Task Interleaving and Orientation Estimation for High-Precision Oriented Object Detection in Aerial Images

Qi Ming [1], Lingjuan Miao[1], Zhiqiang Zhou[1], Junjie Song[1], Yunpeng Dong[1], Xue Yang[2]

[1] *School of Automation, Beijing Institute of Technology*
[2] *Department of Computer Science and Engineering, Shanghai Jiao Tong University*

Abstract

Oriented object detection in aerial images has received extensive attention due to its wide range of application scenarios. Although great success has been achieved, current methods still suffer from inferior high-precision detection performance. Firstly, the classification scores cannot truly represent the localization accuracy of the predictions. Secondly, the orientation prediction in these detectors is not accurate enough for high-precision object detection. In this paper, we propose a **T**ask **I**nterleaving and **O**rientation **E**stimation **Det**ector (TIOE-Det) for high-quality oriented object detection in aerial images. Specifically, a posterior hierarchical alignment (PHA) label is proposed to optimize the detection pipeline. TIOE-Det adopts PHA label to integrate fine-grained posterior localization guidance into classification task to address the misalignment between classification and localization subtasks. Then, a balanced alignment loss is developed to solve the imbalance localization loss contribution in PHA prediction. Moreover, we propose a progressive orientation estimation (POE) strategy to approximate the orientation of objects with n-ary codes. On this basis, an angular deviation weighting strategy is proposed to achieve accurate evaluation of angle deviation in POE strategy. TIOE-Det achieves significant gains on high-precision detection performance. Extensive experiments on multiple datasets prove the superiority of our approach.

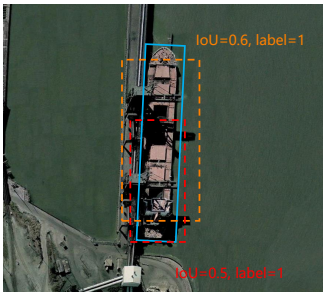
Keywords: Oriented object detection, Aerial images, Convolutional neural network, Misaligned tasks, Orientation estimation

1. Introduction

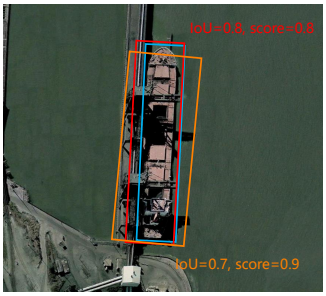
Object detection in aerial images has been a hot topic in recent years. As the available satellite data increased rapidly, efficient detection of objects of interest in aerial images has become a crucial issue. Traditional methods usually use handcraft features for object detection [1, 2], which are both time-consuming and not accurate enough.

Over the past few years, the development of deep learning has greatly promoted the progress in generic object detection. The powerful and efficient feature extraction ability of convolutional neural networks (CNNs) enable the detector to have both high speed and high accuracy. A series of advanced detectors have been proposed to achieve high-performance detection with horizontal bounding box (HBB), such as Faster R-CNN [3] and YOLO series [4, 5, 6]. These detectors decouple the object detection task into a category recognition subtask and a position regression subtask, and then design independent branches to complete the respective tasks.

Objects in aerial images often have arbitrary orientations. The horizontal bounding box used in generic object detection cannot locate these oriented aerial objects well. Therefore, rotation detectors use the oriented bounding box (OBB) to represent the ground-truth (GT) objects in the aerial images[7, 8, 9]. The GT object is denoted as (cx, cy, w, h, θ) under the OBB representation, in which (cx, cy) denotes the center point of the OBB, (w, h) is the weight and the height of box, θ represents the orientation of object. Recently, many advanced rotation detectors have been proposed to achieve accurate oriented object

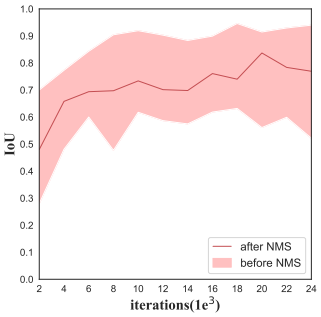


(a) Anchors

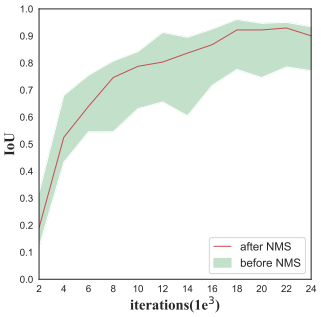


(b) Predictions

Figure 1: Visualization of the ground-truth box (blue), preset anchors (dotted line) and its regression boxes (solid line). The predicted box with higher IoU (0.8>0.7) gets a lower classification score (0.8<0.9), which reveals that the binary classification label cannot distinguish bounding boxes with different localization accuracy.



(a) RetinaNet



(b) TIOE-Det

Figure 2: Illustration of IoU distribution between all detections and GT boxes. The output detections is obtained through NMS from all detections. (a) reveals that many high-quality detections cannot be effectively output due to the inconsistency between classification and regression, which hurts the high-precision detection performance. (b) achieves significant high-precision detection performance through task interleaving and consistency learning.

detection in aerial images[8, 10, 11, 12, 13, 14, 15]. However, high-precision oriented object detection in aerial images remains a challenging task. Most of the existing rotation detectors are developed from the generic object detectors by directly introducing an extra angle prediction. Therefore, the framework does not adapt to oriented object detection.

Firstly, rotation detectors usually adopt the unrelated classification and regression branches to achieve oriented object detection. The independent prediction of the two tasks makes them incompatible, which degrades high-precision detection performance. Specifically, a high classification score of the predicted box cannot guarantee a good localization result. For example, the two anchors in Fig. 1a are regarded as positive samples, and thus ground-truth (GT) labels for classification are set to 1. However, their corresponding predictions in Fig. 1b show that the predicted box with better localization accuracy gets lower predicted classification scores. This high-quality but low-scoring detection would be suppressed in the Non-Maximum Suppression (NMS) process, leading to the weak correlation between classification score and localization accuracy.

We further visualized the Intersection-over-Union (IoU) distribution of detections to confirm the above comments. We trained the rotated RetinaNet [16] on HRSC2016 dataset [17], and then performed inference on testing images and counted the detections with the predicted classification score higher than 0.5. Before NMS, the IoU of predictions has a large variance and includes many potential high-precision detections (see Fig. 2a). However, when NMS is performed based on the classification confidence, high-quality detections are suppressed due to unreliable classification scores.

Secondly, many rotation detectors suffer from inaccurate orientation regression introduced by angle prediction in OBB. The mainstream rotation detectors directly regress the angle of OBB, which gives rise to three issues. Firstly, angle prediction should be paid more attention for high-precision oriented object detection, but most rotation detectors often treat different variables equally in regression loss. Secondly, the boundary of the angle definition leads to a suboptimal angle optimization process. As shown in Fig. 3, with the angle defined in $[0, 180^{o})$, the real angular deviation between the anchor box $(100, 100, 600, 100, 0^{o})$ and the GT box $(100, 100, 600, 100, 175^{o})$ is quite small, there is only an slight angle offset of 5^{o}. But angular deviation would be calculated as 175^{o} due to the boundary of angle definition, which leads to a large angle loss. As a result, the regression loss may oscillate and leads to a suboptimal optimization process as shown in the right of Fig. 3. Thirdly, for the objects with large aspect ratios (such as bridges, ships), a slight angular deviation will cause the

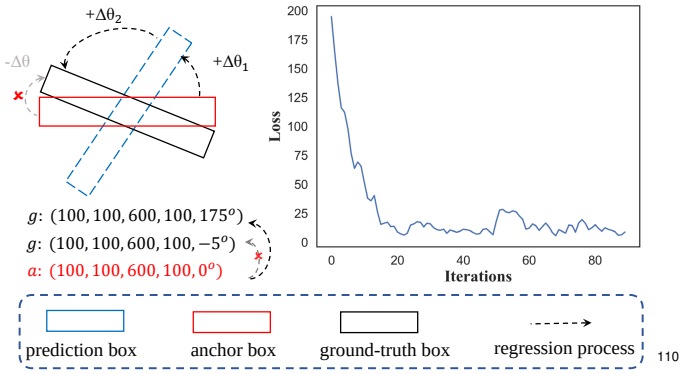


Figure 3: Oscillation of angle regression loss caused by the inadequate orientation representation. Due to the definition boundary of the angle, it's suboptimal to direct regress the angles in OBB representation.

IoU between the predictions and the GT boxes to drop sharply. The angle loss of these objects should not be treated the same as that of square-like objects.

To solve the above-mentioned problems, in this paper, we proposed a novel Task Interleaving and Orientation Estimation detector (TIOE-Det) for high-precision oriented object detection in aerial images. TIOE-Det discards the binary classification branch and adopts a task interleaving branch to interweave class recognition task and OBB regression task under a unified pipeline. Specifically, a posterior hierarchical alignment (PHA) label is designed to introduce fine-grained posterior localization guidance into classification task. Next, we propose a balanced alignment loss (BAL) to solve dominant loss contribution of negatives in PHA prediction. During the inference stage, localization-guided NMS is conducted to obtain high-precision detections based on PHA scores. As shown in Fig. 2b,TIOE-Det outputs credible detections with high IoUs and thus achieves better high-precision detection performance.

To achieve accurate orientation prediction, we propose a progressive orientation estimation (POE) strategy to optimize angle prediction in TIOE-Det. The POE strategy encodes the GT angle into a discrete n-ary code via a progressive approximating manner. Continuous angles are transformed into efficient discrete codes within an acceptable error range. In this way, the suboptimal optimization problem could be solved. Then, an angular deviation weighting (ADW) strategy is designed to further optimize the angle loss under POE representation. The ADW strategy comprehensively considers the aspect ratio, angular error, and gradient optimization to determine the magnitude of angle loss for better convergence.

TIOE-Det achieves superior high-precision detection accuracy, and outperforms many recent advanced rotation detectors. Extensive experiments on multiple aerial image datasets demonstrate the effectiveness of our method. The main contribution of this paper can be summarized as follows:

- A novel TIOE-Det is proposed to achieve high-precision oriented object detection by bridging the inconsistency between subtasks and optimizing the orientation prediction.
- We observed that binary classification task lead to misaligned classification and regression performance. A posterior hierarchical alignment label is then proposed to use fine-grained posterior localization guidance to optimize the detection pipeline in rotation detectors.
- We innovatively represent angles as n-ary codes via a progressive orientation estimation (POE) method for high-precision OBB regression. Meanwhile, the angular deviation weighting strategy is developed to adaptively correct POE deviation to further performance gains.

2. Related Work

2.1. Oriented Object Detection

Object detection is an important topic in the field of computer vision. Over the past decade, a series of detectors have been proposed to detect objects using horizontal bounding box (HBB) [18, 3, 4, 5]. Recently, oriented object detection in aerial images has received more and more attention due to its wide range of application scenarios. The objects in the aerial images are from a bird's-eye view with arbitrary orientations. Therefore, the oriented bounding box (OBB) is used to represent arbitrary-oriented objects. Many advanced rotation detectors have been developed to detect oriented objects in the aerial images [10, 13, 11, 14, 19, 20]. Since there is large variations in angle, scale, and aspect ratio of the objects in aerial

scenes, these methods preset densely laid anchor boxes for accurate detection, such as SCRDet [19], RRPN [21]. The dense anchors bring redundant overhead and lead to imbalance problems. Some methods preset horizontal anchors to alleviate the imbalance issue [20, 8, 10]. For example, RoI Transformer [20] transforms a horizontal RoI into a rotated RoI and then extracts rotation-invariant features for classification and regression. S^2A-Net [10] generates high-quality oriented anchors via anchor refinement and adaptively aligns the convolutional features with the anchors.

Some works focus on feature extraction in oriented object detection [8, 7, 10]. For example, Ming *et al.*[8] suggested that the classification and regression tasks respond differently to features in oriented object detection. Then, a polarized attention mechanism is proposed to extract task-sensitive feature maps. R^3Det [7] builds aligned feature maps to accommodate the localization offsets of the refined bounding boxes.

Optimization of regression loss for oriented object detection is another hot topic recently. The extra angle prediction in OBB representation derives many issues, such as loss oscillation caused by out-of-bounds angle [22], ambiguity of OBB representations [12]. Circular Smooth Label [22] tackles the out-of-bounds angles by transforming orientation regression into a classification task. RIDet [12] treats ambiguous representations as equivalent local minima to optimize angular error with a representation invariance loss. Yang *et al.*[14, 11] adopted Gaussian wasserstein distance and Kullback-Leibler divergence to measure the distance between OBBs, thus avoiding the problems caused by angle prediction.

2.2. Misaligned Classification and Regression

The misalignment between classification and regression indicates that the classification scores of the predictions cannot represent the localization accuracy of the predictions. This issue has been discussed in some previous work in horizontal object detection. For example, some work [23] realigns RoI features to eliminate feature offsets of RoIs, which helps the NMS procedure to select well localized bounding boxes. Miao *et al.* [24] suggested that the misalignment stems from unreasonable training sample selection and designed a dynamic anchor learning strategy to select high-quality positives.

Some work have tried to directly predict the IoU between the detections and GT boxes to guide NMS procedure, and thus bridging the gap between classification and regression [25, 26, 23]. IoU-Net [25] and IoU-uniform R-CNN [23] use an additional IoU prediction branch to evaluate the localization accuracy of the detections for the NMS process. Methods such as VFNet[27] and cleanliness scores [26] combine IoU with classification scores to select high-quality detections. There are some methods obtain credible classification confidence by uncertainty estimation, such as softer-NMS [28], Gaussian YOLOv3 [29].

However, there are still some problems in these methods. Firstly, most of these works ignore that the binary GT labels of the classification task are the culprit of the misaligned classification and regression performance. The binary classification branch is still adopted during training and inference, and thus the predicted confidence is unreliable. Secondly, the semantic information of IoU is obscure and hard to identify. The methods that use IoU prediction branches often design complex IoU regression structures and training strategies, such as IoU-Net [25]. Even so, IoU prediction is still not accurate enough, and the network is hard to converge. Thirdly, the extra IoU prediction branch introduces additional computational overhead and reduces the inference speed. We will make improvements in these areas in this paper.

2.3. Angle Prediction in Rotation Detectors

Objects in aerial images are usually arbitrary-oriented. The simple and effective represent is the oriented bounding box (OBB) denoted as (cx, cy, w, h, θ), which is also the mainstream representation in current rotation detectors [10, 13, 11, 14, 19, 20, 12]. The angle variable introduced in OBB derives many problems.

The boundary of the angle definition leads to a suboptimal angle optimization process, which has been discussed in some

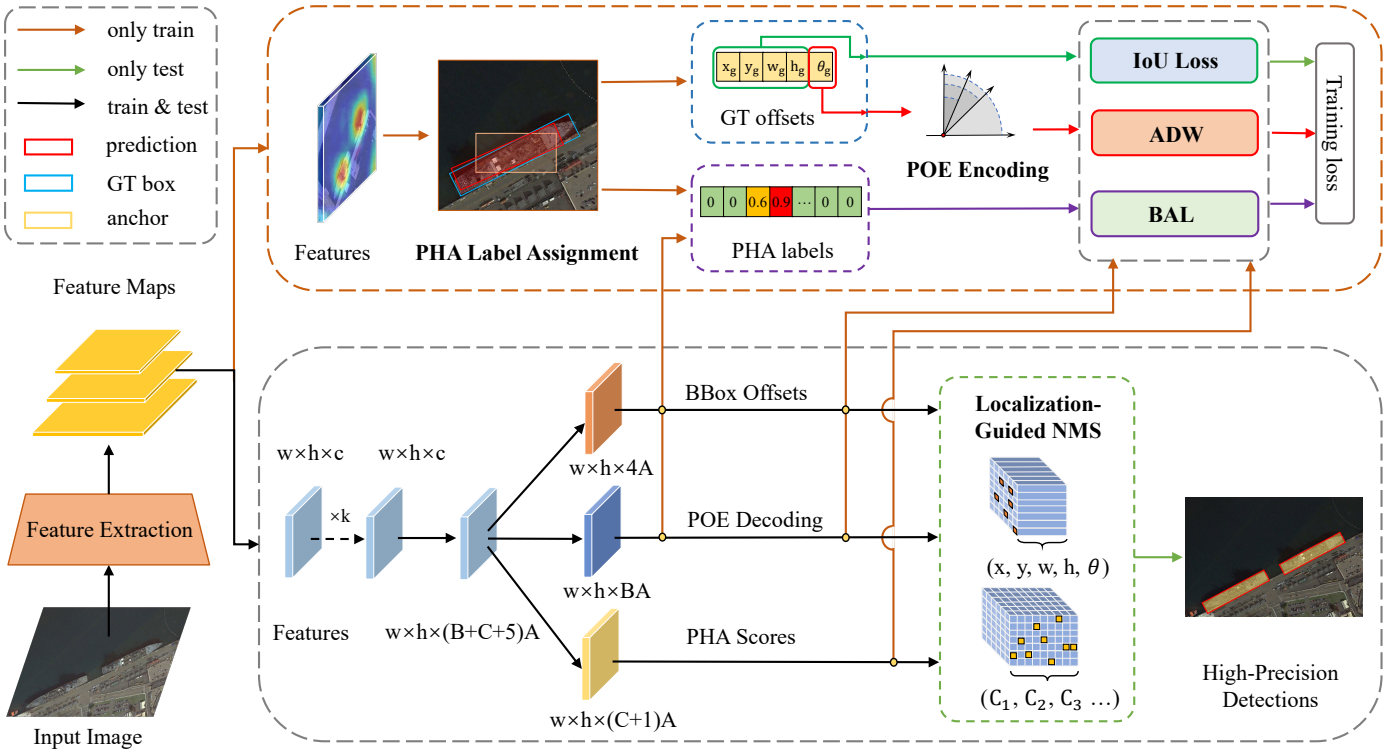


Figure 4: Overview of our proposed TIOE-Det. 'C' is the total number of categories, 'A' denotes the number of anchors laid at each position of the feature map, and 'B' is the length of the POE coding of orientations. 'only train' means it ony works during the training process, likewise for 'only test' and 'train and test'.

previous work [30, 22, 12, 31]. To solve the issue, the work [12] propose a representation invariance loss, which treats redundant OBB representations as equivalent local minima for consistent optimization. Llerena, J. M. *et al.* [11] transform the OBB into a Gaussian distribution, and use a covariance matrix to represent the orientation of the OBB. There are also methods to discretize the angle variable to optimize angle prediction. For example, Circular Smooth Label [22] (CSL) transforms the angle regression into a angle classification task via gaussian window function. But the overly heavy angle classification head brings large computational burden and reduces the inference speed. Densely Coded Labels [30] (DCL) solves the problem by using Binary code and Gray code for efficient angle encoding. However, neither CSL nor DCL considers the impact of different bits in the coding method on IoU variations.

Besides, aerial images often contain a large number of objects with large aspect ratios, such as bridges, trucks. For these objects, a slight angular deviation will cause the IoU between the predictions and the GT boxes to drop sharply, and thus angle prediction should be paid more attention. Zhang *et al.* [32] proposed a aspect ratio guided method for more accurate angle regression for long objects. Zhu *et al.* [33] designed a length-independent IoU to increase the tolerance of long and narrow objects in the label assignment for better angle performance.

3. Proposed Method

The overall framework of our TIOE-Det is shown in Fig. 4. TIOE-Det uses a fully convolutional network to extract multi-scale features. Then, three branches are adopted to locate the objects and conduct class recognition. The HBB regression branch together with POE prediction branch determine the position of the objects in the images. Next, the task interleaving branch predicts PHA scores to determine the class and localization confidence of the predicted boxes. Finally, localization-guided NMS is performed during inference to select high-precision detections based on the predicted PHA scores.

(a) IoU = 0.50 (b) IoU = 0.80 (c) IoU = 0.85

Figure 5: Illustration of different IoU cases between GT box (blue) and corresponding detections (red).

3.1. Posterior Hierarchical Alignment Label

Current rotation detectors often use a unified IoU threshold between GT boxes and preset anchors for training sample selection (also called label assignment). We suggest that various rotated IoUs between two OBBs should be treated differently during training. As illustrated in Fig. 5a, the predicted box with an IoU of 0.5 has a poor spatial alignment with GT box. This detection should not be treated the same as the one with an IoU of 0.8 in Fig. 5b. However, current methods treat them equally as positive samples in classification branch and their class labels are all set to 1. In this case, the classifier cannot select detections with accurate localization results based on the classification scores. Hence it is not conducive to high-quality oriented object detection.

The most intuitive solution is to introduce the posterior localization guidance to the classification task, just like the IoU prediction methods [25, 34, 35, 27]. Although good performance has been achieved by these methods, there are still many problems, which can be summarized into the following three folds:

1) Firstly, the semantic information of IoU is very obscure, and it is hard to predict IoU accurately. Most IoU prediction work designs complex network [34] or independent training strategy [25] to predict IoU, which not only makes the model more complicated, but also still suffers from the hard convergence of IoU prediction.

2) Secondly, it is unnecessary to predict the accurate IoU for low-quality detections (e.g. samples with IoU <0.1). The IoU prediction for low-quality samples does not help improve the performance, but hinders network convergence.

3) Moreover, completely accurate IoU prediction for positives is inefficient and may brings slight performance improvement. For example, The two detections with IoU of 0.80 and 0.85 are almost the same spatially (see Fig. 5b and Fig. 5c). However, the continuous IoU is hard to learning.

Based on the above observations, we propose the posterior hierarchical alignment (PHA) label as an efficient metric for interleaving classification and localization tasks. The classification branch is replaced with a PHA prediction branch. Then, PHA label assignment is conducted based on the prior knowledge from both category and localization to select high-quality samples. Finally, the predicted PHA scores are used for localization-guided NMS for high-precision detections. The detailed definition of PHA label is introduced as follows.

We denote the IoU between a predicted box and its corresponding GT box as o, which also represents posterior spatial alignment of the outputs. T_p and T_n are the IoU thresholds to determine the positives and negatives, respectively. The posterior IoU interval is divided into l intervals:

$$\delta = (1.0 - T_p)/l. \tag{1}$$

Next, the PHA label o^* is defined as follows:

$$o^*(o) = \begin{cases} (\lfloor o/\delta \rfloor + 1) \cdot \delta & \text{if } o > T_p \\ 0 & \text{if } o < T_n, \end{cases} \tag{2}$$

in which $\lfloor \cdot \rfloor$ is the floor function.

PHA label divides posterior IoU of the prediction into fine-grained quality intervals, which effectively characterize localization accuracy. Compared with the overly fine posterior IoU prediction, our discrete but accurate PHA labels are more conducive to network convergence to achieve better performance. Meanwhile, we set PHA label of low-quality samples to be 0 in Eq. (2). Since these samples might produce abnormally high scores during inference stage if no supervision was imposed.

Algorithm 1 Localization-Guided NMS

Input: $\mathcal{B} = \{b_1, b_2, ..., b_n\}$ is a $N \times 5$ matrix of detection boxes. $\mathcal{S} = \{s_1, s_2, ..., s_n\}$ denotes the PHA scores. N_t is the NMS threshold.

Output: $\mathcal{D} = \{d_1, d_2, ..., d_n\}$ is a $N \times 5$ matrix of final detections.

$\mathcal{D} \Leftarrow \{\}$
while $\mathcal{B} \neq \varnothing$ **do**
  $m \Leftarrow \text{argmax}\, \mathcal{S}$
  $\mathcal{M} \Leftarrow b_m$
  $\mathcal{D} \Leftarrow \mathcal{D} \cup \mathcal{M}$; $\mathcal{B} \Leftarrow \mathcal{B} - \mathcal{M}$
  for b_i in $\mathcal{B}$ **do**
    if $IoU(\mathcal{M}, b_i) \geq N_t$ **then**
      $\mathcal{B} \Leftarrow \mathcal{B} - b_i$; $\mathcal{S} \Leftarrow \mathcal{S} - s_i$
    end if
  end for
end while
return $\mathcal{D}, \mathcal{S}$

Besides, we adopted a curriculum learning based training strategy to gradually increase the number of PHA levels for a more smooth and stable training process. The adaptive PHA level is as follows:

$$l(t) = \left\lfloor \frac{t}{l_0} \right\rfloor + 1, \tag{3}$$

in which $t = \frac{iters}{Max\_Iteration}$, and $Max\_Iteration$ is the total number of iterations. l_0 is the number of PHA levels that is finally adopted. The number of levels is gradually increased in different intervals until it reaches l_0. We use fewer PHA intervals in initial training stage so that the network can converge more quickly. As the model converges, the number of levels adaptively increases so that the model can distinguish IoU in different intervals and then recognize high-quality detections.

In the inference stage, we design a localization-Guided NMS (LG-NMS) to select high-quality detections with predicted PHA scores. The category with highest PHA score of a prediction box will be selected as predicted class. Then the predictions with PHA scores less than preset threshold are suppressed. Different from traditional NMS procedure that use classification score for sample selection, LG-NMS adopts the credible PHA score to ensure high-quality detections. In this way, predictions with accurate localization results will be selected as the final detections. Therefore, LG-NMS effectively bridges the inconsistency of classification and regression. The pseudo-code of LG-NMS can be found in Algorithm 1.

3.2. Progressive Orientation Estimation

Accurate angle prediction is quite important for high-precision oriented object detection in aerial images. Angle representation is periodic, which means that many angle representations may represent the same real orientation. It would hinder the model convergence[22]. Intuitively, transforming angle regression into an angle classification task could avoid the problem of redundant representation of angles. Some previous work [22, 30] tried to regard the orientation prediction task as an angle classification task, but still suffer from heavy heads, intolerable errors, or hard optimization. To optimize the angle classification method, we propose a heuristic orientation encoding method into TIOE-Det, which is called progressive orientation estimation (POE). POE is a heuristic and flexible encoding method. On the one hand, POE has practical physical meaning and is easy to learn. On the other hand, we can flexibly adjust the representation of POE according to the angle prediction accuracy requirements of different detection tasks.

Given the pre-defined angle range $R = [R_0, R_1)$ (such as R = $[0, 180^\circ)$), we encode object orientation θ into a discrete n-ary code Θ that has N significant conditions, and $N \in \mathbb{N}^+$. We denote $\Delta R = R_1 - R_0$ as the length of angle range. Both R_0, R_1, and ΔR are converted to degrees. The total length of Θ is as follows:

$$k(R, N) = min\{x \in \mathbb{Z} | N^x \geq \Delta R\}. \tag{4}$$

Note that the n-ary code can represent digits in $[0, N^k)$, which is beyond the real angle range ΔR. We constrain the representation range into the given angle range via a angle unit δ:

$$\delta = \Delta R / N^k. \tag{5}$$

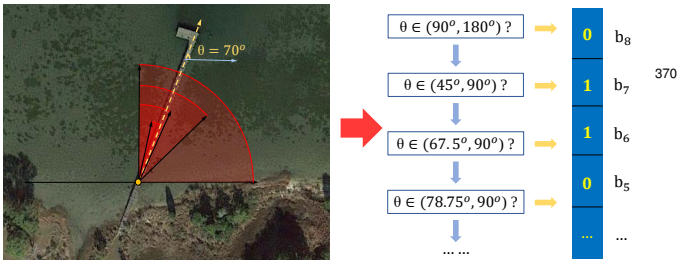


Figure 6: Illustration of the flow of POE coding for a given $\theta = 70°$ with $N = 2$.

The target n-ary code is a tuple denoted as $\Theta = (e_1, e_2, ..., e_k)$, in which:

$$\begin{cases} e_k(\theta) = \left\lfloor (\frac{\theta}{\delta})/N^{k-1} \right\rfloor \\ e_i(\theta) = \left\lfloor \frac{\frac{\theta}{\delta} - \sum_{j=i}^{k-1} (e_{j+1} N^j)}{N^{i-1}} \right\rfloor, i \in \{1, 2, ..., (k-1)\} \end{cases} . \tag{6}$$

where $\lfloor \cdot \rfloor$ is the floor function. Eq. (6) recursively divides the total angle interval and progressively approximates the real orientation of the bounding box. The N significant conditions in n-ary code divide the angle interval into N equal subintervals, and the angle interval is determined by the position of the code symbol.

Shown in Fig. 6 is a special case of POE when $N = 2$. In this case, the GT angle θ is encoded as a binary code Θ and $k = 8$. We first divide the entire angle range $R = [0, 180°)$ into two subintervals $R_l = [0, 90°)$ and $R_r = [90°, 180°)$. Since that $\theta \in R_l$, we know $b_8 = 0$. On this basis, we further divide the R_l into $R'_l = [0, 45°)$ and $R'_r = [45°, 90°)$. Then we know $b_7 = 1$ since $\theta \in R'_r$. Other digits of POE coding are also obtained by recursively refining the interval and approximating GT orientation.

Although POE is an approximate encoding of continuous angles, the angular error is tolerable. The angular error of POE is always less than $1°$ according to the definition in Eq. 4 and Eq. 5. For instance, in the example in Fig. 6, the angle range $[0, 180°)$ is encoded into the binary code. The maximum angular error is the angle unit in Eq. 5, that is, $\delta = \frac{R_1}{N^k} = \frac{180°}{2^8} \approx 0.703°$. Such a small angular deviation could hardly be recognized and would not bring much representation error. We visualize the change of IoU with aspect ratios and scales under angle error of $x0.703°$ as shown in Fig. 7a. The maximum IoU deviation is only about 0.06, which hardly affects detection performance.

During inference stage, the predicted POE vector Θ is decoded into the angle representation as follows:

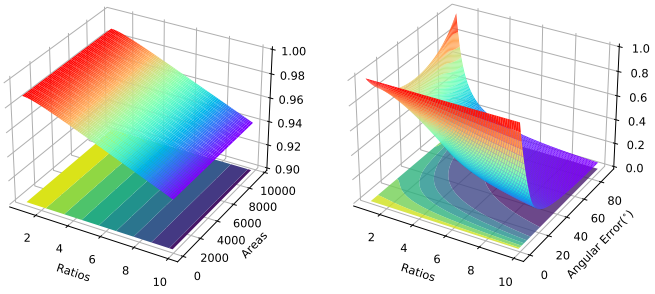
$$\theta(\Theta) = R_0 + \sum_{i=1}^{k} (N^{i-1} e_i) \tag{7}$$

Our method is a heuristic progressive search strategy for orientation encoding, which is a general form of similar angle classification methods. For instance, when $N = 180$ in Eq. 4, our POE is simplified to one-hot angle encoding method [31]. When $N = 2$ in Eq. 4, our encoding is the same as to binary encoding in DCL [30]. POE demonstrates the effectiveness of n-ary codes from the perspective of progressive orientation approximation. In this way, it gives a general and flexible form of angle classification representation and an intuitive explanation of its feasibility.

Furthermore, POE is superior to the previous angle classification methods. The overly heavy classification head leads to slow inference speed, such as CSL [22] and MEBOW [31]. On the contrary, the coarse-grained angle encoding cannot accurately measure the deviation of the angle, such as DCL [30]. Our method is more generalized and flexible, and we can adjust the significant conditions of POE to make a trade-off between accuracy and computational overhead. Besides, some existing angle classification methods deviate from practical meaning and thus they are difficult to learn, such as angle encoding with Gray code [30]. POE coding is an interpretable method inspired by progressive orientation approximation, and thus it is easier to converge.

3.3. Loss Function

We designed two novel loss functions for the proposed PHA prediction branch and POE strategy to further optimize detection performance.



(a) (b)

Figure 7: Visualization of IoU with variation of bounding box area, aspect ratios, and angular errors. Two center-aligned OBBs have the same area. Visualization result in (a) adopts a fixed angular error of 0.703° (N=2 in POE coding), and the IoU deviation is quite small, which ensures that POE strategy does not degrade detection accuracy. The results in (b) show that a slight angular error would lead to a drastic drop in IoU for OBBs with large aspect ratios.

3.3.1. Balanced Alignment Loss

Most of the predictions in the feature maps are background. These massive low-quality anchors with very low PHA scores will dominate the training loss, which makes it hard to optimize the PHA predictions for positives.

To solve the problem, we propose the balanced alignment loss (BAL) to balance the influence of positives and negatives. For a detection box with predicted PHA score $\tilde{o}$, its deviation from PHA label o^* is denoted as Δ:

$$\Delta = |o^* - \tilde{o}| \tag{8}$$

Then, the balanced alignment loss is as follows:

$$BAL(\tilde{o}, o^*) = -\Delta^{\alpha} log(\Delta) \cdot [1 - \chi(o^*)] - \Delta^{\beta} log(\Delta) \chi(o^*) \tag{9}$$

in which $\chi(\cdot)$ is the indicator function:

$$\chi(o^*) = \begin{cases} 1 & \text{if } o^* > T_p \\ 0 & \text{if } o^* < T_n, \end{cases} \tag{10}$$

T_p and T_n are the thresholds for training sample division of positives and negatives, respectively. α and β are modulation parameters to control the contribution of training samples with different PHA contribution to the loss. BAL reduces the loss contribution of the simple background candidates and positives through modulation terms Δ^{α} and Δ^{β}. Meanwhile, by adjusting α and β, we could make a trade-off between the loss contribution of positives and negatives for balanced training.

Then, the BAL for PHA prediction during training is defined as follows:

$$L_{\text{PHA}}(\tilde{\boldsymbol{o}}, \boldsymbol{o}^*) = \frac{1}{N} \sum_{i \in \psi} BAL(\tilde{\boldsymbol{o}}_i, \boldsymbol{o}_i^*), \tag{11}$$

in which ψ indicates the total training samples. o^* is PHA labels assigned to a certain class of objects. $\tilde{o}$ represents the class-specific PHA prediction. Note that $\tilde{o} \in \boldsymbol{R}^{N \times C}$ and $\tilde{o} \in [0, 1]$, N is the number of total anchors and C denotes the classes. In the inference stage, we will select the class with the highest IoU predicted by each anchor and use it as the class prediction result and confidence.

3.3.2. IoU Loss

We decouple the OBB prediction into HBB prediction and POE coding prediction in TIOE-Det (see Fig. 4). The corresponding loss functions are described below.

The scales of objects in aerial images varies greatly. IoU is scale invariant when measuring the spatial gap between two HBBs. Therefore, we use IoU loss for HBB regression in TIOE-Det. For each object g, its OBB representation is $\boldsymbol{b}_i^* = (cx^*, cy^*, w^*, h^*, \theta^*)$, in which (cx^*, cy^*) is center coordinate of OBB, (w^*, h^*, θ^*) denote width, height, and angle of the box, respectively. Its corresponding prediction is denoted as $\boldsymbol{b}_i = (cx, cy, w, h, \theta)$. Then, the IoU loss for HBB is as follows:

$$L_{\text{IoU}}(\boldsymbol{b}, \boldsymbol{b}^*) = \frac{1}{N_p} \sum_{i \in \psi_p} \left[1 - o(\boldsymbol{b}_i^*, \boldsymbol{b}_i)|_{\theta^* = \theta = 0} \right] \tag{12}$$

where N_p indicates the number of positive anchors ψ_p. $\boldsymbol{b} \in \boldsymbol{R}^{N \times 5}$ is total anchors, and $\boldsymbol{b}^* \in \boldsymbol{R}^{N \times 5}$ denotes the corresponding GT box. $o(\cdot)$ calculates the IoU between two boxes.

3.3.3. Angular Distance Weighting

Another major problem is the angular error measurement in POE coding. Previous classification-based rotation detectors suffer from two issues. Firstly, these method usually treat the

impact of different bits equally in angle coding [30, 22, 31], and therefore they produce the same gradients for different angular errors. We suggest that high bits in the POE coding have a greater impact on the angular error. For example, if the model outputs $b_7 = 1$ in Fig. 6, the predicted angle is far away from the correct orientation, while the influence of incorrect b_1 does not hurt so much. Secondly, these methods still suffer from misalignment between angle loss and detection performance. For example, if the angle range $[0, 180^\circ)$ is transformed into 180 angle classes, the cross entropy loss between 46° and 45° degrees is the same as that between 1° and 45°. They all have only two different digits in the encoded labels and lead to same angle loss, which is obviously unreasonable.

We proposed an angular distance weighting (ADW) strategy to optimize the angle classification loss and address the above issues. The ADW consists of two parts: angular offset metric (AOM) and encoding offset metric (EOM). The AOM measures the importance of angular error for accurate localization, while EOM evaluates the importance of different bits within POE coding. Specifically, AOM is defined as follows:

$$AOM(\Delta\theta, r) = \underbrace{(\Delta\theta - \sin\Delta\theta)}_{f(\Delta\theta)} \cdot \underbrace{ln(r + e - 1)}_{g(r)}, \qquad (13)$$

where $\Delta\theta$ denotes the angular error between the orientation of GT box θ^* and that of predicted box θ. r is aspect ratio of GT box. $f(\Delta\theta)$ and $g(r)$ are functions of $\Delta\theta$ and r respectively, and we will introduce them later.

Next, AOM is weighted to angle loss together with an EOM as follows:

$$L_{\text{ANG}}(\boldsymbol{e}, \boldsymbol{e}^*) = AOM(\Delta\theta, r) \cdot \underbrace{\sum_{i=1}^{k} (N^{i-1})^{\gamma} \cdot FL(e_i, e_i^*)}_{EOM}, \quad (14)$$

in which $\boldsymbol{e}$ and $\boldsymbol{e}^*$ denotes POE coding of a prediction and its GT label. $FL(\cdot)$ is focal loss [16] for angle classification. γ is hyperparameter to adjust the contribution of different bits in POE coding to total loss. $(N^{i-1})^{\gamma}$ is the EOM to distinguish different position of POE labels. The accurate prediction of higher significant bits is more important that of lower bits, therefore, EOM weighting is larger for high bits.

Next, the AOM in Eq. (13) is to determine the magnitude of angle loss. A good AOM should satisfy following properties:

Property 1: $g(r)$ is monotonically increasing w.r.t. the aspect ratio r. For objects with large aspect ratios, slight angular deviation would lead to a sharp drop in detection accuracy (see Fig. 7b) and thus require additional attention.

Property 2: $f(\Delta\theta)$ is monotonically increasing w.r.t the angular deviation. That is, a small angle loss should guarantee a small angular error, so that the model converges correctly.

Property 3: The gradient of $f(\Delta\theta)$ w.r.t. angular error is a monotonically increasing function. When the angular error is large, a large gradient is expected for fast convergence. Conversely, a small gradient is required to achieve accurate prediction as angular error is small.

On the basis of above considerations, the AOM is designed as show in Eq. (13). In Eq. (13), $r \in [1, +\infty]$. $\Delta\theta$ is the angular error, and $\Delta\theta \in [0, \pi)$. $g(r) = ln(r + e - 1) = 1$ when $r = 1$ for squre-like objects. As $r \uparrow$, $g(r) \uparrow$, $u \uparrow$, and $L_{\text{ANG}} \uparrow$, therefore the **Property 1** holds. When $\Delta\theta \downarrow$, $f(\Delta\theta) = (\Delta\theta - \sin\Delta\theta) \downarrow$, $L_{\text{ANG}} \downarrow$, and thus the **Property 2** is established. Since $f'(\Delta\theta) = \Delta\theta - \cos(\Delta\theta)$, when $\Delta\theta \downarrow$, $f'(\Delta\theta), \downarrow$, the **Property 3** is also satisfied. Therefore, $AOM(\cdot)$ in Eq. 13 is a good candidate to evaluate angular deviation.

The above two modules in ADW strategy bridge the inconsistency between the angle loss and real angle deviation, and help to achieve fast angular convergence and accurate prediction.

The overall loss of our TIOE-Det combines the above parts, which is denoted as follows:

$$L = \lambda_1 \cdot L_{\text{PHA}} + \lambda_2 \cdot L_{\text{IoU}} + \lambda_3 \cdot L_{\text{ANG}}, \qquad (15)$$

where L_{PHA} , L_{IoU}, L_{ANG} are the PHA prediction loss, HBB prediction loss, and angle loss, respectively. These loss items are balanced via parameters $\lambda_1, \lambda_2, \lambda_3$, ($\lambda_1$=$\lambda_2$=$\lambda_3$=1 in our experiments).

4. Experimental Setup

4.1. Datasets

Extensive experiments are conducted on multiple publicly available aerial image datasets, including DOTA [36], FAIR1M [37], DIOR-R [38], HRSC2016 [17], UCAS-AOD [39], UAV-ROD [40].

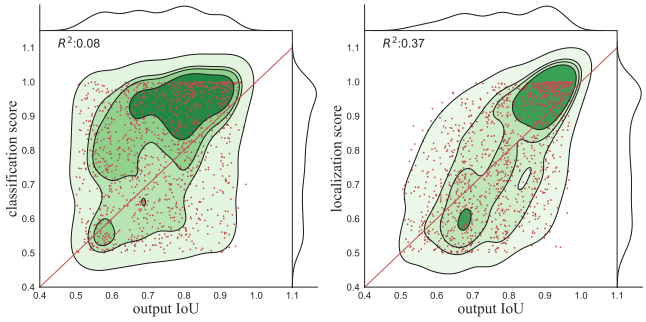
DOTA [36] is a large-scale aerial and satellite imagery datasets with oriented bounding box annotations. It contains 2806 aerial images with 188282 annotated instances. There are 15 categories including plane (PL), baseball diamond (BD), bridge (BR), ground track field (GTF), small vehicle (SV), large 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). The original size of images in the dataset ranges from about 800 $\times$ 800 to about 4,000 $\times$ 4,000 pixels.

FAIR1M [37] is a recent benchmark dataset for fine-grained object recognition in aerial imagery with more than 1 million instances and more than 15,000 images. All objects in the dataset are annotated to 37 categories by oriented bounding boxes, including Boeing 737, Boeing 777, Boeing 747, Boeing 787, Airbus A320, Airbus A220, Airbus A330, Airbus A350, COMAC C919, COMAC ARJ21, other-airplane, passenger ship, motorboat, fishing boat, tugboat, engineering ship, liquid cargo ship, dry cargo ship, warship, other-ship, small car, bus, cargo truck, dump truck, van, trailer, tractor, truck tractor, excavator, other-vehicle, baseball field, basketball court, football field, tennis court, roundabout, intersection, and bridge. The image width in FAIR1M ranges from 1000 to 10,000 pixels.

DIOR-R [38] is a large benchmark for object detection in remote sensing images, which contains 23,463 images and 192,518 instances. There are total 20 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). The size of images in the dataset is 800$\times$800 pixels.

HRSC2016 dataset [17] collects 1061 images from Google Earth for high resolution remote sensing ship detection. HRSC2016 contains lots of ships with large aspect ratios. The image size range from 300$\times$300 to 1500$\times$900. The total dataset is divided into training set, validation set, and test set, including 436, 181, and 444 images, respectively.

UCAS-AOD [39] is an aerial plane and car detection dataset. It contains 1510 images, including 1000 images for planes and 510 images for cars. UAV-ROD [40] is an aerial car dataset which contains 1150 images in the training set and 427 images in the test set.



(a) Baseline (b) TIOE-Det

Figure 8: The correlation between localization accuracy and corresponding confidence, the Pearson correlation coefficients are : (a) 0.08 for baseline model, and (b) 0.37 with our PHA prediction.

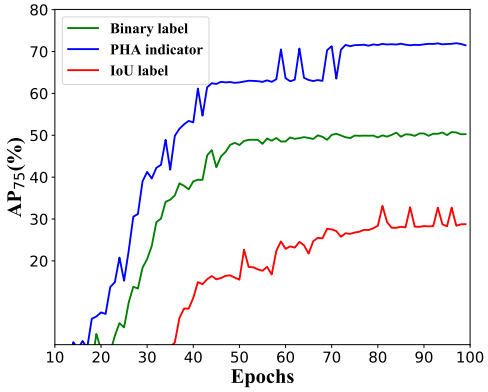


Figure 9: Comparison of high-precision detection performance with different labels.

Table 1: Evaluation of different components in PHA prediction.

Cls Pred.	IoU Pred.	PHA Pred.	BAL	AP_{75}	$AP_{50:90}$
✓				53.68	52.54
	✓			45.15	49.89
✓	✓			57.61	54.29
✓	✓	✓		65.33	56.01
		✓		67.17	56.50
		✓	✓	**71.74**	**57.28**

Table 2: Evaluation of different setting of PHA labels. * indicates using curriculum learning strategy.

Levels	1	3	5	10	100	5*
$AP_{50:90}$	52.2	54.1	55.9	53.3	49.8	**56.5**

4.2. Experimental Setting

All images are resized to 800×800 or 1024×1024 for training and testing in our experiments. Note that images in DOTA and FAIR1M are too large to be fed into the model directly, we crop images into patches of 1024×1024 with a stride of 512. We use the Adam optimizer for training, and the initial learning rate is set to 5×10^{-4}. The models are trained on RTX 3090 GPUs with batch size set to 8. The total training iterations are 600 epochs for HRSC2016, UCAS-AOD, and UAV-ROD. For large-scale remote sensing dataset DOTA, DIOR-R, and FAIR1M, models are trained for 300 epochs. Ablation studies are conducted on HRSC2016 which contains lots of ships with large aspect ratios. We use Random flip, rotation, and scaling for data augmentation.

We use the Average Precision (AP) and mean Average Precision (mAP) to evaluate the detection performance. Specifically, $AP_{50:90}$ means average precision over different IoU thresholds, from 0.5 to 0.9, step 0.1. $AP_{50:90}$ considers high IoU thresholds so it helps to measure high-precision detection performance. We employ the mean Average Orientation Error (mAOE°) to evaluate angular errors for orientation prediction.

5. Experimental Results

5.1. Ablation Study

5.1.1. Evaluation of PHA Prediction

Component-wise ablations of PHA. The component-wise experiments of PHA prediction are shown in Table 1. "Cls Pred." denote using classification branch, "IoU Pred." means adopting IoU branch, and "PHA Pred." means using PHA prediction branch. The baseline model with classification branch reaches the AP_{50} of 86.09%, AP_{75} of 53.68%, and $AP_{50:90}$ of 52.54%.

The variant that directly use IoU branch to replace classification branch leads to sharp performance drops of high-precision detection, AP_{75} drops by 8.53% and $AP_{50:90}$ by 2.65%. It shows that the only prediction of IoU could not work well. Then, combination of classification and IoU regression brings 3.93 points improvement on AP_{75} and 1.75 points on $AP_{50:90}$ compared with the baseline. We suggest that though binary classification is not accurate enough, it is more stable and easy to converge compared with direct IoU prediction. On this basis, when the PHA labels is adopted, the AP_{75} is improved by further 11.65% and $AP_{50:90}$ by 3.47%. Furthermore, after removing the classification branch, the model achieves a gain of 13.49 points on AP_{75} and 3.96 points on $AP_{50:90}$. The improvements confirms that binary classification labels harm the high-precision detection performance. Finally, balanced alignment loss alleviates the imbalance problem in PHA prediction, avoiding too many negative samples to dominate the PHA loss, so the performance is further improved to reach the AP_{75} of 71.74% and $AP_{50:90}$ of 57.28%. In total, our methods improves the AP_{75} and $AP_{50:90}$ of baseline by 18.06% and 4.74% , respectively.

We visualized the correlation between output confidence scores and regression accuracy of detections in Fig. 8. Illustrated in Fig. 8a, there is a weak correlation between classification score and IoU of detections of baseline. The Pearson correlation coefficient is just 0.08. Our method makes the confidence better represent localization accuracy, thereby reaching a Pearson correlation coefficient of 0.37 in Fig. 8b. PHA score helps

Table 3: Analysis of different hyperparameters of balanced alignment loss.

AP_{75} α / β	0.1	0.2	0.5	1.0
2.0	70.4	**71.7**	71.2	68.3
2.5	70.1	70.9	71.4	68.5
3.0	68.8	68.2	69.5	69.3

to achieve more reliable NMS procedure for high-precision detection.

Different PHA levels. We conducted experiments on the PHA label assign strategy to find optimal setting s. The experimental results are shown in Table 2. When the number of PHA levels is equal to 1, the IoU labels are binary just like the classification task, and it reached $AP_{50:90}$ of 52.2%. Whereas the variant with 5 levels achieves $AP_{50:90}$ of 55.9%, which is the best performance reported among all the levels. It shows that the fine-grained IoU intervals help to represent the localization accuracy of detections. As the number of total levels is increased to 100, the IoU interval is 0.05, which is similar to the continuous IoU prediction. As a result, $AP_{50:90}$ dramatically drops to 49.8%, and it is close to the only IoU prediction method (49.89% in Table 1). It further proves that it's hard to predict continuous IoU directly, and fine-grained PHA labels works better. Finally, the curriculum learning strategy achieves smooth model convergence, which improves performance to 56.5%.

As illustrated in Fig. 9, when $l = 2$, binary label in classification branch helps the model converge fast compared with IoU prediction. Direct prediction of continuous IoU hinders model convergence and cannot improve high-precision detection performance in early stages of training. PHA label provides accurate posterior localization information and improves network convergence, therefore achieves fast model convergence and accurate detections.

Hyperparameters in BAL. We further conducted experiments to find the optimal hyperparameters for balanced alignment loss (see Table 3). We found that the best performance would be obtained when $\alpha = 0.2$ and $\beta = 2$. Hyperparameter sensitivity experiments show that balanced alignment loss reports good performance improvements in many parameters. Obviously it is robust to different parameters within a reasonable range, and thus hyperparameters tuning for balanced alignment loss is not troublesome.

Table 4: Evaluation of different significant conditions in POE coding.

N	Reg	2	3	5	8	16
$AP_{50:90}$	52.54	54.03	54.21	53.21	**54.62**	53.76
mAOE	6.81	4.45	4.26	5.82	**4.08**	5.21

Table 5: Performance evaluation of components in ADW strategy.

POE	EOM	AOM	$AP_{50:90}$	mAOE
✓			52.71	5.73
✓	✓		53.35	5.06
✓		✓	54.21	4.63
✓	✓	✓	**54.62**	**4.08**

5.1.2. Evaluation of POE Strategy

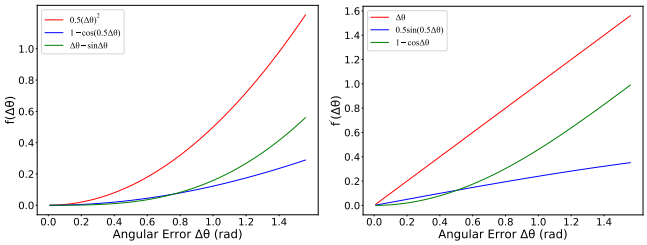
Different significant conditions. The experimental results in Table 4 show that different significant conditions N lead to different performances in POE coding. The baseline model adopts direct angle regression for oriented bounding box prediction. POE coding under different N all get better performance compared with baseline, which proves the superiority of our method. When $N = 8$, POE strategy improves $AP_{50:90}$ by 2.08 points and reduces mAOE by 2.73°. Note that the maximum angular error $\delta \approx 0.352°$ when $N = 8$, while $\delta \approx 0.288°$ when $N = 5$. However, detection performance is even dropped with the smaller theoretical angular error. We suggest that a smaller theoretical angular error means more angle intervals are divided, which makes it hard for the angle classification head to accurately discriminate the tiny angular error. Hence, there is a trade-off between detection accuracy and model convergence.

Table 6: Analysis of hyperparameter in EOM. $N = 8$ in POE coding.

γ	0	0.1	0.3	0.5	0.7	1.0
$AP_{50:90}$	52.71	52.83	**53.35**	48.65	41.35	33.57
mAOE	5.73	5.51	**5.06**	6.93	7.61	10.36

Component-wise Ablations in ADW strategy. We conduct experiments to evaluate the performance of angular distance weighting (ADW) strategy for training. The ADW strategy consists of two parts, angular offset metric (AOM) measures the angular error of predicted POE coding, and encoding offset metric (EOM) distinguishes different bits within POE coding. The experimental results are shown in the Table 5. Both EOM and AOM improve high-precision detection performance and reduce angular error of predictions. Specifically, EOM pays more attention to the high bits in POE encoding, thus making the training process more stable. AOM introduces constraints of angular error, gradient of angular offsets, and aspect ratios into angle classification loss, which helps to achieve faster convergence and accurate detections. Finally, ADW strategy improves $AP_{50:90}$ by 1.91% and reduces the mAOE by 1.65° in total.

Hyperparameters in EOM. In Eq. 14, γ is introduced into EOM to control the loss contribution of different bits in POE coding. Further, we conduct experiments to find the optimal hyperparameter γ. We set $N = 8$ in POE coding for fair comparison. As shown in Table 6, when $\gamma = 0.3$, TIOE-Det achieves $AP_{50:90}$ of 53.35% and mAOE of 5.06°, which is the best performance among the parameters compared. When $\gamma = 0$, EOM, like many current angle classification methods [22, 30], treats every bit in the POE encoding equally. In this way, the wrong prediction of high bits would lead to serious misjudgment of object orientation, thus achieving inferior performance. As γ increases, high bits in POE coding are gradually paid more attention. However, an excessively large γ would cause the loss contribution of high bits to dominate the angle loss. For example, when $\gamma = 1$, EOM weighting to different bits of 8-ary POE coding is $\{8^3, 8^2, 1\}$, which would cause the lower bits to be almost ignored. As a result, it in turn leads to sharp angular errors and inferior accuracy.



(a) (b)

Figure 10: Curves of (a) different $f(\Delta\theta)$ in AOM function and (b) the corresponding gradient.

Comparison with related methods. We compare the proposed POE strategy with other angle classification methods, and the results are shown in Table 7. 'Reg' is the baseline model that adopts angle regression to predict orientation of objects. It achieves $AP_{50:90}$ and mAOE of 52.54% and 6.81°, respectively. 'OneHot'[41] method divides angle range into 180 significant conditions equally, and adopts one-hot labels to represent the orientation. We suggest that too many angle classes make classification branch hard to converge, therefore it achieves inferior performance compared with baseline. CSL[22] introduces error tolerance for adjacent angle classes into one-hot labels, which improve $AP_{50:90}$ by 1.44% and mAOE by 2.43°. However, these two methods introduce a heavy classification head, resulting in a sharp increase in parameters and computational complexity of the model. Recent BCL and GCL [30] use larger interval division to reduce computational cost, but their detection performance is inferior to CSL. Our POE strategy achieves significant performance gains by introducing a small computational cost. It achieves $AP_{50:90}$ of 54.62% and mAOE of 4.08°, which is the best among the compared methods.

Evaluation of different AOMs. We compare the performance of different AOM candidates, and the experimental results are shown in Table 8. The AOM consists of $f(\Delta\theta)$ and $g(r)$, which introduce angular error and aspect ratio information into angle loss, respectively. $g(r)$ improves the loss contribution of large aspect ratio objects, and thus increases $AP_{50:90}$ by 0.17%, and

Table 7: Comparison with other angle classification methods.

Methods	Param.(M)	ΔParam.	GFLOPS	ΔGFLOPS	$AP_{50:90}$	mAOE
Reg	7.25779	——	16.82426	——	52.54	6.81
OneHot[41]	7.74055	+6.65%	18.37297	+9.21%	47.12	7.12
GCL[30]	7.26318	**+0.07%**	16.84157	**+0.11%**	50.67	6.95
BCL[30]	7.26318	**+0.07%**	16.84157	**+0.11%**	53.06	5.34
CSL[22]	7.74055	+6.65%	18.37297	+9.21%	53.98	4.38
POE (ours)	7.27937	+0.29%	16.89348	+0.41%	**54.62**	**4.08**

Figure 11: Detection results on HRSC2016 dataset.

Table 8: Performance evaluation of different AOMs.

$f(\Delta\theta)$	$g(r)$	$AP_{50:90}$	mAOE
——	——	53.35	5.06
——	$ln(r+e-1)$	53.52	4.83
$0.5\Delta\theta^2$	$ln(r+e-1)$	53.69	4.92
$1-\cos(0.5\Delta\theta)$	$ln(r+e-1)$	53.88	4.75
$\Delta\theta-\sin\Delta\theta$	$ln(r+e-1)$	**54.21**	**4.63**

(a) UCAS-AOD

(b) UAV-ROD

Figure 12: Illustrations of detections on (a)UCAS-AOD dataset and (b) UAV-ROD daataset.

improves mAOE by $0.23°$. Furthermore, we compared the performance of different $f(\Delta\theta)$. All candidate of $f(\Delta\theta)$ satisfy **Property 2** and **Property 3** in Section 3.3.2, that is, both $f(\Delta\theta)$ and its gradient are monotonically increasing with angular error (as shown in Fig. 10). Among them, $f(\Delta\theta)=\Delta\theta-\sin\Delta\theta$ achieves $AP_{50:90}$ of 54.21% and mAOE of 4.63%, which is the best among compared functions.

5.2. Comparison with state-of-the-art methods

5.2.1. Results on HRSC2016

The HRSC2016 dataset [17] contains a large number of remote sensing ships with large aspect ratios. We compare the performance of state-of-the-art methods on HRSC2016 dataset. As shown in Table 9, TIOE-Det achieves the mAP of 90.16%, which is the best performance among the compared methods. POE strategy helps to accurately predict the orientation of ships with large aspect ratios, while PHA scores allows for confident selection of high-quality detections. Some detection results are shown in Fig. 11.

Table 9: Comparisons with other methods on HRSC2016 dataset.

Methods	Type	Size	mAP
RRPN [21]	two-stage	800×800	79.08
R^2PN [42]	two-stage	—	79.60
RetinaNet [16]	one-stage	416×416	80.81
RRD [43]	one-stage	384×384	84.30
RoI Trans. [20]	two-stage	512×800	86.20
RSDet [44]	one-stage	800×800	86.50
Gliding Vertex [9]	two-stage	512×800	88.20
OPLD [45]	two-stage	1024×1333	88.44
DAL [24]	one-stage	416×416	88.95
R^3Det [7]	one-stage	800×800	89.26
DCL [30]	one-stage	800×800	89.46
RIDet [12]	one-stage	800×800	89.63
CFC-Net [8]	one-stage	800×800	89.70
GWD [11]	one-stage	800×800	89.85
AProNet [46]	two-stage	512×800	90.03
TIOE-Det (Ours)	one-stage	800×800	**90.16**

Table 10: Comparisons with different methods on UCAS-AOD dataset.

Methods	Size	Car	Airplane	mAP
YOLOv3 [6]	800	74.63	89.52	82.08
RetinaNet [16]	800	84.64	**90.51**	87.57
FR-O [36]	800	86.87	89.86	88.36
RoI Transformer [20]	800	87.99	89.90	88.95
RIDet [12]	800	88.50	89.96	89.23
SLA [13]	800	88.57	90.30	89.44
TIOE-Det(ours)	800	**88.83**	90.15	**89.49**

5.2.2. *Results on UCAS-AOD and UAV-AOD*

UCAS-AOD[39] and UAV-ROD[40] are aerial image datasets with OBB annotations. We conducted experiments on the two datasets, and experimental results are shown in Table 10 and Table 11. UCAS-AOD [39] contains a large number of small-scale cars that are difficult to detect. TIOE-Det achieves accurate car detection with an AP of 88.83%, which is the best among the compared methods. Noting that the detection performance of airplane is slightly lower than that of RetinaNet[16], we suggest that airplanes are annotated with square-like boxes, and thus the POE strategy does not bring a significant performance gain. UAV-ROD [40] is a recent dataset of drone aerial imagery. AP here denotes average precision over different IoU thresholds, from 0.50 to 0.95, step 0.05. AP_{50} of our method is slightly inferior to models such as TS^4Net[40]. We suggest that our method focuses more on improving high-precision detection performance. As a result, TIOE-Det achieves the AP of 77.93% and AP_{75} of 89.64%, which are the best performance among compared models. Some detection results on UCAS-AOD and UAV-ROD are shown in Fig. 12.

Table 11: Comparisons with different methods on UAV-ROD dataset.

Methods	AP	AP_{75}	AP_{50}
RetinaNet [16]	71.46	85.88	97.68
Faster R-CNN [3]	75.79	86.38	98.07
TS^4Net [40]	76.75	88.17	**98.10**
TIOE-Det(ours)	**77.93**	**89.64**	97.89

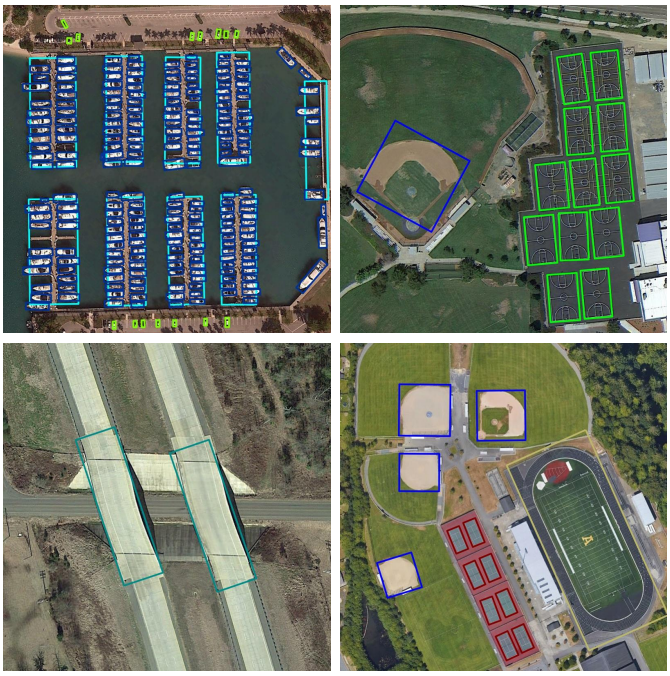


Figure 13: Illustrations of some detections on DIOR-R dataset.

5.2.3. *Results on DIOR-R*

DIOR dataset[38] is a large-scale public dataset for object detection in remote sensing images. DIOR-R shares the same image with the original version DIOR and introduces additional oriented bounding box annotations. As shown in Table 12,

Table 12: Comparison with other models on DIOR dataset.

Methods	APL	APO	BF	BC	BR	CH	ESA	ETS	DAM	GF	GTF	HA	OP	SH	STA	STO	TC	TS	VE	WM	mAP
RetinaNet[16]	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[47]	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
DAL[24]	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
Faster RCNN[3]	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[9]	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
RIDet[12]	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[8]	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
TIOE-Det	**68.65**	28.62	76.68	**84.76**	**39.32**	72.35	72.66	63.87	20.36	75.19	77.41	40.63	47.48	**82.61**	72.58	70.33	81.93	47.86	**52.06**	64.24	**61.98**

Table 13: Performance comparison with state-of-the-arts on the DOTA dataset. The items with red and blue colors indicate the best and second-best results of each column, respectively. 'Ms' means using multi-scale training and testing.

	Methods	Ms	PL	BD	BR	GTF	SV	LV	SH	TC	BC	ST	SBF	RA	HA	SP	HC	mAP
Two Stage	FR-O[36]		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[48]	✓	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
	RoI Trans.[20]	✓	88.64	78.52	43.44	75.92	68.81	73.68	83.59	90.74	77.27	81.46	58.39	53.54	62.83	58.93	47.67	69.56
	CAD-Net[49]	✓	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[19]	✓	**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[9]		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[50]	✓	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[51]	✓	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[22]	✓	**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[45]	✓	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
	AProNet[46]	✓	88.77	84.95	55.27	78.40	76.65	78.54	88.45	90.83	86.56	87.01	65.62	70.29	75.43	78.17	67.28	78.16
One Stage	PIoU[35]		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[8]	✓	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^3Det[7]	✓	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
	DAL[24]		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[13]	✓	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[30]	✓	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[11]	✓	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[12]	✓	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[]	✓	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
	KLD[14]	✓	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**
	TIOE-Det(ours)	✓	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**

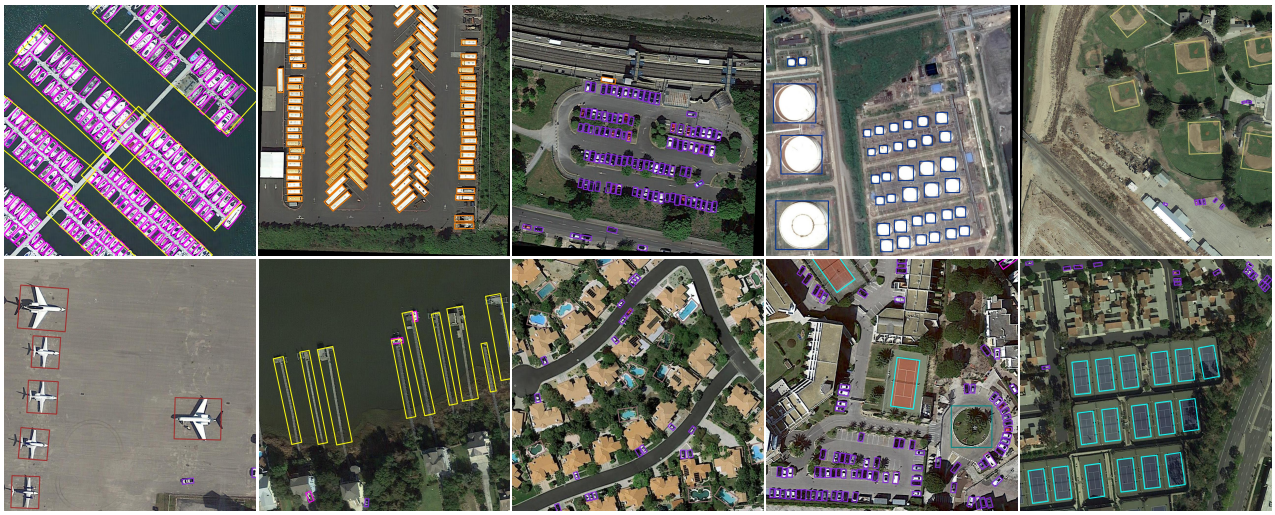


Figure 14: Visualization of detections on DOTA dataset.

Figure 15: Detection results on FAIR1M dataset.

TIOE-Det achieves the mAP of 61.98%, which outperforms many existing methods. Some detections results on DIOR-R are shown in Fig. 13.

5.2.4. Results on DOTA

DOTA[36] is the most commonly used datasets for oriented object detection in remote sensing images. We have reported detection performance of our model on DOTA in Table 13. TIOE-Det achieves the mAP of 78.69%, which outperforms many recent state-of-the-art methos such as AProNet[46]. Some detections on DOTA dataset are visualized in Fig. 14. Our model achieves superior performance on categories with large aspect ratios and densely arranged objects, such as bridge(BR), small vehicle(SV), large vehicle(LV). It shows that TIOE-Det achieves accurate orientation prediction and provides reliable high-precision detections.

5.2.5. Results on FAIR1M

FAIR1M is a recent large-scale dataset for fine-grained object detection in remote sensing imagery. Many classes have high inter-class similarity, such as Boeing 737, Boeing 747, Boeing 777. It is a challenging task to detect objects and identify their categories. For a fair comparison, we reproduce some advanced detectors on FAIR1M dataset in Table 14. Generally, the current two-stage detectors adopt RoI align[53] to extract discriminative features, which greatly improves the accuracy of fine-grained object recognition. Therefore, two-stage detectors in Table 14 (Faster RCNN[3], RoI Transformer[20], Gliding Vertex[9]) achieve better performance than one-stage detectors (such as FCOS[52], CFC-Net[8]). TIOE-Det achieves the mAP of 35.16%, which outperforms the compared one-stage detectors and even some two-stage detectors in Table 14. After using multi-scale training and testing, our method achieves the mAP of 43.87%. Visualization of some detection results is shown in Fig. 15.

5.3. Analysis and Discussion

Our TIOE-Det achieves state-of-the-art performance on multiple datasets. The modules we propose, PHA label, BAL, POE coding, and ADW strategy achieve stable performance gains. Specifically, it can be seen from Fig. 8 and Table Tab. 1 that the PHA label greatly improves the performance of high-precision

Table 14: Comparison with some recent models on FAIR1M dataset. The items with red and blue colors indicate the best and second-best results of each column, respectively. * denotes using multi-scale training and testing.

Method	FCOS[52]	RetinaNet[16]	DAL[24]	RIDet[12]	Faster RCNN[3]	CFC-Net[8]	Gliding Vertex[9]	RoI Trans.[20]	TIOE-Det	TIOE-Det*
mAP	23.70	27.67	29.00	31.58	33.70	34.31	35.86	**38.27**	35.16	**43.87**
Boeing 737	10.34	35.01	32.53	28.25	36.05	30.89	36.32	35.84	**37.62**	**41.65**
Boeing 747	43.54	83.72	74.39	80.62	85.19	83.87	82.61	82.74	**86.71**	**85.39**
Boeing 777	5.96	12.64	**13.14**	12.92	12.45	10.72	11.29	12.81	11.06	**17.53**
Boeing 787	13.67	36.68	39.91	45.28	45.35	38.60	**48.69**	43.90	46.32	**48.53**
C919	0.00	1.44	2.11	0.15	15.45	5.67	**24.48**	15.77	0.00	**24.32**
A220	11.71	45.44	41.32	39.89	**49.50**	42.44	**50.01**	48.68	48.75	45.92
A321	3.95	64.95	58.38	53.69	63.16	50.68	65.27	67.35	**68.49**	**70.21**
A330	15.03	58.52	44.59	62.80	65.89	55.13	**69.98**	65.56	**72.51**	63.09
A350	14.20	71.45	54.88	55.27	62.69	59.20	65.18	62.92	**78.19**	**77.21**
ARJ21	13.75	3.60	1.57	8.53	31.25	5.30	33.24	**33.60**	8.62	**45.32**
passenger ship	10.65	3.83	9.90	6.11	6.24	7.19	8.92	**15.20**	3.73	**12.96**
motorboat	46.21	22.03	53.04	55.20	44.37	**63.38**	52.04	58.04	58.45	**65.39**
fishing boat	**9.59**	2.12	5.71	5.49	3.71	8.72	5.11	9.37	5.12	**10.29**
tugboat	19.81	13.34	21.08	30.15	26.05	19.70	28.49	**30.17**	**30.51**	29.85
engineering ship	**13.24**	9.11	7.11	5.84	6.88	7.67	9.73	10.87	10.38	**11.21**
liquid cargo ship	12.92	4.37	12.05	17.21	9.50	**21.23**	15.67	19.28	5.56	**24.00**
dry cargo ship	**35.08**	14.49	28.41	29.58	17.78	30.54	26.75	33.02	18.71	**36.01**
warship	20.75	3.81	11.91	14.47	6.37	23.21	13.67	**24.90**	2.52	**32.35**
small car	42.56	41.91	48.05	52.73	51.44	62.43	49.53	57.73	**65.89**	**74.86**
bus	15.55	5.55	7.71	15.27	21.00	**34.50**	22.04	31.23	4.73	**53.31**
cargo truck	31.72	20.69	25.04	30.32	32.89	41.15	36.69	**42.46**	36.29	**49.93**
dump truck	23.90	16.54	22.82	29.50	40.04	42.18	39.52	**45.26**	41.31	**55.78**
van	34.59	34.09	43.26	45.01	45.96	51.65	43.65	54.49	**65.89**	**75.08**
trailer	12.14	0.33	2.48	3.82	7.82	11.41	11.65	**15.54**	0.53	**19.62**
tractor	1.07	0.36	1.03	0.05	**3.77**	1.69	2.90	3.55	0.18	**4.00**
excavator	7.90	0.52	5.06	5.03	9.28	10.26	12.49	**12.78**	9.83	**16.62**
truck tractor	1.09	0.01	0.55	0.53	1.71	0.71	**3.66**	**2.59**	0.10	2.18
basketball court	23.09	22.28	38.76	37.47	39.92	40.21	39.85	42.87	**50.23**	**50.90**
tennis court	74.76	78.62	75.37	77.78	76.97	79.41	76.98	78.40	**80.23**	**83.97**
football field	49.64	59.46	46.10	52.69	52.36	58.01	50.79	59.30	**60.70**	**65.29**
baseball field	82.90	86.46	84.66	85.63	**87.56**	84.34	86.85	86.60	**88.57**	85.96
intersection	55.14	57.33	44.06	51.41	57.11	51.98	58.59	58.18	**65.07**	**63.36**
roundabout	**26.46**	20.30	13.96	17.05	**22.28**	18.22	20.49	19.34	21.02	21.50
bridge	**22.79**	9.89	15.08	17.96	7.75	14.31	16.21	20.76	11.94	**28.01**

object detection. We suggest that PHA label efficiently selects high-quality predictions, thereby avoiding the inconsistency between classification scores and localization accuracy during inference. BAL alleviates the imbalance problem in PHA prediction for better performance.

The proposed POE strategy solves the periodicity of angle prediction. However, it introduces the inaccurate angular distance measurement. Therefore, AWD strategy applies AOM and EOM to calculate the angle deviation during training. As shown in Table 5, the ADW strategy significantly reduces the angle prediction error, thereby improving the high-precision detection performance. Also, the flexible POE coding allows for a trade-off between accuracy and speed of the model as shown in Tab. 6 and Tab. 7.

6. Conclusion

High-precision oriented object detection has always been a challenging task. Current mainstream rotation detectors suffer from unreliable detection results and inaccurate orientation prediction. In this paper, we design TIOE-Det for high-precision object detection in remote sensing images. TIOE-Det employs two novel modules: posterior hierarchical alignment (PHA) branch and progressive direction estimation (POE) strategy. Specifically, PHA branch predicts PHA score based on localization accuracy for high-quality detection selection. The POE strategy discretizes the object orientation and adopts interpretable progressive coding to represent orientation of the target. Furthermore, we designed a balanced alignment loss and an angular deviation weighting strategy during loss calculation for two proposed module. TIOE-Det achieves superior performance on multiple publicly available remote sensing datasets. Extensive experimental results demonstrate the effectiveness of our method.

References

[1] Y. Li, X. Sun, H. Wang, H. Sun, X. Li, Automatic target detection in high-resolution remote sensing images using a contour-based spatial model, IEEE Geoscience and Remote Sensing Letters 9 (5) (2012) 886–890.

[2] C. Zhu, H. Zhou, R. Wang, J. Guo, A novel hierarchical method of ship detection from spaceborne optical image based on shape and texture features, IEEE Transactions on geoscience and remote sensing 48 (9) (2010) 3446–3456.

[3] S. Ren, K. He, R. Girshick, J. Sun, Faster r-cnn: Towards real-time object detection with region proposal networks, IEEE transactions on pattern analysis and machine intelligence 39 (6) (2016) 1137–1149.

[4] J. Redmon, S. Divvala, R. Girshick, A. Farhadi, You only look once: Unified, real-time object detection, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 779–788.

[5] J. Redmon, A. Farhadi, Yolo9000: better, faster, stronger, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2017, pp. 7263–7271.

[6] J. Redmon, A. Farhadi, Yolov3: An incremental improvement, arXiv preprint arXiv:1804.02767.

[7] X. Yang, J. Yan, Z. Feng, T. He, R3det: Refined single-stage detector with feature refinement for rotating object, Proceedings of the AAAI Conference on Artificial Intelligence 35 (4) (2021) 3163–3171.

[8] Q. Ming, L. Miao, Z. Zhou, Y. Dong, Cfc-net: A critical feature capturing network for arbitrary-oriented object detection in remote-sensing images, IEEE Transactions on Geoscience and Remote Sensing 60 (2022) 1–14. doi:10.1109/TGRS.2021.3095186.

[9] Y. Xu, M. Fu, Q. Wang, Y. Wang, K. Chen, G.-S. Xia, X. Bai, Gliding vertex on the horizontal bounding box for multi-oriented object detection, IEEE transactions on pattern analysis and machine intelligence.

[10] J. Han, J. Ding, J. Li, G. S. Xia, Align deep features for oriented object detection, IEEE Transactions on Geoscience and Remote Sensing (2021) 1–11doi:10.1109/TGRS.2021.3062048.

[11] X. Yang, J. Yan, Q. Ming, W. Wang, X. Zhang, Q. Tian, Rethinking rotated object detection with gaussian wasserstein distance loss, arXiv preprint arXiv:2101.11952.

[12] Q. Ming, L. Miao, Z. Zhou, X. Yang, Y. Dong, Optimization for arbitrary-oriented object detection via representation invariance loss, IEEE Geoscience and Remote Sensing Letters 19 (2022) 1–5. doi:10.1109/LGRS.2021.3115110.

[13] Q. Ming, L. Miao, Z. Zhou, J. Song, X. Yang, Sparse label assignment for oriented object detection in aerial images, Remote Sensing 13 (14) (2021) 2664.

[14] X. Yang, X. Yang, J. Yang, Q. Ming, W. Wang, Q. Tian, J. Yan, Learning high-precision bounding box for rotated object detection via kullback-leibler divergence, arXiv preprint arXiv:2106.01883.

[15] J. Han, J. Ding, N. Xue, G.-S. Xia, Redet: A rotation-equivariant detector for aerial object detection, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2021, pp. 2786–2795.

[16] T.-Y. Lin, P. Goyal, R. Girshick, K. He, P. Dollár, Focal loss for dense object detection, in: Proceedings of the IEEE international conference on computer vision, 2017, pp. 2980–2988.

[17] Z. Liu, L. Yuan, L. Weng, Y. Yang, A high resolution optical satellite image dataset for ship recognition and some new baselines, in: Proceedings of the International Conference on Pattern Recognition Applications and Methods, Vol. 2, 2017, pp. 324–331.

[18] R. Girshick, Fast r-cnn, in: Proceedings of the IEEE international conference on computer vision, 2015, pp. 1440–1448.

[19] X. Yang, J. Yang, J. Yan, Y. Zhang, T. Zhang, Z. Guo, X. Sun, K. Fu, Scrdet: Towards more robust detection for small, cluttered and rotated objects, in: Proceedings of the IEEE International Conference on Computer Vision, 2019, pp. 8232–8241.

[20] J. Ding, N. Xue, Y. Long, G.-S. Xia, Q. Lu, Learning roi transformer for oriented object detection in aerial images, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2019, pp. 2849–2858.

[21] J. Ma, W. Shao, H. Ye, L. Wang, H. Wang, Y. Zheng, X. Xue, Arbitrary-oriented scene text detection via rotation proposals, IEEE Transactions on Multimedia 20 (11) (2018) 3111–3122.

[22] X. Yang, J. Yan, Arbitrary-oriented object detection with circular smooth label, in: European Conference on Computer Vision, Springer, 2020, pp. 677–694.

[23] L. Zhu, Z. Xie, L. Liu, B. Tao, W. Tao, Iou-uniform r-cnn: Breaking through the limitations of rpn, Pattern Recognition 112 (2021) 107816.

[24] Q. Ming, Z. Zhou, L. Miao, H. Zhang, L. Li, Dynamic anchor learning for arbitrary-oriented object detection, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 35, 2021, pp. 2355–2363.

[25] B. Jiang, R. Luo, J. Mao, T. Xiao, Y. Jiang, Acquisition of localization confidence for accurate object detection, in: Proceedings of the European Conference on Computer Vision (ECCV), 2018, pp. 784–799.

[26] H. Li, Z. Wu, C. Zhu, C. Xiong, R. Socher, L. S. Davis, Learning from noisy anchors for one-stage object detection, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 10588–10597.

[27] H. Zhang, Y. Wang, F. Dayoub, N. Sunderhauf, Varifocalnet: An iou-aware dense object detector, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 8514–8523.

[28] Y. He, X. Zhang, M. Savvides, K. Kitani, Softer-nms: Rethinking bounding box regression for accurate object detection, arXiv preprint arXiv:1809.08545 2 (3).

[29] J. Choi, D. Chun, H. Kim, H.-J. Lee, Gaussian yolov3: An accurate and fast object detector using localization uncertainty for autonomous driving, in: Proceedings of the IEEE International Conference on Computer Vision, 2019, pp. 502–511.

[30] X. Yang, L. Hou, Y. Zhou, W. Wang, J. Yan, Dense label encoding for boundary discontinuity free rotation detection, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021.

[31] C. Wu, Y. Chen, J. Luo, C.-C. Su, A. Dawane, B. Hanzra, Z. Deng, B. Liu, J. Z. Wang, C.-h. Kuo, Mebow: Monocular estimation of body orientation in the wild, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 3451–3461.

[32] C. Zhang, B. Xiong, X. Li, G. Kuang, Aspect-ratio-guided detection for oriented objects in remote sensing images, IEEE Geoscience and Remote Sensing Letters 19 (2021) 1–5.

[33] Y. Zhu, J. Du, X. Wu, Adaptive period embedding for representing oriented objects in aerial images, IEEE Transactions on Geoscience and Remote Sensing 58 (10) (2020) 7247–7257.

[34] S. Wu, X. Li, X. Wang, Iou-aware single-stage object detector for accurate localization, Image and Vision Computing (2020) 103911.

[35] Z. Chen, K. Chen, W. Lin, J. See, H. Yu, Y. Ke, C. Yang, Piou loss: Towards accurate oriented object detection in complex environments, in: European Conference on Computer Vision, Springer, 2020, pp. 195–211.

[36] G.-S. Xia, X. Bai, J. Ding, Z. Zhu, S. Belongie, J. Luo, M. Datcu, M. Pelillo, L. Zhang, Dota: A large-scale dataset for object detection in aerial images, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 3974–3983.

[37] X. Sun, P. Wang, Z. Yan, F. Xu, R. Wang, W. Diao, J. Chen, J. Li, Y. Feng, T. Xu, et al., Fair1m: A benchmark dataset for fine-grained object recognition in high-resolution remote sensing imagery, ISPRS Journal of Photogrammetry and Remote Sensing 184 (2022) 116–130.

[38] K. Li, G. Wan, G. Cheng, L. Meng, J. Han, Object detection in optical remote sensing images: A survey and a new benchmark, ISPRS Journal of Photogrammetry and Remote Sensing 159 (2020) 296–307.

[39] H. Zhu, X. Chen, W. Dai, K. Fu, Q. Ye, J. Jiao, Orientation robust object detection in aerial images using deep convolutional neural network, in: 2015 IEEE International Conference on Image Processing (ICIP), IEEE, 2015, pp. 3735–3739.

[40] J. Zhou, K. Feng, W. Li, J. Han, F. Pan, Ts4net: Two-stage sample selective strategy for rotating object detection, Neurocomputing.

[41] H. A. Rowley, S. Baluja, T. Kanade, Rotation invariant neural network-based face detection, in: Proceedings. 1998 IEEE computer society conference on computer vision and pattern recognition (Cat. No. 98CB36231), IEEE, 1998, pp. 38–44.

[42] Z. Zhang, W. Guo, S. Zhu, W. Yu, Toward arbitrary-oriented ship detection with rotated region proposal and discrimination networks, IEEE Geoscience and Remote Sensing Letters 15 (11) (2018) 1745–1749.

[43] M. Liao, Z. Zhu, B. Shi, G.-s. Xia, X. Bai, Rotation-sensitive regression for oriented scene text detection, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2018, pp. 5909–5918.

[44] W. Qian, X. Yang, S. Peng, Y. Guo, J. Yan, Learning modulated loss for rotated object detection, in: Proceedings of the AAAI Conference on Artificial Intelligence, 2021.

[45] Q. Song, F. Yang, L. Yang, C. Liu, M. Hu, L. Xia, Learning point-guided localization for detection in remote sensing images, IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing.

[46] X. Zheng, W. Zhang, L. Huan, J. Gong, H. Zhang, Apronet: Detecting objects with precise orientation from aerial images, ISPRS Journal of Pho-

togrammetry and Remote Sensing 181 (2021) 99–112.

[47] J. Yu, Y. Jiang, Z. Wang, Z. Cao, T. Huang, Unitbox: An advanced object detection network, in: Proceedings of the 24th ACM international conference on Multimedia, 2016, pp. 516–520.

[48] S. M. Azimi, E. Vig, R. Bahmanyar, M. Körner, P. Reinartz, Towards multi-class object detection in unconstrained remote sensing imagery, in: Asian conference on computer vision, Springer, 2018, pp. 150–165.

[49] G. Zhang, S. Lu, W. Zhang, Cad-net: A context-aware detection network for objects in remote sensing imagery, IEEE Transactions on Geoscience and Remote Sensing 57 (12) (2019) 10015–10024.

[50] J. Wang, J. Ding, H. Guo, W. Cheng, T. Pan, W. Yang, Mask obb: A semantic attention-based mask oriented bounding box representation for multi-category object detection in aerial images, Remote Sensing 11 (24) (2019) 2930.

[51] J. Wang, W. Yang, H.-C. Li, H. Zhang, G.-S. Xia, Learning center probability map for detecting objects in aerial images, IEEE Transactions on Geoscience and Remote Sensing 59 (5) (2020) 4307–4323.

[52] Z. Tian, C. Shen, H. Chen, T. He, Fcos: Fully convolutional one-stage object detection, in: Proceedings of the IEEE/CVF international conference on computer vision, 2019, pp. 9627–9636.

[53] K. He, G. Gkioxari, P. Dollár, R. Girshick, Mask r-cnn, in: Proceedings of the IEEE international conference on computer vision, 2017, pp. 2961–2969.