ISSN 2616-5775 Vol. 6, Issue 4: 68-76, DOI: 10.25236/AJCIS.2023.060409

The GMIoU Loss for Accurate Rotated Object Detection in Remote Sensing Image

Zhiwei Zhang[1,a,*], Wei Sun[1,b]

[1]*College of Information Engineering, Shanghai Maritime University, Shanghai, China*
[a]*13347257557@163.com*, [b]*weisun@shmtu.edu.cn*
[*]*Corresponding author*

Abstract: *How to accurately metric the oriented bounding box loss in remote sensing image object detection has been a hot research topic in this field. The main reason for not being able to accurately calculate the loss of rotating bounding boxes is the complexity of calculating IoU between rotating bounding boxes. This paper is inspired by KFIoU and explores on this basis, and finds that KFIoU needs to be more to metric the merging volume of rotating bounding boxes, and the variation in the value is much different from the actual one. In this paper, we propose a faster and more accurate loss function GMIoU, which calculates the bounding box concatenation volume by Gaussian mixture model and the intersection volume by Kalman filer, and GMIoU solves the problem of inaccuracy of KFIoU in the case of no intersection of rotating bounding boxes and improves the numerator and denominator of KFIoU. The large differences between numerator and denominator of KFIoU in orders of magnitude. Meanwhile, to prevent the angle periodicity problem from affecting the model training effect, a classification-based DCL angle classification module is introduced in the head of the model to improve the model generalization ability and angle prediction accuracy. The experimental results show that GMIoU is closer to SkewIoU regarding variation trend, and the convergence speed is faster compared with, for example, KFIoU. The training effect improves the model's accuracy for large aspect ratio objects and shows good results on square-like objects.*

Keywords: *remote sensing image, rotating object, 2D Gaussian distribution, GMIoU*

1. Introduction

Remote sensing images, as the main information carrier for object detection, contain much useful information. How to accurately detect objects in remote sensing images has important applications in object surveillance, intelligence reconnaissance, intelligent agriculture, environmental monitoring, and other fields [1]. Meanwhile, object detection technology is also the basis for scenario segmentation, object tracking, video object recognition, etc. The study of remote sensing image object detection is significant for social and economic development. In recent years, with the rapid development of deep learning technology, deep learning-based object detection methods have been proposed one after another in the process of researchers' exploration. At the same time, with the release of large remote sensing datasets has made the deep learning based remote sensing image object detection technology gradually become a research hotspot in computer vision, produced some excellent research results. Next, an overview of the research progress in recent years is given.

There are two main categories in conventional object detection methods: two-stage object detection algorithms and single-stage object detection algorithms based on two-stage object detection algorithms. In the development of Two-stage universal object detection models, R-CNN object detection networks, i.e., region traversal convolutional neural networks, were initially pioneered by Girshick et al [2]. Subsequently, domestic scholars Keming He et al [3] proposed the SPP Net object detection network, which used feature pyramids to achieve multi-scale feature fusion, greatly improved the network's effectiveness and speed. After that, Girshick et al. successively proposed classical two-stage object detection algorithms such as Fast R-CNN [4] and Faster R-CNN [5]. Faster R-CNN ensures a high detection accuracy and, at the same time, has a significant improvement in speed compared with previous detection models, reaching 30 FPS per second. Redmon et al [6], from the perspective of regression, abandoned the previous object detection using region search method and replaced it with YOLO, a one-stage object detection model that directly classifies objects and performs bounding box regression operations using regression methods. This model did not need to generate candidate regions, which

ISSN 2616-5775 Vol. 6, Issue 4: 68-76, DOI: 10.25236/AJCIS.2023.060409

greatly saves detection time but loses some detection accuracy. This method significantly improves the accuracy of small object detection while keeping the computational effort unchanged. The above research results explore the deep learning-based object detection methods from different directions, all of which have been improved to different degrees, making the base-general object detection model has reached the industrial-level application level, However it is difficult to achieve ideal results when applied in remote sensing image rotation object detection.

2. Related work

Because of the large difference between their photography circumstances and ordinary direction, remote sensing images have arbitrary orientation, high aspect ratio, dense distribution, and numerous scales, which causes general detection models to fail to recognize rotating objects in remote sensing images[7]. These difficulties have attracted the attention of researchers. As a new hot direction of remote sensing image rotating object detection, some researchers initially tried to extend from the horizontal object detection model. Li et al[8]., proposed R3Det algorithm using rotated candidate region generation network R-RPN to get rotated R-RoI and then feature extraction based on R-RoI to predict the rotated object angle[9]. Yang Xue et al[10]., proposed SCRDet algorithm that achieves feature enhancement for small and dense objects by adding a feature fusion structure and using a supervised attention mechanismto highlight the object and suppress the background [11]. Yang Xue et al., further proposed the CSL and DSL models based on angle classification to convert angle regression into an angle classification problem[12-13]. The angle coding branch is added to the detection header, and the angles are divided into finite ordered classes according to the equal division principle to achieve angle classification. Then the angle values are predicted by the classification method. The method effectively solves the periodicity problem in the angle regression process and improves the model generalization ability and accuracy. The above methods have achieved some results in angle prediction, but their loss functions mainly adopt L1 loss for the bounding box regression problem. However, the index that accurately reflects the bounding box loss is IoU. The loss functions of such inter-parameter independent regressions ignore the correlation between the parameters of the rotated box and especially cannot link the angle loss with the bounding box loss. There are a large number of metrics based on IoU loss during horizontal frame detection, such as GIoU, DIoU, CIoU, and EIoU [14-17]. Because of the inclusion of angles, it makes it challenging to metric the IoU between rotating boxes. Therefore, some researchers have tried to start from the bounding box loss and propose some loss functions that approximately fit SkewIoU, while introducing the angle values into the rotated box regression loss function together so as to achieve a more accurate model for training. Zhiming et al., proposed PIoU loss (Pixels-IoU) using pixel point counting to calculate the IoU loss between the bounding boxes, and the experimental results showed an improvement in both boxes and angle prediction[18]. However, the method is difficult to deploy, and the regression speed is slow. Yang Xue et al., proposed loss functions GWD, KLD and KFIOU based on 2D-Gaussian distribution[19-21]. Among them, KFIoU defines the function expression according to the principle of IoU calculation, which trend level alignment with the SkewIoU loss in numerical trend and FKIoU is convenient to deploy. However, KFIoU cannot reflect the loss when two rotating boxes do not intersect. The main reason is that KFIoU is not accurate enough to calculation of the merged set. Based on the above problems, Our proposes a novel loss function GMIoU based on Gaussian mixture model and Kalman filter, which converges faster, has better training effect and is easy to deploy. Meanwhile, in terms of angle prediction, for further improves the angle prediction accuracy, model generalization ability, and prediction effect on square-like objects by introducing DCL branch.

3. Proposed method

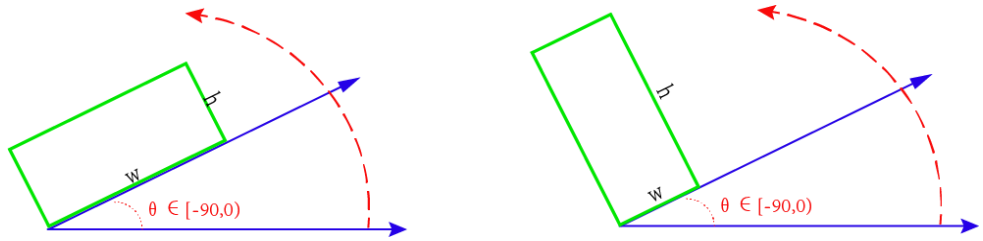
In this section, we give a detailed overview of our proposed method. Firstly, we introduce the definition of rotating boxes and, simultaneously, expose the angle regression and edge exchange problems of this angle definition method. Next we introduce the converting method of rotating boxes to Gaussian and analyze the shortcomings of KFIoU, and finally we propose our solution.

3.1 Rotated Bounding Box Definition

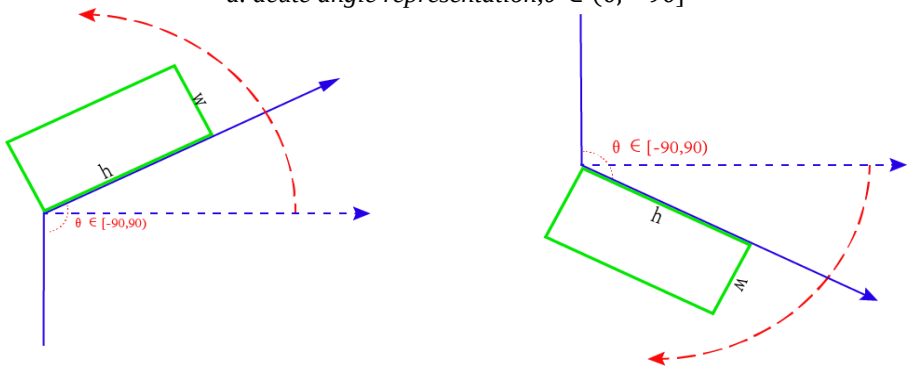
In rotated object detection, the 5-parameter method and the 8-parameter methods are usually used. In the 5-parameter method, the model output parameters are (x,y,w,h,θ), representing the center point coordinates, object aspect, and angle values, respectively. In the 5-parameter method, OpenCV gives two kinds of angle definition ranges. One is the long-edge representation with the angle range [-90, 90) ,and

ISSN 2616-5775 Vol. 6, Issue 4: 68-76, DOI: 10.25236/AJCIS.2023.060409

the other is the short-edge representation with the angle range [-90,0), as shown in Figure 1.



a. acute angle representation, $\theta \in (0, -90]$



b. long side representation, $\theta \in [-90,90)$

Figure 1: Rotated bounding box definition

3.2 Rotated object detection to 2-D Gaussian distribution

In horizontal object detection models, the loss of bounding box regression is often measured by IoU. With the efforts of researchers, a series of IoU-based loss functions have been proposed. The common ones are GIoU, DIoU, CIoU, and EIoU. However, in rotating object detection, the arbitrary angle makes it difficult to design an IoU-based and derivable loss function. The GWD loss function is calculated by converting the rotation frame (x,y,w,h,θ) into a 2D Gaussian distribution N(μ, Σ), and then using the theory in the field of probability statistics for the loss calculation. The method achieves a combination of bounding box regression and angle regression, improves the learning efficiency of the rotating box loss function, and solves the problem of indifferentiability of the IoU loss based on the rotating box. The specific conversion principle is as follows: the bounding box's centroid is converted into a Gaussian distributed fractional mean vector, the length and width are converted into a covariance matrix, the angle is converted into a direction matrix, and the specific expressions are as follows.

$$\Sigma = R\Lambda R^T$$

$$= \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix} \begin{pmatrix} w^2/4 & 0 \\ 0 & h^2/4 \end{pmatrix} \begin{pmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{pmatrix}$$

$$= \begin{pmatrix} \frac{w^2 cos^2\theta + h^2 sin^2\theta}{4} & \frac{w^2-h^2}{4}\sin\theta\cos\theta \\ \frac{w^2-h^2}{4}\sin\theta\cos\theta & \frac{w^2 cos^2\theta + h^2 sin^2\theta}{4} \end{pmatrix} \quad (1)$$

$$\mu = (x, y) \quad (2)$$

Here Σ signifies the covariance matrix of a Gaussian distribution. R symbolizes the direction matrix, which is made up of trigonometric functions of angles, and Λ denotes the eigenvalue matrix. B is the mean vector. The covariance matrix may be simply utilized to compute the size of the area $S_b(\Sigma)$, after transforming the rotating bounding box into a 2D Gaussian distribution, as illustrated in Equation 3.

$$S_b(\Sigma) = 2^n \sqrt{\prod eig(\Sigma)} = 2^n \cdot \left|\Sigma^{\frac{1}{2}}\right| = 2^n \cdot |\Sigma|^{\frac{1}{2}} \quad (3)$$

$S_b(\Sigma)$ denotes the rotating box volume, $eig(\Sigma)$ for the covariance matrix's eigenmatrix, where n = 2, for the dimension. With this equation, we can easily calculate the overlapping and merging regions between rotating boxes in GMIoU.

ISSN 2616-5775 Vol. 6, Issue 4: 68-76, DOI: 10.25236/AJCIS.2023.060409

3.3 GMIoU loss

The details of the suggested approaches will be addressed in this section. To start, we examine the KFIoU Loss's deficiencies. KFIoU calculates the approximate intersection of bounding boxes using Kalman gain, and the union set is defined as the sum of two bounding boxes minus the intersection area, which is defined in the form of $\text{KFIoU} = \frac{C}{A+B-C}$. Here C denotes the intersection region of two boxes, where A and B denote the predicted and ground truth boxes, respectively. Generally, when the predicted box has no intersection with the ground-truth box at all, the IoU value is close to the minimum point 0; when the predicted box is nearly at the position of the ground true box, the IoU value should reach the maximum value 1. However, the value range of FKIoU is $(0, 1/3)$. Through the experiment,we found that it is observed that the value of KFIoU denominator far exceeds the numerator,and the amount of KFIoU change is nearly zero when there is no intersection between the predicted and ground truth box.The method of merging in Euclidean geometric space is adopted in dealing with the merging region by analyzing FKIoU, which can obtain an accurate merging volume in Euclidean spatial plane geometry, but in the field of probability statistics, it is obvious that using a geometric method to solve a probability distribution problem is not the best calculation method. In this paper, we propose a faster and more precise Gaussian-based loss function GMIoU. When two Gaussian distributions are multiplied together to produce a Gaussian distribution, and the Gaussian distribution is exactly the intersection area $\alpha N_k(\mu, \Sigma)$, as seen in Figure 2-a. Equation 4-7 illustrates the calculating technique. where μ_k denotes the mean vector and Σ_k denotes the covariance matrix.

$$\alpha N_k(\mu_k, \Sigma_k) = N_1(\mu_1, \Sigma_1) N_2(\mu_2, \Sigma_2) \tag{4}$$

$$\mu_{pt} = \mu_p + K(\mu_t - \mu_p) \tag{5}$$

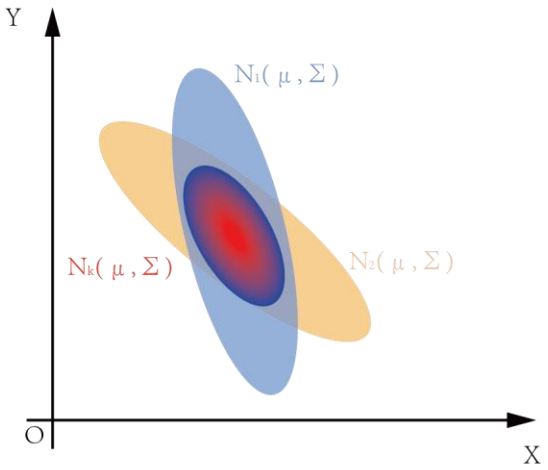
$$\begin{aligned} \Sigma_{pt} &= \Sigma_p - K\Sigma_p \\ &= \Sigma_p - \Sigma_p(\Sigma_p + \Sigma_t)^{-1}\Sigma_p \end{aligned} \tag{6}$$

$$K = \Sigma_p(\Sigma_p + \Sigma_t)^{-1} \tag{7}$$

Here α is the scale factor, α is less when the two distributions are further apart and greater when they are closing a. K is the Kalman gain, Σ_p is the covariance matrix of predicted box, and Σ_t is the covariance matrix of ground-truth box. Lastly, as shown in Equation 8, the size of the intersection region volume $S_k(\Sigma_k)$ may be computed from Σ_k.

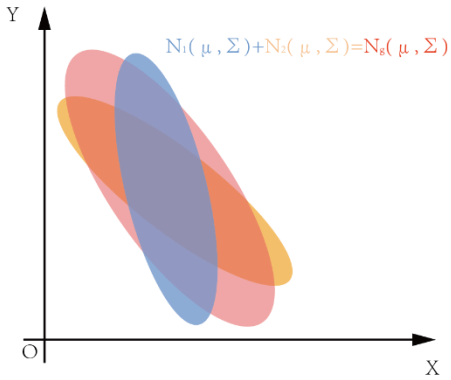
$$S_k(\Sigma_k) = 2^n \sqrt{\prod eig(\Sigma_k)} = 2^n \cdot \left|\Sigma_k^{\frac{1}{2}}\right| = 2^n \cdot |\Sigma_k|^{\frac{1}{2}} \tag{8}$$

Here $S_k(\Sigma_k)$ denotes the size of the intersection area of the two rotating boxes, as shown by the area of the red region in Figure 2-b.



a. $N_k(\mu, \Sigma)$represents the overlap part's Gaussian distribution.

ISSN 2616-5775 Vol. 6, Issue 4: 68-76, DOI: 10.25236/AJCIS.2023.060409



b. $N_g(\mu, \Sigma)$ *is the combined part's Gaussian distribution.*

Figure 2: Conversion diagram of overlapping and merging parts

The computation employs the Gaussian mixed distribution for the combined region. According to Gaussian distribution characteristics, scaling two Gaussian distributions together yields a mixed Gaussian model $N_g(\mu, \Sigma)$, as illustrated in Figure 2-b, and its equation is presented in Equation 9. The region contained by the Gaussian mixture model, may more correctly reflect the merging of the rotating boxes. Eventually, Eq. 10 shows the formula to calculate the merging region size S_g.

$$N_g(\mu, \Sigma) = \sum_{k=1}^{K} \beta_i N(\mu_i, \Sigma_i), \quad \sum_{i=1}^{K} \beta_i = 1 \tag{9}$$

$$S_g = \beta_1 S(\Sigma_p) + \beta_2 S(\Sigma_t), \quad \beta_1 + \beta_2 = 1 \tag{10}$$

The volume enclosed by the Gaussian mixture model is denoted by S_g.The weights in the Gaussian distribution are represented by β_1, β_2. In summary, equation 11 depicts the GMIoU expression.

$$GKIoU = \frac{S_k(\Sigma_k)}{S_g} = \frac{S_k(\Sigma_k)}{\beta_1 S_p(\Sigma_p) + \beta_2 S_t(\Sigma_t)} \tag{11}$$

S_p, S_t, S_k, are the predicted box, ground-truth box, and overlap volume between boxes, respectively. In a vast amount of experiments, we try various combinations of(β_1, β_2), and the optimum model training performance is obtained when β_1, β_2 is selected as 1/2. We can conclude from the analysis that the upper bound of GMIoU in n-dimensional space is $\frac{2^n}{2^{\frac{n}{2}+1}}$, and thus the value domain of GMIoU in 2D-dimensional space is (0,1). It is intuitively discovered that the function graph of GMIoU is closer to SkewIoU in trend, which can better reflect SkewIoU's actual change.

The One-Stage network RetinaNet is utilized as the backbone network in this study, and the angle prediction component presents the DCL module based on angle classification to avoid the issue of angle regression periodicity while also improving angle prediction accuracy. For the regression parameters, the 5-parameter method is employed, with an angle definition range of [-90, 90). In conclusion, Equation 12 summarizes the multi-task loss function.

$$L_{total} = \lambda_1 \sum_{n=1}^{N_{pos}} L_{GKIOU}\big(N(p_n), N(gt_n)\big) + \frac{\lambda_2}{N} \sum_{n=1}^{N} L_{DCL}(\theta'_n, \theta_n) + \frac{\lambda_3}{N} \sum_{n=1}^{N} L_{cls}(p_n, gt_n) + \frac{\lambda_3}{N} \sum_{n=1}^{N} L_1(v(p_n), v(gt_n)) \tag{12}$$

$$L_{GKIOU} = 1 - GKIOU \tag{13}$$

Where λ_i is the weight value, taking values in the range of (0, 1]. N_{pos} and N are the numbers of anchors that contain the object and the total number of anchors, respectively. b_n denotes the predicted boxes ,and gt_n denotes the ground truth boxes. The 2D-dimensional Gaussian distribution converted by the rotated boxes is denoted by $N(\cdot)$. L_{GKIOU} is the bounding boxes regression loss, L_1 is the center point loss, and the smooth L1 loss function is employed, where $v(\cdot)$ indicates the center point coordinates. L_{DCL}, L_{cls} denote the angle classification loss and the category classification loss, which are both calculated using the focal loss function. The detector training procedure is divided as follows:1. The model output the offset$(t'_x, t'_y, t'_w, t'_h, t'_\theta)$; 2. decoding the bias quantity to yield the predicting boxes; 3. converting the bounding boxes to 2-D Gaussian distributions; 4. Calculating $S_i(\Sigma_i)$ and GMIoU

ISSN 2616-5775 Vol. 6, Issue 4: 68-76, DOI: 10.25236/AJCIS.2023.060409

values; 5. computing the total loss before performing backpropagation to update the model parameters. Equation 14 represents regression of (x, y, w, h).

$$\begin{aligned} t_x &= (x - x_a)/w_a, \ \ t_y = (y - y_a)/h_a, \\ t_w &= \log(w/w_a), \ \ t_h = \log(h/h_a), \\ t_\theta &= (\theta - \theta_a) \cdot \pi/180; \\ t_x^* &= (x^* - x_a)/w_a, t_y^* = (y^* - y_a)/h_a, \\ t_w^* &= \log(w^*/w_a), t_h^* = \log(h^*/h_a), \\ t_\theta^* &= (\theta^* - \theta_a) \cdot \pi/180. \end{aligned} \tag{14}$$

Here x, x_a, x^* denote the ground-truth box, anchor box, and predicted box, others as above. The center coordinates, width, height, and angle of the bounding box are denoted as x, y, w, h, θ ,respectively.

4. Experiments

In this section, the exceptional performance of GMIoU in model training is demonstrated by comparative experiments.

4.1 Implement details

When building the network model, we implemented the Pytorch-based MMRotate architecture. The detector is deployed on a specialized deep learning server with 4 GeForce RTX 2080 Ti and 11G memory. The backbone network uses RetinaNet and inherits the official weight file for the network initialization; anchor settings remain the same as the original RetinaNet and FPN models. We used SGD optimization, in which the learning rate is set to 0.002, the weight decay set to 0.0001, and the momentum are set to 0.9, batch size 8, 2 images per GPU, total 4 GPU. The entire training duration for the DOTA v1.0 dataset was 20 epochs, with the learning rate being reduced by a factor of 10 at the 11th and 16th epochs.

4.2 Datasets

As the experimental dataset in this work, we use the open-source remote sensing image dataset DOTA. The collection includes 2806 remote sensing images, each of which is over 4000×4000 in size. Most of the objects in the DOTA dataset are characterized by large aspect ratio, tiny pixels, and dense distribution, which demands a detection model of high performance. The provider of DOTA v1.0 annotated 15 types of objects: Plane, Baseball dia-mond, Bridge, Ground field track, Smallve-hicle, Large vehicle, Ship, Tennis court, Basketball court, Storage tank, Soccer ball field,Roundabout, Harbor, Swimming pool, and Helicopter, with a total of 188,282 rotating bounding boxes. We divided the data set 1/2 for training, 1/6 for validation, and 1/3 for testing. In the data preprocessing stage, we utilize the cropping approach to divide each image into 1024*1024 subgraphs with a 150-pixel overlap to avoid losing too many objects. Finally, we have 15749 images in the training set, 5297 in the validation set, and 10594 in the test set.

4.3 Comparison experiments

First, research on GMM coefficients for better Gaussian mixture model coefficients, the difference between GMIOU values and SkewIoU in terms of numerical variation for different aspect ratios was examined by trying different β combinations. The findings demonstrate that whether (β_1, β_2) is assumed to be biased towards the prediction boxes or the ground-truth boxes, the patterns of IoU changes are equivalent. The results indicate that the GMIoU trend plots are comparable when (β_1, β_2) is taken as $(1/3, 2/3)$ or$(2/3, 1/3)$. However, it is certain that (β_1, β_2) is chosen as $(1/2, 1/2)$ when the variation trend is near to SKewIoU. Hence, (β_1, β_2) is treated as $(1/2, 1/2)$ in all subsequent experiments.

1) Comparison of different loss functions

To demonstrate the performance of GMIoU, we performed experiments to compare the training results of different models in the DOTA 1.0 dataset. The mAP 50 and the accuracy of 15 categories were employed as the experiment's evaluation indicators. The experimental results were displayed in Table 1,

ISSN 2616-5775 Vol. 6, Issue 4: 68-76, DOI: 10.25236/AJCIS.2023.060409

and the results of other models were applied in the FKIoU paper. Images of test results for different detection models, as shown in Figure 3. The experimental results show that models trained using the GMIoU loss function are more accurate for detecting plane, baseball diamond, tennis court ,and basketball court objects. Compared to KFIoU, the model trained using GMIoU outperforms it on plane, bridge, small car, tennis court, basketball court, roundabout, and harbor objects. The study mentioned above showed that the models enhance the detection of small items and large aspect ratio objects. It can be concluded that GMIoU is more sensitive to small and large aspect ratio objects and can train models more effectively. It has also been observed that the prediction performance of square-like planes and roundabouts has improved due to the addition of the DCL angle classification approach, which effectively eliminates angle periodicity and improves angle prediction accuracy. Besides that, the experimental findings show that the model trained with GMIoU loss has higher slippage in identifying massive vehicle objects. Furthermore, the experimental findings show that the model trained with the GMIoU loss has higher slippage in detecting of large vehicle objects. When viewing the image of the test result, it is evident that the large vehicle is recognized as two objects, large vehicle and small vehicle, indicating that more work is needed to better the detection performance of the large vehicle.

Table 1: Method denotes the various models, such as H-104 for ResNet and R-101 for ResNet-101. In DOTA v1.0, the table column shows 15 AP of various objects, and the model's mAP value. The color red denotes the best performance.

Method	PIoU (2020)	O^2-DNet	DAL (2021d)	P-RSDet (2020)	BBAVectors (2021)	DRN (2020)	DCL (2021a)	GWD (2021c)	KFIoU	GMIoU (our)
Backbone	DLA-34	H-104	R-101	R-101	R-101	H-104	R-152	R-152	R-152	R-152
PL	80.90	89.31	88.61	88.58	88.35	89.71	89.1	86.96	89.80	89.65
BD	69.70	82.14	79.69	77.83	79.96	82.34	84.13	83.88	85.80	84.95
BR	24.10	47.33	46.27	50.44	50.69	47.22	50.15	54.36	58.90	56.4
GTF	60.20	61.21	70.37	69.29	62.18	64.10	73.57	77.53	81.30	79.10
SV	38.30	71.32	65.89	71.10	78.43	76.22	71.48	74.41	26.70	76.53
LV	64.40	74.03	76.10	75.79	78.98	74.43	58.13	68.48	67.40	66.80
SH	64.80	78.62	78.53	78.66	87.94	85.84	78.00	80.34	77.90	79.92
TC	90.90	90.76	90.84	90.88	90.85	90.57	90.89	86.62	90.80	91.23
BC	77.20	82.23	79.98	80.10	83.58	86.18	86.64	83.41	86.70	89.60
ST	70.40	81.36	78.41	81.71	84.35	84.89	86.78	85.55	68.50	85.47
SBF	46.50	60.93	58.71	57.92	54.13	57.65	67.97	73.47	64.80	72.86
RA	37.10	60.17	62.02	63.03	60.24	61.93	67.25	67.77	59.00	70.12
HA	57.10	58.21	69.23	66.30	65.22	69.30	65.63	72.57	74.60	76.10
SP	61.90	66.98	71.32	69.77	64.28	69.63	74.06	75.76	65.60	73.00
HC	64.00	61.03	60.65	63.13	55.70	58.48	67.05	73.40	69.49	61.70
mAP	60.50	71.04	71.78	72.3	72.32	73.23	74.06	76.30	70.00	74.50

The detection performance of the model trained with different loss functions is represented in the image below. The Figure 3 categories are mostly focused on ship, harbor, and vehicle. The detection performance of the model trained with L1 loss and KFIoU is represented by groups a and b, respectively, while the model trained with GMIoU is represented by group c.

(a) L1 loss

(b) KFIoU

ISSN 2616-5775 Vol. 6, Issue 4: 68-76, DOI: 10.25236/AJCIS.2023.060409

(c) Our (GMIoU)

Figure 3: Shows a comparison of visualizations using the DOTA 1.0 dataset, with RetinaNet.

2) Comparison of convergence speed

In order to verify the stability and convergence speed of GMIoU, this experiment uses GMIoU and KFIoU methods to train the network in the same network model and compares the training process of both. The experimental environment is the same as above, with 12 epochs trained on the DOTA 1.0 dataset. The experimental results are shown in Figure 4, from which we can see that GMIoU loss decreases smoothly and faster. FKIoU, on the other hand, has anomalies at the 3rd epoch and fails to converge until the end of training. It can be inferred that KFIoU requires more strict set of training parameters, and the convergence process is not stable. We also find that the KFIoU loss value is slightly higher than the GMIoU loss in the first three training epochs, which is because GMIoU can fit the actual SkewIoU variation more accurately. From this, it can be inferred that GMIoU has better generalization ability, faster convergence and more stable training process.

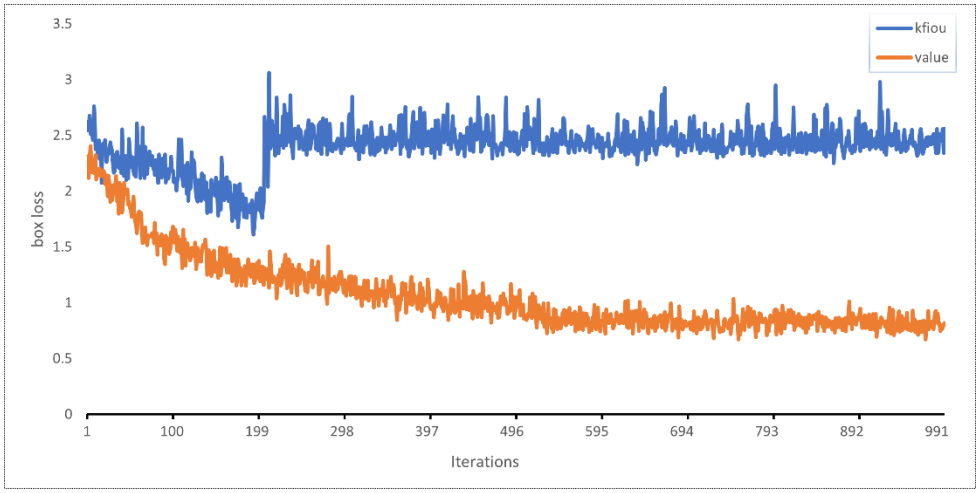


Figure 4: Comparison of the training processes at KFIoU and GMIoU

5. Conclusion

In this article, we propose a combination of angle classification and Gaussian distribution transformation method to improve the detection performance of square-like and large aspect ratio objects, which is 0.44% better than DCL and 4.5% better than KFIoU in mAP. Moreover, we propose GMIoU, a more efficient Gaussian-based loss function for rotating object recognition that has been empirically proven to be more accurate in capturing the trend of IoU between rotating bounding boxes and has a greater improvement in training speed than KFIoU. The model already has good results at 12 epochs when training the DOTA dataset with the addition of GMIoU. Despite the good results of GMIoU, the loss function still has many things that could be improved, mainly in the following three aspects .Firstly, how to accurately calculate the area of the Gaussian mixture model; second, a more scientific method has not been explored in the setting of hyperparameter β; thirdly, how to effectively integrate the centroid loss into GMIoU. These urgent problems will become the next major point of research direction.

References

[1] NIE Guang tao, HUANG Hua, A Survey of Object Detection in Optical Remote Sensing Images. ACTA AUTOMATICA SINICA 47.08(2021):1749-1768. doi:10.16383/j.aas.c200596.
[2] Girshick Ross, et al. "Rich feature hierarchies for accurate object detection and semantic

ISSN 2616-5775 Vol. 6, Issue 4: 68-76, DOI: 10.25236/AJCIS.2023.060409

segmentation." Proceedings of the IEEE conference on computer vision and pattern recognition. 2014.
[3] He Kaiming, et al. "Spatial pyramid pooling in deep convolutional networks for visual recognition." IEEE transactions on pattern analysis and machine intelligence 37.9 (2015): 1904-1916.
[4] Girshick Ross. "Fast r-cnn." Proceedings of the IEEE international conference on computer vision. 2015.
[5] Ren Shaoqing, et al. "Faster r-cnn: Towards real-time object detection with region proposal networks." Advances in neural information processing systems 28 (2015).
[6] Redmon Joseph, et al. "You only look once: Unified, real-time object detection." Proceedings of the IEEE conference on computer vision and pattern recognition. 2016.
[7] LIAO Yurong1, et al. Research progress of deep learning-based object detection of optical remote sensing image. Journal on Communications 43.05(2022):190-203.
[8] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In Advances in Neural Information Processing Systems,pp. 91–99, 2015.
[9] Yang X, Liu Q, Yan J, et al. R3Det: Refined single-stage detector with feature refinement for rotating object[J/OL].(2020-2-21)[2021-10-09].
[10] Yang Xue, et al. "Scrdet: Towards more robust detection for small, cluttered and rotated objects." Proceedings of the IEEE/CVF International Conference on Computer Vision. 2019.
[11] LI Z H, WANG Z P, HE Y T. Aerial photography dense small target detection algorithm based on adaptive cooperative attention mechanism[J]. Acta Aeronautica et Astronautica Sinica, 2023, 44(8):327944(in Chinese). doi: 10.7527/S1000- 6893.2022.27944
[12] YANG X, YAN J C, HE T, et al. On the arbitrary-oriented object detection: Classification based approaches revisited [J/OL]. (2021-3-27)[2021-10-09].
[13] YANG X, HOU L P, ZHOU Y, et al. Dense label encoding for boundary discontinuity free rotation detection[J/OL].(2021-5-25).2021-10-09]
[14] Rezato Figurehi H, Tsoi N, Gwak J Y, et al. Generalized intersection over union: A metric and a loss for bounding box regression[C]//Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2019: 658-666.
[15] Chen Dong, Miao Duoqian, Control Distance IoU and Control Distance IoU Loss for Better Bounding Box Regression, Pattern Recognition, Volume 137,2023,109256, ISSN 0031-3203,
[16] Zheng Z., et al. "Distance-IoU Loss: Faster and Better Learning for Bounding Box Regression." arXiv (2019).
[17] Zhang Y F, Ren W, Zhang Z, et al. Focal and efficient IOU loss for accurate bounding box regression [J]. arXiv preprint arXiv:2101.08158, 2021.
[18] Chen Z. et al. (2020). PIoU Loss: Towards Accurate Oriented Object Detection in Complex Environments. In: Vedaldi, A., Bischof, H., Brox, T., Frahm, JM. (eds) Computer Vision – ECCV 2020. ECCV 2020. Lecture Notes in Computer Science, vol 12350. Springer, Cham.
[19] Xue Yang, Junchi Yan, Qi Ming, Wentao Wang, Xiaopeng Zhang, and Qi Tian. Rethinking rotated object detection with gaussian wasserstein distance loss. In International Conference on Machine Learning, pp. 11830–11841. PMLR, 2021c.
[20] Xue Yang, Xiaojiang Yang, Jirui Yang, Qi Ming, Wentao Wang, Qi Tian, and Junchi Yan. Learning high-precision bounding box for rotated object detection via kullback-leibler divergence. Advances in Neural Information Processing Systems, 34, 2021d.
[21] Yang X., Zhou Y., Zhang G., Yang J., Wang W., Yan J. & Tian Q. (2022). The kfiou loss for rotated object detection. arXiv preprint arXiv:2201. 12558, 2022