Constrained-SIoU: A Metric for Horizontal Candidates in Multi-Oriented Object Detection

Yanan Zhang, Haichang Li, Rui Wang, *Member, IEEE*, Mengya Zhang, and Xiaohui Hu

***Abstract*—Intersection over union (IoU) has been widely adopted to evaluate and select candidate regions in multi-oriented object detection. Intuitively, overlaps between candidates and multi-oriented ground-truth boxes make more sense when assessing the quality of horizontal candidates. However, the horizontal minimum bounding box (HMBB) of the ground-truth box is generally used for the IoU calculation in practice, bringing about biased results. In this article, we propose a novel Splicing Intersection over Union (SIoU) to provide a more preferable metric for horizontal candidate selection when detecting multi-oriented objects. By computing the intersection between the candidate region and the ground-truth box rather than its HMBB, SIoU provides a better description of how much object information a candidate contains. Furthermore, we introduce two variants of constraints for the center of each candidate to ensure its location focusing on the objects. Candidates whose centers deviate too far from the objects will be penalized. We integrate the constraint with SIoU, denoted as constrained-SIoU, to select horizontal candidates more efficiently. Comparative experiments on two datasets of aerial images, DOTA and HRSC2016, demonstrate the effectiveness of the proposed method.**

***Index Terms*—Aerial images, label assignment, multi-oriented object detection, remote sensing.**

I. Introduction

In the past few years, remote sensing image processing has made rapid progress with the development of deep learning. Remote sensing images have some characteristics that are different from natural images, and therefore, require special researches. For example, they tend to have a lower resolution than natural images and are usually bird's-eye views. The research issues that have attracted a lot of attention mainly include image classification [1]–[3], object detection [4]–[6], change detection [7]–[9], etc. Object detection can be divided into supervised object detection [6], [10], [11], weakly supervised object detection [4], [5], [12], and unsupervised object detection. According to the orientation of objects, it can also be divided into horizontal object detection and multi-oriented object detection. In this article, we mainly focus on the supervised multi-oriented object detection.

Recently, multi-oriented object detection, which aims at describing objects with oriented bounding boxes (OBBs), has achieved promising performances [6], [10], [11], [13]–[15]. OBB generally includes rotated bounding box (RBB) and quadrilateral bounding box (QBB). As a subproblem of general object detection, multi-oriented object detection is first explored in scene text detection, and then, gradually extended to remote sensing imagery processing. Currently, most multi-oriented object detection methods [10], [11], [16] are based on the state-of-the-art horizontal detectors, e.g., Faster R-CNN [17], feature pyramid network (FPN) [18], and RetinaNet [19]. Compared with the horizontal rectangles, bounding boxes with orientation information indicate more accurate locations.

Anchor-based approaches [17], [19], [22], [23] for general object detection often preset external indicators, which are known as anchors [17], [19] or default boxes [23], to provide candidates for subsequent predictions. In two-stage methods, proposals generated in the first stage are also regarded as candidates. Candidates are compared with the ground-truth boxes to determine the targets of classification and localization through a process named label assignment. Label assignment plays a crucial role in the performance of detector and has been studied extensively [24]–[26]. Most assignment methods use IoU to measure the overlaps of ground-truth boxes and candidates. Note that though intersection over union (IoU) is able to gauge the similarity between two boxes of any shape, IoU in this article refers specifically to the overlap between two horizontal boxes in order to distinguish it from SkewIoU. Borrowing from axis-aligned detection algorithms, a lot of methods focusing on multi-oriented objects also generate horizontal anchors in advance, whether the final prediction format is OBB [16], [27] or QBB [6]. Since the ground-truth box and the candidate are generally of different types (one is axis-aligned and the other is multi-oriented), the HMBB of the ground-truth box is used to calculate IoU.

However, we observe that IoU based on the HMBB may lead to unreasonable overlaps and biased assignment results ulteriorly. The HMBB contains not only the features in the ground-truth box but also the information in the background area. As shown in Fig. 1(a), an equivalent IoU value may exist

Manuscript received September 13, 2021; revised November 8, 2021; accepted December 8, 2021. Date of publication December 23, 2021; date of current version January 17, 2022. This work was supported in part by the National Natural Science Foundation of China under Grant 61802380, in part by the National Key Research and Development Program of China under Grant 2019YFB1405100, and in part by the Strategic Priority Research Program of the Chinese Academy of Sciences under Grant XDA19020500. *(Corresponding author: Yanan Zhang.)*

Yanan Zhang is with the Science and Technology on Integrated Information System Laboratory, Institute of Software, Chinese Academy of Sciences, Beijing 100190, China, and also with the University of Chinese Academy of Sciences, Beijing 100049, China (e-mail: yanan2018@iscas.ac.cn).

Haichang Li, Rui Wang, Mengya Zhang, and Xiaohui Hu are with the Science and Technology on Integrated Information System Laboratory, Institute of Software, Chinese Academy of Sciences, Beijing 100190, China (e-mail: haichang@iscas.ac.cn; wangrui@iscas.ac.cn; mengya@iscas.ac.cn; hxh@iscas.ac.cn).

Digital Object Identifier 10.1109/JSTARS.2021.3137552

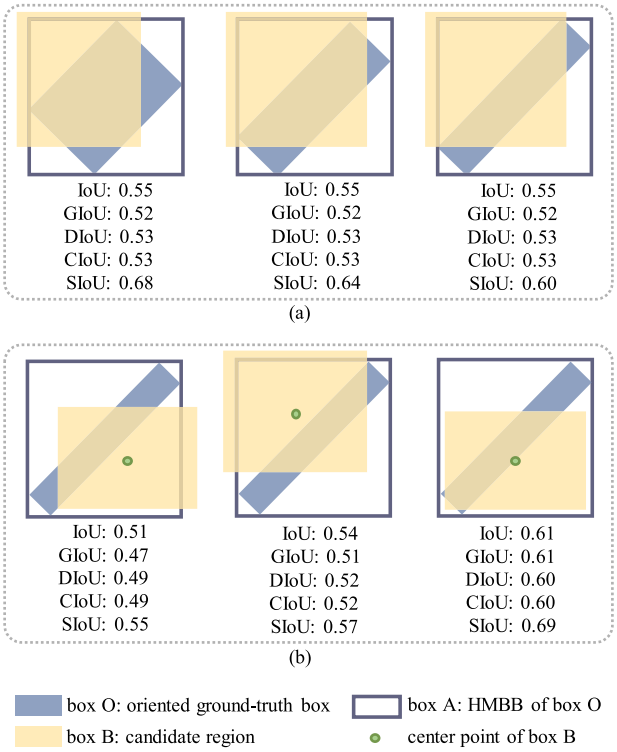

Fig. 1. Two sets of examples. (a) Three cases have the same IoU value, GIoU [20] value, DIoU [21] value, and CIoU [21] value. However, they have different SIoU values. Each candidate contains different part of the object. (b) All candidates have large IoU and SIoU values, and two of them also have large GIoU, DIoU, and CIoU values, but their centers are outside the respective objects.

in completely different situations. Aspect ratios and orientations of the objects greatly affect the actual overlaps between the candidate regions and the objects. However, IoU just considers information about the HMBB, so it is inaccurate to assess the quality of anchors only based on IoU. Similarly, GIoU [20], DIOU [21], and CIoU [21] do not differentiate multi-oriented objects with the same HMBB, which can also lead to inaccurate results. Furthermore, we notice that the centers of some positive candidates selected based on IoU or SIoU are far away from the centers of objects. In some extreme cases, they are even outside the respective objects. In the fixed label assignment method, 0.5 is a conventional threshold to select positive samples. As shown in Fig. 1(b), a candidate with a high IoU value and a high SIoU value may not focus on the object. This will hurt the detector's performance. DIoU [21] and CIoU [21] add a penalty term based on the distance between centers of two boxes, so the case of center deviation is reduced to some extent. However, since they are not designed specifically for either rotated ground-truth boxes or label assignment, CIoU and DIoU cannot handle all center deviations. Some candidates with high DIoU and CIoU values still have centers outside the respective objects, as shown in the second and third examples in Fig. 1(b). For axis-aligned objects, IoU between a ground-truth box and an anchor can generally evaluate the anchor's quality. Unfortunately, if the ground-truth box is with orientation, IoU will be not suitable to be applied.

In this article, we come up with a new method to address the problem in label assignment. First, we propose a novel metric named Splicing Intersection over Union (SIoU) to measure the overlap between an axis-aligned candidate and a multi-oriented ground-truth box. SIoU takes advantage of the following two factors:

1) overlap between the candidate and the ground-truth box; and
2) overlap between the candidate and the HMBB of the ground-truth box.

It not only reflects how much valuable information a candidate contains, but also takes into account the necessary redundancy it must have. As shown in Fig. 1, we call features contained in box O as valuable information, and the necessary redundancy refers to the features contained in A but not in O. In addition, we introduce two variants of constraints for the center of each candidate to avoid it being too far from the object. In multi-oriented object detection, the combination of SIoU and the center constraint, constrained-SIoU, can serve as the metric to assign labels for horizontal candidates.

The main contributions of this article are summarized as follows.

1) We analyze the problem of IoU used in label assignment and propose a new metric named SIoU. SIoU depicts the overlap between an axis-aligned box and a multi-oriented box. It helps to select proper horizontal candidates. The anchors and proposals obtained by the proposed SIoU are more precise than those selected by IoU.
2) We observe that the centers of some selected positive samples are far away from the objects, so we introduce constraints toward the centers. Biased centers may cause classification errors and inaccurate locations. Through penalizing the anchors whose centers are apart from the objects in label assignment, we effectively reduce the number of improper predictions.
3) We conduct experiments on two benchmarks. Without bells and whistles, our proposed method achieves significant performance improvements.

II. Related Work

A. Anchor-Based Algorithms in Object Detection

As its name implies, anchor-based algorithms, whether for general horizontal object detection or multi-oriented object detection, commonly preset anchors. Differently, anchor-free methods, e.g., [28]–[32], do not need anchors. In this section, we mainly discuss anchor-based methods in object detection.

1) Axis-Aligned Object Detection: Algorithms in object detection can be generally divided into two types: single-stage methods and two-stage methods. In single-stage methods, e.g., YOLO [33], [34], SSD [23], and RetinaNet [19], anchors are directly used for subsequent predictions. In two-stage methods [17], [22], [35]–[37], proposals generated in the first stage are fed into the second subnetwork to obtain their categories and more precise locations. The most popular network to produce proposals is Region Proposal Network (RPN), which is proposed in Faster R-CNN [17] and has been widely used in two-stage

networks [6], [27] and multi-stage networks [38]. The RPN first enumerates a certain number of aspect ratios and scales to crop anchors, then classifies the anchors and refines their locations and shapes to produce proposals. Anchors and proposals in axis-aligned object detection are still axis-aligned, so IoU is suitable to evaluate and select candidates.

2) Multi-Oriented Object Detection: Multi-oriented object detection algorithms are mostly built upon the axis-aligned detectors. The difference is that anchors and region proposals for detecting oriented objects can be either horizontal or multi-oriented. Most algorithms adopt horizontal anchors, e.g., see [6], [10], [16], [27], and [39]–[41]. Rotational Region CNN (R^2 CNN) [39], SCRDet [10], and gliding vertex [6] are based on Faster R-CNN [17]. They use the RPN to generate axis-aligned proposals, and then, produce rotated bounding boxes or quadrangles for multi-oriented objects. Horizontal candidates do not often match objects well. To solve this problem, Ding *et al.* [27] transform the horizontal regions of interest (HRoIs) into rotated regions of interest (RRoIs) through a subnet named RoI Transformer. R^3 Det [16] adjusts the regression head of RetinaNet [19] to predict multi-oriented bounding boxes. It adds multiple refinement stages and introduces the feature refinement module to modify the inaccurate features in the refinement. As rotated anchors can provide more precise features for classification and better initial values for regression, some approaches [42]–[46] crop anchors with orientation. Both anchors and objects are multi-oriented, so SkewIoU is used for label assignment in this case. The RRPN [42] creatively introduces rotated anchors into the RPN network based on Faster R-CNN [17] to detect multi-oriented texts. DRBox [47] analyzes the disadvantages of the horizontal bounding boxes and proposes to use multi-angle prior RBoxes (essentially multi-oriented anchors) to produce rotated bounding boxes. FFA [46] adopts Oriented Region Proposal Network (RPN-O) to generate rotated proposals and uses RoI-O pooling for feature extraction. In addition to modifying the network, some researchers study loss functions to solve the problem of regression loss discontinuity. Regression loss discontinuity is caused by the periodicity of angles. The fundamental problem is that a rotated box can be represented in several different ways. Modulated Rotation Loss [48] enumerates all possible representations for ground-truth boxes as well as the resulting loss values and selects the minimum one to get regression loss. SkewIoU Loss is the ideal loss function like IoU Loss [49] in general object detection. However, the calculation of SkewIoU is indifferentiable. PIoU Loss [50] uses statistics of pixels in each box to approximate SkewIoU. Gaussian Wasserstein Distance (GWD) [51] models each box as a 2-D Gaussian Wasserstein Distance, and uses their GWD as the regression loss.

B. Label Assignment

Fixed label assignment is a traditional assignment method that sets two fixed thresholds to select positive and negative samples. We will describe it in detail in Section III-A. Generally, a fixed assignment method lacks flexibility and may not bring optimal assignment results. Adaptive Training Sample Selection (ATSS) [26] and Dynamic R-CNN [52] dynamically determine thresholds based on IoU statistics. ATSS [26] uses mean and standard deviation, and Dynamic R-CNN [52] uses top-K fractions. Some methods take into account the prediction results for label assignment. Li *et al.* [53] introduce the cleanliness score to measure the anchors' quality. The cleanliness score is the weighted mean of the localization accuracy and classification confidence. After selecting a fixed number of positive candidates, the cleanliness score is used to generate soft labels for classification and weights of loss terms for regression. Similarly, Dynamic Anchor Learning (DAL) [54] selects the candidates according to the prediction results. In addition to output IoU, it also considers the input IoU to stabilize the training. In anchor-based detection methods, some objects with special aspect ratios or scales cannot get enough positive candidates. High-quality anchor mining strategy (HAMBox) [55] utilizes the prediction results to select high-quality anchors for this part of objects. In addition, OneNet [56], FCOS_{PSS} (PSS refers to positive sample selector) [57], and DeFCN [58] explore one-to-one label assignment methods to achieve detection without postprocessing. Unlike these studies, we mainly focus on how to measure the similarity of boxes with different orientations. Although the experiments are based on the fixed label assignment method, actually our proposed constrained-SIoU is able to be integrated into any label assignment algorithms that need to calculate IoU, as long as the candidates and the objects are oriented differently.

Algorithm 1: IoU-based Fixed Threshold Label Assigner.

Input: O is a set which contains m ground-truth boxes. B is a set which consists of n candidates. pos_iou_thr is the threshold for positive samples, and neg_iou_thr is the threshold for negative samples.

Output: $results$ with shape (n,) is the results of assignment.

for $i = 1 \rightarrow n$ **do**
  Calculate the overlaps $overlaps_i$ between the i-th candidate and the ground-truth boxes.
  $max\_overlap \leftarrow \max(overlaps_i)$
  $index \leftarrow argmax(overlaps_i)$
  if $max\_overlap >= pos\_iou\_thr$ **then**
    $results_i \leftarrow index$
  else if $max\_overlap < neg\_iou\_thr$ **then**
    $results_i \leftarrow 0$
  else
    $results_i \leftarrow -1$
  end if
end for

III. Background

We first briefly introduce the fixed label assignment method in Section III-A, and then, analyze the necessity of horizontal candidate regions in multi-oriented object detection in Section III-B. In Section III-C, we discuss problems of the metrics used in label assignment.

A. Overview of the Fixed Threshold Label Assigner

In this article, we adopt the method of label assignment proposed by [17]. After cropping anchors, overlaps between these anchors and ground-truth boxes are calculated. Based on some preset fixed thresholds, positive and negative samples are selected. This strategy is widely applied in anchor-based algorithms such as Cascade R-CNN [38], SSD [23], and RetinaNet [19]. We sketch out the method in Algorithm 1. For a more concise representation, we omit the part of low-quality matching. In the assignment results, numbers greater than 0 represent the index of the object matched by the candidate, 0 means the corresponding region is a negative sample, and -1 shows the ignored one.

In multi-oriented object detection, many algorithms follow the label assignment method in Algorithm 1. If axis-aligned anchors are adopted, overlaps between candidates and the HMBBs of ground-truth boxes may lead to inaccurate results. We propose a new method to calculate the overlaps. The details will be described in Section IV.

B. Comparison of Horizontal and Rotated Candidates

In practice, some algorithms use rotated anchors, e.g., [42] and [47], but many more use horizontal ones, e.g., [10], [16], and [27]. High efficiency and satisfactory recall are the main reasons why horizontal candidates are more popular. First of all, generating rotated candidates will cause extra time consumption. In training, encoding the orientation information will result in a multiplication of anchors. For example, there are six orientations in the RRPN [42], which means that the number of anchors will be six times as many as in the horizontal case. The large number of anchors seriously slow down the label assignment, which uses SkewIoU as the primary basis. Actually, in the experiments, we find that the massive computation of SkewIoU without CUDA for acceleration will have a devastating impact on the training speed. Unluckily, there will be a great burden on the memory if CUDA is used. Ding *et al*. [27] also discussed the degradation of matching efficiency caused by the dramatic increase of anchors' number. In addition, as the overlap between two multi-oriented boxes is largely affected by the orientation, some objects will have difficulty in finding anchors with sufficient overlaps. To some extent, adding orientation information does not necessarily improve the recalls, even with more anchors. Yang *et al*. [16] discovered that though rotated anchors can perform better in dense scenes, horizontal anchors can achieve higher recalls in fewer quantities. Although cropping anchors at a smaller interval of orientation can improve the performance, it will increase the time consumption even further. In summary, it does take more time and is not necessarily better to crop rotated anchors, so we still generate horizontal anchors and assign labels for them based on the new metric we proposed.

C. Drawbacks of Existing Metrics

There are two kinds of metrics commonly used in label assignment when detecting multi-oriented objects. The first one is IoU shown in (1). As discussed earlier, it is a relatively crude indicator with a bias in evaluating the quality of horizontal candidates. Another one is for oriented boxes named SkewIoU, which is proposed in the RRPN [42] and also used in RoI Transformer [27]. Unfortunately, SkewIoU, which is attained by (2), is still not suitable for calculating the overlap between a horizontal box and a rotated one. Generally, we intend to find candidates similar to the HMBBs of the ground-truth boxes. However, as Fig. 2(a) and (b) shows, SkewIoU between a non-horizontal object and its HMBB will never be 1, and the upper limit value of SkewIoU is largely influenced by the orientation and aspect ratio of the object. Moreover, from Fig. 2(b) and (c), we can learn that the SkewIoU of an object and its HMBB is not necessarily the highest one, so it is also not viable to assign labels according to the ranking of SkewIoU. In summary, IoU for two axis-aligned boxes cannot explicitly reflect how much valid information an anchor contains, and SkewIoU ignores the necessary redundancy one horizontal anchor must have.

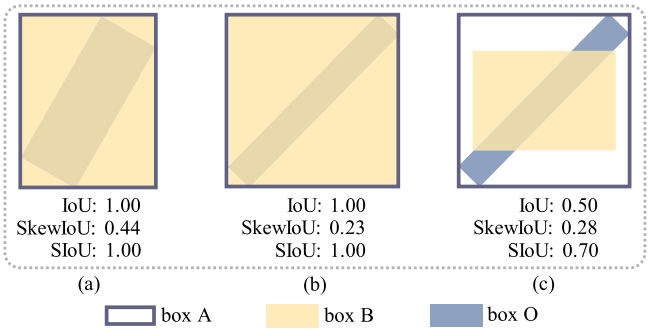

Fig. 2. Comparison of three metrics. For different objects in (a) and (b), the HMBBs have different SkewIoU values. For the same ground-truth box in (b) and (c), the candidate in (b) that is closer to the HMBB of O has a smaller SkewIoU value.

$$\text{IoU} = \frac{|A \cap B|}{|A \cup B|} \tag{1}$$

$$\text{SkewIoU} = \frac{|O \cap B|}{|O \cup B|}. \tag{2}$$

In addition, there are three other metrics in object detection. Based on IoU Loss [49], Rezatofighi *et al.* [20] propose another important metric named Generalized Intersection over Union (GIoU) and GIoU Loss. GIoU is shown in (3), and C in the equation is the smallest rectangle enclosing the union of A and B. GIoU is equivalent to adding a penalty term for unnecessary redundancy in the candidate box. If the candidate is totally inside the ground-truth box, GIoU is equal to IoU. As a loss function, GIoU Loss still works in the case of non-overlapping bounding boxes. However, as we do not care how far apart two non-overlapping boxes are in label assignment, the role of GIoU is limited. Distance-IoU (DIoU) [21], which is shown in (4) considers the distance between the centers of two boxes. ρ in the equation is the Euclidean distance between the centers of two boxes, and c in the equation is the diagonal length of C in GIoU. DIoU Loss converges faster and still plays a role in the cases where GIoU Loss degrades into IoU Loss. In our opinion, the penalty term in DIoU is closer to our constraint

toward the centers. They are different in the calculation method of the penalty term. Complete IoU (CIoU) [21] adds another penalty term to DIoU to get the consistency of aspect ratios. Readers are referred to [21] for more details.

$$\text{GIoU} = \text{IoU} - \frac{|C \backslash (A \cup B)|}{|C|} \tag{3}$$

$$\text{DIoU} = \text{IoU} - \frac{\rho^2(b, b_{gt})}{c^2}. \tag{4}$$

IoU, GIoU [20], DIoU [21], and CIoU [21] are all for horizontal objects. SkewIoU measures the overlap between two multi-oriented boxes. To the best of our knowledge, the metric for boxes of different types has not come up yet.

IV. Proposed Method

Constrained-SIoU consists of two parts, SIoU and center constraint, described in Sections IV-A and IV-B, respectively.

A. Splicing Intersection Over Union

As we have discussed in Section III-C, for a multi-oriented object O, IoU between the candidate B and O's HMBB A is relatively rough to evaluate the quality of B. Besides, direct use of SkewIoU, which is calculated by (2), is also problematic. At present, there is still a lack of ideal methods to evaluate the quality of horizontal candidates in multi-oriented object detection.

To address this issue, we propose a new method named SIoU to calculate the overlap between a horizontal box and a multi-oriented box.

First, IoU for $A, B \subseteq S \in R^n$ can also be expressed as

$$\text{IoU} = \begin{cases} \frac{1}{\frac{|A|}{|A \cap B|} + \frac{|B|}{|A \cap B|} - 1} & |A \cap B| \neq 0 \\ 0 & |A \cap B| = 0. \end{cases} \tag{5}$$

When $|A \cap B| \neq 0$, the first term in the denominator, $|A|/|A \cap B|$, can be read as a multiple of the first box's size and the intersection's size, and the second term is the same. If A is the box enclosing the object, the first term indicates how much valuable information B contains about A, and the second term describes how much redundancy B has. As shown in Fig. 3, the intersection between O and B is sometimes quite different with the intersection of A and B. In multi-oriented object detection, the first term may be inaccurate because the real object may only occupy a small part of region in A. Suppose that A is the HMBB of O, we can replace A in the first term with O so that the first term is able to better represent how much information B contains about the object. B cannot be exactly the same with O, unless O is axis aligned, so B has to contain some redundancy. In order to keep the overlaps from being affected by the necessary redundancy, the second term remains unchanged. In addition, it is intuitive that the overlap of two non-overlapping boxes should be 0. In this way, we can obtain a method to calculate the overlap between a horizontal box and a rotated one, just as (6) shows. The area of intersection in (6) can be calculated like the intersection area in SkewIoU. The entire calculation process refers to [42].

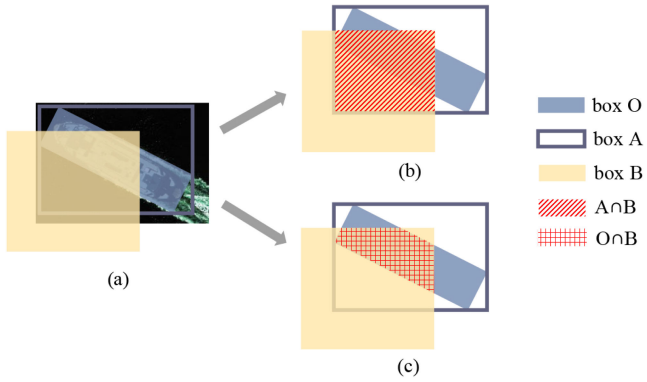

Fig. 3. Intersections in SIoU. (a) Overall relationship between the ground-truth box O, its HMBB A, and the horizontal candidate B. (b) Region of intersection between A and B. (c) Intersection of O and B.

$$\text{SIoU} = \begin{cases} \frac{1}{\frac{|O|}{|O \cap B|} + \frac{|B|}{|A \cap B|} - 1} & |O \cap B| \neq 0 \\ 0 & |O \cap B| = 0 \end{cases}. \tag{6}$$

The properties that both IoU and SIoU have are as follows.

1) SIoU is nonnegative, and the range of SIoU is still $[0, 1]$. When B and A are exactly the same, SIoU between O and B is 1.
2) SIoU is still invariant to the scale.

The differences between IoU and SIoU include the following.

1) IoU between A and B has symmetry, but SIoU between O and B does not.
2) If A and B are not identical, there is no definite relationship between the values of IoU and SIoU. If O and B do not overlap but A and B do, SIoU will still be 0 even if IoU between A and B is larger than 0.
3) The process of IoU calculation is differentiable. Unfortunately, there are nondifferentiable parts in the calculation of SIoU, so we cannot simply use $1 - SIoU$ or $-\ln SIoU$ as the loss function.

In summary, SIoU retains most of the good properties of IoU and increases the ability to evaluate boxes of different types. When assigning labels for horizontal candidates in multi-oriented object detection, SIoU can be used just like IoU.

B. Center Constraint

Though there are a few objects with acentric features, information contained in the center of a candidate is critical for classification and regression in most cases. The satisfactory performances of FCOS [59] and CenterNet [30] demonstrate the importance of the central feature. However, in multi-oriented object detection, some candidates with both high IoU values and great SIoU values have centroids far away from the objects' centers. As shown in Fig. 1(b), some of them are even centered outside the objects. Intuitively, this may spell trouble for predictions.

The simple and straightforward idea is to suppress the positive samples whose centers are not close to the centers of objects. We merely regard these samples as negative if the distances between their centers and the objects' centers exceed the predefined

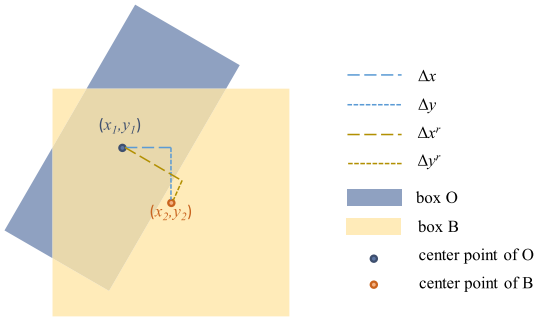

Fig. 4. Center constraint. If the center of box B is far from the center of the box O, we calculate the horizontal and vertical distances between two centers and convert them to be aligned with the orientation of O.

range. We call it hard constraint (abbreviated HC). We will give a formal representation of the constraint in the following.

As shown in Fig. 4, suppose that the center of the ground-truth box O is (x_1, y_1), and the center of the horizontal box B is (x_2, y_2), then their horizontal and vertical distances are Δx and Δy, respectively. The differences along the direction of object, Δx^r and Δy^r, can be obtained according to the orientation of O. The angle is denoted by θ.

$$\Delta x^r = \cos\theta * \Delta x + \sin\theta * \Delta y \tag{7}$$

$$\Delta y^r = -\sin\theta * \Delta x + \cos\theta * \Delta y. \tag{8}$$

The penalty term for HC is shown in (9). Suppose the width of the object is w, and the height of the object is h, then the two parts of the P^{hard} are expressed by (10) and (11), respectively. λ is the hyperparameter, which decides the center's range of candidates in reserve. For example, if $\lambda = 1$, the value of the penalty term will be 1 if the center of the candidate is outside the object, and then, the overlap of this pair of boxes will be 0. If $\lambda < 1$, the range will be smaller and the constraint will be tighter. We will discuss how to combine SIoU and our penalty term in Section IV-C.

$$P^{\text{hard}} = \max\left(P_x^{\text{hard}}, P_y^{\text{hard}}\right) \tag{9}$$

$$P_x^{\text{hard}} = \begin{cases} 1 & \frac{\Delta x^r}{0.5*w} - \lambda \geq 0 \\ 0 & \frac{\Delta x^r}{0.5*w} - \lambda < 0 \end{cases} \tag{10}$$

$$P_y^{\text{hard}} = \begin{cases} 1 & \frac{\Delta y^r}{0.5*h} - \lambda \geq 0 \\ 0 & \frac{\Delta y^r}{0.5*h} - \lambda < 0. \end{cases} \tag{11}$$

HC eliminates samples that are not centered in the range we set, but it has a drawback: samples whose centers are outside the range sometimes are able to produce reasonable results, especially when their overlaps are large enough, but HC excludes these samples.

To handle this problem, we propose a new constraint named soft constraint (abbreviated SC). If the center of a sample is outside the range we set, a penalty term P^{soft} will be generated according to the distance of two centers. P^{soft} also consists of two parts, representing the two directions along the width and height of the object, respectively.

$$P^{\text{soft}} = P_x^{\text{soft}} + P_y^{\text{soft}}. \tag{12}$$

Algorithm 2: Matching Matrix for Label Assignment.

Input: Set of oriented boxes $S^o = \{O_i\}_{i=1}^{N^o}$, set of horizontal boxes $S^b = \{B_j\}_{j=1}^{N^b}$, the type of penalty term pt.
Output: The matching matrix M

```
for i = 1 → N^o do
  for j = 1 → N^b do
    Calculate A_i, the HMBB of O_i
    Calculate SIoU by (6) using O_i, A_i, and B_j
    if pt == "HC" then
      Calculate penalty term P by (9), (10), and (11)
    else if pt == "SC" then
      Calculate penalty term P by (12), (13), and (14)
    else
      P = 0
    end if
    M_{i,j} ← max(0, SIoU − P)
  end for
end for
```

Just like in HC, w is the width of the object, h is the height of the object, and λ is a hyperparameter which is no less than 0, then the penalty term will be

$$P_x^{\text{soft}} = \max\left(0, \frac{\Delta x^r}{0.5 * w} - \lambda\right) \tag{13}$$

$$P_y^{\text{soft}} = \max\left(0, \frac{\Delta y^r}{0.5 * h} - \lambda\right). \tag{14}$$

Suppose $\lambda = 1$, $\Delta x^r/(0.5 * w) - 1 > 0$ is equivalent to $\Delta x^r - 0.5 * w > 0$, and the center of the horizontal box, (x_2, y_2), will be outside the rotated box. In this case, we will give it a penalty term. Based on $0.5 * w$, the bigger Δx^r is, the larger P_x will be. If (x_2, y_2) is inside the rotated box, the penalty term is 0. It is the same with P_y. If $\lambda < 1$, there will be less scope for impunity. Similarly, if $\lambda > 1$, there will be more samples whose penalty values are 0.

Compared with HC, SC is more flexible and friendlier to samples near the penalty boundary.

C. Matching Matrix for Label Assignment

In summary, as the Algorithm 2 shows, the new metric mainly depends on the overlaps between horizontal candidates and oriented ground-truth boxes. The penalty term is optional. We can calculate matching matrix only use SIoU, or give hard or soft constraint to the center of candidates. The matching matrix we get will be applied in label assignment. Note that if the ground-truth box is quadrangular, it is not easy to calculate the relative difference between two centers. We can first get the rotated minimum bounding box (RMBB) of the ground-truth box, and then, calculate their distance. Generally, most quadrangles are quite similar to their RMBBs, and SIoU can be calculated using quadrangles, so the matching matrix we get is still accurate. Moreover, if we use HC and set λ=1, we can

directly exclude samples whose centers are outside the quadrangular ground-truth boxes. In most cases, λ=1 can produce good results. Although ground-truth boxes in our experiments are rotated rectangles, constrained-SIoU can be applied to detectors of quadrangles [6].

Our method consumes about an additional 10% time in the training process. When it is applied in RetinaNet-H [16] with three scales, in each epoch, SIoU takes about 600 s to calculate, and the time required for center constraint is negligible, while the whole training time of the model using IoU is about 100 min. Luckily, we do not need to assign labels during the inference, so our method does not add any testing time.

V. Experiments

We evaluate the proposed method on two benchmarks: DOTA [60] and HRSC2016 [61]. Experiments are conducted based on two methods: Faster R-CNN [17] and RetinaNet [19]. The ablation study is carried on both DOTA [60] and HRSC2016 [61]. Furthermore, we also compare our performance with other methods. Note that to clarify the category of constraint used, we use "SIoU with SC" and "SIoU with HC" to represent constrained-SIoU.

A. Datasets

DOTA [60] is one of the largest datasets with orientation annotations. It contains 2806 images with 188 282 instances. The 1.0 version of the dataset we used consists of 15 categories: plane, baseball diamond (BD), bridge, ground field track (GTF), small vehicle (SV), large vehicle (LV), ship, tennis court (TC), basketball court (BC), storage tank (ST), soccer-ball field (SBF), roundabout (RA), harbor, swimming pool (SP), and helicopter (HC). The official public indicator is mAP, and evaluation server is provided.

Training and validation sets are used during the training, and the testing set is used for testing. Images are cropped into subimages of 1024×1024 with an overlap of 200 pixels. In total, we get 15 749 patches in the training set, 5 297 patches in the validation set, and 10 833 patches in the testing set. 10 276 of patches in the training set are with objects, and 3 281 of patches in the validation set are with objects. During the training, we only use patches with objects. For data augmentation, we use two scales, 0.5 and 1.0, on training and validation sets. Three scales, 0.5, 1.0, and 1.5, are applied to the testing set. We crop images to subimages of 1024×1024 with step 500.

HRSC [61] is a dataset with the orientation annotations for ship detection. 1070 images in this dataset are from six harbors. There are 436 images in the training set, 181 images in the validation set, and 453 images in the testing set. Training set is used during the training, and testing set is used in the test. Since the dataset is relatively small, we use random flip, rotation, and photometric distortions for data augmentation in all experiments.

B. Implementation Details

We mainly use two baseline methods in the experiments, Faster R-CNN trained on oriented bounding boxes (FR-O) [60] and RetinaNet trained on oriented bounding boxes (RetinaNet-H, "H" means horizontal anchors) [16]. The former one is a classical two-stage detection method, and the last one is an important one-stage algorithm.

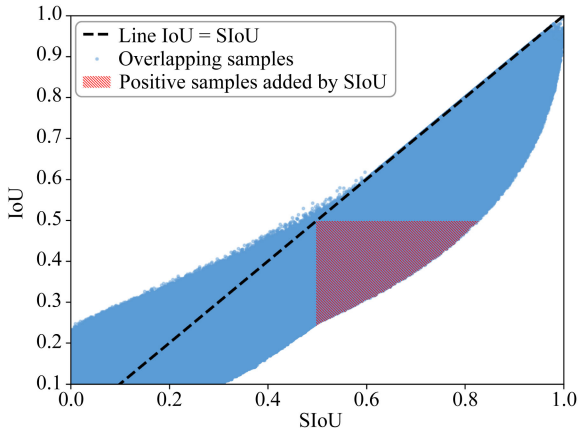

Fig. 5. Relationships between SIoU values and IoU values. Samples come from the model of RetinaNet-H [16] with three scales based on DOTA [60].

We implement our algorithm based on the public codebase "*AerialDetection*,"[1] which is modified from mmdetection [62]. In all experiments, we use 1 TITAN xp GPU to train our model. The basic learning rate is 0.00125 for one GPU and one image per GPU. In the experiments of DOTA [60], the batch size is 2 and the learning rate is initialized as 0.0025. For FR-O [60], models are trained by 12 epochs, and the learning rate is dropped tenfold at the 9th and 12th epoch. For RetinaNet-H [16], the total number is 24 epochs, and the learning rate decreases at the 17th and the 23th epoch. For HRSC [61], the batch size is 1, so the learning rate is 0.00125. Models are trained by 120 epochs, and the learning rate is dropped tenfold at the 81th and 111th epoch. During the training, we use the SGD to optimize. The momentum is 0.9, and the weight decay is 0.0001. Unless otherwise specified, all experiments are conducted with ResNet50 and FPN. All the hyperparameters follow the default settings in "AerialDetection," and our method only add one new hyperparameter λ. We simply set $\lambda = 1$ unless specifically mentioned.

C. Ablation Study

1) Results About SIoU on DOTA [60]: By applying SIoU to FR-O [60], the performance increases 1.19%. For RetinaNet-H [16] with three scales on each feature map, SIoU results in a 1% performance drop. We quantitatively plot the relationships between SIoU values and IoU values of all training samples in RetinaNet-H [16] with three scales. For ease of display, samples with IoU values less than 0.1 are removed. As shown in Fig. 5, the distribution of SIoU is different from IoU. If 0.5 is taken as the boundary between positive and negative samples, SIoU adds more positive samples while retaining almost all positive ones selected by IoU. Since enough anchors are predefined, the number of positive samples may have been met. Positive samples added by SIoU may have negative effects, because most of them

[1][Online]. Available: https://github.com/dingjiansw101/AerialDetection.git

TABLE I
ABLATION RESULTS ON DOTA [60]

Method	Metric	Plane	BD	Bridge	GTF	SV	LV	Ship	TC	BC	ST	SBF	RA	Harbor	SP	HC	mAP
FR-O [60]	IoU	88.69	75.02	45.57	59.42	73.86	69.74	77.53	90.58	81.93	82.63	46.38	62.36	62.82	65.65	52.18	68.96
	IoU with HC	88.58	73.30	46.27	59.70	73.56	73.21	83.99	90.36	79.09	82.64	44.46	64.22	63.70	66.99	62.20	70.15(**+1.19**)
	IoU with SC	88.79	74.90	45.36	63.97	77.52	73.62	84.00	90.78	78.57	82.39	42.65	60.29	63.89	67.52	54.34	69.90(**+0.94**)
	SIoU	88.90	80.35	44.72	62.61	73.54	70.56	77.67	90.81	78.50	81.79	52.44	61.26	64.08	66.44	57.86	70.10(**+1.14**)
	SIoU with HC	88.93	79.24	47.24	62.79	73.84	73.65	78.08	90.76	81.48	81.39	53.77	62.96	64.61	67.96	57.38	70.94(**+1.98**)
	SIoU with SC	89.24	80.70	44.34	65.99	73.45	73.55	83.52	90.80	81.42	80.57	55.65	60.03	64.87	67.49	63.17	71.65(**+2.69**)
†RetinaNet-H [16]	IoU	87.81	80.43	42.71	66.54	70.53	58.07	73.73	90.87	79.37	71.25	50.43	62.34	62.06	66.68	53.32	67.74
	[a]IoU with HC	88.72	81.62	44.22	67.53	75.30	73.87	78.77	90.66	80.37	71.57	53.68	64.74	63.64	66.11	52.35	70.21(**+2.47**)
	[b]IoU with SC	88.89	79.53	43.11	65.71	76.87	74.04	83.93	90.65	84.36	81.50	50.88	64.19	63.92	67.18	51.33	71.07(**+3.33**)
	SIoU	82.68	74.92	42.42	66.63	68.57	57.18	73.20	90.72	77.51	67.23	54.41	62.82	61.90	65.23	55.74	66.74(**-1.00**)
	[e]SIoU	87.10	79.95	42.61	69.98	72.39	60.94	74.54	90.86	81.01	69.83	55.39	63.36	62.96	67.06	46.18	68.28(**+0.54**)
	[c]SIoU with HC	89.05	79.15	43.66	70.03	74.67	73.06	78.45	90.80	79.96	71.33	52.06	62.71	64.26	68.33	51.92	69.96(**+2.22**)
	[d]SIoU with SC	89.27	78.80	40.92	68.96	76.79	73.42	83.01	90.67	82.73	80.96	52.81	62.15	63.15	67.76	52.93	70.95(**+3.21**)
⋆RetinaNet-H [16]	IoU	83.70	70.93	44.25	60.20	76.24	61.90	74.77	90.68	77.49	71.44	39.53	65.01	60.71	62.12	53.36	66.15
	IoU with HC	82.61	71.93	43.18	62.61	77.29	71.18	77.52	90.59	79.34	71.01	43.45	63.13	62.02	63.75	52.03	67.44(**+1.29**)
	IoU with SC	85.71	72.66	44.47	58.35	77.13	72.08	82.49	90.75	80.58	79.68	41.23	63.50	62.57	64.80	54.91	68.73(**+2.58**)
	SIoU	87.27	73.92	44.76	65.71	75.54	61.46	74.87	90.90	79.70	72.59	48.29	61.03	62.68	67.16	48.29	67.61(**+1.46**)
	[e]SIoU	88.24	72.10	43.42	67.53	77.31	65.07	76.25	90.87	81.40	70.87	45.46	62.24	61.65	67.03	54.94	68.29(**+2.14**)
	SIoU with HC	87.99	75.06	45.58	69.62	77.88	73.51	83.56	90.87	83.65	72.39	49.13	59.94	63.51	68.43	52.20	70.22(**+4.07**)
	SIoU with SC	88.81	75.62	45.45	69.41	78.53	73.09	83.72	90.89	80.97	80.44	47.54	61.99	63.56	67.12	53.72	70.72(**+4.57**)

SC means soft center constraint, and HC is hard center constraint.
Models marked with ⋆ are using 1 scale every feature layer to generate anchors. Three scales are applied in models with †.
In the model marked with a, $\lambda = 0.7$. In the model marked with b, $\lambda = 0.6$. In the model marked with c, $\lambda = 0.7$. In the model marked with d, $\lambda = 0.3$.
Models marked with e are with the modified thresholds to make sure that the number of positive and negative samples is the same as the baseline method.

are not selected by IoU and the quality is not high enough. When we set the number of scales per feature map is 1, the performance of the benchmark method decreases, but we get a performance gain of 1.46% by using SIoU. We speculate that there are two important factors affecting the performance in label assignment, one is the metric that is used to measure the quality of anchors, and the other is the method for selecting positive and negative samples. Although SIoU is a suitable metric, the variation in its distribution makes 0.5 an inappropriate positive threshold. The negative effect of the unsuitable threshold is greater than the positive action of the good metric, so the performance decreases when SIoU is applied to the model of RetinaNet-H [16] with three scales. To verify our conjecture, we modify the thresholds in label assignment to make sure that each experiment marked with e in Table I has the same number of positive and negative samples as the baseline methods. As shown in Table I, the performance of the experiment using SIoU with modified thresholds is significantly better than that without modification, whether the setting is one scale or three scales. In addition, perhaps a more flexible approach in label assignment can help SIoU to be more effective.

2) Results About Center Constraint on DOTA [60]: We combine the center constraint with both IoU and SIoU based on FR-O [17], RetinaNet-H [19] with one scale and RetinaNet-H [19] with three scales. From Table I, we learn that SC improves the mAP by 1.55%, while HC increases by 0.84% for FR-O [17] based on SIoU. For RetinaNet-H [19] with three scales, we use the training set to train the model with different λ and test it on the validation set in order to find the optimal hyperparameter λ. The results are shown in Fig. 6. As shown in Fig. 6, the performances of SC are generally better than those of HC. The curve of SC is smoother, indicating that SC is more robust than HC to different thresholds λ. When λ drops below 0.3, the performances of HC deteriorate rapidly due to

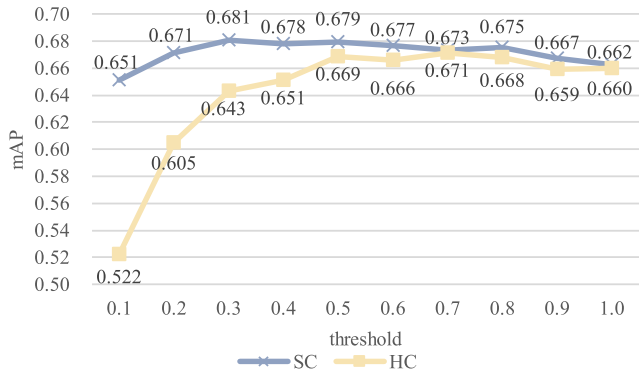

Fig. 6. Curve of the model performance as a function of the threshold λ based on RetinaNet-H [16] with three scales.

the decrease in the number of positive samples matched, while the performances of SC change relatively steadily. We believe that this is mainly because SC can retain samples whose centers are outside our predefined range but have relatively high overlap with the object. The optimal value of λ for SC is 0.3, while that for HC is 0.7. To some extent, this also confirms the ability of SC to retain high quality positive samples outside the range of center.

We visualize some of the results, and compare our new metric to the baseline method. From Fig. 7, we find that false positives tend to occur near objects with large aspect ratios like harbors and large vehicles. In the model using SIoU with SC, the number of false positives decreases significantly.

3) Results on HRSC2016 [61]: We apply our new metric in FR-O [60] and RetinaNet-H [16] on HRSC2016 [61]. As shown in Table II, applying the new method to label assignment can significantly improve the performance of both baseline methods. Alhough IoU with HC leads to the best model regarding mAP_{50},

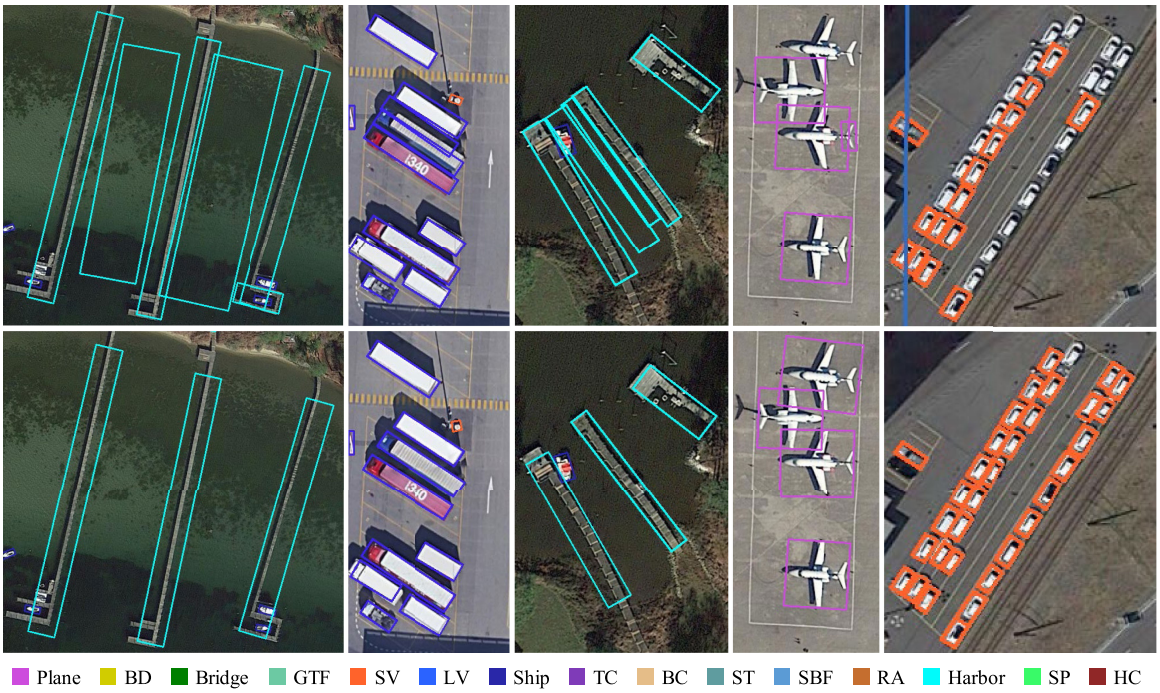

Fig. 7. Comparisons with the baseline results of FR-O [60] on DOTA [60]. The first line is from the baseline method, and the second line shows the results from the model that assigns labels based on SIoU with SC.

TABLE II
ABLATION RESULTS ON HRSC2016 [61]

Method	Metric	Backbone	Size	$mAP_{50}(07)$	$mAP_{50}(12)$	$mAP_{75}(07)$	$mAP_{75}(12)$
RetinaNet-H [16]	IoU	ResNet-50	416×416	79.75	80.40	47.14	45.30
	IoU with HC	ResNet-50	416×416	88.13	91.16	61.63	59.99
	IoU with SC	ResNet-50	416×416	87.95	90.77	60.37	58.84
	SIoU	ResNet-50	416×416	75.73	77.78	48.03	47.62
	SIoU with HC	ResNet-50	416×416	87.58	90.17	63.11	62.91
	SIoU with SC	ResNet-50	416×416	87.80	90.30	**64.19**	64.17
FR-O [60]	IoU	ResNet-50	416×416	86.52	89.16	53.12	56.43
	IoU with HC	ResNet-50	416×416	**89.14**	**92.89**	63.53	63.44
	IoU with SC	ResNet-50	416×416	88.52	91.65	62.74	62.75
	SIoU	ResNet-50	416×416	87.18	90.21	59.25	57.60
	SIoU with HC	ResNet-50	416×416	88.76	92.84	63.16	63.22
	SIoU with SC	ResNet-50	416×416	88.79	92.55	63.72	**65.13**
FR-O [60]	SIoU with SC	ResNet-101	800×800	**89.53**	**96.73**	**75.07**	**77.95**

All models use data augmentation and multiscale training.
The best two results are shown in bold font.

it is clear that the performance of SIoU with SC is better when mAP_{75} is considered. Our method can produce detection results of higher quality.

D. Comparison With Other Metrics

1) Results About GIoU [20] on DOTA [60]: Since GIoU does not explicitly consider the center distance between two boxes, we mainly compare it to the baseline method. As shown in Table III, when we set three scales, the performance of the model using GIoU is slightly improved with unchanged thresholds. With modified thresholds, its performance decreases a little bit. If the model contains only one scale per feature map, its performance changes little when using GIoU. In summary, GIoU focuses on non-overlapping boxes in regression, which is not important in label assignment. Therefore, GIoU does not have a significant impact on the results when applied in label assignment.

2) Results About DIoU [21] and CIoU [21]: The penalty term in DIoU and CIoU is a little bit like the "SC" we proposed, so we compare their performances to "IoU with SC." As shown in Table III, no matter the model is with three scales or one scale, they do not behave better than "IoU with SC." The center constraint we proposed consists of two terms, corresponding to the two directions of a 2-D rotated rectangle. The penalty term in DIoU only considers the Euclidean distance of two centers, so it is not suitable for objects with large aspect ratios,

TABLE III
COMPARISON BETWEEN DIFFERENT METRICS ON DOTA [60]

Method	Metric	Plane	BD	Bridge	GTF	SV	LV	Ship	TC	BC	ST	SBF	RA	Harbor	SP	HC	mAP
†RetinaNet-H [16]	IoU	87.81	80.43	42.71	66.54	70.53	58.07	73.73	90.87	79.37	71.25	50.43	62.34	62.06	66.68	53.32	67.74
	GIoU	88.03	76.38	43.25	65.35	73.61	62.52	76.34	90.83	81.61	69.93	52.54	61.70	61.46	66.51	51.80	68.12(**+0.38**)
	^{e}GIoU	86.63	76.22	43.75	66.53	64.94	56.65	73.49	90.76	81.00	72.07	53.22	61.06	62.05	68.48	53.05	67.33(**-0.41**)
	IoU with SC	88.89	79.53	43.11	65.71	76.87	74.04	83.93	90.65	84.36	81.50	50.88	64.19	63.92	67.18	51.33	71.07
	DIoU	88.08	77.16	43.14	60.45	72.91	61.53	74.78	90.82	80.96	78.23	50.43	64.37	61.49	66.06	48.77	67.95(**-3.12**)
	CIoU	89.09	77.81	42.91	62.54	74.31	61.42	75.88	90.87	80.20	79.78	53.45	63.23	62.28	67.74	50.27	68.78(**-2.29**)
⋆RetinaNet-H [16]	IoU	83.70	70.93	44.25	60.20	76.24	61.90	74.77	90.68	77.49	71.44	39.53	65.01	60.71	62.12	53.36	66.15
	GIoU	86.07	70.79	43.01	60.02	76.96	66.58	75.97	90.73	77.48	70.60	40.19	62.83	60.41	65.19	45.29	66.14(**-0.01**)
	^{e}GIoU	84.76	73.39	45.64	60.99	76.91	62.52	75.83	90.81	78.01	72.73	38.81	63.45	61.00	65.52	48.56	66.59(**+0.44**)
	IoU with SC	85.71	72.66	44.47	58.35	77.13	72.08	82.49	90.75	80.58	79.68	41.23	63.50	62.57	64.80	54.91	68.73
	DIoU	85.90	72.53	42.06	60.83	77.30	65.91	76.05	90.72	74.65	80.31	44.92	61.02	59.99	64.48	51.39	67.20(**-1.53**)
	CIoU	85.82	72.23	42.48	57.07	77.49	65.21	76.52	90.60	76.98	80.14	39.26	63.82	60.25	65.96	47.49	66.75(**-1.98**)

Models marked with ⋆ are using 1 scale every feature layer to generate anchors. Three scales are applied in models with †.
Models marked with e are with the modified thresholds to make sure that the number of positive and negative samples is the same as the baseline method.

TABLE IV
COMPARISON WITH OTHER METHODS ON DOTA [60]

Method	Backbone	Plane	BD	Bridge	GTF	SV	LV	Ship	TC	BC	ST	SBF	RA	Harbor	SP	HC	mAP
PIoU [50]	DLA-34	80.9	69.7	24.1	60.2	38.3	64.4	64.8	**90.9**	77.2	70.4	46.5	37.1	57.1	61.9	64.0	60.5
R^{2}CNN [39]	ResNet101	80.94	65.67	35.34	67.44	59.92	50.91	55.81	90.67	66.92	72.39	55.06	52.23	55.14	53.35	48.22	60.67
RRPN [42]	ResNet101	88.52	71.20	31.66	59.30	51.85	56.19	57.25	90.81	72.84	67.38	56.69	52.84	53.08	51.94	53.58	61.01
RoI Transformer [27]	ResNet101	88.64	78.52	43.44	75.92	68.81	73.68	83.59	90.74	77.27	81.46	58.39	53.54	62.83	58.93	47.67	69.56
SCRDet [10]	ResNet101	89.98	80.65	52.09	68.36	68.36	60.32	72.41	90.85	**87.94**	**86.86**	65.02	66.68	66.25	68.24	65.21	72.61
R^{3}Det [16]	ResNet152	89.49	81.17	50.53	66.10	70.92	**78.66**	78.21	90.81	85.26	84.23	61.81	63.77	68.16	69.83	67.17	73.74
Gliding Vertex [6]	ResNet101	89.64	85.00	52.26	77.34	73.01	73.14	86.82	90.74	79.02	86.81	59.55	**70.91**	72.94	70.86	57.32	75.02
OSSDet [64]	ResNeXt101	89.49	81.10	51.23	71.30	**76.80**	76.97	**87.27**	90.79	83.43	84.71	60.55	64.92	71.21	70.44	66.00	75.08
FFA [46]	ResNet101	90.1	82.7	54.2	75.2	71.0	**79.9**	83.5	90.7	83.9	84.6	61.2	68.0	70.7	**76.0**	63.7	75.7
APE [65]	ResNeXt101	89.96	83.62	53.42	76.03	74.01	77.16	79.45	90.83	**87.15**	84.51	**67.72**	60.33	**74.61**	**71.84**	65.55	75.75
CSL (FPN based) [11]	ResNet152	**90.25**	**85.53**	**54.64**	75.31	70.44	73.51	77.62	90.84	86.15	86.69	**69.60**	68.04	73.83	71.10	**68.93**	76.17
OPLD [15]	ResNet101	89.37	**85.82**	54.10	**79.58**	75.00	75.13	**86.92**	90.88	86.42	86.62	62.46	**68.41**	**73.98**	68.11	63.69	**76.43**
[a]SIoU with SC (RetinaNet-H [16])	ResNet50	**90.15**	83.78	49.64	**77.42**	**78.53**	78.43	**86.92**	90.79	85.40	85.16	65.67	65.21	**73.98**	70.60	63.25	76.33
SIoU with SC (FR-O [60])	ResNet50	89.39	84.00	**54.76**	73.71	76.22	78.31	83.73	**90.87**	84.96	**87.37**	62.38	67.15	73.86	69.44	**71.98**	**76.54**

In the model marked with a, $\lambda = 0.3$.

as shown in Fig. 1(b). Furthermore, we set the distance within a specific range to avoid punishment, which also improves the performance.

E. Comparison With the State-of-The-Art

1) DOTA [60]: With data augmentation, we obtain competitive results on DOTA [60]. As shown in Table IV, our model stands out in categories whose objects are with dense arrangements and large aspect ratios, like small vehicle (SV), ship, and large vehicle (LV). Combined with the visualized results, we learn that our new method helps to reduce the number of low-quality candidate regions via center constraint, and further reduce the results of false positive.

2) HRSC2016 [61]: We utilize data augmentation in all experiments to increase data volume for FR-O [60] and RetinaNet-H [16]. As shown in Table V, our performance is comparable with the best method using VOC07 metric to measure, and is better when VOC12 metric is used.

TABLE V
COMPARISON WITH OTHER METHODS ON HRSC2016 [61]

Method	mAP$_{50}$(07)	mAP$_{50}$(12)
RoI Transformer [27]	86.20	-
RSDet [48]	86.50	-
CenterMap [66]	-	92.8
SBD [67]	-	93.70
Gliding Vertex [6]	88.20	-
OPLD [15]	88.44	-
PIoU [50]	89.20	-
SLA [68]	89.51	-
S^{2}A-Net [69]	**90.17**	95.01
R^{3}Det [16]	89.26	96.01
FPN-CSL [11]	89.62	**96.10**
DAL [54]	89.77	-
OSSDet [64]	**89.91**	-
SIoU with SC (FR-O [17])	89.53	**96.73**

VI. CONCLUSION

In this article, we introduce a new metric named constrained-SIoU to describe the overlap between a horizontal box and a multi-oriented object. Compared to IoU, SIoU is a better indicator of how much information a horizontal box contains about the object. Combined with the center constraint, SIoU can be applied to label assignment and help improve the model's performance. Our experiments show that our proposed constrained-SIoU has positive effects in both single-stage detectors and two-stage detectors. In the future, for one thing, we will work on the adaptive label assignment for the assignment method we adopt is based on the fixed thresholds. For another, we will explore the application of SIoU to loss functions.

ACKNOWLEDGMENT

The authors would like to thank Z. Lian for the suggestions on the writing.

REFERENCES

[1] G.-S. Xia *et al.*, "AID: A benchmark data set for performance evaluation of aerial scene classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 7, pp. 3965–3981, Jul. 2017.

[2] 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.

[3] G. Cheng *et al.*, "SPNet: Siamese-prototype network for few-shot remote sensing image scene classification," *IEEE Trans. Geosci. Remote Sens.*, to be published, doi: 10.1109/TGRS.2021.3099033.

[4] X. Yao, X. Feng, J. Han, G. Cheng, and L. Guo, "Automatic weakly supervised object detection from high spatial resolution remote sensing images via dynamic curriculum learning," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 1, pp. 675–685, Jan. 2020.

[5] X. Feng, J. Han, X. Yao, and G. Cheng, "Progressive contextual instance refinement for weakly supervised object detection in remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 11, pp. 8002–8012, Nov. 2020.

[6] 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. 1442–1459, Apr. 2021.

[7] M. Zhang, G. Xu, K. Chen, M. Yan, and X. Sun, "Triplet-based semantic relation learning for aerial remote sensing image change detection," *IEEE Geosci. Remote Sens. Lett.*, vol. 16, no. 2, pp. 266–270, Feb. 2019.

[8] M. Gong, Y. Yang, T. Zhan, X. Niu, and S. Li, "A generative discriminatory classified network for change detection in multispectral imagery," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 12, no. 1, pp. 321–333, Jan. 2019.

[9] X. Zheng, X. Chen, X. Lu, and B. Sun, "Unsupervised change detection by cross-resolution difference learning," *IEEE Trans. Geosci. Remote Sens.*, to be published, doi: 10.1109/TGRS.2021.3079907.

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

[11] X. Yang and J. Yan, "Arbitrary-oriented object detection with circular smooth label," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 677–694.

[12] H. Bilen and A. Vedaldi, "Weakly supervised deep detection networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 2846–2854.

[13] P. Wang, X. Sun, W. Diao, and K. Fu, "FMSSD: Feature-merged single-shot detection for multiscale objects in large-scale remote sensing imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 5, pp. 3377–3390, May 2020.

[14] H. Wei, Y. Zhang, Z. Chang, H. Li, H. Wang, and X. Sun, "Oriented objects as pairs of middle lines," *Int. Soc. J. Photogrammetry Remote Sens.*, vol. 169, pp. 268–279, 2020. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0924271620302690

[15] Q. Song, F. Yang, L. Yang, C. Liu, M. Hu, and L. Xia, "Learning point-guided localization for detection in remote sensing images," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 14, pp. 1084–1094, Jan. 2021.

[16] X. Yang, J. Yan, Z. Feng, and T. He, "R3Det: Refined single-stage detector with feature refinement for rotating object," *Proc. Conf. Assoc. Advance. Artif. Intell.*, vol. 35, no. 4, pp. 3163–3171, May 2021. [Online]. Available: https://ojs.aaai.org/index.php/AAAI/article/view/16426

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

[18] 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.

[19] 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.

[20] H. Rezatofighi, N. Tsoi, J. Gwak, A. Sadeghian, I. Reid, and S. Savarese, "Generalized intersection over union: A metric and a loss for bounding box regression," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 658–666.

[21] Z. Zheng, P. Wang, W. Liu, J. Li, R. Ye, and D. Ren, "Distance-IoU loss: Faster and better learning for bounding box regression," in *Proc. Conf. Assoc. Adv. Artif. Intell.*, 2020, pp. 12993–13000.

[22] J. Dai, Y. Li, K. He, and J. Sun, "R-FCN: Object detection via region-based fully convolutional networks," in *Proc. 30th Int. Conf. Adv. Neural Inf. Process. Syst.*, 2016, pp. 379–387.

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

[24] X. Zhang, F. Wan, C. Liu, R. Ji, and Q. Ye, "FreeAnchor: Learning to match anchors for visual object detection," in *Proc. 33rd Int. Conf. Neural Inf. Process. Syst.*, 2019, pp. 147–155.

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

[26] 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.

[27] 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 Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 2849–2858.

[28] H. Law and J. Deng, "CornerNet: Detecting objects as paired keypoints," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 734–750.

[29] K. Duan, S. Bai, L. Xie, H. Qi, Q. Huang, and Q. Tian, "CenterNet: Keypoint triplets for object detection," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2019, pp. 6569–6578.

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

[31] H. Qiu, Y. Ma, Z. Li, S. Liu, and J. Sun, "BorderDet: Border feature for dense object detection," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 549–564.

[32] X. Pan *et al.*, "Dynamic refinement network for oriented and densely packed object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 11 207–11 216.

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

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

[35] 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.*, 2015, pp. 580–587.

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

[37] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 2961–2969.

[38] Z. Cai and N. Vasconcelos, "Cascade R-CNN: Delving into high quality object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 6154–6162.

[39] Y. Jiang *et al.*, "R2CNN: Rotational region CNN for orientation robust scene text detection," 2017, *arXiv:1706.09579*.

[40] M. Liao, B. Shi, and X. Bai, "TextBoxes: A single-shot oriented scene text detector," *IEEE Trans. Image Process.*, vol. 27, no. 8, pp. 3676–3690, Aug. 2018.

[41] M. Liao, Z. Zhu, B. Shi, G.-S. Xia, and X. Bai, "Rotation-sensitive regression for oriented scene text detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 5909–5918.

[42] J. Ma *et al.*, "Arbitrary-oriented scene text detection via rotation proposals," *IEEE Trans. Multimedia*, vol. 20, no. 11, pp. 3111–3122, Nov. 2018.

[43] 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.

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

[45] S. M. Azimi, E. Vig, R. Bahmanyar, M. Körner, and P. Reinartz, "Towards multi-class object detection in unconstrained remote sensing imagery," in *Proc. Asian Conf. Comput. Vis.*, 2018, pp. 150–165.

[46] 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," *Int. Soc. J. Photogrammetry Remote Sens.*, vol. 161, pp. 294–308, 2020.

[47] L. Liu, Z. Pan, and B. Lei, "Learning a rotation invariant detector with rotatable bounding box," 2017, *arXiv:1711.09405*.

[48] W. Qian, X. Yang, S. Peng, J. Yan, and Y. Guo, "Learning modulated loss for rotated object detection," in *Proc. AAAI Conf. Artif. Intell.*, May 2021, vol. 35, no. 3, pp. 2458–2466. [Online]. Available: https://ojs.aaai.org/index.php/AAAI/article/view/16347

[49] J. Yu, Y. Jiang, Z. Wang, Z. Cao, and T. Huang, "UnitBox: An advanced object detection network," in *Proc. 24th ACM Int. Conf. Multimedia*, 2016, pp. 516–520.

[50] Z. Chen *et al.*, "PioU loss: Towards accurate oriented object detection in complex environments," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 195–211.

[51] X. Yang, J. Yan, Q. Ming, W. Wang, X. Zhang, and Q. Tian, "Rethinking rotated object detection with Gaussian Wasserstein distance loss," *Int. Conf. Mach. Learning*, pp. 11830–11841, 2021.
[52] H. Zhang, H. Chang, B. Ma, N. Wang, and X. Chen, "Dynamic R-CNN: Towards high quality object detection via dynamic training," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 260–275.
[53] H. Li, Z. Wu, C. Zhu, C. Xiong, R. Socher, and L. S. Davis, "Learning from noisy anchors for one-stage object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 10 588–10 597.
[54] Q. Ming, Z. Zhou, L. Miao, H. Zhang, and L. Li, "Dynamic anchor learning for arbitrary-oriented object detection," *Proc. AAAI Conf. Artif. Intell.*, vol. 35, no. 3, pp. 2355–2363, May 2021. [Online]. Available: https://ojs.aaai.org/index.php/AAAI/article/view/16336
[55] Y. Liu, X. Tang, J. Han, J. Liu, D. Rui, and X. Wu, "HamBox: Delving into mining high-quality anchors on face detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 13043–13051.
[56] P. Sun, Y. Jiang, E. Xie, Z. Yuan, C. Wang, and P. Luo, "OneNet: Towards end-to-end one-stage object detection," 2020, *arXiv:2012.05780*.
[57] Q. Zhou, C. Yu, C. Shen, Z. Wang, and H. Li, "Object detection made simpler by eliminating heuristic NMS," 2021, *arXiv:2101.11782*.
[58] J. Wang, L. Song, Z. Li, H. Sun, J. Sun, and N. Zheng, "End-to-end object detection with fully convolutional network," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 15849–15858.
[59] Z. Tian, C. Shen, H. Chen, and T. He, "FCOS: Fully convolutional one-stage object detection," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2020, pp. 9626–9635.
[60] 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.
[61] 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.
[62] K. Chen *et al.*, "MMDetection: Open mmlab detection toolbox and benchmark," 2019, *arXiv:1906.07155*.
[63] 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.
[64] S.-B. Chen, B.-M. Dai, J. Tang, B. Luo, W. Wang, and K. Lu, "A refined single-stage detector with feature enhancement and alignment for oriented objects," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 14, pp. 8898–8908, 2021.
[65] Y. Zhu, J. Du, and X. Wu, "Adaptive period embedding for representing oriented objects in aerial images," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 10, pp. 7247–7257, Oct. 2020.
[66] J. Wang, W. Yang, H.-C. Li, H. Zhang, and G.-S. Xia, "Learning center probability map for detecting objects in aerial images," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 5, pp. 4307–4323, May 2021.
[67] Y. Liu, S. Zhang, L. Jin, L. Xie, Y. Wu, and Z. Wang, "Omnidirectional scene text detection with sequential-free box discretization," in *Proc. 28th Int. Joint Conf. Artif. Intell.*, 2019, pp. 3052–3058. [Online]. Available: https://doi.org/10.24963/ijcai.2019/423
[68] Q. Ming, L. Miao, Z. Zhou, J. Song, and X. Yang, "Sparse label assignment for oriented object detection in aerial images," *Remote Sens.*, vol. 13, no. 14, 2021. [Online]. Available: https://www.mdpi.com/2072-4292/13/14/2664
[69] J. Han, J. Ding, J. Li, and G. S. Xia, "Align deep features for oriented object detection," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, pp. 1–11, 2021.

Yanan Zhang received the B.E. degree in software engineering from Xiamen University, Fujian, China, in 2018. She is currently working toward the Ph.D. degree with the Science and Technology on Integrated Information System Laboratory, Institute of Software, Chinese Academy of Sciences, Beijing, China, and also with the University of Chinese Academy of Sciences, Beijing. She is currently majoring in deep learning and object detection.

Haichang Li received the B.S. degree in information and computing science from Shandong University, Shandong, China, in 2007, and the M.S. degree in computer application technology and Ph.D. degree in pattern recognition and intelligent system from the Institute of Automation, Chinese Academy of Sciences, Beijing, China, in 2011 and 2016, respectively.

He is currently an Associated Professor with Science and Technology on Integrated Information System Laboratory, Institute of Software, Chinese Academy of Sciences. His research interests include computer vision, machine learning, and remote sensing.

Rui Wang received the B.E. degree in computer science and technology from the Ocean University of China, Shandong, China, in 2009, and the M.E. degree in computer software and theory from Shandong University, Shandong, in 2012.

She is currently a Senior Engineer with Science and Technology on Integrated Information System Laboratory, Institute of Software, Chinese Academy of Sciences, Beijing, China. Her research interests include deep reinforcement learning and multimedia technology and systems.

Mengya Zhang received the B.E. degree in automation from the University of Science and Technology Beijing, Beijing, China, in 2016, and the M.E. degree in signal and information processing from the Institute of Electronics, Chinese Academy of Sciences, Beijing, in 2019.

She is currently an Engineer with Science and Technology on Integrated Information System Laboratory, Institute of Software, Chinese Academy of Sciences. Her research interests include deep learning, object detection, and semantic segmentation.

Xiaohui Hu received the Ph.D. degree in computer application technology from Beihang University, Beijing, China, in 2003.

He is a Professor with the Science and Technology on Integrated Information System Laboratory, Institute of Software, Chinese Academy of Sciences, Beijing. His research interests include intelligent information processing, Big Data analytics, and cooperative multiagent systems.