Rotated Object Detection via Scale-Invariant Mahalanobis Distance in Aerial Images

Siyang Wen, Wei Guo, Yi Liu, and Ruijie Wu

Abstract—Rotated object detection in aerial images is a meaningful yet challenging task as objects are densely arranged and have arbitrary orientations. The eight-parameter (coordinates of box vectors) methods in rotated object detection usually use $l_2$-norm losses (L1 loss, L2 loss, and smooth L1 loss) as loss functions. As $l_p$-norm losses are mainly based on non-scale-invariant Minkowski distance, using $l_p$-norm losses will lead to inconsistency with the detection metric rotational Intersection-over-Union (IoU) and training instability. To address the problems, we use Mahalanobis distance to calculate loss between the predicted and the target box vertices’ vectors, proposing a new loss function called Mahalanobis distance loss (MDL) for eight-parameter rotated object detection. As Mahalanobis distance is scale-invariant, MDL is more consistent with detection metric and more stable during training than $l_p$-norm losses. To alleviate the problem of boundary discontinuity like other eight-parameter methods, we further take the minimum loss value to make MDL continuous at boundary cases. We achieve state-of-the-art performance on DOTA-v1.0 with the proposed value to make MDL continuous at boundary cases. We achieve eight-parameter methods, we further take the minimum loss to make MDL continuous at boundary cases. We achieve state-of-the-art performance on DOTA-v1.0 with the proposed method MDL. Furthermore, compared to the experiments using smooth L1 loss and approximate SkewIoU loss, we find that MDL performs better in rotated object detection.

Index Terms—Aerial images, Mahalanobis distance, rotated object detection.

I. INTRODUCTION

Object detection in remote sensing has a wide range of applications in city planning, disaster rescue, and military field, which has been developing rapidly nowadays [1], [2], [3]. Different from detecting horizontal objects in general images, object detection in aerial images focuses more on objects’ orientations as objects are densely arranged and have arbitrary orientations. Therefore, rotated object detection now is applied to aerial images for high-precision detection.

Rotated object detection originates from horizontal object detection and uses the same frameworks but requires some changes due to the extra property orientation. Horizontal object detection generally uses the coordinates of the center, height, and width to represent horizontal boxes, which can easily get the Intersection-over-Union (IoU) between the ground truth and the predicted box. As the IoU is the way of evaluating predicted results and is differentiable, IoU loss [4] (calculated by 1-IoU) and its improved methods (e.g., Generalized IoU (GIoU) loss [5] and Distance-IoU (DIoU) loss [6]) are widely used as loss functions. However, the same idea does not work in rotated object detection as the rotational IoU (aka., SkewIoU) is much more complex and is not differentiable due to the uncertainty of the intersection of two rotated boxes. Therefore, $l_p$-norm losses (L1 loss, L2 loss, and smooth L1 loss [7]) mainly based on Minkowski distance are frequently used when regressing rotated box parameters.

There are usually two ways to describe the oriented bounding box (OBB) of a target object: 1) the coordinates of the center, width, height, and angle of the box (five parameters) and 2) the coordinates of box vectors (eight parameters) [8]. When using $l_p$-norm losses for the regression of five parameters, prediction accuracy may get hurt as the orientation is in the measurement system different from others and a slight deviation will cause a drastic change in SkewIoU. As for eight-parameter methods, although the coordinates of box vectors have the same unit, it can still damage prediction performance to some extent as $l_p$-norm losses are not scale-invariant.

Fig. 1 shows that the values of 1-SkewIoU (same idea as IoU loss) keep invariant, while $l_p$-norm losses change significantly when the heights and widths of the OBBs are doubled. It reveals that the scale of the OBB has a great influence on $l_p$-norm losses, which exacerbates the inconsistency with the detection metric SkewIoU. Moreover, it can also lead to training instability as loss fluctuates when the scale varies greatly. As it is very common for aerial images to have targets with different scales, the impact of these problems cannot be ignored.

Fig. 1. Changes of 1-SkewIoU and $l_p$-norm losses when double the scale.

Manuscript received 7 April 2022; revised 1 July 2022; accepted 28 July 2022. Date of publication 8 August 2022; date of current version 19 August 2022. This work was supported by the National Key Technologies Research and Development Program of China under Grant 2016YFC0802500.

(Corresponding author: Wei Guo.)

Siyang Wen and Ruijie Wu are with the State Key Laboratory of Information Engineering in Surveying, Mapping and Remote Sensing, Wuhan University, Wuhan 430072, China (e-mail: siyangwen@whu.edu.cn; jerrywu@whu.edu.cn).

Wei Guo is with the State Key Laboratory of Information Engineering in Surveying, Mapping and Remote Sensing, Wuhan University, Wuhan 430072, China, and also with the Nanjing Beidou Innovation and Applied Technology Research Institute Company, Ltd., Nanjing 211500, China (e-mail: guowei-lmars@whu.edu.cn).

Yi Liu is with the School of Geodesy and Geomatics, Wuhan University, Wuhan 430072, China (e-mail: yliu@sgg.whu.edu.cn).

Digital Object Identifier 10.1109/LGRS.2022.3197617

1558-0571 © 2022 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.
See https://www.ieee.org/publications/rights/index.html for more information.
In order to reduce the impact of the above issues, in this letter, we use Mahalanobis distance [9] to calculate loss between the predicted and ground truth box vertices’ vectors. Different from Minkowski distance, Mahalanobis distance is scale-invariant as it takes into account the correlations of the data. Therefore, our proposed loss function Mahalanobis distance loss (MDL) is more consistent with the detection metric SkewIoU and more stable during training. However, due to angle periodicity, MDL still has the boundary discontinuity problem like all other eight-parameter methods, which can cause a loss upsurge when two close boxes are on both sides of an axis [8]. To deal with the issue, we take the minimum loss value [8] to make MDL continuous at boundary cases. The effectiveness of MDL is confirmed by experiments implemented on DOTA-v1.0 [10].

In summary, the contributions of this letter are as follows.

1) We introduce a new loss function called MDL for eight-parameter rotated object detection.
2) MDL can promote the consistency with detection metric SkewIoU under the same condition show that MDL performs better.
3) The effectiveness of MDL is confirmed by experiments implementing on DOTA-v1.0 [10].

In this section, we first introduce our new loss function called MDL for eight-parameter rotated object detection, then analyze its advantages in rotated object detection. Finally, we demonstrate the overall loss function design with our architecture.

A. Mahalanobis Distance Loss for Oriented Bounding Box

Inspired by the fact that \( l_n \)-norm losses are mainly based on Minkowski distance, we propose a new loss function MDL based on Mahalanobis distance. Mahalanobis distance, which is scale-invariant, can measure the standard deviations between the predicted and ground truth box vectors. The Mahalanobis distance between \( \mathbf{m} \) and \( \mathbf{n} \) is defined as

\[
\text{MDL}(\mathbf{m}, \mathbf{n}) = \sqrt{(\mathbf{m} - \mathbf{n})^T \Sigma^{-1} (\mathbf{m} - \mathbf{n})}
\]

where \( \Sigma \) refers to the covariance matrix of the distribution.

An oriented box can be viewed as a 2-D distribution that consists of four vertices of the box. Suppose the vertices’ vectors of the oriented box \( B(a, b, c, d) \) are \( a = (x_a, y_a), b = (x_b, y_b), c = (x_c, y_c), \) and \( d = (x_d, y_d) \), then the covariance matrix \( \Sigma \) of the distribution can be calculated as

\[
\Sigma = \begin{pmatrix}
\frac{\sum_{i=a}^{d} (x_i - \bar{x})(x_i - \bar{x})}{N - 1} & \frac{\sum_{i=a}^{d} (x_i - \bar{x})(y_i - \bar{y})}{N - 1} \\
\frac{\sum_{i=a}^{d} (y_i - \bar{y})(x_i - \bar{x})}{N - 1} & \frac{\sum_{i=a}^{d} (y_i - \bar{y})(y_i - \bar{y})}{N - 1}
\end{pmatrix}
\]

where \( N \) refers to the number of points (\( N = 4 \) here), and \( \bar{x} \) and \( \bar{y} \) refer to the average value of vectors’ \( x \) values and \( y \) values, respectively. Note that \( \Sigma \) does not change regardless of the starting point of the vectors.

Thus, MDL between the predicted oriented box \( B(a, b, c, d) \) and the target oriented box \( B^*(a^*, b^*, c^*, d^*) \) can be expressed as

\[
\text{MDL}(B, B^*) = \frac{1}{N} \sum_{i=a}^{d} \text{MD}(i, i^*)
\]

\[
= \frac{1}{N} \sum_{i=a}^{d} \sqrt{((x_i, y_i) - (x_i^*, y_i^*))^T \Sigma^{-1} ((x_i, y_i) - (x_i^*, y_i^*))}
\]

where \( N = 4 \) as we use the mean value of four vertices’ \( \text{MDL} \) for stable training, \( \text{MDL} \) is scale-invariant, which makes training more stable than \( l_n \)-norm losses.

B. Analysis of Mahalanobis Distance Loss

By using the covariance matrix \( \Sigma \), Mahalanobis distance between two points is independent of the measurement units of the original data, which thus is scale-invariant [9]. Derived from Mahalanobis distance, MDL is also scale-invariant. Fig. 2 plots different loss curves when only the scale is changing. We can find that the curve of MDL keeps invariant as same as 1-SkewIoU, while L1 loss and smooth L1 loss get larger and larger with the scale, which indicates that MDL is more consistent with detection metric and more stable during training at such scale cases than \( l_n \)-norm losses.

Moreover, we also compare MDL with \( l_n \)-norm losses at different angles, center shifting, and aspect ratio cases by changing only one factor. As shown in Fig. 3, the trends of MDL loss curves are more consistent with 1-SkewIoU and more stable in all cases, while the curves of L1 loss and smooth L1 loss are more steep than 1-SkewIoU.

Overall, the advantages of using MDL in rotated object detection can be summarized as follows.

1) MDL is more consistent with the detection metric SkewIoU than \( l_n \)-norm losses at different scale, angle, center shifting, and aspect ratio cases.
2) MDL can get loss values that do not fluctuate much due to its property of scale invariance, which makes training more stable than \( l_n \)-norm losses.

C. Overall Loss Function Design

As anchor-based methods suffer from hyperparameters for setting anchor boxes [12], we choose the representative...
For, the Gaussian kernel \( \exp \) directly penalize points near the target center point which gets the target center point is in CenterNet. However, it is unable to achieve faster speed at comparable accuracy [1]. With the architecture shown in Fig. 4, our overall loss function can be divided into three parts: heatmap, box vertices, and offset.

1) Heatmap: Heatmap is used to demonstrate where the target center point is in CenterNet. However, it is unable to directly penalize points near the target center point which get large IoU between the predicted box and the target box. Therefore, the Gaussian kernel \( \exp(-((x - \bar{x})^2 + (y - \bar{y})^2/2\sigma^2)) \) is used to generate the ground truth and the deviation \( \sigma \) is adapted to the target object size. As we only view center points as the positive samples, while other points are negative samples in training, we use an improved Focal loss to balance positive and negative samples as in [14]

\[
L_h = -\sum_i \left\{ \begin{array}{ll}
(1 - \hat{Y}_i)^\alpha \log(\hat{Y}_i), & \text{if } Y_i = 1 \\
(1 - \hat{Y}_i)^\beta (1 - \hat{Y}_i), & \text{otherwise}
\end{array} \right.
\] (4)

where \( \hat{Y}_i \) and \( Y_i \) refer to the predicted and the ground truth value, respectively, \( i \) indicates the pixel locations on the heatmap, and \( \alpha \) and \( \beta \) are hyperparameters for balancing difficult and easy samples. The values of \( \alpha \) and \( \beta \) are empirically set to 2 and 4.

2) Box Vertices: We use the four vectors from box vertices to the center point to describe the OBB, containing top-left \( a \), top-right \( b \), bottom-right \( c \), and bottom-left \( d \), as shown in Fig. 5. We use the proposed method MDL to replace the common \( l_2 \)-norm losses due to MDL’s property of scale-invariance and its high consistency with detection metric. However, like other eight-parameter methods, MDL still cannot solve the problem of boundary discontinuity due to the periodicity of the angle. As for this problem, we are inspired by the modulated rotation loss introduced in RSDet [8]. RSDet first sorts the predicted points and then takes the minimum loss of sorted points themselves, the points moved forward and backward by one place. We extend this method by dropping the sorting step and taking the minimum loss of four losses to make MDL continuous at boundary cases

\[
L_b = \min_k \left\{ \begin{array}{l}
\text{MDL} (B_k(a, b, c, d), B_k^*(a^*, b^*, c^*, d^*)) \\
\text{MDL} (B_k(b, c, d, a), B_k^*(a^*, b^*, c^*, d^*)) \\
\text{MDL} (B_k(c, d, a, b), B_k^*(a^*, b^*, c^*, d^*)) \\
\text{MDL} (B_k(d, a, b, c), B_k^*(a^*, b^*, c^*, d^*))
\end{array} \right. \] (5)

where \( B(a, b, c, d) \) and \( B^*(a^*, b^*, c^*, d^*) \) refer to the predicted and the ground truth box, respectively, \( k \) indicates the objects, and MDL(\( B, B^* \)) is according to (3).

3) Offset: As the image is downsampled by the factor of 4 in CenterNet, there will be an accuracy damage when the feature map is remapped to the original image. Therefore, an additional offset is used to compensate for the damage. The ground truth is the difference between the target center point’s coordinates downsampled by 4 and their integer values. Although offset loss is optional, for higher precision, we choose to adopt this. We also use Mahalanobis distance to make the offset loss adapted to the size of the OBB. The loss between the predicted offset \( O = (x, y) \) and the ground truth offset \( O^* = (x^*, y^*) \) is calculated as follows:

\[
L_o = \sum_k \text{MDL} (O_k, O_k^*) \\
= \sum_k \sqrt{(x_k - x_k^*)^2 + (y_k - y_k^*)^2} \Sigma^{-1} \left( (x_k, y_k) - (x_k^*, y_k^*) \right)
\] (6)

where \( \Sigma \) refers to the same covariance matrix as in Box vertices, and \( k \) indicates the objects.

In summary, our overall loss function is designed as follows:

\[
L = \frac{1}{N} (L_h + L_b + L_o)
\] (7)

where \( N \) indicates the number of ground truth objects.
III. Experiments

A. Dataset

We use DOTA-v1.0 [10] to validate the effectiveness of our method. DOTA-v1.0 contains 2806 aerial images and a total of 188,282 instances in 15 categories: plane (PL), ship (SH), storage tank (ST), baseball diamond (BD), tennis court (TC), basketball court (BC), ground track field (GTF), harbor (HB), bridge (BR), large vehicle (LV), small vehicle (SV), helicopter (HC), roundabout (RA), soccer ball field (SBF), and swimming pool (SP). We crop the images into 600 × 600 patches with a gap of 100 at the scales of 0.5 and 1.0. After cropping, we get a trainval set of 69,337 images and a test set of 35,777 images.

B. Implementation and Testing Details

1) Implementation Details: The experiments are implemented using PyTorch. We use Adam [15] as our optimizer with an initial learning rate of 1.25 × 10⁻⁴. To make the network converge better, we bind an exponentially decaying learning rate scheduler to the optimizer with the decay factor of 0.96. For increasing the number and diversity of training samples, images are preprocessed by data augmentation including random cropping and random flipping. We use a batch size of 48 over six GeForce RTXTM 3090 GPUs to implement our experiments.

2) Testing Details: All the results are derived from the 50 epoch models trained on DOTA-v1.0. From the output heatmap, we adopt the top-500 points whose scores are more than 0.1 as the center points of objects. To obtain the coordinates of the predicted box, we first add the coordinates of the center point and the corresponding offset to get accurate center coordinates, then add the center coordinates and the box vertices’ vectors, and finally multiply the added values by the downsampling factor 4. As we use multiscale images (0.5 and 1.0) for testing, we apply nonmaximum-suppression (NMS) [16] to the output results with the threshold of 0.1 to get the final merged results.

C. Ablation Studies

1) MDL Forms: As mentioned before, the covariance matrix Σ used in MDL can be calculated over the ground truth box vectors (MDL-t) or the predicted box vectors (MDL-p). Table I shows that the mAP of MDL-p is 1.32% higher than that of MDL-t, indicating that using Σ calculated by predicted vectors can further facilitate the regression of predicted values.

2) Minimum Loss Value: Minimum loss value can solve the problem of boundary discontinuity. As can be seen in Table I, the minimum loss value improves the performance of MDL-t and MDL-p by 0.97% and 1.48%, respectively.

3) Comparison With Smooth L1 Loss: To prove the effectiveness of MDL, we use smooth L1 loss to calculate the loss of box vertices and offset and adopt the minimum loss value of box vertices to avoid boundary discontinuity. Table I shows that MDL-t’ and MDL-p’ achieve 0.35% and 2.18% improvement over smooth L1 loss*. Furthermore, Fig. 5 indicates that MDL is more stable than smooth L1 loss during training.

4) Comparison With SkewIoU Loss: An approximate SkewIoU loss is proposed in R²Det [11], which solves boundary discontinuity and inconsistency with SkewIoU. We apply its core idea to our eight-parameter baseline method via smooth L1 loss and find that using SkewIoU function −ln(SkewIoU) has the best performance of 75.23% compared to other SkewIoU functions. It can be seen that in Table I, its best performance is still 0.93% lower than MDL-p*.

D. Further Comparison

For further comparing the proposed method MDL with other methods, we choose both five-parameter and eight-parameter methods with different architectures, as shown in Table II. Most of these methods use the smooth L1 loss as their loss functions, except for SCRDet [17] and RSDet [8], which introduce IoU-Smooth L1 loss based on smooth L1 loss and use the modulated rotation loss based on L1 loss, respectively. From Table II, we find that most results of eight-parameter methods are better than five-parameter methods, which implies that eight-parameter methods do solve the problem of measurement discontinuity in five-parameter methods to some extent. Among these methods, SCRDet gains over other five-parameter methods, which implies that eight-parameter methods do solve the problem of measurement discontinuity in five-parameter methods. These methods all manage to solve the boundary discontinuity problem, which indicates that boundary discontinuity should be addressed for better performance.

By using Mahalanobis distance to calculate loss and taking the minimum loss value to solve the problem of boundary discontinuity, the proposed method MDL-p* achieves 76.16% in mAP, exceeding all the above methods. As most of these methods use the smooth L1 loss as the loss function, the results of MDL-p* show the superiority of MDL. Besides, MDL-p* gains the best or second-best results in most categories. In particular, MDL-p* outperforms other methods in terms of harbor (73.39%), large vehicle (81.25%), and helicopter (68.31%). For further inspection, the visualization results of MDL-p* on DOTA-v1.0 are illustrated in Fig. 6. Although DOTA-v1.0 consists of complex images with targets of different sizes and types, our proposed method can still achieve good performance.
TABLE II

| Five-parameter methods | PL | SH | ST | BD | TC | BC | GTF | HB | BR | LV | SV | IC | RA | SBF | SP | mAP |
|------------------------|----|----|----|----|----|----|-----|----|----|----|----|----|----|-----|----|-----|
| FR-O [10]              | 79.42 | 37.16 | 59.28 | 77.13 | 89.41 | 69.64 | 64.04 | 47.89 | 17.7 | 38.02 | 35.3 | 46.3 | 52.19 | 50.3 | 47.4 | 54.13 |
| ROITrans [18]          | 89.64 | 66.57 | 76.75 | 78.52 | 90.95 | 79.46 | **75.92** | 62.54 | 43.44 | 62.97 | 68.81 | 55.56 | 56.73 | 59.04 | 61.29 | 69.56 |
| CAD-Net [19]           | 87.89 | 76.60 | 73.39 | 82.40 | 90.90 | 79.20 | 73.50 | 62.00 | 49.40 | 63.50 | 71.10 | 62.20 | 60.90 | 48.40 | 67.00 | 69.90 |
| SCRDet [17]            | 89.98 | 72.41 | **86.86** | 89.65 | 90.85 | **87.94** | 68.36 | 66.25 | 52.99 | 60.32 | 68.36 | 65.21 | **66.68** | **65.02** | 68.24 | 72.61 |
| Eight-parameter methods | ICN [2] | 81.36 | 69.98 | 78.20 | 74.30 | 90.76 | 79.06 | 70.32 | 67.02 | **47.70** | **67.82** | **64.89** | 50.23 | 62.90 | 53.64 | 64.17 | 68.16 |
| O2-DNet [20]           | 89.30 | 78.70 | 82.90 | 83.30 | 90.90 | 79.90 | 72.10 | 64.60 | 50.10 | **80.40** | **78.26** | 63.96 | 65.00 | 60.20 | 68.90 | 72.80 |
| RSDet [8]              | **90.1** | 73.6 | 84.7 | 82.0 | **91.2** | 87.1 | 68.5 | 66.1 | 53.8 | 78.7 | 70.2 | 63.7 | **68.2** | **64.3** | 69.3 | 74.1 |
| BBAVectors [1]         | 88.63 | **88.06** | 86.39 | 84.06 | 90.87 | 87.23 | **74.08** | 67.10 | 52.10 | 80.40 | 78.26 | 63.96 | 65.00 | 60.20 | 68.90 | 72.80 |
| MDL-p* (ours)          | 88.75 | **87.58** | 86.94 | **84.96** | 90.84 | 87.85 | 70.96 | 73.39 | 53.20 | 81.20 | **76.46** | **68.31** | 63.47 | 57.65 | 71.87 | **76.16** |

![Fig. 6. Visualization results of MDL-p on DOTA-v1.0.](image)

IV. CONCLUSION

In this letter, we propose a new loss function MDL for eight-parameter rotated object detection. The proposed method MDL calculates the loss between predicted and target box vertices’ vectors via scale-invariant Mahalanobis distance, which can thus alleviate the problems of inconsistency with the detection metric SkewIoU and training instability when using $l_p$-norm losses. The experimental results on DOTA-v1.0 confirm the effectiveness of MDL. Besides, the comparative experiments show that in rotated object detection MDL performs better than the widely used smooth L1 loss and the approximate SkewIoU loss. However, like all other eight-parameter methods, MDL still faces the problem of boundary discontinuity, which we manage to solve by taking the minimum loss value. Therefore, in the future, we would like to explore a way to fundamentally solve boundary discontinuity for all eight-parameter methods. Furthermore, we would like to apply MDL to quadrilateral or polygon object detection as MDL can also be used on point sets.

REFERENCES

[1] J. Yi, P. Wu, B. Liu, Q. Huang, H. Qu, and D. Metaxas, “Oriented object detection in aerial images with box boundary-aware vectors,” in Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV), Jan. 2021, pp. 2150–2159.

[2] S. M. Azimi, E. Víg, R. Bahmanyar, M. Körner, and P. Reintrert, “Towards multi-class object detection in unconstrained remote sensing imagery,” in Proc. Asian Conf. Comput. Vis., 2018, pp. 150–165.

[3] F. Shi, T. Zhang, and T. Zhang, “Orientation-aware vehicle detection in aerial images via an anchor-free object detection approach,” IEEE Trans. Geosci. Remote Sens., vol. 59, no. 6, pp. 5221–5233, Jun. 2020.

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

[5] H. Rezatofighi, N. Tsoi, J. Gwak, A. Sadeghian, I. Reid, and S. Savarese, “Generalized intersection over union: A metric and a loss for bounding box regression,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2019, pp. 658–666.

[6] Z. Zheng, P. Wang, W. Liu, J. Li, R. Ye, and D. Ren, “Distance-IoU loss: Faster and better learning for bounding box regression,” in Proc. AAAI Conf. Artif. Intell., vol. 34, no. 7, 2020, pp. 12993–13000.

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

[8] W. Qian, X. Yang, S. Peng, Y. Guo, and J. Yan, “Learning modulated loss for rotated object detection,” 2019, arXiv:1911.08299.

[9] R. De Maesschalck, D. Jouan-Rimbaud, and D. L. Massart, “The Mahalanobis distance,” Chemometrics Intell. Lab. Syst., vol. 50, no. 1, pp. 1–18, 2000.

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

[11] X. Yang, J. Yan, Z. Feng, and T. He, “R3Det: Refined single-stage detector with feature refinement for rotating object,” in Proc. AAAI Conf. Artif. Intell., vol. 35, no. 4, May 2021, pp. 3163–3171.

[12] K. Duan, S. Bai, L. Xie, H. Qi, Q. Huang, and Q. Tian, “CenterNet: Keypoint triplets for object detection,” in Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV), Oct. 2019, pp. 6569–6578.

[13] X. Zhou, D. Wang, and P. Krähenbühl, “Objects as points,” 2019, arXiv:1904.07850.

[14] H. Law and I. Deng, “CornerNet: Detecting objects as paired keypoints,” in Proc. Eur. Conf. Comput. Vis. (ECCV), 2018, pp. 734–750.

[15] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” 2014, arXiv:1412.6980.

[16] A. Neubeck and L. Van Gool, “Efficient non-maximum suppression,” in Proc. Int. Conf. Pattern Recognit., vol. 3, Aug. 2006, pp. 850–855.

[17] X. Yang et al., “SCRDet: Towards more robust detection for small, cluttered and rotated objects,” in Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV), Oct. 2019, pp. 8232–8241.

[18] J. Ding, N. Xue, Y. Long, G.-S. Xia, and Q. Lu, “Learning RoI transformer for oriented object detection in aerial images,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2019, pp. 2849–2858.

[19] G. Zhang, S. Lu, and W. Zhang, “CAD-Net: A context-aware detection network for objects in remote sensing imagery,” IEEE Trans. Geosci. Remote Sens., vol. 57, no. 12, pp. 10015–10024, Aug. 2019.

[20] H. Wei, Y. Zhang, Z. Chang, H. Li, H. Wang, and X. Sun, “Oriented objects as pairs of middle lines,” ISPRS J. Photogramm. Remote Sens., vol. 169, pp. 268–279, Nov. 2020.