# Gradient Calibration Loss for Fast and Accurate Oriented Bounding Box Regression

Qi Ming, Lingjuan Miao, Zhiqiang Zhou, *Member, IEEE*, Junjie Song, and Aleksandra Pižurica, *Senior Member, IEEE*

***Abstract*— Oriented object detection has a very wide range of application scenarios. In recent years, a lot of rotation detectors have been designed to achieve high-performance oriented object detection. Intersection-over-union (IoU) is the commonly used indicator to evaluate the accuracy of detection performance. Many methods introduce IoU into the bounding box regression loss to achieve the aligned training and evaluation process for better performance. However, in this article, we demonstrate several drawbacks of rotated IoU loss through both experiments and theoretical derivation: 1) there is a negative correlation between the loss gradient and the angular error and 2) the optimization process of rotated IoU loss suffers from scale sensitivity, which is not conducive to model convergence. To solve the problems, we propose a gradient calibration loss (GCL) that optimizes the rotated IoU loss via gradient analysis and correction. We construct the optimized gradient in GCL to avoid IoU loss oscillation and scale sensitivity, thereby accelerating model convergence. Models supervised by GCL have a more stable training process, faster convergence, and better performance. Moreover, GCL can be easily introduced into the existing rotation detectors to achieve performance gains without extra inference overhead. Extensive experiments on multiple oriented object detection datasets and models demonstrate the superiority of our method. Our method achieves state-of-the-art performance on the mainstream benchmark datasets. The source code and models are available at https://github.com/ming71/GCL.**

***Index Terms*— Convolutional neural network, gradient analysis, loss function, oriented object detection.**

## I. Introduction

REMOTE sensing object detection aims to extract objects of interest from remote sensing images, such as ships, vehicles, bridges, and roundabouts. With the development of Earth observation technology, the available high-resolution remote sensing image data are growing explosively. This provides great opportunities for the development of object detection in remote sensing imagery. As an important way to interpret remote sensing imagery, remote sensing object detection has been widely applied to military reconnaissance [1], environmental monitoring [2], urban planning [3], and other fields.

Early remote sensing object detection algorithms were based on manually designed features. Manually designed features are extracted from visual features clues such as color, texture, and contour of the objects [4], [5]. They are highly interpretable, but have poor representation ability, are not robust enough, and are very time-consuming. With the breakthrough in deep learning, the powerful feature extraction capability of convolutional neural networks (CNNs) greatly improves detection performance [6], [7], [8]. Remote sensing images often have complex backgrounds, in which objects have large aspect ratios and large-scale variations. Many existing object detectors cannot achieve good performance on remote sensing imagery. Therefore, a large number of solutions have been proposed to improve the performance of object detection in the field of remote sensing [9], [10], [11].

Generic object detection predicts horizontal bounding box (HBB) to locate objects [6], [7], [12], [13], [14], [15]. However, objects in remote sensing imagery are often arbitrary-oriented. Therefore, oriented bounding box (OBB) is adopted to accurately represent the location of objects to achieve better detection performance. The mainstream methods in oriented object detection inherit from generic object detection framework with an additional angle prediction being introduced to predict OBB [9], [10], [11], [16], [17]. A series of methods are proposed to improve label assignment [18], [19], angle representation [10], [20], [21], and loss function [16], [22] in oriented object detection to achieve accurate detections.

Despite great success, the above methods still suffer from misalignment between evaluation metric and regression loss [16], [22], [23]. The current mainstream regression loss is $L_n$-norm loss, while average precision (AP) is the most commonly used evaluation metric in these methods. Yang et al. [16] suggested that a smaller regression loss cannot guarantee a better performance. That is, the values of $L_n$-norm loss and AP metric are not completely positively related. In this case, optimizing the training loss of the model may not bring performance gains. Intersection-over-union (IoU) is a good candidate to bridge regression loss and evaluation criteria. IoU has been extensively used as regression loss in horizontal object detection, such as GIoU loss [24] and DIoU loss [25]. IoU-based losses inherently help bridge the gap between

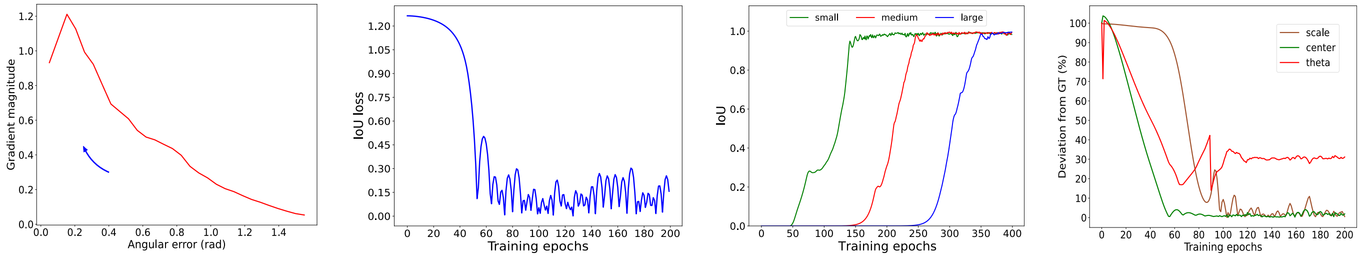
Manuscript received 4 November 2023; revised 23 December 2023 and 22 January 2024; accepted 12 February 2024. Date of publication 19 February 2024; date of current version 28 February 2024. *(Corresponding author: Zhiqiang Zhou.)*

Qi Ming, Lingjuan Miao, Zhiqiang Zhou, and Junjie Song are with the School of Automation, Beijing Institute of Technology, Beijing 100081, China (e-mail: chaser.ming@gmail.com; miaolingjuan@bit.edu.cn; zhzhzhou@bit.edu.cn; bitsjj@gmail.com).

Aleksandra Pižurica is with the Faculty of Engineering and Architecture, Ghent University, 9000 Ghent, Belgium (e-mail: aleksandra.pizurica@ugent.be).

This article has supplementary downloadable material available at https://doi.org/10.1109/TGRS.2024.3367294, provided by the authors.

Digital Object Identifier 10.1109/TGRS.2024.3367294



(a) (b) (c) (d)

Fig. 1. Visualization of the loss variation and convergence speed in oriented object detection. (a) Gradient variation in rotated IoU loss with respect to angular error during training. (b) Oscillation of the rotated IoU loss during regression. (c) Convergence speed of objects at different scales under the supervision of rotated IoU loss. (d) Convergence process of variables in OBB representation.

regression loss and performance evaluation. Intuitively, IoU can also be applied into OBB regression loss for high-quality oriented object detection.

However, we found through both the experimental and theoretical analyses that the IoU-based loss functions suffer from many difficulties in oriented object detection. First, we find through experiments that a large angular deviation between the prediction and ground truth (GT) box leads to small gradients of the rotated IoU loss [see Fig. 1(a)], which would lead to slow convergence of angle regression. Furthermore, as the angular error converges, the gradient magnitude would increase. These two cases would cause the regression loss to oscillate and make the model hard to converge. It can be seen from Fig. 1(a) that when the predicted box approaches the GT box, the angular error decreases gradually, but it produces an increasing gradient. The anomalous gradient changes can cause the oscillation of angle prediction and even induce nonconvergence of the loss [see Fig. 1(b)]. Second, the gradient of the rotated IoU loss is negatively correlated with the size of the objects. As shown in Fig. 1(c), large objects have small gradients, which leads to slow convergence. On the contrary, small objects have large gradients and are difficult to be precisely predicted. Although the rotated IoU is scale-invariant, the optimization process of the IoU loss is abnormally sensitive to the scale variations, i.e., rotated IoU loss still suffers from scale sensitivity. We suggest that these properties are the causes that make the rotated IoU loss not well-applied to achieve consistent training and evaluation for oriented object detection. The detailed theoretic proof of the above observations will be given in Section III.

In this article, we demonstrate through gradient analysis methods that abnormal gradient changes cause the above problems. On this basis, we propose the gradient calibration loss (GCL) for fast and accurate OBB regression. We solve the drawbacks of rotated IoU loss by reconstructing gradient convergence in GCL. Specifically, GCL alleviates the negative correlation between the loss gradient and angular error. In this way, a faster and more stable angle convergence can be ensured. Furthermore, a gradient scaling strategy is introduced into GCL to adapt to bounding boxes in different sizes. It helps make objects of all scales to be well-regressed and endows the IoU loss with scale invariance during the regression process. We apply our method to many mainstream rotation detectors to demonstrate its generality. Extensive experiments on multiple datasets confirm that our GCL loss achieves steady and significant performance gains. Our method achieves state-of-the-art results on multiple datasets for oriented object detection.

The main contributions of this article are summarized as follows.

1) We demonstrate several drawbacks of rotated IoU loss in oriented object detection through both experiments and theoretical derivation. First, there is a negative correlation between the loss gradient and the angular error. Second, the optimization process of rotated IoU loss suffers from scale sensitivity. These factors hinder the convergence of the model, making it hard to produce high-quality detections.
2) We summarize some properties of a good regression loss for oriented object detection. The gradient calibration loss (GCL) is accordingly designed to resolve the oscillation of loss function and the difficulty in model convergence by correcting the gradient of regression loss.
3) We analyze the scale invariance in regression loss. Then, a gradient scaling strategy is introduced into GCL for adaptation of object scales.
4) Our method can be flexibly embedded into the existing detectors without introducing additional computational cost. Experiments on multiple datasets demonstrate the superiority of our method. GCL achieves steady performance improvements over multiple state-of-the-art methods.

## II. Related Work

### A. Generic Object Detection

In the past decades, considerable achievements have been made in the field of object detection [6], [7], [12], [13], [14], [26]. Generic object detection methods use HBB to represent objects in images. The mainstream detectors can be divided into two categories: one-stage detectors [12], [13], [14], [26] and two-stage detectors [6], [7]. Two-stage detectors first generate some proposals, and then perform feature extraction and prediction on them to obtain detection results. For example, Faster R-CNN [7] uses a region proposal network (RPN) to generate a series of proposals that might contain objects. Region of Interest (RoI) pooling is then applied to extract the features of the proposals for subsequent classification and regression. The one-stage detectors treat the object detection as a regression problem. For example, YOLO [12] and SSD [14] are convolutional network that simultaneously predict bounding boxes and the corresponding class probabilities in one step. In general, the two-stage detectors have higher detection accuracy while the one-stage detector has faster inference speed.

There are some work trying to optimize the training loss to improve the detection performance [27], [28]. Lin et al. [27] focused on the extreme imbalance between the foreground and the background in object detection and designed the focal loss for one-stage detectors. The focal loss increases the loss contribution of positive samples to avoid the classification loss being dominated by massive negative samples. Yu et al. [28] suggested that evaluation metric could be used to construct a regression loss, and proposed the IoU loss to achieve the consistency of training and evaluation. The IoU loss ensures that the optimization of the training loss can bring stable performance gains and is widely used to achieve high-precision object detection. Recently, some variants of IoU loss have been proposed. For example, Rezatofighi et al. [24] proposed GIoU loss to optimize the IoU calculation for nonoverlapping case of two bounding boxes to accelerate loss convergence. DIoU loss [25] imposes additional supervision on the distance between the center points of two bounding boxes to speed up the loss convergence.

### B. Oriented Object Detection

Objects in remote sensing images are arbitrarily oriented. When HBB is used to represent a rotated remote sensing object, the bounding box will contain a lots of background information, which cannot accurately represent the location of the object and affects the subsequent feature extraction process. In contrast, OBB fully surrounds the rotated object in remote sensing imagery and can represent it more accurately. Therefore, oriented object detection has received increasing attention in recent years. A series of rotation detectors have been proposed to solve the problem [9], [10], [17], [18]. Rotation detectors use OBBs to accurately represent rotated objects. Many methods are inherited from generic object detection methods and additionally predict the orientation for oriented object detection in remote sensing imagery [17], [19], [29], [30]. For example, rotation RPN (RRPN) was proposed [31] to generate inclined proposals with orientation angle information. Zand et al. [32] introduced angle prediction and box refinement into the classic one-stage detector to detect oriented remote sensing objects.

Recently, research in remote sensing object detection field has gradually focused on the optimization of detection pipeline based on the properties of rotated objects. Some studies optimize the label assignment for rotated objects [18], [33]. Specifically, Ming et al. [18] observed the inconsistent performance of rotated anchors before and after regression in aerial imagery, and then proposed a dynamic anchor learning strategy to adaptively select high-quality anchors. Feature extraction for rotated objects is also improved to obtain better performance in remote sensing object detection [9], [17], [34], [35], [36]. ReDet [35] designed rotation-invariant features and rotation-equivariant features for the classification and regression subtasks of OBB prediction, respectively. R$^3$Det [34] and S$^2$A-Net [17] aligned the OBBs and their features to obtain accurate localization.

There are also some work that focuses on the design of loss functions for OBB regression [10], [16], [19], [20], [22], [30]. For example, Yang and Yan [10] found that the periodicity of the angle in OBB representation makes the common $L_n$-norm loss hard to converge. Then, they proposed circular smooth label (CSL) to turn the angle regression task into the angle classification task to avoid the problem. Miao et al. [20] suggested that the redundant representations of an OBB derived from angle periodicity can be treated as equivalent local minima for regression task. Subsequently, they proposed a representation invariant loss (RIL) to uniformly optimize redundant representations to achieve better performance. Some methods attempt to solve the above problems with novel representations of OBB. Yan et al. [16], [22] approximate the rotated object with a Gaussian distribution, and then use Gaussian Wasserstein distance [16] or Kullback–Leibler divergence [22] to measure the distance between OBBs. However, these metrics are just approximations of IoU, and the corresponding loss drop cannot fully guarantee the performance gain [37], [38]. In particular, Yan et al. [37] observed discrepancies in the convergence trends between IoU approximation metrics and actual IoU, and therefore, the regression loss fails to guarantee improved detection performance. In summary, there is still much room for further improvement.

## III. Revisit of Rotated IoU Loss in Oriented Object Detection

### A. Preliminaries

In this section, we will review IoU loss in OBB regression and demonstrate its potential problems.

IoU, also known as Jaccard index, is the most popular evaluation metric to measure the detection performance in object detection. Meanwhile, IoU can also be used as the regression loss to ensure consistency between performance evaluation and training loss. The most intuitive form of IoU loss is as follows:

$$L_{\mathrm{IoU}}\big(\boldsymbol{b},\boldsymbol{b}^*\big)=1-\mathrm{IoU}\big(\boldsymbol{b},\boldsymbol{b}^*\big). \tag{1}$$

There is also another commonly used nonlinear IoU loss

$$L_{\mathrm{lnIoU}}\big(\boldsymbol{b},\boldsymbol{b}^*\big)=-\ln\big[\mathrm{IoU}\big(\boldsymbol{b},\boldsymbol{b}^*\big)\big] \tag{2}$$

where $\boldsymbol{b}$ and $\boldsymbol{b}^*$ are the vectors of predictions and GT boxes, respectively. Specifically, $\boldsymbol{b}=(x,y,w,h,\theta)$, where $(x,y)$ denotes the center point, and $(w,h,\theta)$ are the width, height, and angle of OBB, respectively. $\mathrm{IoU}(\cdot)$ is used to calculate the rotated IoU between $\boldsymbol{b}$ and $\boldsymbol{b}^*$.

Calculation of rotated IoU between two OBBs is much more complicated than that of HBBs. However, we suggest that the rotated IoU loss can be simplified for the study in this article. As shown in Fig. 1(d), the center point converges very quickly under the supervision of the rotated IoU loss. The angle and the box shape, however, converge slowly and even oscillate. As a result, we can conclude that the optimization of center point does not cause slow convergence of the rotated IoU loss, and therefore only consider a simplified rotated IoU loss

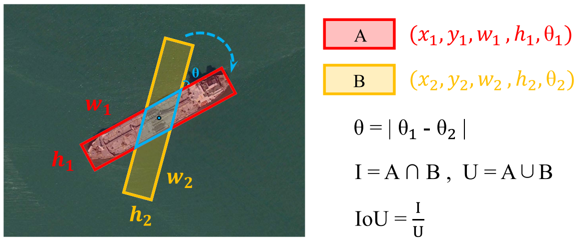

A
$(x_1, y_1, w_1, h_1, \theta_1)$
B
$(x_2, y_2, w_2, h_2, \theta_2)$
$\theta = |\theta_1 - \theta_2|$
$I = A \cap B$, $U = A \cup B$
$IoU = \frac{I}{U}$
$w_1$
$h_1$
$w_2$
$h_2$
$\theta$

Fig. 2. Illustration of the geometric relationship between two OBBs with aligned center points.

between two center-aligned OBBs (see Fig. 1) for theoretically analysis.

For simplicity, we consider the case of analyzing a pair of samples. Given two center-aligned OBBs $A$ and $B$, their representation vectors are $\boldsymbol{b}^* = (x_1, y_1, w_1, h_1, \theta_1)$ and $\boldsymbol{b} = (x_2, y_2, w_2, h_2, \theta_2)$, where $x_1 = x_2$ and $y_1 = y_2$. $\boldsymbol{b}^*$ and $\boldsymbol{b}$ are the vectors of length 5. Note that $\boldsymbol{b^*}$ is the GT box and $\boldsymbol{b}$ denotes predicted box. The geometric relationship between two center-aligned OBBs is shown in Fig. 2. In this case, the following geometric relationship can be obtained:

$$\begin{cases} I = \dfrac{h_1 h_2}{\sin\theta} \\ U = w_1 h_1 + w_2 h_2 - I \end{cases} \tag{3}$$

where $I$ and $U$ are the intersection and union of two OBBs, respectively. And then

$$\text{IoU} = \frac{I}{U} = \frac{h_1 h_2}{(w_1 h_1 + w_2 h_2) \cdot \sin\theta - h_1 h_2} \tag{4}$$

in which $\theta$ is the angular deviation between OBBs in Fig. 2 and $\theta = |\theta_1 - \theta_2|$. Note that (4) only holds when the intersection between prediction box and GT box is a parallelogram, and this situation accounts for the majority of scenarios during the regression process. Specifically, $\theta \in [\theta_0, (\pi/2)]$, and $\theta_0$ is the boundary conditions under which the above assumptions hold. $\theta_0$ can be easily calculated from the geometric relationship. Meanwhile, as shown in Fig. 1(a), the gradient of the IoU loss with respect to the angular error would converge well when the angular error is small. It can be verified through numerical calculations that these well-converged cases appear when $\theta \in [0, \theta_0]$. Therefore, we only consider the abnormal situations with $\theta \in [\theta_0, (\pi/2)]$.

### B. Analysis of Gradient With Respect to Angular Error

The linear IoU loss in (1) is

$$L_{\text{IoU}} = 1 - \text{IoU}\big(\boldsymbol{b}, \boldsymbol{b}^*\big) = 1 - \frac{h_1 h_2}{(w_1 h_1 + w_2 h_2)\sin\theta - h_1 h_2}. \tag{5}$$

Next, combining (4) and (5), the partial derivative of $L_{\text{IoU}}$ with respect to $\theta$ can be obtained as

$$\frac{\partial L_{\text{IoU}}}{\partial\theta} = \text{IoU} \cdot (1 + \text{IoU}) \cdot \cot\theta. \tag{6}$$

As the regression loss converges, $\theta \downarrow$, $\cot\theta \uparrow$, and IoU $\uparrow$, therefore $(\partial L_{\text{IoU}}/\partial\theta) \uparrow$. It shows that the drop in rotated IoU loss leads to the increase in its gradients with respect to angular error, which is consistent with the results in Fig. 1(a). Therefore, the abnormal gradients cause loss oscillation and slow convergence of the model [see Fig. 1(b)]. This issue degrades detection performance more severely for objects with large aspect ratios, such as ships and vehicles. For these objects, a slight angular deviation will result in a sharp drop in rotated IoU.

The above problem also exists in nonlinear IoU loss. Combining (2) and (4), the partial derivative of nonlinear rotated IoU loss $L_{\text{lnIoU}}$ with respect to $\theta$ is

$$\frac{\partial L_{\text{lnIoU}}}{\partial\theta} = (1 + \text{IoU}) \cdot \cot\theta. \tag{7}$$

When $\theta \downarrow$, $\cot\theta \uparrow$, and thus $(\partial L_{\text{lnIoU}}/\partial\theta) \uparrow$, which would also lead to loss oscillation and hinders model convergence. The comparison between (6) and (7) reveals that the gradient of the linear IoU loss increases faster than that of the nonlinear IoU loss. This perhaps can explain, in another perspective, why the nonlinear IoU loss usually achieves relatively better detection performance than the linear IoU loss.

### C. Analysis of Gradient With Respect to Scale

The scale invariance of regression loss has been widely discussed in many literature [22], [24], [25], [28], [39], [40]. We suggest in this article that scale invariance property for regression loss actually includes two aspects.

*Property 1*: Scale invariance of loss metrics.

*Property 2*: Scale invariance of loss optimization process.

*Prop. 1* indicates that the same spatial overlap ratios of objects of different scales produce similar loss values. Generally, IoU loss is considered to be scale-invariant since it satisfies *Prop. 1*. *Prop. 2* means that the optimization steps are adapted to object scales during training. Specifically, the regression loss produces large-step optimization for large objects and small-step optimization for small objects. As a result, even if large-scale objects produce large deviations, the consequent large-step optimization ensures fast correction of predictions.

Few current works consider regression loss from the latter property. The experimental results shown in Fig. 1(c) reveal that the rotated IoU loss does not satisfy *Prop. 2*. Consequently, large objects converge slowly, while small objects converge relatively faster in Fig. 1(c). In the following, we analyze the gradient of rotated IoU loss to further verify this point.

Supposing $s$ is the size of the OBB ($w$ or $h$), we know $h_2, w_2 \propto s$, $I \propto s^2$, and $U \propto s^2$. For linear IoU loss in (5), the partial derivative of $L_{\text{IoU}}$ with respect to $s$ is as follows:

$$\begin{aligned} \frac{\partial L_{\text{IoU}}}{\partial w_2} &= \text{IoU} \cdot \frac{h_2}{U} \propto \frac{1}{s} \\ \frac{\partial L_{\text{IoU}}}{\partial h_2} &= -\text{IoU} \cdot \frac{S_1}{U h_2} \propto \frac{1}{s} \end{aligned} \tag{8}$$

where $S_1 = w_1 h_1$ denotes the area of $\boldsymbol{b}^*$. Therefore, we know $(\partial L_{\text{IoU}}/\partial s) \propto (1/s)$. It means that large-scale OBBs produce small gradients, causing the loss to converge slowly. Conversely, there are relatively large gradients for small objects, which makes the IoU loss hard to converge steadily and even oscillate.

Next, we discuss nonlinear rotated IoU loss in (2)

$$\begin{aligned}\frac{\partial L_{\mathrm{lnIoU}}}{\partial w_2} &= \frac{h_2}{U} \propto \frac{1}{s}\\ \frac{\partial L_{\mathrm{lnIoU}}}{\partial h_2} &= -\frac{S_1}{U h_2} \propto \frac{1}{s}.\end{aligned} \tag{9}$$

We get $(\partial L_{\mathrm{lnIoU}}/\partial s) \propto (1/s)$ for nonlinear rotated IoU loss. Therefore, nonlinear IoU loss still suffers from scale sensitivity, making regression of large objects converge slowly, while small objects hard to locate precisely due to large gradients. In summary, we prove that the rotated IoU loss does not satisfy *Prop. 2*, which would make the optimization process suffer from scale sensitivity.

## IV. Methodology

### *A. Gradient Calibration Loss*

We propose the GCL to improve rotated IoU loss by correcting its gradient. GCL builds corrected gradient with respect to angular error and box scale, and then obtains the optimized regression loss through integration. Then, we will first introduce the gradient correction strategy for angle regression, and then present gradient scaling strategy in Section III.

First, we need to establish criteria for evaluating regression loss functions. For a good regression loss in oriented object detection, we suggest that the following two conditions should be satisfied.

*Condition 1:* The loss is monotonically decreasing with respect to both IoU and angular error.

*Condition 2:* The gradient of the loss with respect to angular error is a nondecreasing function of angular error.

*Cond. 1* ensures the alignment between training loss and evaluation criteria, i.e., better performance lead to lower regression loss. *Cond. 2* ensures stable convergence of the model to obtain high-quality detection. Most of the existing regression losses for oriented object detection only satisfy *Cond. 1*, but do not consider *Cond. 2*, which would lead to unstable training process.

Obviously, the rotated IoU loss satisfies *Cond. 1*, but *Cond. 2* does not hold. Therefore, we construct a series of GCLs to optimize the regression process. An intuitive approach is to directly design an ideal gradient. The ideal gradient of the loss $L$ with respect to angular error $\theta$ should be positively correlated or irrelevant to angular error. For example, we choose the expected gradient as follows:

$$\frac{\partial L}{\partial \theta} = C \cdot \theta \sin\theta \tag{10}$$

in which $f(\theta) = \theta \sin\theta$ is a monotonically increasing function that satisfies *Cond. 2*. $C = g(h_1, h_2, w_1, w_2)$ is independent of $\theta$. Furthermore, we manually define $C = (U/h_1 h_2)$,

then the designed gradient is

$$\frac{\partial L}{\partial \theta} = C \cdot \theta \sin\theta = \frac{U}{h_1 h_2} \cdot \theta \sin\theta. \tag{11}$$

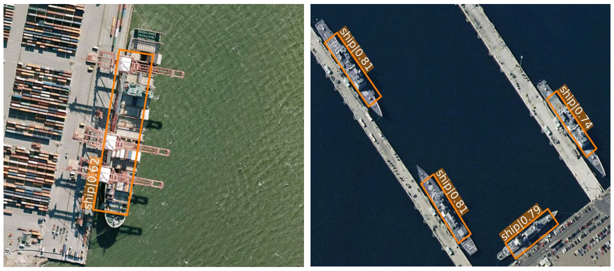


Fig. 3. Some failure detection cases for preliminary construction of GCL.

In this way, we would obtain the IoU-related item in the regression loss, which helps bridge the gap between regression loss and evaluation metric.

Next, we get the initial version of GCL through integration (denoted as $L'$)

$$\begin{aligned}L' &= \int \left(\frac{U}{h_1 h_2} \cdot \theta \sin\theta\right) d\theta\\ &= \frac{U}{h_1 h_2} \cdot \sin\theta \cdot \left(1 - \frac{\theta}{\tan\theta}\right)\\ &= \left(1 - \frac{\theta}{\tan\theta}\right) / \mathrm{IoU}\end{aligned} \tag{12}$$

where we denote $g(\theta) = 1 - (\theta/\tan\theta)$, and $g(\theta)$ is a monotonically increasing function when $\theta \in [\theta_0, (\pi/2)]$. When $\theta \downarrow$, we know $g(\theta) \downarrow$ and $IoU \uparrow$, and thus, $L' \downarrow$. Therefore, *Cond. 1* also holds, so (12) is a good candidate for GCL.

However, the model supervised by (12) focuses more on angle prediction rather than overall IoU improvement. Therefore, the angle converges well, but the scales and center points of OBBs are not well-optimized (see Fig. 3). To solve this problem, we further introduce a hyperparameter into GCL

$$L' = \left(1 - \frac{\theta}{\tan\theta}\right) / \mathrm{IoU}^{\alpha} \tag{13}$$

where $\alpha$ ($\alpha \geq 1$) is the hyperparameter for adjustment on IoU. Note that the modification of GCL would lead to changes in the gradient, so we need to check whether the improved GCL still satisfies *Cond. 1* and *Cond. 2*. Obviously, $L_{\mathrm{GCL}}$ in (13) is a monotonically increasing function with respect to $IoU$ and $\theta$, and thus *Cond. 1* holds. Then we can also prove that $(\partial L_{\mathrm{GCL}}/\partial \theta)$ is a monotonically increasing function with respect to $\theta$ when $\theta > 0$ (see supplementary material for detailed proof). Therefore, $L_{\mathrm{GCL}}$ in (13) also satisfies *Cond. 2*. Increasing $\alpha$ would make the regression loss more sensitive to IoU changes, which helps achieve more accurate detections. Despite that, a too large $\alpha$ may lead to nonconvergence of the network. We found through experiments that the optimal value of $\alpha$ is 2.

Equations (12) and (13) directly construct the monotonically increasing gradient of loss with respect to angular error. To further ensure the consistency between the regression loss and detection accuracy, we integrate the above loss term into the initial rotated IoU loss to correct the abnormal gradient for more stable training and better performance. Taking the nonlinear rotated IoU loss as the example, the final corrected loss is denoted as $L_{\text{GCL}}$, which is shown as follows:

$$L_{\text{GCL}} = \lambda \cdot \left(1 - \frac{\theta}{\tan\theta}\right)/\text{IoU}^{\alpha} - \ln(\text{IoU}) \tag{14}$$

where $\lambda$ is the hyperparameter to adjust the contribution of different parts ($\lambda = 0.1$ in our experiments). Compared with (13), GCL in (14) performs gradient calibration on the original rotated IoU loss, which has the following advantages: 1) the independent IoU loss helps alleviate the excessive attention of GCL to the angle optimization; 2) the calibrated gradients are properly boosted when the angular error is small, which avoid the slow convergence caused by small optimization step; and 3) the original rotated IoU loss provides stable convergence for cases where the premise in Fig. 2 does not hold.

From the above analysis, we suggest that GCL actually denotes a series of regression losses constructed from the perspective of gradient optimization. The above formulas provide some feasible approaches. More generally, we conclude the general form of GCL for calibrating the rotated IoU loss. We denote the general from of GCL as $L^*_{\text{GCL}}$. The calibrated gradient is as follows:

$$\frac{\partial L^*_{\text{GCL}}}{\partial\theta} = f(\theta) + \frac{\partial L_{\text{IoU}}}{\partial\theta} \tag{15}$$

where $f(\theta)$ denotes the compensated gradient and should be a monotonically increasing function. $(\partial L_{\text{IoU}}/\partial\theta)$ denotes the gradient of IoU-based loss with respect to angular error. There are many alternatives for $f(\theta)$, such as $(\theta\sin\theta)$, $(1-\cos\theta)$, $((\theta/\sin\theta)-1)$. Therefore, GCL is a very flexible method for optimizing oriented objects. For example, supposing the initial regression loss is nonlinear IoU loss, we choose $f(\theta) = \theta\sin\theta$ here, thereby we have

$$\frac{\partial L^*_{\text{GCL}}}{\partial\theta} = \theta\cdot\sin\theta + \frac{1}{\tan\theta}. \tag{16}$$

Next, another form of GCL is obtained as follows:

$$L^*_{\text{GCL}} = \int \frac{\partial L^*_{\text{GCL}}}{\partial\theta} d\theta = \sin\theta\cdot\left(1 - \frac{\theta}{\tan\theta}\right) - \ln(\text{IoU}). \tag{17}$$

Obviously, $(\partial L_{\text{GCL}}/\partial\theta) > 0$ when $\theta \in (0, (\pi/2)]$, thus $L_{\text{GCL}}$ is a monotonically increasing function with respect to $\theta$. In addition, $L_{\text{GCL}}$ is a monotonically decreasing function with respect to IoU. Therefore, *Cond. 1* holds, that is, lower regression loss means smaller angular error and larger IoU during regression. GCL in (17) is parameter-free, and thus, there is no need to manually fine-tune the hyperparameters. Moreover, the compensated gradient mitigates the rise of the gradients during the regression process and helps achieve stable angle optimization. Equation (17) also shows that GCL can consistently improve detection performance by applying additional supervision on angle prediction. Consequently, our method can be applied to other detectors without introducing extra inference overhead.

### B. Gradient Scaling Strategy

In this section, we will discuss the calibration of the gradient of the IoU loss with respect to box scale in detail. As shown above, we present the general form of the gradient of GCL [in (15)] and two specific forms of loss [see (14) and (17)]. However, the gradient compensation term is independent of box scale in (15)

$$\frac{\partial f(\theta)}{\partial h_2} = \frac{\partial f(\theta)}{\partial w_2} = 0. \tag{18}$$

The partial derivative of $L_{\text{GCL}}$ with respect to scale is the same as initial rotated IoU loss and still negatively correlated with box scale. Therefore, the loss in (17) still does not satisfy *Prop. 2* mentioned in Section III-C.

The gradient compensation term of GCL in (12) and (13) introduce information about box scale, but they still suffer from abnormal scale sensitivity. Taking (12), for example, the gradients of GCL with respect to box scale are as follows:

$$\begin{aligned}\frac{\partial L_{\text{GCL}}}{\partial w_2} &= \sin\theta\cdot\left(1 - \frac{\theta}{\tan\theta}\right)/h_1 \propto \frac{1}{s}\\ \frac{\partial L_{\text{GCL}}}{\partial h_2} &= \sin\theta\cdot\left(1 - \frac{\theta}{\tan\theta}\right)/h_2 \propto \frac{1}{s}.\end{aligned} \tag{19}$$

Note that the ideal gradient should be proportional to the size of the bounding box as suggested in *Prop. 2*. We need to ensure $C \propto s^2$ in (10). Since $C = (U/h_1h_2) \propto s^0$ in (16), the gradients of GCL with respect to box size still lead to inferior regression process of objects with different scales. However, if we introduce $C \propto s^2$ in (15) to calibrate the gradients, GCL is no longer scale-invariant in term of loss metric and thus does not satisfy *Prop. 1* any more.

As a result, we need to correct the loss gradient while maintaining the scale invariance of the loss value. To make GCL satisfy both *Prop. 1* and *Prop. 2*, we calibrate the GCL by scaling the gradient of the loss during training. The update rule of gradient descent is modified as follows:

$$\Delta s = -\eta\cdot\frac{\partial L_{\text{GCL}}}{\partial s}\cdot U \tag{20}$$

in which $s$ represents the size of the box ($h_2$ or $w_2$), and $U$ is the union area of prediction box and GT box. $\Delta s$ is the step size in the gradient descent algorithm, and $\eta$ is the learning rate. We know from (9) that $(\partial L_{\text{lnIoU}}/\partial s) \propto (1/s)$. Then, $(\partial L_{\text{GCL}}/\partial s) \propto (1/s)$ in GCL shown in (14) and (17). Since $U \propto s^2$, $(\partial L_{\text{GCL}}/\partial s)\cdot U \propto s$. In this way, the gradient of GCL with respect to box scale is positively related to the box size. Large objects update parameters with large steps during the regression process, while small objects take small ones. *Prop. 2* is also established in GCL.

The gradient scaling strategy endows GCL with the ability to adapt to object scale variations. At the same time, this strategy maintains the scale invariance of loss metric in GCL, that is, *Prop. 1* also holds. The gradient scaling of GCL slightly modifies the backpropagation strategy and can easily be applied to the existing oriented detectors.

Overall, gradient calibration is a flexible approach to design a family of GCL for fast and stable model convergence. The analysis and conclusions in this section can also be used to explore better regression losses in other models and tasks.

### C. Training

We use RetinaNet [27] as the baseline model and apply additional angle regression to predict OBBs. The classification subtask is supervised by the focal loss [27], while the

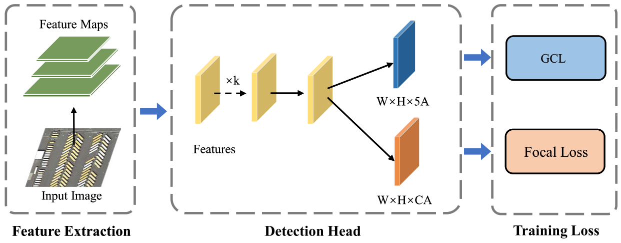


Fig. 4. Overall framework of the model. "$A$" denotes the number of anchors, and "$C$" is the number of classes.

regression loss is replaced by our proposed GCL. The overall frame is shown in Fig. 4. The total training loss is as follows:

$$L = \frac{1}{N}\sum_i \mathrm{FL}(p_i, p_i^*) + \frac{1}{N_p}\sum_i L_{\mathrm{GCL}}(\boldsymbol{b_i}, \boldsymbol{b_i^*}) \qquad (21)$$

in which $FL()$ denotes focal loss [27] denoted as follows:

$$\mathrm{FL}(p_i, p_i^*) = -(1-\hat{p}_i)^{\gamma} \log(\hat{p}_i). \qquad (22)$$

We have

$$\hat{p}_i = \begin{cases} p_i, & \text{if } p_i^* = 1 \\ 1 - p_i, & \text{otherwise.} \end{cases} \qquad (23)$$

In (23), $p_i^*$ and $\boldsymbol{b_i^*}$ are the GT labels for classification and regression, while $p_i$ and $\boldsymbol{b_i}$ are the corresponding predictions, respectively. In (21), $L_{\mathrm{GCL}}$ is the GCL for OBB regression. $N$ is the total number of training samples, and $N_p$ denotes the number of positive samples. In the early stage of training, for predictions whose center points are outside the assigned GT boxes, we will penalize the center distance to ensure a better convergence.

Our GCL is flexible and can be extended to the existing detectors to achieve further performance improvements. We also conduct extensive experiments on some state-of-the-art detectors such as S$^2$ANet [17] and ORCNN [41] to demonstrate the superiority of GCL.

## V. Experiments

### A. Dataset

We have conducted experiments on mainstream large-scale datasets for oriented object detection, including HRSC2016, UCAS-AOD, DOTA, UAV-ROD, DIOR-R, and FAIR1M.

HRSC2016 [42] is a high-resolution ship detection dataset with a total of 1061 images. The dataset is divided into training set, validation set, and test set, including 436, 181, and 444 images, respectively. UCAS-AOD [43] is collected for plane and car detection in aerial images. It contains 1510 images, including 1000 plane images and 510 car images. UAV-ROD [44] consists of 1577 aerial images and 30 090 instances for car detection. ICDAR 2015 [45] (IC15) contains 1500 images, which is used for incidental scene text detection.

DOTA [46] is a large-scale dataset for object detection in remote sensing images. DOTA-v1.0 includes 2806 large-scale aerial images with 188 282 annotated instances. It contains 15 categories, including plane (PL), baseball diamond (BD), bridge (BR), ground track field (GTF), small vehicle (SV), large vehicle (LV), ship (SH), tennis court (TC), basketball court (BC), storage tank (ST), soccer-ball field (SBF), roundabout (RA), harbor (HA), swimming pool (SP), and helicopter (HC). DOTA-v1.5 and DOTA-v1.0 share the same images, just with an extra category container crane (CC) and more annotations for small objects.

DIOR [3] is a large-scale benchmark for object detection in aerial images. It contains 23 463 images, covering 20 object classes, including airplane (APL), airport (APO), baseball field (BF), basketball court (BC), bridge (BR), chimney (CH), expressway service area (ESA), expressway toll station (ETS), dam (DAM), golf field (GF), ground track field (GTF), harbor (HA), overpass (OP), ship (SH), stadium (STA), storage tank (STO), tennis court (TC), train station (TS), vehicle (VE), and windmill (WM). DIOR-R shares the same images with the DIOR dataset. OBBs are annotated for objects in the dataset to adapt to the oriented detection task.

FAIR1M [47] is a recent large-scale dataset for fine-grained object recognition in high-resolution remote sensing imagery. It includes more than one million instances and more than 15 000 images. Objects in the FAIR1M dataset are annotated with respect to 37 categories.

### B. Implementation Details

For ablation experiments, we adopt ResNet-50 [48] as the backbone. All the images are scaled to 416 × 416 for HRSC2016, 800 × 800 for UCAS-AOD, UAV-ROD, DIOR-R, and FAIR1M, and 1024 × 1024 for DOTA-v1.0 and DOTA-v1.5. Random flips and rotation are adopted for data augmentation.

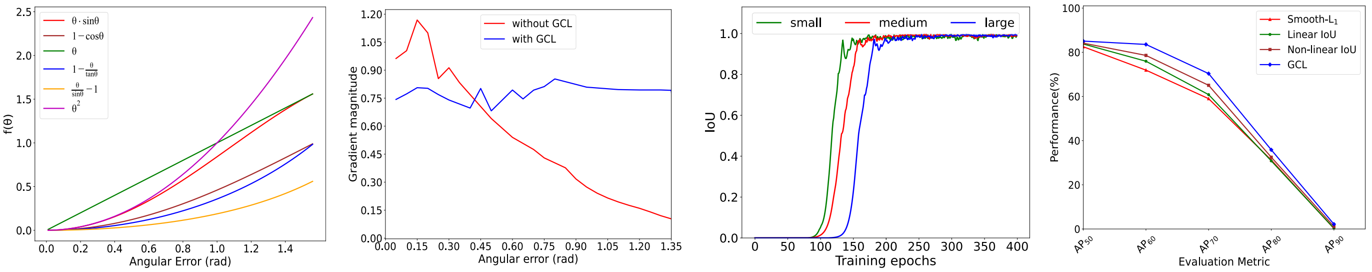
We used SGD optimizer to train the model with the batch size set to 8 on RTX 3090Ti GPUs. The learning rate is set to 0.01. The total iterations are 72 epochs for HRSC2016, 36 epochs for UCAS-AOD and UAV-ROD, and 12 epochs for DIOR-R, FAIR1M, DOTA-v1.0, and DOTA-v1.5. Note that images in DOTA and FAIR1M are too large, and we split images into 1024 × 1024 patches with an overlap of 500 pixels.

If not specially specified, all "GCL" in the tables refers to the GCL method with RetinaNet-R50 as the baseline. We also integrate GCL into the existing detectors for further performance improvements. These models are denoted as "X+GCL," where "X" represents other advanced detectors.

### C. Ablation Study

*1) Evaluation of Hyperparameters of GCL:* We conducted experiments to analyze the effect of different hyperparameters of GCL. The GCL in (14) contains two hyperparameters $\lambda$ and $\alpha$, while GCL in (17) is parameter-free. The experimental results are reported in Table I. AP$_{50:90}$ means AP over different IoU thresholds, from 0.5 to 0.9, step 0.1. AP$_{50:90}$ considers more higher IoU thresholds so it effectively measures high-precision detection performance.

The best performances are AP$_{50}$ of 85.41% and AP$_{50:90}$ of 50.96% in (14). The optimal settings is $\alpha = 2$ and $\lambda = 0.1$. $\alpha$ controls the IoU convergence in gradient compensation term, while $\lambda$ determines the contribution of the gradient compensation in regression loss. As $\alpha$ increases, the convergence of



(a) (b) (c) (d)

Fig. 5. Visualization of some properties and performance improvements of GCL. (a) Gradient variation in different compensation terms in GCL. (b) Comparison of gradient variation in the rotated IoU loss and GCL. (c) Convergence speed of objects at different scales with GCL. (d) Comparison of different losses in detecting long-strip ships.

TABLE I

ANALYSIS OF HYPERPARAMETERS OF GCL

| Gradient Compensation Term | $\alpha$ | $\lambda$ | AP$_{50}$ | AP$_{50:90}$ |
|---|---|---|---|---|
| $\lambda \cdot \left(1 - \frac{\theta}{\tan\theta}\right) / IoU^{\alpha}$ | 1 | 0.1 | 83.06 | 42.54 |
| | | 0.5 | **85.41** | 50.86 |
| | | 1.0 | 84.09 | 44.31 |
| | 2 | 0.1 | 85.11 | **50.96** |
| | | 0.5 | 84.16 | 40.65 |
| | 3 | 0.05 | 81.11 | 38.98 |
| | | 0.1 | 76.65 | 30.26 |
| $\sin\theta \cdot \left(1 - \frac{\theta}{\tan\theta}\right)$ | —— | —— | 85.23 | 50.71 |

TABLE II

ABLATION EXPERIMENTS ON GCL

| with GCL ? | $f(\theta)$ | with GS ? | AP$_{50}$ | AP$_{50:90}$ |
|---|---|---|---|---|
| × | —— | × | 83.13 | 48.76 |
| ✓ | $\frac{\theta}{\sin\theta} - 1$ | × | 83.38 | 49.12 |
| | $\theta$ | × | 83.82 | 48.61 |
| | $1 - \cos\theta$ | × | 84.46 | 49.57 |
| | $\theta^2$ | × | 84.71 | 50.38 |
| | $\theta \sin\theta$ | × | 85.23 | 50.71 |
| | | ✓ | **85.88** | **51.13** |

TABLE III

COMPARISON OF PERFORMANCE WITH DIFFERENT REGRESSION LOSSES FOR OBJECTS WITH LARGE ASPECT RATIOS

| Model | Reg. Loss | BR | SV | LV | SH | HA | mAP |
|---|---|---|---|---|---|---|---|
| RetinaNet [27] | IoU-SM [49] | **44.32** | 63.03 | 51.25 | 72.78 | 56.21 | 57.52 |
| | CSL [10] | 42.25 | 68.28 | 54.51 | 72.85 | 53.10 | 58.20 |
| | RIL [20] | 40.81 | 67.63 | 55.45 | 72.42 | 55.49 | 58.36 |
| | GWD [16] | 44.07 | 71.92 | 62.56 | 77.94 | 60.25 | 63.35 |
| | KLD [22] | 44.00 | 74.45 | **72.48** | 84.30 | **65.54** | 68.15 |
| | GCL | 44.27 | **78.83** | 70.84 | **85.40** | 62.73 | **68.41** |
| S$^2$ANet [17] | smooth-L$_1$ | 55.52 | 80.95 | 84.17 | 88.67 | 76.98 | 77.25 |
| | -$ln(IoU)$ | 53.86 | 79.13 | 83.71 | 88.42 | 75.55 | 76.13 |
| | GCL | **59.36** | **81.95** | **85.33** | **88.88** | **79.62** | **79.03** |

IoU is more emphasized, but it may lead to potential gradient divergence. Hence, a smaller $\lambda$ needs to be adopted to reduce its impact. Conversely, when $\alpha$ is small, it may cause the gradient compensation term to pay too much attention to the angle regression, and thus, the IoU is not well-optimized. Therefore, these two parameters need to be fine-tuned carefully to achieve a better tradeoff.

In contrast, gradient compensation term of (17) achieves AP$_{50}$ of 85.23% and AP$_{50:90}$ of 50.71%, which is slightly inferior to that of (14). However, there are no parameters in (17). It is more flexible and convenient, and we will adopt and explore this form of GCL in subsequent experiments. Certainly, we could introduce a $\lambda$ term in the first component of (17) to modulate the amplitude of gradient compensation. The experimental findings indicate that (17) exhibits a low sensitivity to $\lambda$. Consequently, our primary emphasis lies in the selection of the function $f(\theta)$.

*2) Evaluation of Different Components:* We have tried different gradient compensation terms for parameter-free GCL in (17), and the experimental results are reported in Table II. "GS" denotes the gradient scaling strategy. The baseline model is trained with nonlinear IoU loss, which achieves AP$_{50}$ of 83.13% and AP$_{50:90}$ of 48.76%. The gradient variations in these gradient compensation terms are shown in Fig. 5(a). Most variants with GCL achieve better performance than the baseline, which proves the effectiveness of the method. Among them, when $f(\theta) = \theta \sin\theta$ as in (16), the model achieves the best improvements of AP$_{50}$ of 2.1% and AP$_{50:90}$ of 1.95%.

Various gradients of loss with respect to the angular error have different effects on performance gains. If the compensated gradient grows too slowly with the increase in angular error, the gradient is not well-corrected, and the improvement is relatively small, such as the case of $f(\theta) = (\theta/\sin\theta) - 1$. Variant such as $f(\theta) = \theta$ linearly calibrates the gradients which is also not optimal. As a result, AP$_{50:90}$ of this variant even degrades by 0.15%.

We further visualized the gradient changes during training in Fig. 5(b). As the angular error decreases, the normal IoU loss (red curve) suffers a sharp increase in gradient, which will lead to slow angle convergence or even loss oscillation. Relatively, the model trained with GCL (blue curve) corrects the abnormal angular gradient variation. GCL helps achieve fast model convergence and smooth gradient descent, thus ensuring a stable regression process. Performance comparison in Fig. 5(d) shows that our method outperforms the initial IoU loss and smooth-$L_1$ loss in various metrics.

*3) Evaluation of Gradient Scaling Strategy:* The experimental results in Table II show that the gradient scaling

TABLE IV
PERFORMANCE IMPROVEMENTS OF GCL ON DIFFERENT DATASETS

| Model | Reg. Loss | HRSC2016 | FAIR1M | DIOR-R | DOTA-v1.0 | DOTA-v1.5 | UCAS-AOD | UAV-ROD |
|---|---|---|---|---|---|---|---|---|
| RetinaNet [27] | smooth-L$_1$ | 83.32 | 27.67 | 54.21 | 69.86 | 68.31 | 81.05 | **97.67** |
| | $1 - IoU$ | 84.34 | 25.96 | 56.13 | 70.75 | 68.93 | 85.81 | 97.59 |
| | -$ln(IoU)$ | 85.09 | 28.55 | 57.91 | 71.23 | 69.32 | 87.60 | 97.48 |
| | GCL | **86.15** +1.06 | **29.89** +1.34 | **59.89** +1.98 | **72.69** +1.46 | **70.24** +0.92 | **88.54** +0.94 | 97.56 +0.08 |

TABLE V
PERFORMANCE IMPROVEMENTS OF GCL COMBINED WITH STATE-OF-THE-ART MODELS

| Method | Metric | Reg. Loss | HRSC2016 | | DOTA-v1.0 | | DOTA-v1.5 | | DIOR-R | | FAIR1M | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| S$^2$ANet [17] | AP$_{50}$ | -ln(IoU) | 88.68 | +1.07 | 66.17 | +1.72 | 61.06 | +1.12 | 60.51 | +1.20 | 36.12 | +1.45 |
| | | GCL | **89.75** | | **67.89** | | **62.18** | | **61.31** | | **37.57** | |
| | AP$_{75}$ | -ln(IoU) | 54.96 | +4.18 | 37.12 | +3.18 | 31.57 | +2.32 | 35.85 | +1.44 | —— | |
| | | GCL | **59.14** | | **40.30** | | **33.89** | | **37.29** | | | |
| ORCNN [41] | AP$_{50}$ | -ln(IoU) | 88.97 | +0.61 | 67.13 | +0.98 | 64.92 | +0.56 | 59.16 | +0.52 | 39.83 | +0.52 |
| | | GCL | **89.58** | | **68.11** | | **65.48** | | **59.68** | | **40.35** | |
| | AP$_{75}$ | -ln(IoU) | 59.78 | +7.17 | 37.11 | +1.83 | 33.21 | +1.33 | 34.31 | +0.79 | —— | |
| | | GCL | **66.95** | | **38.94** | | **34.54** | | **35.10** | | | |
| RoI Transformer [29] | AP$_{50}$ | -ln(IoU) | 89.18 | +0.51 | 71.72 | +0.80 | 68.20 | +0.51 | 59.27 | +0.76 | 38.27 | +2.37 |
| | | GCL | **89.69** | | **72.52** | | **68.71** | | **60.03** | | **40.64** | |
| | AP$_{75}$ | -ln(IoU) | 68.95 | +2.25 | 36.07 | +1.92 | 35.11 | +1.25 | 33.76 | +1.15 | —— | |
| | | GCL | **71.20** | | **37.99** | | **36.36** | | **34.91** | | | |

strategy can further improve the performance on the basis of GCL. It improves AP$_{50}$ by 0.65 points and AP$_{50:90}$ by 0.42 points. We visualize the loss gradient with respect to box size during training in Fig. 5(c). The gradient scaling strategy effectively corrects the abnormal gradient of initial rotated IoU loss and helps achieve consistent and fast convergence for objects at different scales. These improvements benefit from in-depth analysis and design of the loss gradient.

*4) Comparison of Different Regression Losses:* We have mentioned in Section III-B that rotated objects with large aspect ratios are more susceptible to abnormal loss gradient with respect to angular error. Therefore, we conducted experiments on the DOTA dataset and selected the performance of classes with large aspect ratios, including BR, SV, LV, SH, and HA. The experimental results are shown in Table III. S$^2$ANet [17] is a recently advanced rotation detector for object detection, which uses smooth-L$_1$ loss for OBB regression. As shown in the last three rows of Table III, when the regression loss is replaced with nonlinear IoU loss, the performance on objects with large aspect ratios drops by 1.12% instead. We suggest that the gradient of IoU loss with respect to angle increases abnormally during optimization, which hinders the accurate regression of narrow objects. When GCL is applied to S$^2$ANet, the gradients are corrected and angle regression is more accurate, and therefore, a substantial performance improvement of 2.9% is achieved.

In addition, we compare the proposed method with some recent advanced regression losses for oriented object detection as shown in Table III. Our method outperforms other well-designed losses with a simple and intuitive gradient correction method. The above experimental results further demonstrate the effectiveness of our method.

We also report the experimental results on multiple datasets to further prove the superiority of GCL. As shown in Table IV, since IoU-based losses achieve consistent training and evaluation, these losses achieve better performance compared with smooth-L$_1$ loss on most datasets. Moreover, GCL helps learn more accurate orientations and to consistently regress objects at different scales. Therefore, our method achieves the best performance gains on most datasets in the last row of Table IV. Since the cars in the UAV-ROD dataset are captured at a fixed height view with little scale variation, smooth-L$_1$ loss converges faster and performs better than IoU losses and our method.

Compared with the traditional rotated IoU losses, GCL achieves better performance in the above experiments. On one hand, the traditional rotated IoU losses suffer from abnormal gradient changes during angle regression, which leads to slow convergence and inaccurate prediction. Especially for objects with large aspect ratios (such as ships and bridges), slight angle deviations would cause a sharp decrease in IoU. GCL corrects the abnormal angle gradient and endows a smoother angle regression process. As a result, GCL speeds up the model convergence and improves the accuracy. On the other hand, we suggest that due to the abnormal gradients, the traditional IoU losses have different convergence speeds for objects of different scales. It is hard to balance the optimization process of these objects by directly setting an appropriate learning rate. GCL introduces a gradient scaling strategy, which retains the scale-independent advantages of the IoU loss and solves the abnormal gradient changes (see Fig. 6).

Note that GCL does not modify the inference process of the model, and there is no additional inference overhead introduced. Therefore, GCL can be applied to current detectors without introducing extra computational cost.

### *D. Incorporations With State-of-the-Arts*

To demonstrate the generality of the proposed method, we apply GCL to some advanced detectors. The experimental results are shown in Table V. All the models are trained on

TABLE VI

COMPARISON WITH STATE-OF-THE-ARTS ON THE DOTA-V1.0 DATASET. THE ITEMS WITH RED AND BLUE COLORS INDICATE THE BEST AND SECOND-BEST RESULTS OF EACH COLUMN, RESPECTIVELY. "MS" DENOTES MULTISCALE TRAINING AND TESTING

| | Methods | Backbone | Ms | PL | BD | BR | GTF | SV | LV | SH | TC | BC | ST | SBF | RA | HA | SP | HC | mAP |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Two Stage Detectors | FR-O [46] | R-101 | | 79.09 | 69.12 | 17.17 | 63.49 | 34.20 | 37.16 | 36.20 | 89.19 | 69.60 | 58.96 | 49.40 | 52.52 | 46.69 | 44.80 | 46.30 | 52.93 |
| | ICN [50] | R-101 | ✓ | 81.40 | 74.30 | 47.70 | 70.30 | 64.90 | 67.80 | 70.00 | 90.80 | 79.10 | 78.20 | 53.60 | 62.90 | 67.00 | 64.20 | 50.20 | 68.20 |
| | CAD-Net [51] | R-101 | ✓ | 87.80 | 82.40 | 49.40 | 73.50 | 71.10 | 63.50 | 76.70 | 90.90 | 79.20 | 73.30 | 48.40 | 60.90 | 62.00 | 67.00 | 62.20 | 69.90 |
| | SCRDet [49] | R-101 | ✓ | 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 |
| | Gliding Vertex [52] | R-101 | | 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 |
| | Mask OBB [53] | RX-101 | ✓ | 89.56 | **85.95** | 54.21 | 72.90 | 76.52 | 74.16 | 85.63 | 89.85 | 83.81 | 86.48 | 54.89 | 69.64 | 73.94 | 69.06 | 63.32 | 75.33 |
| | CenterMap-Net [54] | R-101 | ✓ | 89.83 | 84.41 | 54.60 | 70.25 | 77.66 | 78.32 | 87.19 | 90.66 | 84.89 | 85.27 | 56.46 | 69.23 | 74.13 | 71.56 | 66.06 | 76.03 |
| | CSL [10] | R-152 | ✓ | **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 [55] | R-101 | ✓ | 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 |
| | RoI Trans. [29] | R-50 | ✓ | 87.89 | 85.01 | 57.83 | 78.55 | 75.22 | 84.37 | 88.04 | 90.88 | 87.28 | 85.79 | 71.04 | 69.67 | 79.00 | **83.29** | 73.43 | 79.82 |
| | ReDet [35] | ReR-50 | ✓ | 88.81 | 82.48 | 60.83 | 80.82 | 78.34 | **86.06** | 88.31 | 90.87 | **88.77** | 87.03 | 68.65 | 66.90 | 79.26 | 79.71 | 74.67 | 80.10 |
| | ORCNN [41] | R-50 | ✓ | 89.84 | 85.43 | **61.09** | 79.82 | 79.71 | 85.35 | 88.82 | 90.88 | 86.68 | **87.73** | 72.21 | 70.80 | 82.42 | 78.18 | 74.11 | 80.87 |
| | RoI Trans. + GCL | Swin-T | ✓ | 88.99 | 85.51 | **62.52** | **82.45** | 79.89 | **85.93** | 88.31 | 90.84 | 86.13 | 87.39 | **73.09** | 69.86 | **84.23** | 81.54 | **78.63** | **81.69** |
| One Stage Detectors | IENet [56] | R-101 | ✓ | 80.20 | 64.54 | 39.82 | 32.07 | 49.71 | 65.01 | 52.58 | 81.45 | 44.66 | 78.51 | 46.54 | 56.73 | 64.40 | 64.24 | 36.75 | 57.14 |
| | PIoU [30] | DLA-34 | | 80.90 | 69.70 | 24.10 | 60.20 | 38.30 | 64.40 | 64.80 | 90.90 | 77.20 | 70.40 | 46.50 | 37.10 | 57.10 | 61.90 | 64.00 | 60.50 |
| | CFC-Net [9] | R-101 | ✓ | 89.08 | 80.41 | 52.41 | 70.02 | 76.28 | 78.11 | 87.21 | 90.89 | 84.47 | 85.64 | 60.51 | 61.52 | 67.82 | 68.02 | 50.09 | 73.50 |
| | $R^3$Det [34] | R-152 | ✓ | 89.80 | 83.77 | 48.11 | 66.77 | 78.76 | 83.27 | 87.84 | 90.82 | 85.38 | 85.51 | 65.67 | 62.68 | 67.53 | 78.56 | 72.62 | 76.47 |
| | FANet [57] | R-50 | ✓ | **91.24** | 84.42 | 56.53 | 68.12 | 79.82 | 83.48 | 88.91 | 90.56 | 85.72 | 86.35 | 61.96 | 64.85 | 74.08 | 69.14 | 63.35 | 76.57 |
| | DAL [18] | R-50 | | 89.69 | 83.11 | 55.03 | 71.00 | 78.30 | 81.90 | 88.46 | 90.89 | 84.97 | 87.46 | 64.41 | 65.65 | 76.86 | 72.09 | 64.35 | 76.95 |
| | SLA [19] | R-50 | ✓ | 88.33 | 84.67 | 48.78 | 73.34 | 77.47 | 77.82 | 86.53 | 90.72 | 86.98 | 86.43 | 58.86 | 68.27 | 74.10 | 73.09 | 69.30 | 76.36 |
| | DCL [58] | R-152 | ✓ | 89.26 | 83.60 | 53.54 | 72.76 | 79.04 | 82.56 | 87.31 | 90.67 | 86.59 | 86.98 | 67.49 | 66.88 | 73.29 | 70.56 | 69.99 | 77.37 |
| | GWD [16] | R-152 | ✓ | 89.06 | 84.32 | 55.33 | 77.53 | 76.95 | 70.28 | 83.95 | 89.75 | 84.51 | 86.06 | **73.47** | 67.77 | 72.60 | 75.76 | 74.17 | 77.43 |
| | RIDet [20] | R-50 | ✓ | 89.31 | 80.77 | 54.07 | 76.38 | 79.81 | 81.99 | **89.13** | 90.72 | 83.58 | 87.22 | 64.42 | 67.56 | 78.08 | 79.17 | 62.07 | 77.62 |
| | RDD [33] | R-101 | ✓ | 89.15 | 83.92 | 52.51 | 73.06 | 77.81 | 79.00 | 87.08 | 90.62 | 86.72 | 87.15 | 63.96 | 70.29 | 76.98 | 75.79 | 72.15 | 77.75 |
| | $AF^2$Det [59] | Res2Net | ✓✓ | 89.71 | 84.08 | 53.84 | 74.76 | 79.05 | 85.18 | 88.43 | 90.82 | **88.71** | 87.10 | 67.74 | 67.30 | 76.86 | 71.93 | 71.98 | 78.30 |
| | KLD [22] | R-50 | ✓ | 88.91 | 85.23 | 53.64 | **81.23** | 78.20 | 76.99 | 84.58 | 89.50 | 86.84 | 86.38 | 71.69 | 68.06 | 75.95 | 72.23 | 75.42 | 78.32 |
| | AQE-Det [60] | R-101 | ✓ | 89.49 | **85.89** | 55.46 | 77.32 | 74.17 | 80.33 | 87.65 | 90.82 | 86.99 | 86.52 | 66.58 | 69.06 | 77.12 | 78.35 | 69.51 | 78.35 |
| | TIOE-Det [21] | R-50 | ✓ | 89.76 | 85.23 | 56.32 | 76.17 | 80.17 | 85.58 | 88.41 | 90.81 | 85.93 | 87.27 | 68.32 | 70.32 | 68.93 | 78.33 | 68.87 | 78.69 |
| | $S^2$ANet [17] | R-50 | ✓ | 89.28 | 84.11 | 56.95 | 79.21 | 80.18 | 82.93 | **89.21** | 90.86 | 84.66 | 87.61 | 71.66 | 68.23 | 78.58 | 78.20 | 65.55 | 79.15 |
| | GCL | R-50 | ✓ | 89.06 | 83.21 | 43.34 | 76.35 | 79.06 | 77.47 | 83.11 | 90.14 | 84.08 | 85.12 | 64.21 | 65.52 | 66.17 | 75.39 | 58.69 | 74.73 |
| | GCL | Swin-T | | 89.53 | 84.33 | 43.06 | 76.18 | 80.97 | 78.21 | 86.66 | **90.91** | 87.17 | 85.03 | 64.05 | 65.85 | 66.99 | 77.20 | 59.02 | 75.68 |
| | GCL | Swin-T | ✓ | 89.65 | 85.13 | 43.25 | 77.41 | **81.25** | 77.93 | 86.69 | **90.90** | 86.93 | 84.49 | 64.13 | 65.77 | 68.14 | 78.51 | 61.31 | 76.10 |
| | $S^2$ANet + GCL | Swin-T | | 89.02 | 84.72 | 57.21 | 74.72 | 80.68 | 84.11 | 88.68 | 90.87 | 87.36 | 87.20 | 65.60 | 70.08 | 76.91 | 78.55 | 65.35 | 78.73 |
| | $S^2$ANet + GCL | Swin-T | ✓ | 89.40 | 85.19 | 60.39 | 79.88 | **81.37** | 85.80 | 88.99 | 90.85 | 85.73 | **88.02** | 70.38 | **73.52** | **83.56** | **82.49** | **75.63** | **81.41** |

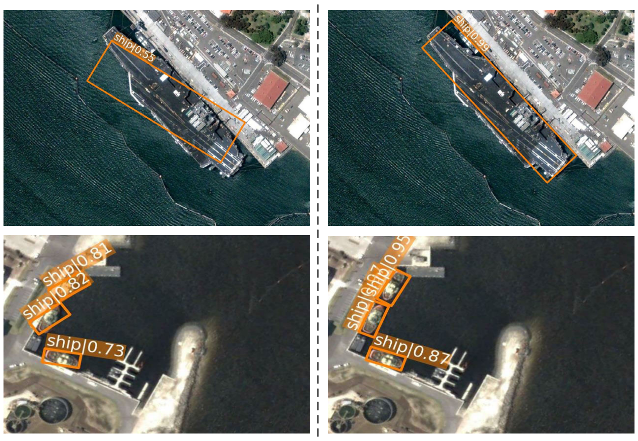


Fig. 6. Visualization of detection results of extremely large-scale objects and small objects with RIoU loss and GCL supervision, respectively.

TABLE VII

COMPARISON OF DIFFERENT TRAINING SETTINGS ON DOTA-V1.0

| Baselline | Backbone | + GCL | Ms | mAP |
|---|---|---|---|---|
| RetinaNet [27] | R-50 | | ✓ | 72.25 |
| | | ✓ | ✓ | 74.73 |
| | Swin-T | | | 74.64 |
| | | ✓ | | 75.68 |
| | | ✓ | ✓ | **76.10** |
| $S^2$ANet [17] | R-50 | | ✓ | 79.15 |
| | Swin-T | | | 78.01 |
| | | ✓ | | 78.73 |
| | | ✓ | ✓ | **81.41** |

the training set and testing and evaluated on the validation set. Multiscale training and testing are only used on the FAIR1M dataset. Note that the FAIR1M dataset just releases the training set and test set, so only $AP_{50}$ can be evaluated by submitting the detections on the test set to its official server.

$S^2$ANet [17], ORCNN [41], and RoI Transformer [29] are the recent state-of-the-art frameworks for oriented object detection. First, we replaced the smooth-$L_1$ loss of these detectors with the initial nonlinear IoU loss for regression. Generally, IoU loss has been shown to achieve better performance [17]. Next, we applied GCL to these models and obtained performance gains as shown in Table V. Specifically, HRSC2016 [42] contains a large number of ships with large aspect ratios. GCL accelerates the angle optimization and therefore achieves substantial $AP_{75}$ improvements. Besides, the scales of objects in DOTA, DIOR-R, and FAIR1M are highly variable. The gradient scaling strategy of GCL helps consistently optimize objects of different scales and achieve fast convergence. Extensive performance gains are achieved on multiple benchmarks with the proposed GCL, which proves its superiority.

### E. Comparison With State-of-the-Arts

*1) Results on DOTA:* DOTA is currently the most widely used large-scale dataset for oriented object detection in aerial

TABLE VIII

COMPARISON WITH STATE-OF-THE-ARTS ON THE DOTA-V1.5 DATASET

| Methods | Backbone | Ms | PL | BD | BR | GTF | SV | LV | SH | TC | BC | ST | SBF | RA | HA | SP | HC | CC | mAP |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| RetinaNet [27] | R-50 | | 71.43 | 77.64 | 42.12 | 64.65 | 44.53 | 56.79 | 73.31 | 90.84 | 76.02 | 59.96 | 46.95 | 69.24 | 59.65 | 64.52 | 48.06 | 0.83 | 59.16 |
| FR-O | R-50 | | 71.89 | 74.47 | 44.45 | 59.87 | 51.28 | 68.98 | 79.37 | 90.78 | 77.38 | 67.50 | 47.75 | 69.72 | 61.22 | 65.28 | 60.47 | 1.54 | 62.00 |
| DAL | R-50 | | 73.25 | 78.95 | 41.13 | 65.51 | 51.05 | 72.54 | 81.12 | 90.86 | 78.51 | 68.91 | 47.95 | 71.42 | 59.15 | 71.21 | 45.32 | 0.42 | 62.33 |
| Mask R-CNN [61] | R-50 | | 76.84 | 73.51 | 49.90 | 57.80 | 51.31 | 71.34 | 79.75 | 90.46 | 74.21 | 66.07 | 46.21 | 70.61 | 63.07 | 64.46 | 57.81 | 9.42 | 62.67 |
| HTC | R-50 | | 77.80 | 73.67 | 51.40 | 63.99 | 51.54 | 73.31 | 80.31 | 90.48 | 75.12 | 67.34 | 48.51 | 70.63 | 64.84 | 64.48 | 55.87 | 5.15 | 63.40 |
| CFC-Net [9] | R-50 | | 79.15 | 79.54 | 45.19 | 61.15 | 52.26 | 77.33 | 85.92 | **90.89** | 76.69 | 66.18 | 49.77 | 68.65 | 61.30 | 70.78 | 51.45 | 2.78 | 63.69 |
| RoI Trans. [29] | R-50 | | 71.92 | 76.07 | 51.87 | 69.24 | 52.05 | 75.18 | 80.72 | 90.53 | 78.58 | 68.26 | 49.18 | 71.74 | 67.51 | 65.53 | 62.16 | 9.99 | 65.03 |
| ReDet [35] | ReR-50 | | 79.20 | 82.81 | 51.92 | 71.41 | 52.38 | 75.73 | 80.92 | 90.83 | 75.81 | 68.64 | 49.29 | 72.03 | 73.36 | 70.55 | 63.33 | 11.53 | 66.86 |
| $S^2$ANet [17] | Swin-T | | 81.00 | 84.97 | 54.79 | 73.29 | 64.53 | 79.84 | 88.80 | 90.87 | 83.87 | 77.68 | 60.70 | 76.02 | 75.07 | 73.93 | 65.32 | 23.47 | 72.13 |
| ReDet [35] | ReR-50 | ✓ | **88.51** | **86.45** | **61.23** | **81.20** | 67.60 | **83.65** | **90.00** | 90.86 | 84.30 | 75.33 | **71.49** | 72.06 | 78.32 | 74.73 | 76.10 | 46.98 | 76.80 |
| GCL | R-50 | ✓ | 85.88 | 85.40 | 45.49 | 79.57 | 58.53 | 73.14 | 86.52 | 90.87 | 81.91 | 79.03 | 67.55 | 76.16 | 64.70 | 75.12 | 71.58 | 2.42 | 70.24 |
| $S^2$ANet + GCL | Swin-T | ✓ | 87.28 | 85.00 | 59.42 | 80.88 | **75.90** | 83.31 | 89.85 | 90.78 | **84.90** | **85.67** | 69.59 | **76.64** | **82.08** | **78.14** | **79.89** | **49.51** | **78.68** |

TABLE IX

COMPARISON WITH STATE-OF-THE-ARTS ON THE DIOR-R DATASET. * DENOTES A LONGER TRAINING SCHEDULE

| | Methods | APL | APO | BF | BC | BR | CH | ESA | ETS | DAM | GF | GTF | HA | OP | SH | STA | STO | TC | TS | VE | WM | mAP |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| One Stage Detectors | RetinaNet [27] | 61.49 | 28.52 | 73.57 | 81.17 | 23.98 | 72.54 | 58.20 | **72.39** | 19.94 | 69.25 | 79.54 | 32.14 | 44.87 | 77.71 | 67.57 | 61.09 | 81.46 | 47.33 | 38.01 | 60.24 | 57.55 |
| | IoU loss [27] | 62.73 | 22.62 | **75.96** | 81.40 | 24.30 | 72.68 | 75.70 | 59.11 | 21.63 | 77.02 | 79.34 | 37.33 | 38.79 | 69.96 | 72.53 | 59.06 | 81.46 | 46.57 | 37.54 | 62.54 | 57.91 |
| | SLA [19] | 67.85 | 21.40 | 74.24 | 80.75 | 31.26 | 72.33 | 61.83 | 56.06 | 17.47 | 65.99 | 76.55 | 38.02 | 46.59 | 80.63 | 69.94 | 67.61 | 81.25 | 46.50 | 46.50 | 55.57 | 57.92 |
| | DAL [18] | 62.70 | 25.42 | 71.77 | 80.92 | 34.88 | 72.63 | 69.07 | 60.52 | 22.15 | 68.23 | 76.71 | 39.81 | 48.66 | 80.91 | 72.83 | 62.19 | 81.27 | 48.67 | 42.60 | 62.77 | 59.24 |
| | RIDet [20] | 62.90 | 32.43 | **77.58** | 81.09 | 37.27 | 72.58 | 76.17 | 64.95 | 24.42 | 55.22 | 81.12 | 43.61 | 50.88 | 81.05 | 73.16 | 60.45 | 81.49 | 49.02 | 43.35 | 62.48 | 60.56 |
| | CFC-Net [9] | 64.94 | 33.43 | 75.16 | 81.25 | 36.14 | 71.75 | 70.13 | 63.57 | 18.01 | 68.15 | 80.82 | 41.58 | 52.30 | 80.95 | 68.72 | 69.61 | **83.73** | 47.06 | 47.91 | 57.86 | 60.65 |
| | S2ANet* [17] | 66.46 | 43.75 | 71.54 | 81.44 | 43.14 | 73.98 | 78.74 | 68.81 | 27.22 | 74.96 | 81.47 | 46.07 | 57.30 | **87.79** | 67.12 | 70.40 | **86.53** | 55.32 | **53.60** | 63.87 | 64.98 |
| | GCL | **68.87** | 26.04 | 75.29 | 81.49 | 27.50 | 72.71 | 75.91 | 59.09 | 22.14 | 76.78 | 78.67 | 39.18 | 41.61 | 79.44 | 72.47 | 64.41 | 81.46 | 47.02 | 44.52 | 63.18 | 59.89 |
| | S2ANet*+GCL | **69.70** | 45.53 | 75.93 | 81.42 | 42.85 | **75.04** | 79.36 | 68.90 | 29.20 | 76.33 | 80.77 | 45.91 | 57.55 | **86.69** | 69.76 | 70.62 | 81.57 | 53.26 | **53.04** | 65.25 | **65.43** |
| Two Stage Detectors | Faster RCNN [7] | 62.79 | 26.80 | 71.72 | 80.91 | 34.20 | 72.57 | 65.75 | 66.45 | 18.95 | 66.63 | 79.24 | 34.95 | 48.79 | 81.14 | 64.34 | **71.21** | 81.44 | 47.31 | 50.46 | 65.21 | 59.54 |
| | Gliding Vertex [52] | 65.35 | 28.87 | 74.96 | 81.33 | 33.88 | 74.31 | 64.70 | 70.72 | 19.58 | 72.30 | 78.68 | 37.22 | 49.64 | 80.22 | 69.26 | 61.13 | 81.49 | 44.76 | 47.71 | 65.04 | 60.06 |
| | RoI Trans. [29] | 63.34 | 37.88 | 71.78 | **87.53** | 40.68 | 72.60 | 68.09 | **78.71** | 26.86 | 68.96 | 82.74 | **47.71** | 55.61 | 81.21 | **78.23** | 70.26 | 81.61 | 54.86 | 43.27 | 65.52 | 63.87 |
| | RoI Trans* [29] | 62.96 | **45.98** | 71.90 | 81.22 | **45.04** | 72.68 | **79.87** | 70.12 | 30.18 | **77.45** | 81.72 | 48.53 | **58.08** | 81.19 | **75.44** | 62.69 | 81.39 | 55.37 | 48.65 | **66.46** | 64.85 |
| | ORCNN* [41] | 62.92 | 42.07 | 71.80 | 81.22 | 42.84 | 72.69 | 79.31 | 67.55 | 27.89 | 77.38 | 77.04 | 47.37 | 56.48 | 81.13 | 69.51 | 62.11 | 81.54 | 54.70 | 43.84 | 66.31 | 63.28 |
| | AOPG [62] | 62.39 | 37.79 | 71.62 | **87.63** | 40.90 | 72.47 | 77.99 | 65.42 | **31.08** | 73.20 | 81.94 | 42.32 | 54.45 | 81.17 | 72.69 | **71.31** | 81.49 | **60.04** | 52.38 | **69.99** | 64.41 |
| | ORCNN*+GCL | 62.45 | 41.57 | 71.76 | 81.11 | 43.81 | 72.65 | 79.52 | 67.84 | 29.14 | 75.16 | **83.22** | 44.79 | 57.59 | 81.13 | 74.73 | 62.19 | 81.47 | 54.84 | 43.35 | 65.56 | 63.70 |
| | RoI Trans.*+GCL | 63.08 | **48.87** | 72.03 | 81.33 | **45.38** | **80.01** | **80.42** | 70.38 | **34.16** | **78.41** | **83.12** | 47.13 | **59.42** | 81.23 | 73.40 | 62.47 | 81.45 | **56.77** | 43.85 | 66.16 | **65.45** |

images. The experimental results on DOTA are shown in Tables VI–VIII.

We compare the proposed method with some state-of-the-art methods on DOTA-v1.0 OBB task, and the results are reported in Table VI. GCL adopts the naive RetinaNet [27] as the baseline, and the backbone is replaced with a tiny Swin Transformer [63]. GCL achieves superior performance of 76.10%. RoI transformer [29] is an advanced rotation detector. Our method also helps further improve accuracy of RoI transformer, achieving a state-of-the-art performance of 81.69%.

Extensive experiments in Table V prove the generality of GCL with pure CNN-based backbone network. We also demonstrate the generality of GCL across different backbone networks. Swin Transformer [63] is a recent architecture based on multihead attention mechanism. It achieves state-of-the-arts results on multiple vision tasks. We introduce it into oriented object detection as the backbone network, and the results are shown in Table VII. The naive RetinaNet [27] reaches an mAP of 72.25% on the DOTA-v1.0 with ResNet-50 [48] as backbone network. However, even the single-scale model of the Swin Transformer achieves a higher performance of 74.64%. Next, GCL further improves mAP by 1.04% on variant with Swin Transformer. Ablation experiments on $S^2$ANet also yield similar results.

The experimental results on DOTA-v1.5 are shown in Table VIII. The annotations of DOTA-v1.5 contain many extremely small instances which are difficult to detect. Our GCL help avoid the loss oscillations for small objects, ensuring a stable and fast bounding box regression. Finally, GCL achieves 78.68% when applied to $S^2$ANet, which is the best among the compared methods.

Some visualizations of detections on DOTA-v1.0 are shown in Fig. 7. There are three types of objects that are difficult to detect in oriented object detection: 1) objects with large aspect ratios, such as harbors and ships in the first column; 2) objects under extreme scale variation, such as large-scale roundabout and small vehicles in the second column; and 3) missed detections in densely arranged scenarios, such as vehicles and ships in the third column. Obviously, our proposed GCL can handle the above scenarios well and achieve accurate detections in complex environments.

*2) Results on DIOR-R:* The results on DIOR-R are reported in Table IX. All the models use ResNet-50 as the backbone network. GCL achieves the mAP of 59.89%, which outperforms many recently proposed one-stage detectors such as SLA [19] and DAL [18]. Since IoU-based loss converges slower compared with smooth-$L_1$ loss, we train 36 epochs on some models to achieve better results. Finally, GCL improves $S^2$ANet by 0.45%–65.43%, which is the best performance among the compared one-stage detectors. Meanwhile, GCL

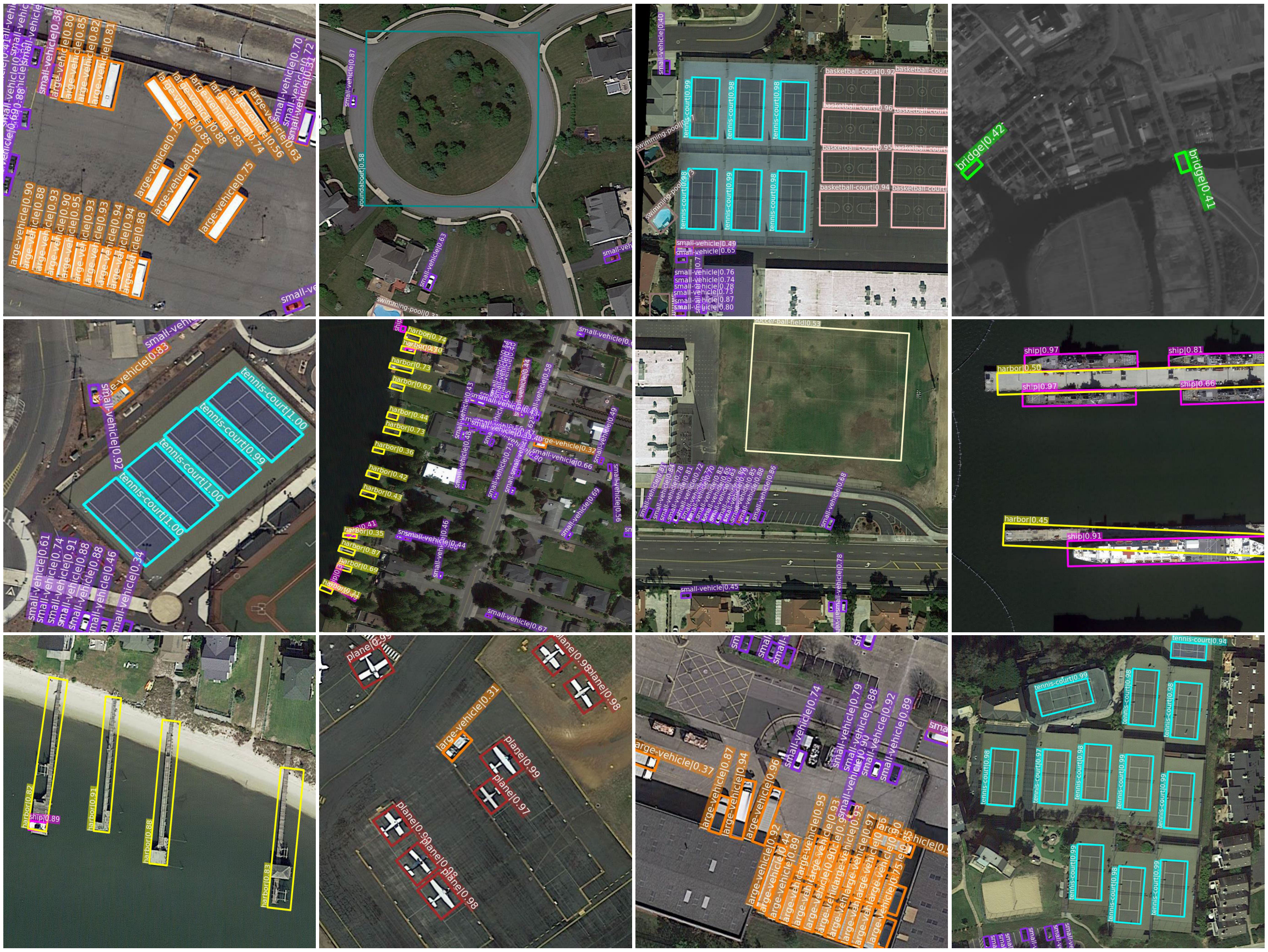


Fig. 7. Visualizations of detection results on DOTA-v1.0 obtained with our method. Bounding boxes of different colors represent objects belonging to different categories. The visualization results demonstrate that our method accurately detects objects in challenging remote sensing scenarios. This includes objects with large aspect ratios or significant scale variations, as well as small targets in dense scenes.

TABLE X

COMPARISON WITH STATE-OF-THE-ARTS ON THE HRSC2016 DATASET

| Methods | RRPN [31] | RetinaNet [27] | RRD [36] | RoI Trans. [29] | Gliding Vertex [52] | OPLD [55] |
|---|---|---|---|---|---|---|
| mAP | 79.08 | 80.81 | 84.3 | 86.2 | 88.2 | 88.44 |
| Methods | DAL [18] | RIDet [20] | CFC-Net [9] | GWD [16] | $S^2$ANet [17] | GCL (ours) |
| mAP | 88.95 | 89.63 | 89.7 | 89.85 | 90.17 | **90.19** |

improves RoI Transformer by 0.6%–65.45%, which is the best among the listed two stage methods.

*3) Results on HRSC2016:* We compare GCL with other methods on HRSC2016 in Table X. For long and narrow ships in HRSC216, accurate orientation prediction is especially important. GCL eliminates the abnormal variation in angular gradient in IoU loss and therefore achieves accurate ship detection. Our method achieves the best performance among the compared methods, achieving the mAP of 90.19%.

We adopt the same training schedule and compare the detection results of IoU loss and GCL. As shown in Fig. 6, the large bounding box converges not well under the supervision of IoU loss, and the localization of small ships is not accurate enough. In contrast, GCL accelerates the regression of large objects for better performance. In addition, the angle prediction no longer suffers from gradient oscillation, and therefore, small objects are also detected well.

*4) Results on FAIR1M:* FAIR1M is the largest dataset for fine-grained oriented object detection in remote sensing images. It contains a total of 37 categories, and the official server provides performance evaluations for 34 categories among them. Since the benchmark is recently released, there are not many experimental results. For a fair comparison, we have trained and evaluated many state-of-the-art models on FAIR1M with our unified codebase. All the models use ResNet-50 as the backbone network. The experimental results are shown in Table XI.

The single-scale GCL achieves the mAP of 29.89%, which is 2.22% higher than the baseline model. Meanwhile, GCL improves the performance of $S^2$ANet and RoI Transformer by 2.37% and 1.45%, respectively. With multiscale training and testing, the variant of $S^2$ANet incorporating GCL achieves the performance of 45.99%, which is the best among the listed methods.

TABLE XI
COMPARISON WITH OTHER METHODS ON THE FAIR1M DATASET

| Method | FCOS [64] | RetinaNet [27] | DAL [18] | RIDet [20] | Faster RCNN [7] | CFC-Net [9] | Gliding Vertex [52] | $S^2$ANet [17] | RoI Trans. [29] | GCL | GCL | RoI Trasn. +GCL | RoI Trasn. +GCL | $S^2$ANet +GCL | $S^2$ANet +GCL |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Ms | | | | | | | | | | | ✓ | | ✓ | | ✓ |
| mAP | 23.70 | 27.67 | 29.12 | 31.58 | 33.70 | 34.31 | 35.86 | 36.12 | 38.27 | 29.89 | 37.74 | 40.64 | **44.68** | 37.57 | **45.99** |
| Boeing 737 | 10.34 | 35.01 | 32.53 | 28.25 | 36.05 | 30.89 | 36.32 | 36.04 | 35.84 | 33.06 | 38.19 | **38.70** | **45.30** | 34.67 | 45.44 |
| Boeing 747 | 43.54 | 83.72 | 74.39 | 80.62 | 85.19 | 83.87 | 82.61 | 84.42 | 82.74 | 81.99 | **86.98** | 85.02 | 83.53 | 84.12 | **85.70** |
| Boeing 777 | 5.96 | 12.64 | 13.14 | 12.92 | 12.45 | 10.72 | 11.29 | 12.09 | 12.81 | 10.00 | 11.49 | **16.48** | 15.29 | 10.83 | **18.94** |
| Boeing 787 | 13.67 | 36.68 | 39.91 | 45.28 | 45.35 | 38.60 | 48.69 | **52.32** | 43.90 | 41.32 | 46.98 | **51.84** | 47.59 | 37.65 | 49.15 |
| C919 | 0.00 | 1.44 | 2.11 | 0.15 | 15.45 | 5.67 | 24.48 | 5.81 | 15.77 | 1.08 | 1.59 | 20.00 | **25.46** | 1.12 | **25.09** |
| A220 | 11.71 | 45.44 | 41.32 | 39.89 | 49.50 | 42.44 | **50.01** | 46.34 | 48.68 | 43.50 | 49.87 | 48.47 | **51.14** | 48.80 | 49.38 |
| A321 | 3.95 | 64.95 | 58.38 | 53.69 | 63.16 | 50.68 | 65.27 | 67.80 | 67.35 | 65.59 | 69.24 | 67.08 | **70.81** | 68.05 | **71.42** |
| A330 | 15.03 | 58.52 | 48.59 | 62.80 | 65.89 | 55.13 | 69.98 | 66.31 | 65.56 | 51.99 | **73.21** | **70.45** | 68.67 | 63.00 | 67.51 |
| A350 | 14.20 | 71.45 | 54.88 | 55.27 | 62.69 | 59.20 | 65.18 | 72.31 | 62.92 | 70.62 | **78.93** | 71.30 | 70.94 | 74.39 | **78.01** |
| ARJ21 | 13.75 | 3.60 | 1.57 | 8.53 | 31.25 | 5.30 | 33.24 | 2.40 | **33.60** | 2.62 | 10.29 | 30.07 | 29.19 | 10.10 | **46.22** |
| passenger ship | 10.65 | 3.83 | 9.90 | 6.11 | 6.24 | 7.19 | 8.92 | 6.93 | 15.20 | 5.60 | 6.29 | **17.28** | **18.29** | 9.11 | 14.12 |
| motorboat | 46.21 | 22.03 | 53.04 | 55.20 | 44.37 | 63.38 | 52.04 | 52.58 | 58.04 | 41.57 | 62.00 | 58.68 | **68.99** | 57.55 | **68.61** |
| fishing boat | 9.59 | 2.12 | 5.71 | 5.49 | 3.71 | 8.72 | 5.11 | 7.36 | 9.37 | 3.13 | 7.44 | 9.41 | **11.73** | 9.78 | **11.95** |
| tugboat | 19.81 | 13.34 | 21.08 | 30.15 | 26.05 | 19.70 | 28.49 | 16.09 | 30.17 | 18.27 | 31.52 | **37.85** | **33.49** | 26.16 | 31.02 |
| engineering ship | **13.24** | 9.11 | 7.11 | 5.84 | 6.88 | 7.67 | 9.73 | 10.05 | 10.87 | 9.20 | 12.45 | 12.80 | **13.69** | 8.84 | 13.03 |
| liquid cargo ship | 12.92 | 4.37 | 12.05 | 17.21 | 9.50 | 21.23 | 15.67 | 22.88 | 19.28 | 5.27 | 9.56 | 24.85 | **30.29** | 23.72 | **27.60** |
| dry cargo ship | 35.08 | 14.49 | 28.41 | 29.58 | 17.78 | 30.54 | 26.75 | 37.82 | 33.02 | 19.96 | 25.94 | 36.67 | 37.14 | **39.67** | **41.69** |
| warship | 20.75 | 3.81 | 11.91 | 14.47 | 6.37 | 23.21 | 13.67 | 27.81 | 24.90 | 4.30 | 17.09 | 25.10 | **32.40** | 31.95 | **39.01** |
| small car | 42.56 | 41.91 | 48.05 | 52.73 | 51.44 | 62.43 | 49.53 | 65.12 | 57.73 | 51.63 | 68.48 | 57.78 | **71.51** | 66.43 | **76.26** |
| bus | 15.55 | 5.55 | 7.71 | 15.27 | 21.00 | 34.50 | 22.04 | 13.85 | 31.23 | 8.85 | 12.82 | 40.26 | **47.77** | 23.58 | **55.56** |
| cargo truck | 31.72 | 20.69 | 25.04 | 30.32 | 32.89 | 41.15 | 36.69 | 38.94 | 42.46 | 26.09 | 42.20 | 43.23 | **50.29** | 39.45 | **53.47** |
| dump truck | 23.90 | 16.54 | 22.82 | 29.50 | 40.04 | 42.18 | 39.52 | 41.80 | 45.26 | 18.25 | 45.31 | 45.53 | **57.17** | 40.96 | **58.32** |
| van | 34.59 | 34.09 | 43.26 | 45.01 | 45.96 | 51.65 | 43.65 | 60.84 | 54.49 | 46.91 | 67.39 | 53.93 | **72.45** | 62.20 | **75.94** |
| trailer | 12.14 | 0.33 | 2.48 | 3.82 | 7.82 | 11.41 | 11.65 | 5.08 | 15.54 | 1.02 | 4.74 | 14.53 | **19.60** | 8.31 | **22.15** |
| tractor | 1.07 | 0.36 | 1.03 | 0.05 | 3.77 | 1.69 | 2.90 | 1.39 | 3.55 | 1.20 | 1.81 | 1.45 | **5.38** | 1.22 | **5.59** |
| excavator | 7.90 | 0.52 | 5.06 | 5.03 | 9.28 | 10.26 | 12.49 | 8.60 | 12.78 | 3.12 | 13.95 | 11.04 | **20.87** | 11.93 | **19.21** |
| truck tractor | 1.09 | 0.01 | 0.55 | 0.53 | 1.71 | 0.71 | **3.66** | 0.58 | 2.59 | 0.08 | 0.58 | 1.01 | **5.99** | 1.04 | 3.52 |
| basketball court | 23.09 | 22.28 | 38.76 | 37.47 | 39.92 | 40.21 | 39.85 | 35.47 | 42.87 | 26.55 | 51.78 | **53.74** | **56.36** | 47.87 | 52.69 |
| tennis court | 74.76 | 78.62 | 75.37 | 77.78 | 76.97 | 79.41 | 76.98 | 78.55 | 78.40 | 79.07 | 81.47 | 80.37 | **85.89** | 81.72 | **86.08** |
| football field | 49.64 | 59.46 | 46.10 | 52.69 | 52.36 | 58.01 | 50.79 | 62.07 | 59.30 | 63.43 | 63.17 | **66.55** | 65.30 | 62.98 | **67.91** |
| baseball field | 82.90 | 86.46 | 84.66 | 85.63 | 87.56 | 84.34 | 86.85 | **88.80** | 86.60 | 87.31 | **88.97** | 87.23 | 87.28 | 87.28 | 87.50 |
| intersection | 55.14 | 57.33 | 44.06 | 51.41 | 57.11 | 51.98 | 58.59 | 55.67 | 58.18 | 59.57 | **66.34** | 61.14 | 63.47 | 58.23 | **64.70** |
| roundabout | **26.46** | 20.30 | 13.96 | 17.05 | 22.28 | 18.22 | 20.49 | **23.18** | 19.34 | 20.06 | 21.57 | 18.77 | 22.24 | 17.12 | 22.19 |
| bridge | 22.79 | 9.89 | 15.08 | 17.96 | 7.75 | 14.31 | 16.21 | 20.86 | 20.76 | 14.10 | 13.38 | 23.38 | **33.75** | 27.43 | **28.57** |

## VI. CONCLUSION

In this article, we analyze the IoU loss in oriented object detection. From the perspective of gradient analysis, we point out that the IoU loss suffers from anomalous gradient variations during the regression process. Hence, we propose a GCL to optimize the OBB regression by correcting the gradient of the regression loss. GCL constructs stable gradient variations as the model converges, which ensures accurate angle prediction. Meanwhile, a gradient scaling strategy is designed to balance the regression loss for objects with different scales. Our method achieves sustained performance gains on multiple remote sensing dataset. However, this article only considers the optimization for the main case of aligning two OBBs at their centers. In various corner cases during the early training stages, the rotated IoU loss still suffers from slow convergence. Therefore, in the future, we will attempt to design a more elegant loss to systematically address the convergence bottleneck of the rotated IoU loss for further performance improvement. In addition, we will investigate whether gradient anomalies exist in other losses and extend gradient analysis methods to optimize other regression losses, enhancing the performance of regression tasks in more domains.

## REFERENCES

[1] L. Li, S. Zhang, and J. Wu, "Efficient object detection framework and hardware architecture for remote sensing images," *Remote Sens.*, vol. 11, no. 20, p. 2376, Oct. 2019.

[2] X. Wu, W. Li, D. Hong, R. Tao, and Q. Du, "Deep learning for unmanned aerial vehicle-based object detection and tracking: A survey," *IEEE Geosci. Remote Sens. Mag.*, vol. 10, no. 1, pp. 91–124, Mar. 2022.

[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. Photogramm. Remote Sens.*, vol. 159, pp. 296–307, Jan. 2020.

[4] S. S. Kumar and M. U. Selvi, "Sea object detection using colour and texture classification," *Int. J. Comput. Appl. Eng. Sci.*, vol. 1, no. 1, pp. 59–63, 2011.

[5] C. Zhu, H. Zhou, R. Wang, and J. Guo, "A novel hierarchical method of ship detection from spaceborne optical image based on shape and texture features," *IEEE Trans. Geosci. Remote Sens.*, vol. 48, no. 9, pp. 3446–3456, Sep. 2010.

[6] R. Girshick, "Fast R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 1440–1448.

[7] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 6, pp. 1137–1149, Jun. 2016.

[8] Q. Ming and X. Xiao, "Towards accurate medical image segmentation with gradient-optimized dice loss," *IEEE Signal Process. Lett.*, vol. 31, pp. 191–195, 2024.

[9] Q. Ming, L. Miao, Z. Zhou, and Y. Dong, "CFC-Net: A critical feature capturing network for arbitrary-oriented object detection in remote-sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5605814.

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

[11] W. Li, W. Wei, and L. Zhang, "GSDet: Object detection in aerial images based on scale reasoning," *IEEE Trans. Image Process.*, vol. 30, pp. 4599–4609, 2021.

[12] 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.*, Jun. 2016, pp. 779–788.

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

[14] W. Liu et al., "SSD: Single shot MultiBox detector," in *Proc. Eur. Conf. Comput. Vis.*, Amsterdam, The Netherlands. Berlin, Germany: Springer, Oct. 2016, pp. 21–37.

[15] W. Zou, Z. Zhang, Y. Peng, C. Xiang, S. Tian, and L. Zhang, "SC-RPN: A strong correlation learning framework for region proposal," *IEEE Trans. Image Process.*, vol. 30, pp. 4084–4098, 2021.

[16] X. Yang, J. Yan, Q. Ming, W. Wang, X. Zhang, and Q. Tian, "Rethinking rotated object detection with Gaussian Wasserstein distance loss," in *Proc. Int. Conf. Mach. Learn.*, 2021, pp. 11830–11841.

[17] 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.

[18] 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.

[19] 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, p. 2664, Jul. 2021.

[20] Q. Ming, L. Miao, Z. Zhou, X. Yang, and Y. Dong, "Optimization for arbitrary-oriented object detection via representation invariance loss," *IEEE Geosci. Remote Sens. Lett.*, vol. 19, pp. 1–5, 2022.

[21] Q. Ming, L. Miao, Z. Zhou, J. Song, Y. Dong, and X. Yang, "Task interleaving and orientation estimation for high-precision oriented object detection in aerial images," *ISPRS J. Photogramm. Remote Sens.*, vol. 196, pp. 241–255, Feb. 2023.

[22] X. Yang et al., "Learning high-precision bounding box for rotated object detection via Kullback–Leibler divergence," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 34, 2021, pp. 18381–18394.

[23] Y. Yu, H. Guan, D. Li, T. Gu, E. Tang, and A. Li, "Orientation guided anchoring for geospatial object detection from remote sensing imagery," *ISPRS J. Photogramm. Remote Sens.*, vol. 160, pp. 67–82, Feb. 2020.

[24] 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/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2019, pp. 658–666.

[25] 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. AAAI Conf. Artif. Intell.*, Apr. 2020, vol. 34, no. 7, pp. 12993–13000.

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

[27] 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.*, Oct. 2017, pp. 2980–2988.

[28] 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.

[29] 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.*, Jun. 2019, pp. 2849–2858.

[30] Z. Chen et al., "PIoU Loss: Towards accurate oriented object detection in complex environments," in *Proc. Eur. Conf. Comput. Vis.* Berlin, Germany: Springer, 2020, pp. 195–211.

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

[32] M. Zand, A. Etemad, and M. Greenspan, "Oriented bounding boxes for small and freely rotated objects," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 4701715.

[33] B. Zhong and K. Ao, "Single-stage rotation-decoupled detector for oriented object," *Remote Sens.*, vol. 12, no. 19, p. 3262, Oct. 2020.

[34] X. Yang, J. Yan, Z. Feng, and T. He, "R$^3$Det: Refined single-stage detector with feature refinement for rotating object," in *Proc. AAAI Conf. Artif. Intell.*, May 2021, vol. 35, no. 4, pp. 3163–3171.

[35] 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. (CVPR)*, Jun. 2021, pp. 2786–2795.

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

[37] X. Yang et al., "The KFIoU loss for rotated object detection," in *Proc. 11th Int. Conf. Learn. Represent.*, 2023, pp. 1–18. [Online]. Available: https://openreview.net/forum?id=qUKsCztWlKq

[38] X. Yu, J. Lu, M. Lin, L. Zhou, and L. Ou, "MKIoU loss: Toward accurate oriented object detection in aerial images," *J. Electron. Imag.*, vol. 32, no. 3, Jun. 2023, Art. no. 033030.

[39] H. Peng and S. Yu, "A systematic IoU-related method: Beyond simplified regression for better localization," *IEEE Trans. Image Process.*, vol. 30, pp. 5032–5044, 2021.

[40] Q. Ming et al., "Deep dive into gradients: Better optimization for 3D object detection with gradient-corrected IoU supervision," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2023, pp. 5136–5145.

[41] 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.*, Oct. 2021, pp. 3520–3529.

[42] Z. Liu, L. Yuan, L. Weng, and Y. Yang, "A high resolution optical satellite image dataset for ship recognition and some new baselines," in *Proc. Int. Conf. Pattern Recognit. Appl. Methods*, vol. 2, 2017, pp. 324–331.

[43] H. Zhu, X. Chen, W. Dai, K. Fu, Q. Ye, and J. Jiao, "Orientation robust object detection in aerial images using deep convolutional neural network," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2015, pp. 3735–3739.

[44] J. Zhou, K. Feng, W. Li, J. Han, and F. Pan, "TS$^4$Net: Two-stage sample selective strategy for rotating object detection," *Neurocomputing*, vol. 501, pp. 753–764, Aug. 2022.

[45] D. Karatzas et al., "ICDAR 2015 competition on robust reading," in *Proc. 13th Int. Conf. Document Anal. Recognit. (ICDAR)*, 2015, pp. 1156–1160.

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

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

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

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

[50] 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.*, Perth, WA, Australia. Berlin, Germany: Springer, 2018, pp. 150–165.

[51] 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.

[52] 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. 2021.

[53] J. Wang, J. Ding, H. Guo, W. Cheng, T. Pan, and W. Yang, "Mask OBB: A semantic attention-based mask oriented bounding box representation for multi-category object detection in aerial images," *Remote Sens.*, vol. 11, no. 24, p. 2930, Dec. 2019.

[54] 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.

[55] 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, 2021.

[56] Y. Lin, P. Feng, J. Guan, W. Wang, and J. Chambers, "IENet: Interacting embranchment one stage anchor free detector for orientation aerial object detection," 2019, *arXiv:1912.00969*.

[57] Y. Zhang, W. Guo, C. Wu, W. Li, and R. Tao, "FANet: An arbitrary direction remote sensing object detection network based on feature fusion and angle classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 61, 2023, Art. no. 5608811.

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

[59] D. Yu et al., "An anchor-free and angle-free detector for oriented object detection using bounding box projection," *IEEE Trans. Geosci. Remote Sens.*, vol. 61, 2023, Art. no. 5618517.

[60] G. Wang et al., "High-quality angle prediction for oriented object detection in remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 61, 2023, Art. no. 5614714.

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

[62] G. Cheng et al., "Anchor-free oriented proposal generator for object detection," 2021, *arXiv:2110.01931*.

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

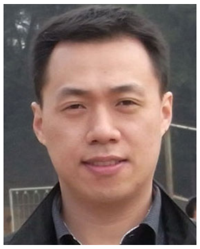
[64] Z. Tian, C. Shen, H. Chen, and T. He, "FCOS: Fully convolutional one-stage object detection," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 9627–9636.

**Qi Ming** received the B.S. degree in automation from the School of Automation, Beijing Institute of Technology (BIT), Beijing, China, in 2018, where he is currently pursuing the Ph.D. degree in navigation, guidance, and control.

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

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

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



**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 Post-Doctoral Researcher at the Institute of Automation, Chinese Academy of Sciences, Beijing. He is currently an Associate Professor at the School of Automation, BIT. His research interests include information fusion, pattern recognition, digital image processing, and vision-based navigation.

**Junjie Song** received the B.S. degree in electrical engineering and the automatization from China University of Mining and Technology, Xuzhou, China, in 2020. He is currently pursuing 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.

**Aleksandra Pižurica** (Senior Member, IEEE) received the Diploma degree in electrical engineering from the University of Novi Sad, Novi Sad, Serbia, in 1994, the M.Sc. degree in telecommunications from the University of Belgrade, Belgrade, Serbia, in 1997, and the Ph.D. degree in engineering from Ghent University, Ghent, Belgium, in 2002.

She is currently a Professor of statistical image modeling with Ghent University. Her research interests include the area of signal and image processing and machine learning, including multiresolution statistical image models, Markov random field models, sparse coding, representation learning, and image and video reconstruction, restoration, and analysis.

Prof. Pižurica is a member of the EURASIP Technical Area Committee Signal and Data Analytics for Machine Learning. She received the Scientific Prize "de Boelpaepe" from the Royal Academy of Science, Letters and Fine Arts of Belgium for her contributions to statistical image modeling and applications to digital painting analysis, in 2015. She received numerous other recognitions for her work, among which as a co-recipient of the David Hestenes Prize from AGACSE 2018 and the Best Paper Award of the IEEE Geoscience and Remote Sensing Society (GRSS) Data Fusion contest, in 2013 and 2014. She served as the TPC Co-Chair for the 30th EUSIPCO Conference (in 2022), an Europe Liaison for IEEE ICIP 2020 and ICIP 2024, the Plenary Co-Chair for EUSIPCO 2024, and elected as the TPC Co-Chair for IEEE ICIP 2026. She served as an Associate Editor for IEEE TRANSACTIONS ON IMAGE PROCESSING from 2012 to 2016 and IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY from 2016 to 2019. She is a Senior Area Editor of IEEE TRANSACTIONS ON IMAGE PROCESSING from 2016 to 2019 and since 2022.