet-50+FPN	✓	
CSA			✓
mAP	40.22	41.46	42.23

(RetinaNet-obb) and ORCNN as our baseline method. First, we directly adopt the official implementation to reproduce the RetinaNet-obb and ORCNN, and then use RB-FPN to replace FPN in the model. As shown in Table II, in ORCNN with ResNet-50, using RB-FPN can achieve 42.23% mAP, about 2.01% mAP higher than the baseline method. With a stronger backbone ResNet-101, RB-FPN further improves the performance by 1.81% mAP. RB-FPN consistently increases the accuracy of RetinaNet-obb and ORCNN with different backbones. In addition, we compare our RB-FPN with traditional FPN during training. Table III shows that the FLOPs increase by 0.3 G, but performance improved by 2.02% mAP, and the results demonstrate a large effect gain with a few parameters and computational load. As shown in Fig. 7, the feature maps of each layer in FPN contain a large amount of noisy information and do not effectively distinguish between background and foreground information. The integrated feature map refined by DKQP attention module has very high response on the objects, which effectively eliminate the background information and enhance the semantic information. RB-FPN balances the semantic information of each layer in the FPN and focuses on regions that may contain objects, thus extracting features that are more beneficial to the detector.

Effectiveness of CSA: We also perform ablation experiments on the online verification of ISPRS Benchmark to evaluate the effectiveness of our proposed CSA. CA (center aware) consider only the *priori* information of the center distance between anchor and GT box to adaptively adjust the IoU threshold, CSA introduce the prior information of the aspect ratio of the GT box based on the CA. Experimental results for different label assignment strategies are shown in Table IV. CA label assignment strategy achieves 41.46% mAP, about 1.2% mAP higher than the baseline method. With CSA label assignment, our method obtain 42.23% mAP, which brings about 2.01% mAP gain over the baseline. Compared with the baseline, these two label assignment strategies yield significant improvements in performance, which also proves that the center distance and the aspect ratio of GT box are important information for label assignment. In addition, we compare our CSA scheme with the Max-IoU scheme during training. As shown in Table III, CSA label assignment strategy is not only an effective method with high detection accuracy but also an efficient scheme in both speed and parameters. As shown in Fig. 8, the visualization results show that the baseline method tends to generate false negatives or cannot accurately detect oriented objects [see Fig. 8 (1)], while our approach has better performance to those oriented objects. We argue that adopting CSA label assignment makes it easier for the network to select enough positive samples, thus making the detector more robust. The experimental results show that our CSA label assignment is effective in compensating for potential samples, and the detector performance is effectively improved due to the center distance and aspect ratio-guided IoU thresholds.

V. CONCLUSION

In this article, we propose a refined and balanced feature pyramid network (RB-FPN), which aims to eliminate the problem of complex background information and enhance the semantic feature information in the FPN. Specifically, RB-FPN focuses more on potential object regions to enhance the semantic information and balance the feature maps in the FPN to provide more pure information for subsequent classification tasks. And the proposed CSA label assignment strategy fully utilize the statistical characteristics of oriented objects. The CSA label assignment strategy dynamically adjust the IoU threshold during the training process, which alleviates the problem of angle sensitivity of IoU for narrow oriented objects. When performing sample selection, the CSA label assignment strategy allows narrow oriented objects to retain more potential samples and prevents high-quality samples from being filtered out. Moreover, a comprehensive and extensive evaluation of FAIR1M dataset indicates that our approach yields consistent and substantial gains compared to the baseline approach.

REFERENCES

[1] G. Cheng and J. Han, "A survey on object detection in optical remote sensing images," *ISPRS J. Photogrammetry Remote Sens.*, vol. 117, pp. 11–28, 2016.
[2] Y. Li, H. Zhang, X. Xue, Y. Jiang, and Q. Shen, "Deep learning for remote sensing image classification: A survey," *Wiley Interdiscipl. Rev.: Data Mining Knowl. Dis.*, vol. 8, no. 6, 2018, Art. no. e1264.

[3] 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 J. Photogrammetry Remote Sens.*, vol. 159, pp. 296–307, 2020.

[4] G. Cheng, C. Yang, X. Yao, L. Guo, and J. Han, "When deep learning meets metric learning: Remote sensing image scene classification via learning discriminative CNNs," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 5, pp. 2811–2821, May 2018.

[5] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2015, pp. 91–99.

[6] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 779–788.

[7] W. Liu et al., "SSD: Single shot multibox detector," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 21–37.

[8] J. Dai, Y. Li, K. He, and J. Sun, "R-FCN: Object detection via region-based fully convolutional networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2016, pp. 379–387.

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

[10] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes (VOC) challenge," *Int. J. Comput. Vis.*, vol. 88, no. 2, pp. 303–338, 2010.

[11] 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 Geosci. Remote Sens. Lett.*, vol. 13, no. 8, pp. 1074–1078, Aug. 2016.

[12] Z. Liu, J. Hu, L. Weng, and Y. Yang, "Rotated region based CNN for ship detection," in *Proc. IEEE Int. Conf. Image Process.*, 2017, pp. 900–904.

[13] G. Zhang, S. Lu, and W. Zhang, "CAD-Net: A context-aware detection network for objects in remote sensing imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 12, pp. 10015–10024, Dec. 2019.

[14] X. Yang et al., "SCRDet: Towards more robust detection for small, cluttered and rotated objects," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2019, pp. 8232–8241.

[15] Z. Guo, C. Liu, X. Zhang, J. Jiao, X. Ji, and Q. Ye, "Beyond bounding-box: Convex-hull feature adaptation for oriented and densely packed object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 8792–8801.

[16] K. Fu, Z. Chang, Y. Zhang, G. Xu, K. Zhang, and X. Sun, "Rotation-aware and multi-scale convolutional neural network for object detection in remote sensing images," *ISPRS J. Photogrammetry Remote Sens.*, vol. 161, pp. 294–308, 2020.

[17] Y. Li, "Detecting lesion bounding ellipses with Gaussian proposal networks," in *Proc. Int. Workshop Mach. Learn. Med. Imag.*, 2019, pp. 337–344.

[18] J. Ding, N. Xue, Y. Long, G.-S. Xia, and Q. Lu, "Learning ROI transformer for oriented object detection in aerial images," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 2849–2858.

[19] X. Feng, J. Han, X. Yao, and G. Cheng, "TCANet: Triple context-aware network for weakly supervised object detection in remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 8, pp. 6946–6955, Aug. 2020.

[20] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 2117–2125.

[21] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 2980–2988.

[22] S. Zhang, C. Chi, Y. Yao, Z. Lei, and S. Z. Li, "Bridging the gap between anchor-based and anchor-free detection via adaptive training sample selection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 9759–9768.

[23] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2014, pp. 580–587.

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

[25] J. Redmon and A. Farhadi, "YOLOv3: An incremental improvement," 2018, *arXiv:1804.02767*.

[26] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, "YOLOv4: Optimal speed and accuracy of object detection," 2020, *arXiv:2004.10934*.

[27] J. Redmon and A. Farhadi, "YOLO9000: Better, faster, stronger," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 7263–7271.

[28] G. Ghiasi, T.-Y. Lin, and Q. V. Le, "NAS-FPN: Learning scalable feature pyramid architecture for object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 7036–7045.

[29] M. Tan, R. Pang, and Q. V. Le, "EfficientDet: Scalable and efficient object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 10781–10790.

[30] S. Qiao, L.-C. Chen, and A. Yuille, "Detectors: Detecting objects with recursive feature pyramid and switchable atrous convolution," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 10213–10224.

[31] M. Hu, Y. Li, L. Fang, and S. Wang, "A2-FPN: Attention aggregation based feature pyramid network for instance segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 15343–15352.

[32] S. Liu, L. Qi, H. Qin, J. Shi, and J. Jia, "Path aggregation network for instance segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 8759–8768.

[33] S. Liu, D. Huang, and Y. Wang, "Learning spatial fusion for single-shot object detection," 2019, *arXiv:1911.09516*.

[34] K. Kim and H. S. Lee, "Probabilistic anchor assignment with IOU prediction for object detection," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 355–371.

[35] B. Zhu et al., "AutoAssign: Differentiable label assignment for dense object detection," 2020, *arXiv:2007.03496*.

[36] Z. Ge, S. Liu, Z. Li, O. Yoshie, and J. Sun, "OTA: Optimal transport assignment for object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 303–312.

[37] Y. Xu et al., "Gliding vertex on the horizontal bounding box for multi-oriented object detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 4, pp. 1452–1459, Apr. 2020.

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

[39] J. Han, J. Ding, N. Xue, and G.-S. Xia, "ReDet: A rotation-equivariant detector for aerial object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 2786–2795.

[40] Q. Ming, Z. Zhou, L. Miao, H. Zhang, and L. Li, "Dynamic anchor learning for arbitrary-oriented object detection," in *Proc. AAAI Conf. Artif. Intell.*, 2021, vol. 35, no. 3, pp. 2355–2363.

[41] C. Zhang, B. Xiong, X. Li, and G. Kuang, "Aspect-ratio-guided detection for oriented objects in remote sensing images," *IEEE Geosci. Remote Sens. Lett.*, vol. 19, 2021, Art. no. 8024805.

[42] X. Sun et al., "FAIR1M: A benchmark dataset for fine-grained object recognition in high-resolution remote sensing imagery," *ISPRS J. Photogrammetry Remote Sens.*, vol. 184, pp. 116–130, 2022.

[43] C. Zhang, B. Xiong, X. Li, J. Zhang, and G. Kuang, "Learning higher quality rotation invariance features for multioriented object detection in remote sensing images," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 14, pp. 5842–5853, 2021.

[44] X. Yao, H. Shen, X. Feng, G. Cheng, and J. Han, "R2IPoints: Pursuing rotation-insensitive point representation for aerial object detection," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5623512.

[45] X. Xie, G. Cheng, J. Wang, X. Yao, and J. Han, "Oriented R-CNN for object detection," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2021, pp. 3520–3529.

[46] Z. Dai, Z. Yang, Y. Yang, J. Carbonell, Q. V. Le, and R. Salakhutdinov, "Transformer-XL: Attentive language models beyond a fixed-length context," 2019, *arXiv:1901.02860*.

[47] X. Zhu, D. Cheng, Z. Zhang, S. Lin, and J. Dai, "An empirical study of spatial attention mechanisms in deep networks," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2019, pp. 6688–6697.

[48] J. Dai et al., "Deformable convolutional networks," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 764–773.

[49] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 770–778.

[50] J. Han, J. Ding, J. Li, and G.-S. Xia, "Align deep features for oriented object detection," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2021, Art. no. 5602511.

[51] G.-S. Xia et al., "DOTA: A large-scale dataset for object detection in aerial images," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 3974–3983.

[52] Z. Liu et al., "Swin transformer: Hierarchical vision transformer using shifted windows," in *Proc. IEEE/CVF Int.Conf. Comput. Vis.*, 2021, pp. 10012–10022.

Junjie Song received the B.S. degree in electrical engineering and the automatization, in 2020, from the China University of Mining and Technology, Xuzhou, China, where he is currently working toward the M.S. degree in control science and engineering with the School of Automation, Beijing Institute of Technology, Beijing, China.

His research interests include computer vision and deep learning in remote sensing applications.

Lingjuan Miao received the B.S. and M.S. degrees in control theory and engineering from the Harbin Institute of Technology, Harbin, China, in 1986 and 1989, respectively, and the Ph.D. degree in control theory and engineering from the China Academy of Launching Vehicle Technology, Beijing, China, in 2001.

Since 1992, she has been with the Beijing Institute of Technology, Beijing, China, first as a Lecturer, an Associate Professor, since 1996, and a Professor, since 2001. Her research interests include GPS, inertial navigation systems, INS/GPS integrated navigation, and multisensor fusion technique.

Qi Ming received the B.S. degree in automation, in 2018, from the School of Automation, Beijing Institute of Technology, Beijing, China, where he is currently working toward the Ph.D. degree in navigation, guidance, and control with the School of Automation.

His research interests include computer vision, object detection, and remote sensing image analysis.

Zhiqiang Zhou (Member, IEEE) received the B.S. degree in automation and the Ph.D. degree in control science and engineering from the School of Automation, Beijing Institute of Technology (BIT), Beijing, China, in 2004 and 2009, respectively.

From 2009 to 2012, he was a Postdoctoral Researcher with the Institute of Automation, Chinese Academy of Sciences, Beijing, China. He is currently an Associate Professor with the School of Automation, BIT. His research interests include information fusion, pattern recognition, digital image processing, and vision-based navigation.

Yunpeng Dong received the B.S. degree in automation from Hebei University, BaoDing, China, in 2020. He is currently working toward the M.S. degree in control engineering with the School of Automation, Beijing Institute of Technology, Beijing, China.

His research interests include object detection and FPGA for deep learning.