electronics

MDPI

Article

Precise Orientation Estimation for Rotated Object Detection Based on a Unit Vector Coding Approach

Chi-Yi Tsai * and Wei-Chuan Lin

Department of Electrical and Computer Engineering, Tamkang University, 151 Yingzhuan Road, Tamshui District, New Taipei City 251, Taiwan; 609460109@o365.tku.edu.tw
* Correspondence: chiyi_tsai@mail.tku.edu.tw

Abstract: Existing rotated object detection methods usually use angular parameters to represent the object orientation. However, due to the symmetry and periodicity of these angular parameters, a well-known boundary discontinuity problem often results. More specifically, when the object orientation angle approaches the periodic boundary, the predicted angle may change rapidly and adversely affect model training. To address this problem, this paper introduces a new method that can effectively solve the boundary discontinuity problem related to angle parameters in rotated object detection. Our approach involves a novel vector-based encoding and decoding technique for angular parameters, and a cosine distance loss function for angular accuracy evaluation. By utilizing the characteristics of unit vectors and cosine similarity functions, our method parameterizes the orientation angle as components of the unit vector during the encoding process and redefines the orientation angle prediction task as a vector prediction problem, effectively avoiding the boundary discontinuity problem. The proposed method achieved a mean average precision (mAP) of 87.48% and an average cosine similarity (CS) of 0.997 on the MVTec test set. It also achieved an mAP score of 90.54% on the HRSC2016 test set, which is better than several existing state-of-the-art methods and proves its accuracy and effectiveness.

Keywords: rotated object detection; orientation angle prediction; angle parameterization; cosine distance; loss function

check for updates

Citation: Tsai, C.-Y.; Lin, W.-C. Precise Orientation Estimation for Rotated Object Detection Based on a Unit Vector Coding Approach. *Electronics* **2024**, *13*, 4402. https://doi.org/10.3390/electronics13224402

Academic Editor: Silvia Liberata Ullo

Received: 15 October 2024
Revised: 3 November 2024
Accepted: 8 November 2024
Published: 10 November 2024

Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

Object detection is a key task in the field of computer vision, aiming to identify and locate objects in images. In recent years, with the advancement of deep learning technology, object detection methods based on Convolutional Neural Networks (CNNs) have made significant progress and have been widely used in various fields. Rotated object detection is a derivative task of traditional object detection methods, which has important value in certain application fields, such as remote sensing image detection [1,2], text recognition [3,4], industrial applications [5,6], 3D object detection [7], and object segmentation [8,9]. Rotated object detection can provide users with a more comprehensive understanding of the orientation and actual size of detected objects in images.

Rotated objects are usually distributed in images at arbitrary orientation angles ranging from 0° to 360°, and the task of rotated object detection is to predict the Rotated Bounding Box (RBB) that best fits the object. Figure 1 illustrates different bounding box representations. As shown in Figure 1a, when using a Horizontal Bounding Box (HBB) to detect an object, if the object is not vertically or horizontally aligned with the image, redundant background information will be included in the bounding box. When objects in an image are closely spaced, HBBs may fail to accurately match the objects [1]. To address these issues, rotated object detection methods were developed. As shown in Figure 1b, several neural network-based rotated object detection methods [1,2,10–12] use five parameters to represent the RBB with box rotation information, providing a more precise object

bounding box, but without its orientation information. Subsequently, refs. [5,13] expanded the application of rotated object detection, not only maintaining the original rotated object detection task, but also estimating its orientation at the same time, as shown in Figure 1c.

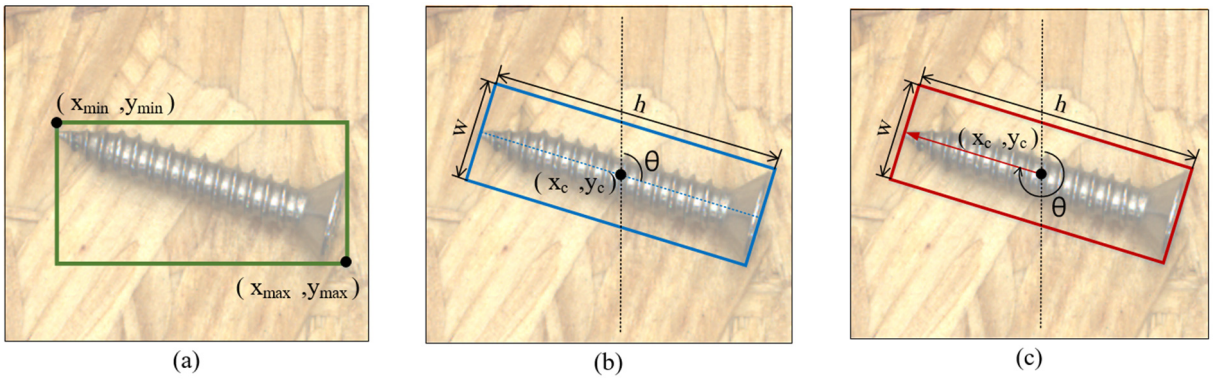
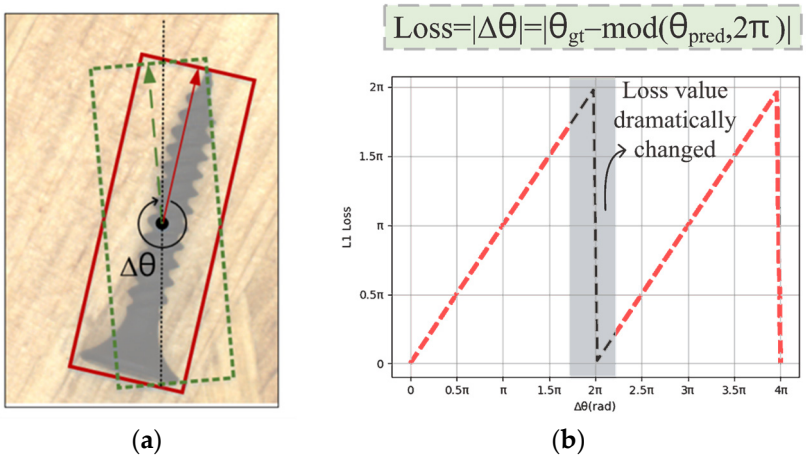


Figure 1. Example of using different parameters to describe the bounding box of an object: (**a**) The HBB representation uses four parameters (x_{min}, y_{min}, x_{max}, y_{max}); (**b**) the RBB with box rotation information uses five parameters (x_c, y_c, w, h, θ), where θ is the rotation angle of the bounding box; (**c**) The RBB with object orientation information also uses five parameters (x_c, y_c, w, h, θ), where θ is the orientation angle of the detected object.

Figure 2 shows the boundary discontinuity problem caused by the periodicity of angle parameters in the rotated object detection task. This problem will cause drastic changes in the loss function under certain conditions, thus affecting the stability of the model's convergence during training and reducing its performance [14,15]. Therefore, boundary discontinuity has become a key issue in rotating object detection. In order to solve this problem, some studies [1,16,17] try to mitigate the sharp changes in the loss function by limiting or changing the calculation of the loss function. For example, [1] proposed an Intersection-over-Union (IoU) smooth L1 loss to smooth the regression branch loss near the boundary, while [16,17] converted the bounding box to a 2D Gaussian distribution to reduce the sensitivity of the model to angle changes during training. Although these designs mitigate the impact of boundary discontinuities on model performance and are proven to improve detection accuracy, they do not fundamentally solve the boundary discontinuity problem due to the inherent characteristics of the angle parameters in the definition method.



(**a**) (**b**)

Figure 2. Illustration of the boundary discontinuity problem: (**a**) When the predicted rotation angle (the dashed arrow) approaches the corresponding ground truth (solid arrow), boundary discontinuity occurs; (**b**) the loss function shows sharp changes at the period boundary of the rotation angle error.

Recently published methods [15,18,19] proposed to use an encoding–decoding mechanism to convert the angle parameter prediction task into a classification task, thereby imparting continuity to the angular parameters and effectively solving the boundary discontinuity problem. However, this encoding–decoding method increases the number of model parameters, which affects the computational load of the network and thus reduces the detection speed. Therefore, avoiding boundary discontinuities without affecting detection efficiency is still a challenge worthy of further exploration.

In this paper, we propose a vector-based encoding–decoding method for rotated object detection, aiming to solve the boundary discontinuity problem. The proposed method combines Unit Vector Coding (UVC) to parameterize object orientation and cosine distance loss (CDL) to evaluate orientation angle estimation accuracy. The main contributions of this study are as follows:

(1) We propose a novel UVC encoding and decoding method that parameterizes object orientation through vector components. The encoded parameters exhibit continuous and reversible characteristics, thereby overcoming the boundary discontinuity and improving the accuracy of object orientation estimation.
(2) We propose a novel CDL function as the loss function of the orientation angle prediction branch in model training to evaluate the predicted angle of rotated objects. Experimental results show that the design of this loss function significantly improves the accuracy of rotated object detection tasks.

On the rotated object detection task of the MVTec screws dataset [20], our method achieves a high mAP of 87.48% at 53.2 frames per second (FPS), enabling real-time rotated object detection. Furthermore, it achieves an mAP of 90.54% on the High-Resolution Ship Collections 2016 (HRSC2016) dataset [21], verifying that the proposed method can improve the detection performance of different rotated object detection tasks.

The remaining sections of this paper are organized as follows: Section 2 provides a review of related research, focusing on rotated object detection, angle encoding, and loss function design. In Section 3, we present the proposed UVC encoding–decoding and CDL function in detail. Section 4 presents experimental results, demonstrating the detection performance of the proposed rotated object detection model on existing public datasets, thereby verifying the impact of the proposed method on the accuracy of object orientation estimation. In Section 5, we summarize the main contributions of this paper and discuss potential directions for future research.

2. Related Research

In this section, we review the literature related to this research, including rotated object detection, parameterization of RBB, and boundary discontinuity problem.

2.1. Rotated Object Detection

In the rotated object detection task [20–22], the object bounding box is usually represented as a RBB in the image, as shown in Figure 1b,c. Several rotated object detection methods are extensions of the classic object detection network [23–28], where the parameters of a HBB (x, y, w, h) are extended to those of a RBB (x, y, w, h, θ). This section introduces the methods for performing rotated object detection and discusses bounding box representations used in the literature.

The existing rotated object detection methods can be divided into two-stage and one-stage methods. In the two-stage object detection methods [1,3,12], the feature maps from the first-stage neural network are input into a Region Proposal Network (RPN), which is based on predefined anchor boxes to capture candidate regions that may contain objects. In the second stage, the detector performs bounding box regression and predicts the category of candidate regions. For example, [1] proposed a rotated Region-of-Interest (RoI) pooling layer based on RoI-Transformer to replace the traditional RoI pooling layer. Reference [3] extends the traditional RPN to rotated RPN to improve the performance of rotated object detection. Reference [12] proposed the ReDet detector, which uses rotation-invariant RoI

pooling to predict the orientation of rotated objects, achieving excellent results when detecting objects with fixed orientations in images. However, the complex parameter settings in the two-stage detection method limit its generalization, and the design of anchor boxes is computationally intensive, reducing the inference efficiency. Therefore, recent research has focused on developing one-stage detection methods to perform rotated object detection more efficiently.

One-stage object detection methods [11,29,30] directly generate object bounding boxes from the input image. These methods divide the input image into multiple non-overlapping grids and predict multiple bounding boxes in each grid cell. In rotated object detection, R3Det [11] introduced a feature refinement layer to handle densely distributed objects in complex scenes. In the R2YOLOX [29] method, the detector outputs both coarse and refined bounding box information, which are then processed through an aligned convolutional layer to produce the final RBB. S2A-Net [30] proposed using a lightweight fully connected layer for feature alignment to achieve improved detection performance. Compared to two-stage detection methods, one-stage detectors bypass the need to generate region proposals and directly perform object detection on the original image. As a result, one-stage detectors exhibit significantly faster inference speeds than two-stage detectors, making them more competitive in real-world applications.

2.2. Representation of RBB

In rotated object detection, there are two different methods to represent the RBB, namely point-based and regression-based representations. Figure 3 illustrates these two representations of the RBB used in most rotated object detection techniques. As shown in Figure 3a, the point-based method defines a RBB using the coordinates of its four corners $(x_1, y_1, x_2, y_2, x_3, y_3, x_4, y_4)$ and outputs it in the form of a quadrilateral. This representation method is commonly used in text recognition tasks [4,31].

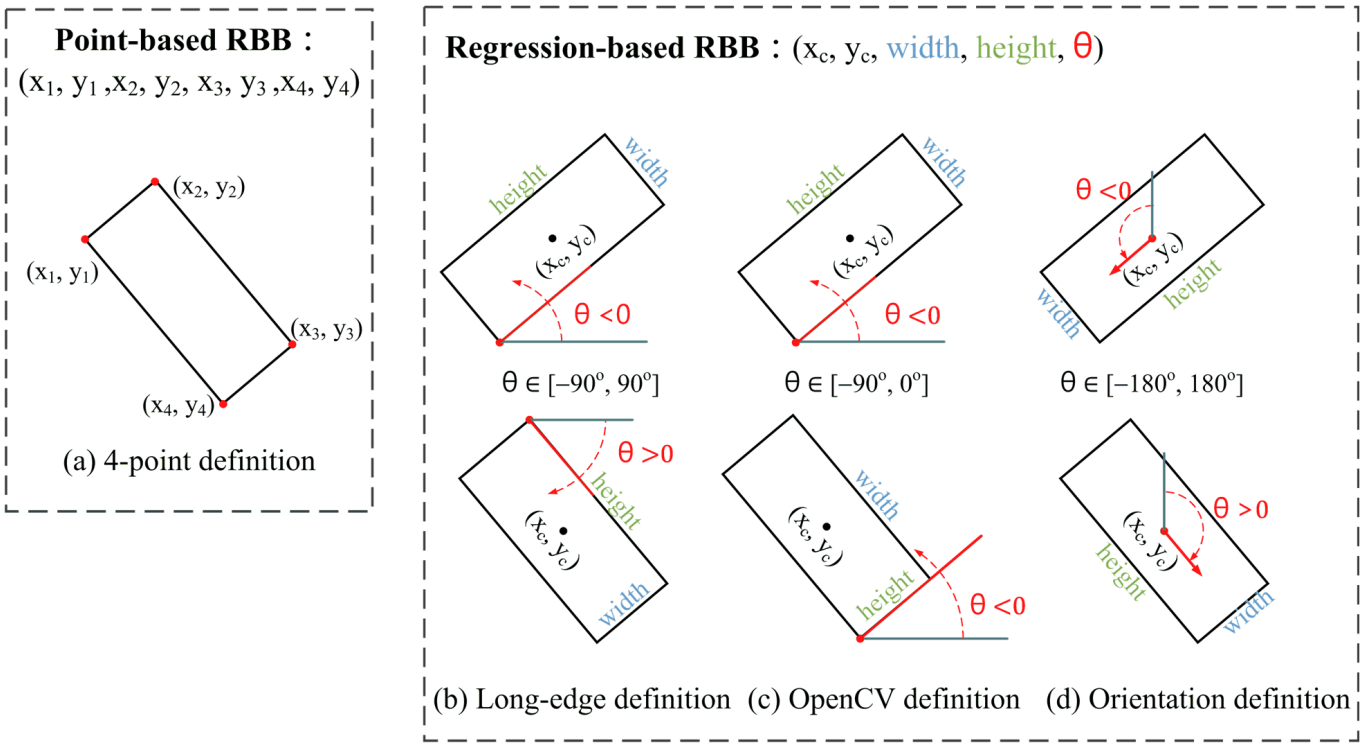


Figure 3. Point-based and regression-based representations of the RBB commonly used in rotated object detection techniques: (**a**) 4-point definition, (**b**) long-edge definition, (**c**) OpenCV definition, and (**d**) orientation definition.

The regression-based representation, as shown in Figure 3b–d, uses five parameters (x, y, w, h, θ) to define the RBB, where (x, y) represents the center position of the box, (w, h) is the width and height of the box, and θ is the rotation angle of the box. This representation method outputs a rotated rectangle and is commonly used in remote sensing [21,22] and industrial scenes [20].

Regression-based representation methods can be further subdivided into the long-edge definition [1,14], the OpenCV definition [2,10], and the orientation-based definition [5,32]. As shown in Figure 3b, the angle parameter θ in the long-edge definition represents the angle between the horizontal axis and the long edge of the bounding box, with $\theta \in [-90^\circ, 90^\circ]$. As shown in Figure 3c, the angle θ in the OpenCV definition represents the angle between the horizontal axis and the closest adjacent edge of the bounding box, with $\theta \in [-90^\circ, 0^\circ]$. Finally, the orientation-based definition, as shown in Figure 3d, defines the angle θ as the clockwise angle between the reference axis and the orientation axis, with $\theta \in [-180^\circ, 180^\circ]$. This definition provides a full 360° range and describes the orientation of a rotated object.

2.3. Boundary Discontinuity Problem

As shown in Figure 2, due to the periodicity of the angle parameters, rapid changes in parameter values occur when the bounding box angles approach the periodic boundary, which is the boundary discontinuity problem. For example, in the long-edge definition, when two bounding boxes share the same center coordinates and dimensions, but the actual angle is 89° and the predicted angle is −89°, the two bounding boxes are almost identical, but the actual angle is different from the predicted angle. This issue arises when the angle parameter nears the defined boundary range, affecting the stability of angle prediction. Therefore, addressing the boundary discontinuity problem has become a significant challenge in the field of rotated object detection in recent years.

Currently, there are works that have proposed solutions to the boundary discontinuity problem from different perspectives. For example, SCRDet [2] introduced IoU-smooth L1 loss, which employs a smoother loss function to mitigate the impact of boundary discontinuity. However, this method only reduces the effect of the discontinuity without fundamentally resolving the issue. Gaussian Wasserstein Distance (GWD) [16] and Kullback–Leibler Divergence (KLD) [33] methods convert the RBBs into 2D Gaussian distributions and use the transformed information to calculate the IoU loss. This design effectively prevents the loss from oscillating due to angle changes. Nevertheless, because of the nature of Gaussian distribution, this approach struggles to accurately predict angle information for nearly square objects, resulting in improved mAP_{50} performance but decreased accuracy in mAP_{75} in experimental results. Circular Smooth Label (CSL) [15] proposed encoding the angle parameters as labels and converting the angle prediction task into a classification problem, using Gaussian focal loss [34] with a dynamic weighting mechanism to calculate the angle classification loss. However, these designs may increase model size, thereby affecting inference efficiency, and the excessive hyperparameter adjustments during training impact model convergence. Densely Coded Labels (DCL) [18], inspired by CSL, converted labels into Gray code, significantly reducing the number of output channels and speeding up model inference time. Phase-Shifting Coder (PSC) [19], inspired by phase shifting, proposed converting angle prediction into the prediction of multiple cosine parameters, transforming the angle parameters into a boundary-free form to offer a different solution to the boundary discontinuity problem.

Zhao et al. proposed a Variant Gaussian Label (VGL)-generating method [35] to handle the periodicity of the angle and large variety of aspect ratios. Ming et al. proposed a Representation Invariance Detection (RIDet) method [36], which employs a novel Representation Invariance Loss (RIL) to improve RBB regression in remote sensing images by treating multiple representations as equivalent local minima, enhancing optimization and alignment with localization accuracy. Cheng et al. proposed an Anchor-free Oriented Proposal Generator (AOPG) [37] that abandons horizontal box-related operations from the network architecture, eliminating the reliance on horizontal boxes to reduce noise and

improve detector robustness. Xu et al. proposed a Circular Gaussian Distribution (CGD)-based method [38] to enhance angular prediction in rotated object detection, addressing multi-solution and boundary issues in traditional methods. Recently, Zhao et al. proposed a novel Angular Boundary Discontinuity Free Loss (ABFL) [39], leveraging the von Mises distribution to handle angle periodicity and solve the angular boundary discontinuity problem when detecting rotated objects. Xu et al. proposed an Angle Correct Module (ACM) [40], which introduces a dual-optimization paradigm to address the boundary discontinuity problem in oriented object detection. The ACM method significantly improves typical IoU-like methods, achieving seamless angular prediction and improved performance across multiple datasets.

3. The Proposed Rotated Object Detection Method

In recent years, numerous studies have proposed deep learning-based methods for rotated object detection. Inspired by these existing methods, we propose a novel CNN-based detection model for rotated object detection. This paper introduces a UVC encoding method and a CDL function design to address the boundary discontinuity problem and improve detection performance. Figure 4 illustrates the framework of the proposed method. This section first provides a brief overview of the detection model's basic architecture and the angle parameter definition. Then, the design of the rotated object detector is explained in detail, followed by an introduction to the loss functions designed for each model branch.

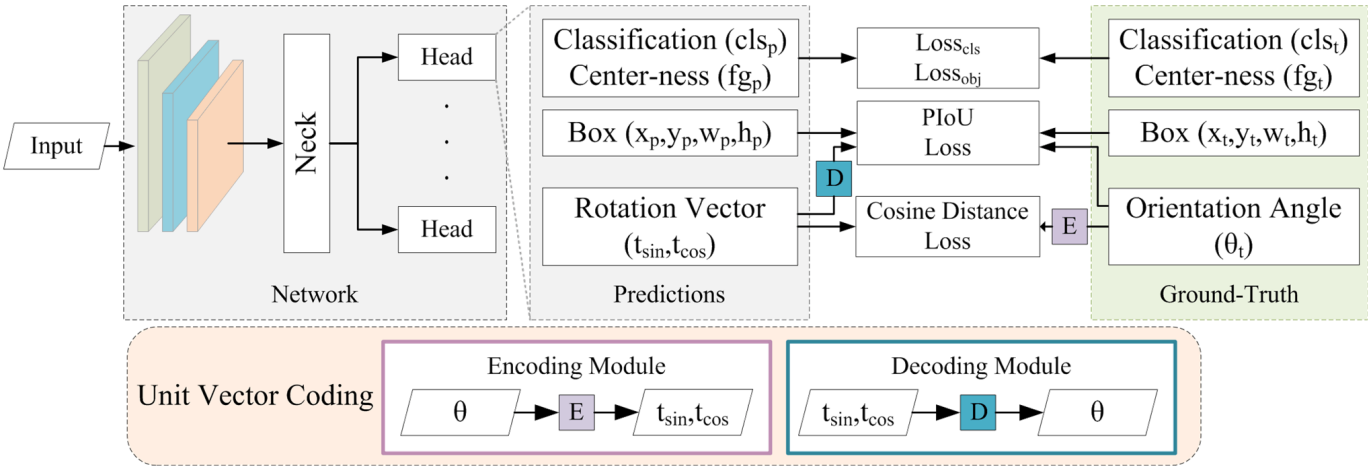


Figure 4. The framework of the proposed method: The input image is processed through the backbone network and neck layer to predict object class, center-ness, bounding box regression, and angle parameters from multi-scale feature maps. The predicted parameters and ground-truth values are then fed into three loss functions to compute the error in RBB predictions.

3.1. Baseline

3.1.1. Orientation Angle Representation

The proposed method utilizes the orientation angle representation defined in the MVTec screws dataset [20] for detecting the orientation of objects, as shown in Figure 5. In this definition, the angle parameter θ has periodic properties and is defined by a periodic constant P, where $\theta \in [-P\pi, P\pi]$. For instance, in the MVTec application scenario, the periodic constant for rotated objects is P = 1, meaning the orientation angle falls within the range $\theta \in [-\pi, \pi]$, with a periodicity of 2π.

3.1.2. Network Architecture

The proposed method extends the You Only Look Once X-s (YOLOX-s) detector [23] for rotated object detection. YOLOX is an anchor-free object detection network featuring a Cross-Stage Partial Darknet53 (CSPDarknet53) backbone and a Path Aggregation Feature Pyramid Network (PAFPN) feature fusion layer. Using the anchor-free detector in our

method provides robustness to detect objects of different sizes without requiring predefined anchor boxes, which is challenging for detecting both small and large objects. The PAFPN extracts feature maps of multiple scales, which are subsequently fed into the rotation detection head module. Figure 6 illustrates the architecture of the proposed rotation detection head module, which processes feature maps of size W×H through 2D convolutional layers and finally outputs W×H sets of RBB information through four branches:

(1) **Regression output**: this branch predicts the center (t_x, t_y), width (t_w), and height (t_h) of the RBB.
(2) **Angle output**: this branch predicts the orientation angle by decomposing it into components (t_{cos}, t_{sin}) in the form of a unit vector.
(3) **Center-ness output**: this branch predicts the center-ness of the object, with an output channel number of 1.
(4) **Classification output**: this branch predicts the class of the object within the RBB, where the number of output channels corresponds to the number of classes in the dataset.

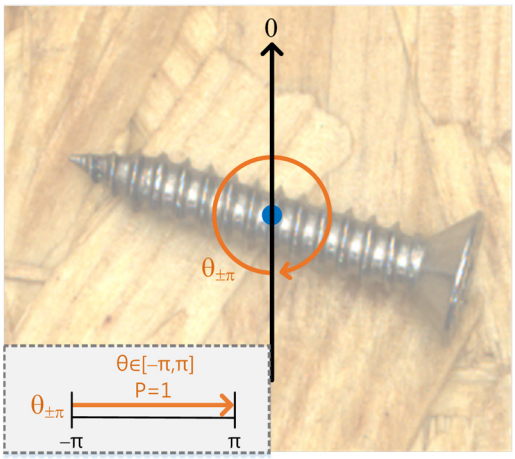


Figure 5. The orientation angle representation used in the proposed method, where the orientation angle range is $\theta \in [-\pi, \pi]$ with a periodicity of 2π.

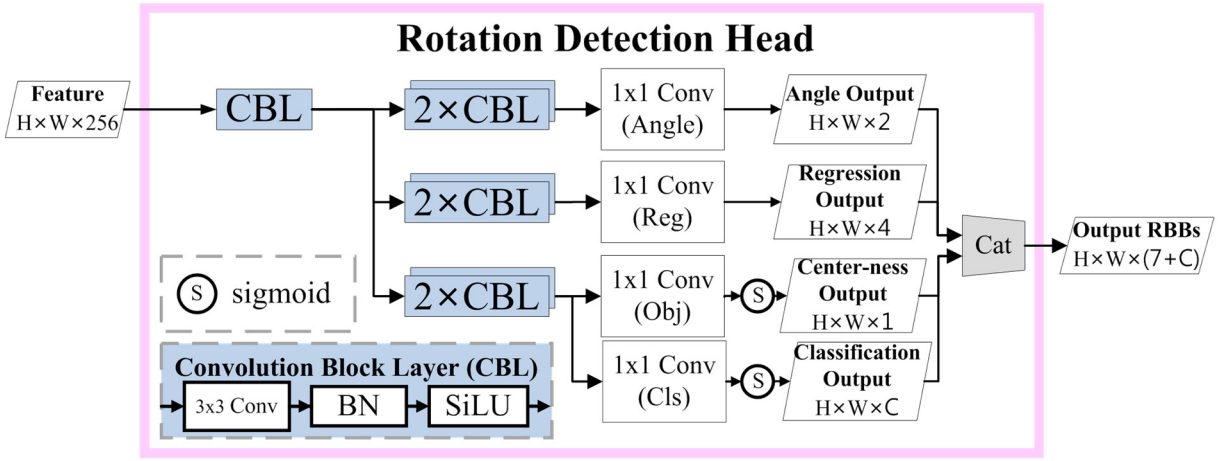


Figure 6. Architecture of the proposed rotation detection head module for the rotated object detection model: the proposed head model consists of three Convolution Block Layer (CBL) modules and four 1 × 1 convolutional (1 × 1 Conv) layers to form the angle, regression, center-ness, and classification prediction branches. The number of channels for each branch is {2, 4, 1, C}, where C represents the number of classes in the dataset. The CBL module consists of a 3 × 3 convolution layer, a Batch Normalization layer, and a SiLU activation function layer.

The proposed method modifies the regression branch of the traditional object detector by extending the output from (t_x, t_y, t_w, t_h) to include the angle prediction branch, transforming it into a rotated object detector (t_x, t_y, t_w, t_h, t_{cos}, t_{sin}). As shown in Figure 4, during the model training, the total loss is computed by weighting the loss functions of each prediction branch and is defined as follows:

$$loss_{total} = \lambda_{reg} loss_{PIoU} + \lambda_{ang} loss_{CDL} + loss_{cls} + loss_{obj}, \quad (1)$$

where $loss_{PIoU}$, $loss_{CDL}$, $loss_{cls}$, and $loss_{obj}$ represent the loss functions for the regression, angle, classification, and center-ness prediction branches, respectively. The weights λ_{reg} and λ_{ang} are used for the regression and angle prediction loss functions. The weight λ_{reg} is defined in YOLOX and set to 3.0. Regarding the setting of the weight λ_{reg}, its value is chosen based on the analysis reported in Section 4.2.

3.2. Unit Vector Coding (UVC)

The proposed detector introduces an angle parameter encoding method based on unit vector properties. The proposed UVC decomposes the angle parameter into vector components, transforming the angle prediction task into a component prediction task. This encoding method is specifically designed for parameters with periodic characteristics, such as angles. By decoding the angle into two independent values with continuous properties, UVC effectively avoids the boundary discontinuity problem, thereby enhancing the accuracy and stability of the model when performing rotated object detection tasks.

The UVC method describes the angle output of the bounding box in terms of the unit vector components t_{cos} and t_{sin}, with the relationship between the output components (t_{cos}, t_{sin}) and orientation angle θ defined as follows:

$$t_{\cos} = \cos(\frac{\theta}{P}) \text{ and } t_{\sin} = \sin(\frac{\theta}{P}), \quad (2)$$

where the periodic constant P defines the range of the orientation angle $\theta \in [-P\pi, P\pi]$. A detailed definition is provided in Figure 7. The output components from the head layer are transformed into the predicted orientation angle θ such that

$$\theta = \begin{cases} P \times \cos^{-1}(\frac{t_{\cos}}{\sqrt{t_{\cos}^2 + t_{\sin}^2}}), & t_{\sin} \geq 0, \\ -P \times \cos^{-1}(\frac{t_{\cos}}{\sqrt{t_{\cos}^2 + t_{\sin}^2}}), & t_{\sin} < 0. \end{cases} \quad (3)$$

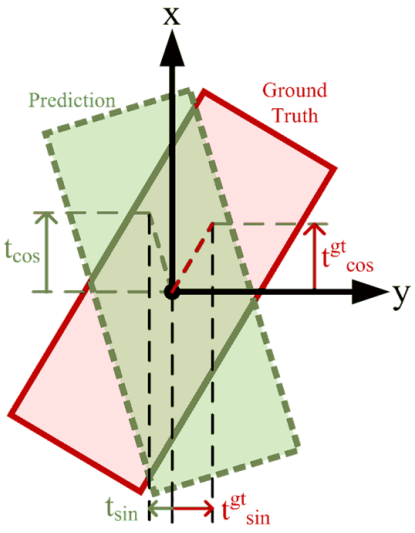


Figure 7. The proposed UVC method transforms the angle prediction task into a component prediction task for the unit vector components (t_{cos}, t_{sin}).

As illustrated in Figure 7, the proposed UVC method transforms the angle prediction task into two independent component predictions based on unit vectors. This approach mitigates drastic error changes at angle boundaries, thus enhancing the stability of predicting angle parameters with periodic characteristics.

3.3. Loss Functions

3.3.1. PIoU Loss

In object detection, the IoU loss is commonly used to evaluate the accuracy of the outputs from the regression branch of the model. In this study, we adopt Pixels-Intersection over Union (PIoU) loss [41] as the criterion for measuring the error between the ground-truth and predicted RBBs. Unlike the traditional IoU for HBBs, PIoU incorporates the orientation angle information of the object, making it more suitable for rotated object detection tasks. Figure 8 illustrates the PIoU calculation, where IoU is computed by counting the number of pixels in the intersection and union of the two RBB. Consequently, the PIoU loss function is defined by

$$loss_{PIoU} = \frac{\sum_{i=1}^{N_b} 1 - \mathrm{PIoU}(\mathrm{Box}_{gt}^{i}, \mathrm{Box}_{p}^{i})}{N_b}, \tag{4}$$

where Box_{gt} represents the parameters of the ground-truth RBB, while Box_{p} refers to the predicted RBB from the model. N_b denotes the number of candidate boxes generated by the model.

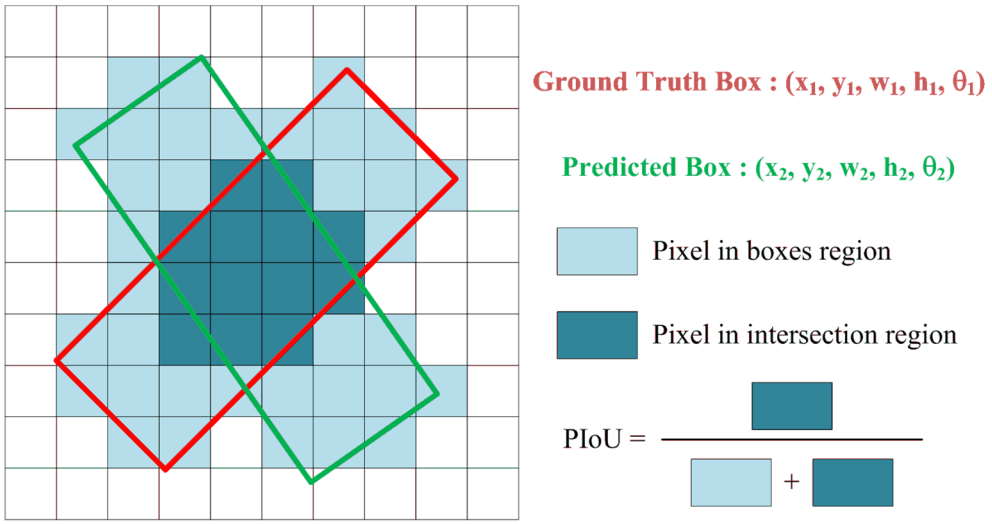


Figure 8. Calculation of PIoU loss.

3.3.2. Cosine Distance Loss (CDL)

The proposed method employs the CDL function for training the angle prediction branch. As shown in Figure 9, this loss function leverages the cosine function's properties to quantify the error between the actual and predicted angles, where a smaller value indicates closer similarity. In this loss function, cosine similarity (CS) is employed to measure the alignment between the predicted and ground-truth angles, as defined by Equation (5):

$$CS(t_{sin}, t_{\cos}, \theta_{gt}) = \frac{t_{\sin} \times \sin(\theta_{gt}) + t_{\cos} \times \cos(\theta_{gt})}{\sqrt{{t_{\sin}}^2 + {t_{\cos}}^2}}, \tag{5}$$

where θ_{gt} denotes the ground-truth angle, while t_{cos} and t_{sin} are the predicted angle components from the model's angle prediction branch. When the CS function equals 1, the two vectors (t_{cos}, t_{sin}) and [$\cos(\theta_{gt})$, $\sin(\theta_{gt})$] are perfectly aligned; conversely, a value of

−1 indicates they are completely opposite. Accordingly, the proposed CDL function for training the angle branch is defined as follows:

$$loss_{CDL} = \sum_{i=1}^{N_b} \frac{[1 - CS(t^i_{\sin}, t^i_{\cos}, \theta^i_{gt})]^2 \times [\left|t^i_{\sin} - \sin(\theta^i_{gt})\right| + \left|t^i_{\cos} - \cos(\theta^i_{gt})\right|]}{N_b}, \quad (6)$$

where the CS function value serves as a dynamic weighting factor to facilitate effective convergence during the component prediction training and enhance the accuracy of the predicted angle parameters for candidate bounding boxes.

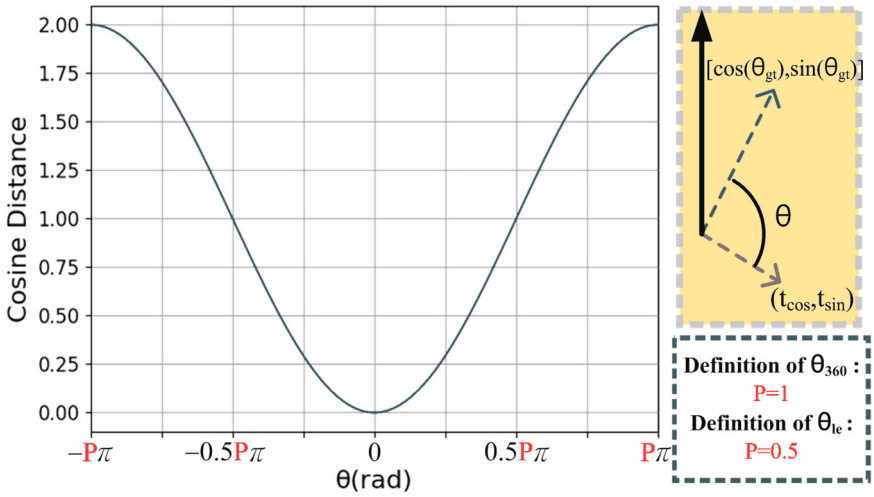


Figure 9. Evolution of the proposed CDL function with respect to the angle θ between two vectors, for θ ∈ [−Pπ, Pπ].

3.4. Dataset and Training Method

3.4.1. Dataset

To validate the performance of the proposed detector in rotated object detection tasks, we used the MVTec screws dataset [20] and the remote sensing dataset HRSC2016 [21] for evaluation. As shown in Figure 10a, the MVTec screws dataset contains objects such as screws and nuts in arbitrary orientations, randomly placed on a wooden background. The RBB information includes the orientation angle of the objects, where θ ∈ [−π, π]. This dataset consists of 327 training samples and 57 test samples, covering a total of 13 object classes.



(**a**) (**b**)

Figure 10. Example of an image sample from (**a**) MVTec screws and (**b**) HRSC2016 datasets.

Figure 10b shows an example of the HRSC2016 dataset, which contains images of various types of ships and offers three levels (L1 to L3) of difficulty for object recognition. In difficulty levels L1 to L3, the ships are categorized into {1, 3, 19} classes, respectively. In our experiments, we selected the L1 task, which focuses on detecting all "ship" objects and predicting the RBB information of the ships. The dataset includes 617 training samples and 444 test samples, encompassing a total of 2976 ship objects.

3.4.2. Training Method

Before training the model, we applied the Mosaic data augmentation strategy [42]. This method involves randomly scaling and rotating four original images to create four sub-images, which are then merged into a single new image used for training. This approach increases the diversity of the training data, enhancing the model's robustness in detecting objects at different scales. During the last 15 epochs of model training, data augmentation is disabled, and the model is trained using the original images to ensure that its predictions align more closely with real-world data distributions.

In this study, the model was trained for 300 epochs using a single NVIDIA GeForce RTX 2080Ti GPU, with a batch size of 16 and an input image size of 640 $\times$ 640. The initial learning rate was set to 0.0025 and gradually decayed as the epochs increased. For optimizer selection, the Stochastic Gradient Descent (SGD) with a momentum value of 0.9 was employed to ensure a fair comparison as it is the default optimizer used in YOLOX detector training.

4. Experimental Results

In the experiment, we adopted two metrics to evaluate the accuracy of rotated object detection tasks, including mAP and CS metrics. The definitions and descriptions of these metrics are outlined as follows:

Mean average precision (mAP): mAP is a key metric to evaluate the performance of object detection models. This metric is commonly used to assess the recall and precision of bounding boxes at various IoU thresholds. For instance, mAP_{50} refers to the IoU threshold of 0.5, where the bounding box is considered a positive sample if the IoU between the predicted and ground-truth RBB exceeds this threshold. The average precision (AP) is then calculated based on the recall and precision of these positive samples, with mAP representing the average AP across all object classes. The mAP_{5095} metric computes the average AP score at IoU thresholds ranging from 0.5 to 0.95, with increments of 0.05. The accuracy metric mAP_{07} follows the precision and recall standards established by PASCAL Visual Object Classes (VOC) 2007 [43].

Cosine similarity (CS): In the MVTec screws dataset, the task involves evaluating the angular difference between the predicted and ground-truth RBBs to assess the accuracy of the angle prediction branch. In this study, we introduced the CS metric, as defined in Equation (5), to quantify the accuracy of the predicted angle values.

4.1. Ablation Study of the Proposed UVC Method Applied to Two IoU Loss Functions

In this section, we present an ablation study to evaluate the effectiveness of the proposed UVC method when applied to two IoU loss functions: the Gaussian distribution-based KLD [33] and PIoU [41]. The objective of this study is to analyze the impact of incorporating the proposed UVC method on model performance and demonstrate its contribution to improving detection accuracy in rotated object detection tasks. The ablation experiments were conducted using the YOLOX-s detector and the MVTec test set, and the results are presented in Table 1. Based on the experimental results, we have the following observations:

(1) Comparing the impact of the KLD loss function and the PIoU loss function on mAP and CS metrics, the experimental results show that PIoU performs better as a regression error metric on this dataset

(2) As shown in Table 1, when the KLD loss function is combined with the proposed UVC method, mAP_{50} and mAP_{5095} increased by 1.8% and 4.5%, respectively. For angle accuracy evaluation, the CS metric increased by 0.013. Similarly, when the PIoU loss function is combined with the UVC method, mAP_{50} and mAP_{5095} increased by 0.2% and 1.1%, respectively, while CS improved by 0.02. These results validate that the proposed UVC method enhances detection accuracy across multiple metrics when integrated with different IoU loss functions. Moreover, the processing speed of the proposed method achieves 53.2 FPS, enabling real-time rotated object detection.

(3) Figure 11 illustrates the visualization results on the MVTec test set. As seen in the figure, training with the proposed UVC method, in combination with both PIoU and KLD loss functions, effectively mitigates the boundary discontinuity problem, thereby improving the model's prediction accuracy and stability.

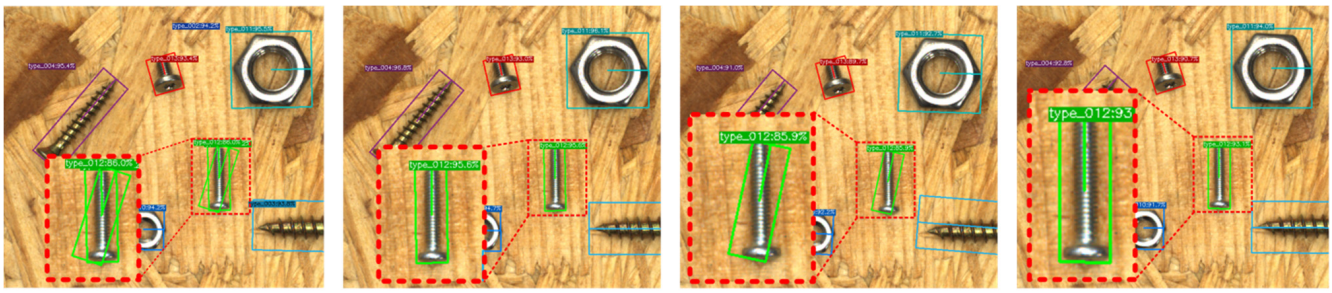


Figure 11. Visualization of experimental results using the proposed UVC method applied to the PIoU and KLD loss functions, evaluated on a sample from the MVTec test set. It is clear that the proposed UVC method effectively mitigates the boundary discontinuity problem, thereby improving the model's prediction accuracy and stability.

Table 1. Ablation study results of the proposed UVC method applied to different IoU loss functions on the MVTec test set.

Detector	IoU Loss	UVC	mAP_{50} (%)	mAP_{5095} (%)	CS	FPS
YOLOX-s	KLD [33]		96.34	75.43	0.983	-
		✓	98.14	79.93	0.996	-
	PIoU [41]		98.55	86.46	0.977	**53.8**
		✓	**98.71**	**87.48**	**0.997**	53.2

The bold font indicates the best result for each column in the table.

4.2. Evaluation on Different Weight Values for the Angle Loss Function

In this section, we investigate the impact of varying weight values for the angle loss function on model training performance in rotated object detection tasks. The objective of these experiments is to determine the optimal balance between the angle and IoU loss functions by adjusting the weight assigned to the angle loss term. Using the YOLOX-s detector and the MVTec test set, we evaluated model performance across different weight values applied to the angle loss function. For this experiment, we selected PIoU and CDL as the loss functions for the regression and angle prediction branches, respectively. Next, we adjusted the angle loss weight λ_{ang} in the total loss function (1) to identify the most effective weight value. As illustrated in Figure 12, the weight value of 2.5 was selected as optimal for the proposed method, as it yielded the best mAP_{5095} and CS scores.

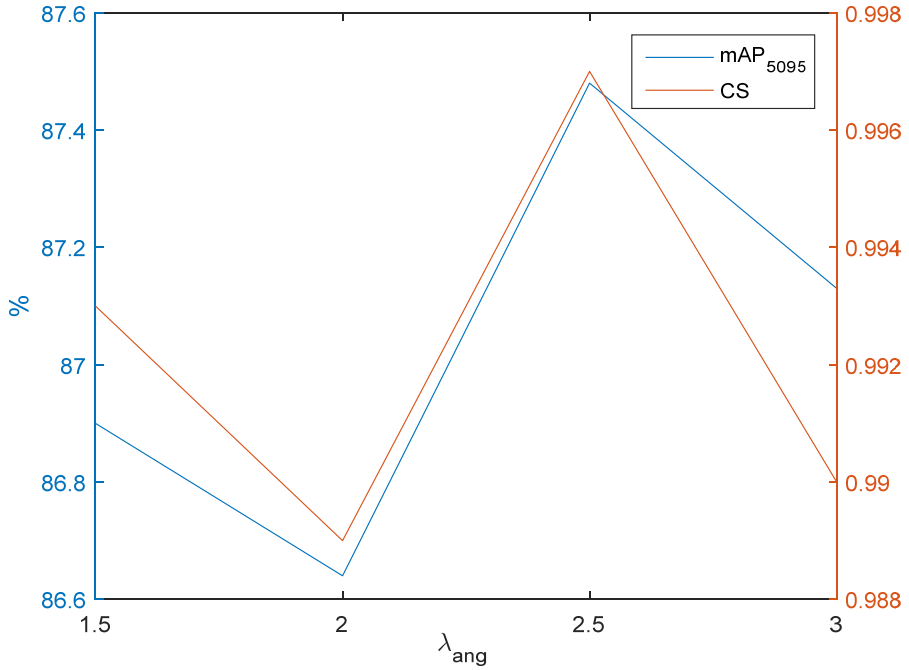


Figure 12. Performance evolution of the weight λ_{ang} from 1.5 to 3.0 on the MVTec test set.

4.3. Performance Comparison of Different Angle Parameter Encoding Methods

In this experiment, we compared the accuracy and computational load of the proposed UVC method with several other encoding methods, as presented in Table 2. Based on the experimental results, we have the following findings:

(1) As shown in Table 2, on the MVTec dataset, the proposed UVC encoding method achieved the highest mAP_{50} and mAP_{5095} scores, significantly outperforming two recently published methods, PSC [19] and ACM [40]. These results demonstrate that the proposed UVC method provides significant performance improvements compared to the current state-of-the-art (SOTA) encoding techniques.

(2) Table 2 also presents a summary of the number of Mega Parameters (MParams) and Giga Floating-point Operations per second (GFLOPs) for each encoding method. The CSL method [15] increases both the parameter number and computational load compared to the method without encoding (None). In contrast, the proposed UVC encoding method introduces negligible changes in both MParams and GFLOPs, while offering superior accuracy performance.

(3) Table 3 presents the performance evaluation of the proposed method for each class on the MVTec test set. Compared to the method without encoding, the proposed method significantly enhances detection accuracy across several classes, including Type01, Type02, Type03, and Type05. These results demonstrate that the proposed method effectively improves the rotated object detection accuracy in terms of mAP5095 and CS metrics compared to the method without encoding.

(4) Figure 13 presents two comparisons of the proposed method with the method without encoding on the MVTec test set. Figure 13a shows the results of the method without encoding. It is clear that when the object's orientation angle approaches the boundary, the boundary discontinuity results in lower-quality bounding boxes, thereby reducing detection accuracy. In contrast, Figure 13b illustrates the results of the proposed method, which effectively addresses the boundary discontinuity issue, leading to enhanced detection accuracy for rotated objects.

Interested readers can refer to the online video [44] for more experimental results of the proposed method on the MVTec test set.

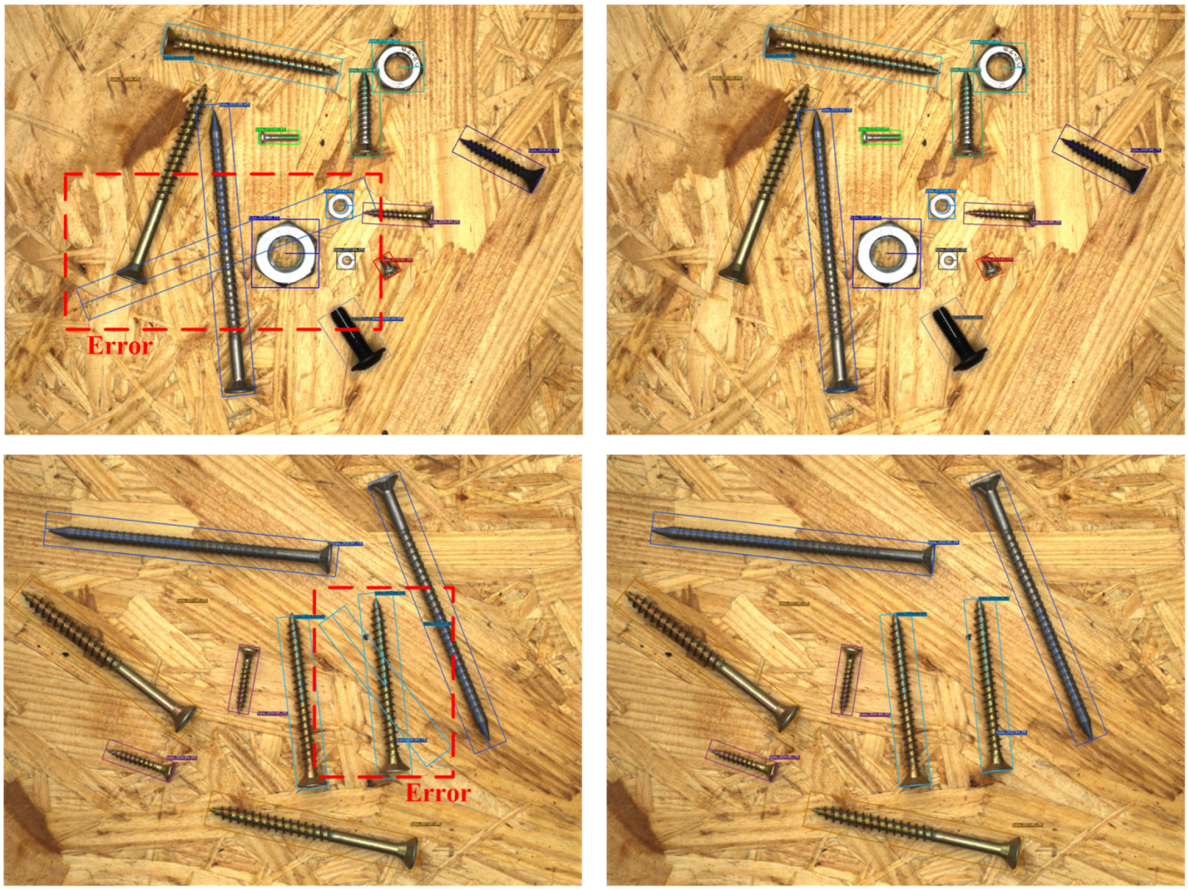
Table 2. Performance comparison of different angle parameter encoding methods on the MVTec test set.

Detector	Encoding	Len	Loss Function	mAP_{50} (%)	mAP_{5095} (%)	CS	MParams	GFLOPs
YOLOX-s	None	1	Smooth L1 loss	98.55	86.46	0.977	9.83	31.76
	CSL [15]	180	Gaussian focal loss	98.65	80.45	0.957	9.90	32.14
		360	Gaussian focal loss	98.24	84.22	0.984	9.97	32.53
	PSC [19]	3	Smooth L1 loss	98.14	86.65	0.989	9.83	31.76
		60	Smooth L1 loss	98.55	86.35	0.601	9.85	31.88
	ACM [40]	2	Smooth L1 loss	98.62	86.96	0.993	9.83	31.76
	UVC	2	CDL (ours)	**98.71**	**87.48**	**0.997**	**9.83**	**31.76**

The bold font indicates the best result for each column in the table.

Table 3. Performance evaluation of the proposed method for each class on the MVTec test set.

Encoding	Metric	Type01	Type02	Type03	Type04	Type05	Type06	Type07	Type08	Type09	Type10	Type11	Type12	Type13
None	AP_{5095}	0.863	0.833	0.844	**0.854**	0.836	0.845	**0.886**	**0.926**	0.924	0.880	0.916	0.822	**0.812**
	CS	1.000	0.830	0.958	1.000	0.965	1.000	**0.999**	1.000	1.000	1.000	1.000	1.000	0.958
UVC	AP_{5095}	**0.888**	**0.860**	**0.864**	0.852	**0.847**	**0.859**	0.862	0.923	**0.939**	**0.884**	**0.927**	**0.858**	0.811
	CS	1.000	**1.000**	**1.000**	1.000	**0.998**	1.000	0.998	1.000	1.000	1.000	1.000	1.000	**0.962**



(**a**) (**b**)

Figure 13. Experimental results on the MVTec test set comparing (**a**) the method without encoding and (**b**) the proposed method. The results demonstrate that the proposed method effectively resolves the boundary discontinuity problem, leading to improved detection accuracy for rotated objects.

4.4. Performance Comparison with SOTA Methods on the HRSC2016 Test Set

We also conducted experiments on the HRSC2016 dataset to evaluate the impact of different periodic parameters on the proposed method. For this dataset, the periodic constant P was set to 0.5, which implies that $\theta \in [-0.5\pi, 0.5\pi]$. Table 4 presents a comparison of the proposed method's performance with several previously introduced methods on the test set, using the PASCAL VOC 2007 metric mAP_{07} as the accuracy evaluation benchmark. The key findings from the results are as follows:

(1) The existing CGD method [38], using ResNet101 as the backbone network, achieved the highest mAP score of 90.61 on the test set. The proposed method, also using the ResNet101 backbone, achieved the second-highest mAP score of 90.54. Additionally, when using the ResNet50 backbone network, the proposed method achieved the third-highest mAP score of 90.44 on the test set. These results confirm that the proposed method achieves comparable performance to the recently published CGD method and outperforms other SOTA methods addressing the boundary discontinuity problem, including KLD [33], CSL [15], and PCS [19].

(2) We also compared our method with the VGL method [35], which employs the Deep-Layer Aggregation network with Deformable Convolutional Networks (DLA34-DCN) backbone. The results indicate that our method, using both ResNet50 and ResNet101 backbones, achieved superior performance, highlighting its efficiency in orientation estimation accuracy.

(3) The detection performance of the proposed method is further evaluated on small and large ship objects in the HRSC2016 test set. In this experiment, a size threshold of 128×128 pixels was established to classify the ground truth into small and large objects. The average precision (AP) for small objects (AP_S) and large objects (AP_L) is then measured. Table 5 presents the results of this evaluation. From Table 5, it is evident that the proposed method achieves comparable AP scores for both small and large objects, indicating that the method exhibits significant scale-invariant detection performance.

(4) Figure 14 presents a comparison between the proposed method and the method without encoding on the HRSC2016 test set. Figure 14a,b show the results of the method without encoding and the proposed method, respectively. The results clearly demonstrate that the proposed UVC encoding method significantly improves the detection robustness of the rotated object detector. This improvement is evident in the more accurate RBBs generated for ships of different sizes, shapes, and orientations, leading to an overall increase in rotated object detection performance and reliability.

Table 4. Performance comparison between the proposed method and SOTA methods on the HRSC2016 test set.

Method	Backbone *	mAP_{07} (%)
VGL [35]	DLA34-DCN	89.78
RIDet [36]	ResNet50	89.47
KLD [33]	ResNet50	89.76
CSL [15]	ResNet50	89.84
PSC [19]	ResNet50	90.06
RoI Transformer [1]	ResNet101	86.20
R3Det-DCL [18]	ResNet101	89.46
RIDet [36]	ResNet101	89.63
R3Det-GWD [16]	ResNet101	89.85
S2A-Net [30]	ResNet101	90.17

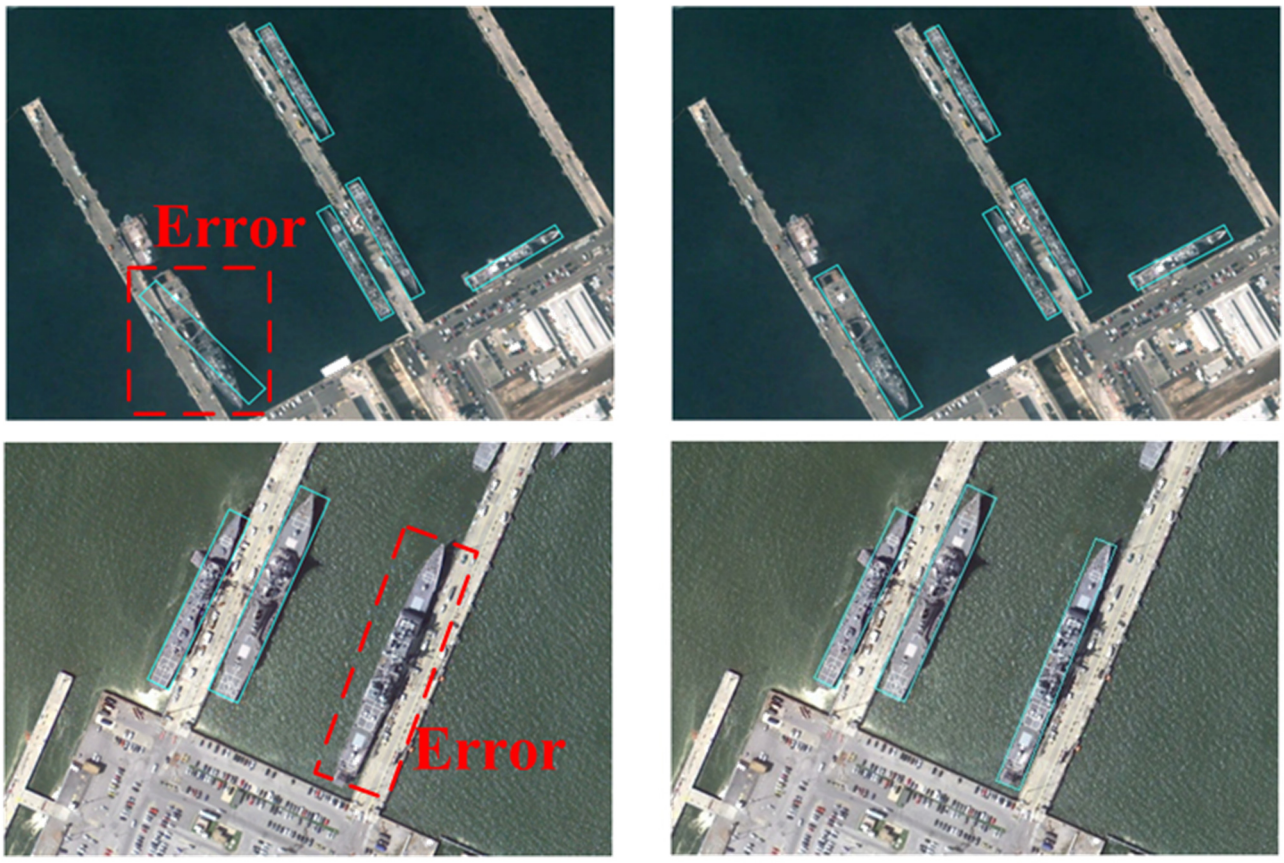
Table 4. *Cont.*

Method	Backbone *	mAP_{07} (%)
ABFL [39]	ResNet101	90.30
AOPG [37]	ResNet101	90.34
CGD [38]	ResNet101	**90.61**
UVC (ours)	ResNet50	90.48
	ResNet101	90.54

*: All backbone networks are equipped with pre-trained weights. The bold font indicates the best result in the table.

Table 5. Performance evaluation of the proposed method on small and large objects in the HRSC2016 test set.

Method	Size Threshold	Backbone	AP_S (%)	AP_L (%)
UVC (ours)	128 × 128	ResNet50	89.48	90.86
		ResNet101	89.65	90.91



(**a**) (**b**)

Figure 14. Experimental results on the HRSC2016 test set comparing (**a**) the method without encoding and (**b**) the proposed method.

5. Conclusions and Future Work

This paper proposes a novel angle parameter encoding method that encodes angular parameters into a continuous and reversible form, effectively addressing the boundary discontinuity problem and enhancing detection accuracy in model predictions. Additionally, the proposed CDL function ensures that angle parameter predictions are better aligned

with the target's orientation. Experimental results demonstrate that the proposed method significantly enhances the model's prediction performance while maintaining minimal computational overhead, making it highly suitable for rotated object detection applications, particularly in industrial settings. Furthermore, experiments conducted on a remote sensing dataset validate the positive impact of the proposed loss function and encoding design across various rotated object detection datasets.

The angle parameter encoding proposed in this study is broadly applicable and can be integrated with various object detection architectures. Future work will focus on applying this method to advanced detection models like YOLOv8 and YOLOv9 to evaluate its generalizability and robustness on large-scale datasets, such as DOTA and FAIR1M. Additionally, the effectiveness of the proposed method for lightweight detection models, including YOLOX-nano and YOLOX-tiny, remains to be studied. Optimizing the design of the rotation detection head module for these lightweight models to improve real-time rotated object detection performance will also be a priority in future research.

Author Contributions: Methodology, C.-Y.T. and W.-C.L.; Software, W.-C.L.; Validation, W.-C.L.; Formal analysis, W.-C.L.; Investigation, C.-Y.T. and W.-C.L.; Resources, C.-Y.T.; Data curation, W.-C.L.; Writing—original draft, C.-Y.T.; Writing—review & editing, C.-Y.T.; Visualization, W.-C.L.; Supervision, C.-Y.T.; Project administration, C.-Y.T.; Funding acquisition, C.-Y.T. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by the National Science and Technology Council of Taiwan under Grant NSTC 112-2221-E-032-036-MY2.

Data Availability Statement: Data are contained within the article.

Conflicts of Interest: The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: Chi-Yi Tsai reports financial support was provided by the National Science and Technology Council of Taiwan. Chi-Yi Tsai reports a relationship with the National Science and Technology Council that includes: funding grants.

References

1. Ding, J.; Xue, N.; Long, Y.; Xia, G.-S.; Lu, Q. Learning RoI Transformer for Oriented Object Detection in Aerial Images. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019; pp. 2849–2858.
2. Yang, X.; Yang, J.; Yan, J.; Zhang, Y.; Zhang, T.; Guo, Z.; Sun, X.; Fu, K. Scrdet: Towards more robust detection for small, cluttered and rotated objects. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019; pp. 8232–8241.
3. Ma, J.; Shao, W.; Ye, H.; Wang, L.; Wang, H.; Zheng, Y.; Xue, X. Arbitrary-oriented scene text detection via rotation proposals CNN: Rotational region CNN for orientation robust scene text detection. In Proceedings of the 2018 24th International Conference on Pattern Recognition (ICPR), Beijing, China, 20–24 August 2018; pp. 3610–3615.
4. Liao, M.; Shi, B.; Bai, X. TextBoxes++: A single-shot oriented scene text detector. *IEEE Trans. Image Process.* **2018**, *27*, 3676–3690. [CrossRef] [PubMed]
5. Wang, Z.; Shen, L.; Li, B.; Yang, J.; Yang, F.; Yuan, K.; Fang, C.; Fanwang, Y. Real-Time Rotated Object Detection Using Angle Decoupling. In Proceedings of the 2021 China Automation Congress (CAC), Beijing, China, 22–24 October 2021; pp. 2772–2778.
6. Nie, K.; von Drigalski, F.; Triyonoputro, J.C.; Nakashima, C.; Shibata, Y.; Konishi, Y.; Ijiri, Y.; Yoshioka, T.; Domae, Y.; Ueshiba, T.; et al. Team O2AS' approach for the task-board task of the World Robot Challenge 2018. *Adv. Robot.* **2020**, *34*, 477–498. [CrossRef]
7. Zhou, D.; Fang, J.; Song, X.; Guan, C.; Yin, J.; Dai, Y.; Yang, R. IoU loss for 2D/3D object detection. In Proceedings of the 2019 International Conference on 3D Vision (3DV), Quebec City, QC, Canada, 16–19 September 2019; pp. 85–94.
8. Liu, S.; Qi, L.; Qin, H.; Shi, J.; Jia, J. Path Aggregation Network for Instance Segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 8759–8768.
9. He, K.; Gkioxari, G.; Dollár, P.; Girshick, R. Mask R-CNN. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 2961–2969.
10. Xie, X.; Cheng, G.; Wang, J.; Yao, X.; Han, J. Oriented R-CNN for object detection. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, QC, Canada, 10–17 October 2021; pp. 3520–3529.
11. Yang, X.; Yan, J.; Feng, Z.; He, T. R3Det: Refined single-stage detector with feature refinement for rotating object. *Proc. AAAI Conf. Artif. Intell.* **2021**, *35*, 3163–3171. [CrossRef]
12. Han, J.; Ding, J.; Xue, N.; Xia, G.S. ReDet: A rotation-equivariant detector for aerial object detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 2786–2795.

13. Wagner, R.; Matuschek, M.; Knaack, P.; Zwick, M.; Geiß, M. IndustrialEdgeML—End-to-end edge-based computer vision system for Industry 5.0. *Procedia Comput. Sci.* **2023**, *217*, 594–603. [CrossRef]
14. Yang, X.; Yan, J.; Liao, W.; Yang, X.; Tang, J.; He, T. SCRDet++: Detecting small, cluttered and rotated objects via instance-level feature denoising and rotation loss smoothing. *IEEE Trans. Pattern Anal. Mach. Intell.* **2022**, *45*, 2384–2399. [CrossRef] [PubMed]
15. Yang, X.; Yan, J. Arbitrary-oriented object detection with circular smooth label. In Proceedings of the 16th European Conference, Glasgow, UK, 23–28 August 2020; pp. 677–694. [CrossRef]
16. Yang, X.; Yan, J.; Ming, Q.; Wang, W.; Zhang, X.; Tian, Q. Rethinking rotated object detection with Gaussian Wasserstein distance loss. In Proceedings of the International Conference on Machine Learning, Virtual, 18–24 July 2021; pp. 11830–11841.
17. Yang, X.; Zhou, Y.; Zhang, G.; Yang, J.; Wang, W.; Yan, J.; Zhang, X.; Tian, Q. The KFIoU loss for rotated object detection. In Proceedings of the International Conference on Learning Representations, Kigali, Rwanda, 1–5 May 2023.
18. Yang, X.; Hou, L.; Zhou, Y.; Wang, W.; Yan, J. Dense label encoding for boundary discontinuity free rotation detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 15819–15829.
19. Yu, Y.; Da, F. Phase-shifting coder: Predicting accurate orientation in oriented object detection. In Proceedings of the 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Vancouver, BC, Canada, 17–24 June 2023; pp. 13354–13363.
20. MVTec Datasets. MVTec Screws Dataset. Available online: https://www.mvtec.com/company/research/datasets/mvtec-screws (accessed on 7 November 2024).
21. Liu, Z.; Yuan, L.; Weng, L.; Yang, Y. A high resolution optical satellite image dataset for ship recognition and some new baselines. In Proceedings of the 6th International Conference on Pattern Recognition Applications and Methods, Porto, Portugal, 24–26 February 2017; pp. 324–331.
22. Ding, J.; Xue, N.; Xia, G.-S.; Bai, X.; Yang, W.; Yang, M.Y.; Belongie, S.; Luo, J.; Datcu, M.; Pelillo, M.; et al. Object Detection in Aerial Images: A Large-Scale Benchmark and Challenges. *IEEE Trans. Pattern Anal. Mach. Intell.* **2022**, *44*, 7778–7796. [CrossRef] [PubMed]
23. Ge, Z.; Liu, S.; Wang, F.; Li, Z.; Sun, J. Yolox: Exceeding YOLO series in 2021. *arXiv* **2021**, arXiv:2107.08430.
24. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
25. Girshick, R. Fast R-CNN. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015; pp. 1440–1448.
26. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards real-time object detection with region proposal networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 1137–1149. [CrossRef] [PubMed]
27. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.Y.; Berg, A.C. SSD: Single shot multibox detector. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, Proceedings, Part I*; Springer International Publishing: Berlin/Heidelberg, Germany, 2016; pp. 21–37.
28. Lin, T.Y.; Goyal, P.; Girshick, R.; He, K.; Dollár, P. Focal loss for dense object detection. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 2980–2988.
29. Liu, F.; Chen, R.; Zhang, J.; Xing, K.; Liu, H.; Qin, J. R2YOLOX: A lightweight refined anchor-free rotated detector for object detection in aerial images. *IEEE Trans. Geosci. Remote. Sens.* **2022**, *60*, 5632715. [CrossRef]
30. Han, J.; Ding, J.; Li, J.; Xia, G.S. Align deep features for oriented object detection. *IEEE Trans. Geosci. Remote Sens.* **2021**, *60*, 5602511. [CrossRef]
31. Xu, Y.; Fu, M.; Wang, Q.; Wang, Y.; Chen, K.; Xia, G.-S.; Bai, X. Gliding vertex on the horizontal bounding box for multi-oriented object detection. *IEEE Trans. Pattern Anal. Mach. Intell.* **2020**, *43*, 1452–1459. [CrossRef] [PubMed]
32. Zhang, F.; Wang, X.; Zhou, S.; Wang, Y.; Hou, Y. Arbitrary-oriented ship detection through center-head point extraction. *IEEE Trans. Geosci. Remote. Sens.* **2021**, *60*, 5612414. [CrossRef]
33. Yang, X.; Yang, X.; Yang, J.; Ming, Q.; Wang, W.; Tian, Q.; Yan, J. Learning high-precision bounding box for rotated object detection via Kullback-Leibler divergence. *Adv. Neural Inf. Process. Syst.* **2021**, *34*, 18381–18394.
34. Wang, J.; Li, F.; Bi, H. Gaussian focal loss: Learning distribution polarized angle prediction for rotated object detection in aerial images. *IEEE Trans. Geosci. Remote. Sens.* **2022**, *60*, 4707013. [CrossRef]
35. Zhao, T.; Liu, N.; Celik, T.; Li, H.-C. An arbitrary-oriented object detector based on variant gaussian label in remote sensing images. *IEEE Geosci. Remote. Sens. Lett.* **2022**, *19*, 8013605. [CrossRef]
36. Ming, Q.; Miao, L.; Zhou, Z.; Yang, X.; Dong, Y. Optimization for arbitrary-oriented object detection via representation invariance loss. *IEEE Geosci. Remote. Sens. Lett.* **2022**, *19*, 8021505. [CrossRef]
37. Cheng, G.; Wang, J.; Li, K.; Xie, X.; Lang, C.; Yao, Y.; Han, J. Anchor-free oriented proposal generator for object detection. *IEEE Trans. Geosci. Remote. Sens.* **2022**, *60*, 5625414. [CrossRef]
38. Xu, H.; Liu, X.; Ma, Y.; Zhu, Z.; Wang, S.; Yan, C.; Dai, F. Rotated object detection with circular gaussian distribution. *Electronics* **2023**, *12*, 3265. [CrossRef]
39. Zhao, Z.; Li, S. ABFL: Angular boundary discontinuity free loss for arbitrary oriented object detection in aerial images. *IEEE Trans. Geosci. Remote. Sens.* **2024**, *62*, 5611411. [CrossRef]

40. Xu, H.; Liu, X.; Xu, H.; Ma, Y.; Zhu, Z.; Yan, C.; Dai, F. Rethinking boundary discontinuity problem for oriented object detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 16–22 June 2024; pp. 17406–17415.
41. Chen, Z.; Chen, K.; Lin, W.; See, J.; Yu, H.; Ke, Y.; Yang, C. Piou loss: Towards accurate oriented object detection in complex environments. In *Computer Vision–ECCV 2020: 16th European Conference, Proceedings, Part V*; Springer International Publishing: Berlin/Heidelberg, Germany, 2020; pp. 195–211.
42. Bochkovskiy, A.; Wang, C.Y.; Liao, H.Y.M. Yolov4: Optimal speed and accuracy of object detection. *arXiv* **2020**, arXiv:2004.10934.
43. Everingham, M. The PASCAL Visual Object Classes Challenge 2007. Available online: http://host.robots.ox.ac.uk/pascal/VOC/voc2007/ (accessed on 7 November 2024).
44. Experimental Results of the Proposed Method on the MVTec Test Set, Results of Precise Orientation Estimation for Rotated Object Detection Based on Unit Vector Coding. Available online: https://youtu.be/ulJX3NIFMDE (accessed on 7 November 2024).

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.