Few Could Be Better Than All: Feature Sampling and Grouping for Scene Text Detection

Jingqun Tang[1], Wenqing Zhang[2], Hongye Liu[1], MingKun Yang[2],
Bo Jiang[1], Guanglong Hu[1], Xiang Bai[2*]
[1]NetEase, [2]Huazhong University of Science and Technology
{jingquntang, liuhongye1998, bjiang002, guanglong.hu}@163.com,
{wenqingzhang, yangmingkun, xbai}@hust.edu.cn

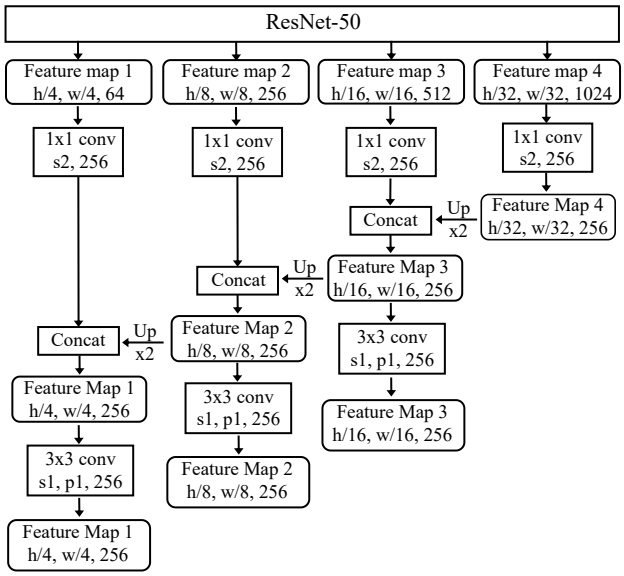

Figure 1. The structure of our feature pyramid network equipped with ResNet-50.

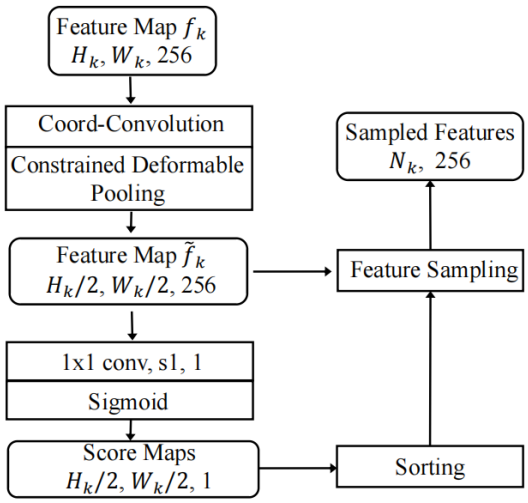

Figure 2. The pipeline of feature sampling for each input feature map f_k.

1. Implementation Details

1.1. Network Architecture

Our proposed transformer-based architecture is composed of a backbone network, a feature sampling network, and a feature grouping network.

The backbone is the basic feature pyramid network (FPN) [8] equipped with ResNet-50 [6] as shown in Fig. 1, The produced feature maps in three different scales (*i.e.* $1/4$, $1/8$, $1/16$) are used for feature sampling.

As shown in Fig. 2, each feature map is first fed into a Coord-Convolution layer [9] to involve position information for the incoming presentation in our feature sampling network. Next, it is down-sampled by a constrained deformable pooling adjusted from [3]. In our implementation, the predicted offsets are obtained by $\triangle \mathbf{p}_{ij} = \lambda \cdot \triangle \widehat{\mathbf{p}}_{ij} \circ (W_k, H_k)$, where $\lambda = Sigmoid(Avg(f_{ij}))$ is a learnable scaling parameter to modulate the predicted offset and f_{ij} is the feature vector at (i, j). The other symbol definitions are consistent with the original ROI deformable pooling [3]. Then, a convolution layer with a 1×1 kernel size and a Sigmoid function are employed to generate confidence score maps to distinguish representative text regions. After that, we select the features with top-N_k scores in each scale layer k, and gather them into a sequence form with a shape $(\sum_k N_k, C)$, where $C = 256$ is the channel number.

In our feature grouping network, the sampled features

*Corresponding Author

Figure 3. The bad cases of "text overlapping" in our method. The red bounding boxes denote the wrong predictions, and the green ones are the right predictions.

Figure 4. The visualization of feature sampling and grouping. We visualize the attention weights for one text instance's features in the last transformer layer. The weight value increases from 0 to 1 as the color changes from blue to red. The output feature for the text instance in a red bounding box is mainly aggregated from the inner text point features.

are first concatenated with position embeddings. Then, we adopt four basic transformer encoder layers as those in DETR [2] to model the feature relationship, and implicitly aggregate the features from the same text instance. Finally, scores and coordinates of rotated bounding boxes are obtained via a text/non-text classification head and a bounding box prediction head, which are composed of full-connected layers and Sigmoid functions.

1.2. Scale-Invariant GWD Loss

To regress the coordinates of rotated bounding boxes, we adapt the Gaussian Wasserstein Distance (GWD) loss [20] into a scale-invariant form to better balance the loss weights of text with different scales. Following the GWD loss, we first convert the rotated bounding box $\mathcal{B}(x, y, h, w, \theta)$ into a 2-D Gaussian distribution representation $\mathcal{N}(\mathbf{m}, \mathbf{\Sigma})$, where $\mathbf{m} = (x, y)$ and $\mathbf{\Sigma}$ is formulated as

$$\mathbf{\Sigma} = \begin{pmatrix} \frac{w}{2}\cos^2\theta + \frac{h}{2}\sin^2\theta & \frac{w-h}{2}\cos\theta\sin\theta \\ \frac{w-h}{2}\cos\theta\sin\theta & \frac{w}{2}\sin^2\theta + \frac{h}{2}\cos^2\theta \end{pmatrix}^2. \tag{1}$$

Then, we use the Wasserstein distance between two instances to formulate d^2 as

$$d^2 = \| \mathbf{m}_1 - \mathbf{m}_2 \|_2^2 + \mathbf{Tr}\left(\mathbf{\Sigma}_1 + \mathbf{\Sigma}_2 - 2(\mathbf{\Sigma}_1^{1/2}\mathbf{\Sigma}_2\mathbf{\Sigma}_1^{1/2})^{1/2}\right). \tag{2}$$

Due to the extreme variance of scales, the loss of small text has a negligible influence on the gradient back-propagation compared with the loss of large text. Hence, we adjust the GWD loss into a scale-invariant form as follows:

$$\widehat{\mathcal{L}}_{rbox} = \frac{1}{N_r}\sum_x (1 - \frac{1}{\tau + f(d^2(\frac{\widehat{u}_x}{|\widehat{t}_x|}, \frac{\widehat{t}_x}{|\widehat{t}_x|}))}), \tag{3}$$

where $\widehat{u}_x$ denotes the predicted rotated bounding box, $\widehat{t}_x$ denotes the target one, and $|\widehat{t}_x|$ denotes its area. N_r is the number of bounding boxes after pair-wise matching. The elements with $\widehat{\ }$ denote the matched bounding boxes or the target ones after pair-wise matching. $f(\cdot)$ represents a non-linear function, and τ is a hyper-parameter to modulate the loss. According to the GWD loss [20], we set $f(d^2) = \log(d^2 + 1)$ and $\tau = 3$. By normalizing $\widehat{u}_x$ and $\widehat{t}_x$ with the area of $\widehat{t}_x$, we can decrease the negative effect of the scale imbalance.

1.3. Training

In the training period, the data argumentation for training data includes: (1) Random Rotation, flipping, and perspective transformation; (2) Color argumentation; (3) Random cropping. In addition, both sides of the training images are randomly resized in the range between 640×640 and 1680×1680 with an interval of 64. In our loss function, we use λ_c, λ_d, and λ_f to adjust the influences of different losses. Specifically, we set λ_c to 0.5 and λ_d to 1. For λ_f, we initialize it to $1e^{-2}$, and decay it by a factor 0.1 at the 35th and 45th epoch, respectively.

Figure 5. The qualitative results of our proposed method in different cases, including multi-oriented text, long text, multi-lingual text, low-resolution text, curved text, dense text. For curved text detection, the Bezier curves' control points are drawn in red.

Method	Sampling Number	F-measure		
		IC15	TD500	MTWI
FPN+FC	64+128+256	85.7	85.5	70.6
FPN+GCN	64+128+256	87.9	87.0	72.5
Ours (RBox)	64+128+256	**89.1**	**88.1**	**75.2**

Table 1. The ablation study on feature grouping with non-transformer structures.

Methods	IC15			MLT17 val		
	P	R	F	P	R	F
Average Pooling	89.5	87.2	88.3	86.6	72.6	79.0
Deformable Pooling	89.9	87.3	88.6	86.8	72.8	79.2
Ours (RBox)	90.9	87.3	**89.1**	86.8	73.4	**79.5**

Table 2. The abalation study on the constrained deformable pooling. "P", "R", and "F" represent Precision, Recall, and F- measure, respectively.

$\widehat{\mathcal{L}}_{rbox}$	IC15			MLT17 val		
	P	R	F	P	R	F
GWD	90.2	86.6	88.4	86.7	72.6	79.0
Ours (RBox)	90.9	87.3	**89.1**	86.8	73.4	**79.5**

Table 3. The ablation study on the loss for rotated bounding boxes.

Transformer Layer	IC15			MLT17 val		
	P	R	F	P	R	F
Basic Layer	90.8	87.3	89.1	86.8	73.4	79.5
Swin Transformer Layer	90.9	88.1	**89.5**	87.2	73.4	**79.7**

Table 4. The experiment on the transformer layers in our feature grouping network.

1.4. Inference

In the inference period, we keep the aspect ratio of test images and resize the shorter sides to 768 (for TD500 and MTWI) or 1024 (for others), while the upper limit of the longer sides is 2048. Moreover, we can easily obtain the detection results without any complex post-processing. By setting a proper threshold, we only keep the predicted boxes with scores higher than the threshold. Specifically, we set it to 0.45 for the IC15 dataset, and 0.5 for other datasets.

2. Experiments

2.1. Qualitative Results

As shown in Fig. 5, we provide more qualitative results for visualization, including multi-oriented text, long text, multi-lingual text, small text, dense text, and curved text. Moreover, we also provide some bad cases of our method shown in Fig. 3. The red bounding boxes are the wrong predictions. It is hard for our method to deal with the case of "text overlapping", because the features of the overlapping text instances are quite complex and tangled. Our feature grouping module sometime fails in these cases.

As shown in Fig. 4, we show the feature grouping results of the predicted rotated bounding boxes in red. We visualize the attention weights for one text instance's features in the last transformer layer. The weight value increases from 0 to 1 as the color changes from blue to red. It means that the output features for text instances in red bounding boxes are mainly aggregated from the inner text point features (red ones).

	Method	Backbone	MS	PL	BD	BR	GTF	SV	LV	SH	TC	BC	ST	SBF	RA	HA	SP	HC	AP_{50}
Two-stage	ICN [1]	R-101	√	81.40	74.30	47.70	70.30	64.90	67.80	70.00	90.80	79.10	78.20	53.60	62.90	67.00	64.20	50.20	68.20
	RoI-Trans. [4]	R-101	√	88.64	78.52	43.44	75.92	68.81	73.68	83.59	90.74	77.27	81.46	58.39	53.54	62.83	58.93	47.67	69.56
	SCRDet [22]	R-101	√	89.98	80.65	52.09	68.36	68.36	60.32	72.41	90.85	**87.94**	86.86	65.02	66.68	66.25	68.24	65.21	72.61
	Gliding Vertex [17]	R-101		89.64	85.00	52.26	77.34	73.01	73.14	86.82	90.74	79.02	86.81	59.55	**70.91**	72.94	70.86	57.32	75.02
	CenterMap OBB [14]	R-101	√	89.83	84.41	54.60	70.25	77.66	78.32	87.19	90.66	84.89	85.27	56.46	69.23	74.13	71.56	66.06	76.03
	FPN-CSL [19]	R-152	√	90.25	**85.53**	54.64	75.31	70.44	73.51	77.62	90.84	86.15	86.69	69.60	68.04	73.83	71.10	68.93	76.17
	RSDet-II [13]	R-152	√	89.93	84.45	53.77	74.35	71.52	78.31	78.12	**91.14**	87.35	86.93	65.64	65.17	75.35	**79.74**	63.31	76.34
	Oriented R-CNN [16]	R-50		89.46	82.12	54.78	70.86	78.93	83.00	88.20	**90.90**	87.50	84.68	63.97	67.69	74.94	68.84	52.28	75.87
		R-101		88.86	83.48	55.27	76.92	74.27	82.10	87.52	**90.90**	85.56	85.33	65.51	66.82	74.36	70.15	57.28	76.28
		R-50	√	89.84	85.43	61.09	79.82	79.71	**85.35**	88.82	90.88	86.68	**87.73**	72.21	70.80	**82.42**	78.18	74.11	**80.87**
		R-101	√	**90.26**	84.74	**62.01**	80.42	79.04	**85.07**	88.52	90.85	87.24	**87.96**	72.26	70.03	**82.93**	78.46	68.05	80.52
Refine-stage	CFC-Net [11]	R-101	√	89.08	80.41	52.41	70.02	76.28	78.11	87.21	90.89	84.47	85.64	60.51	61.52	67.82	68.02	50.09	73.50
	DCL [18]	R-152	√	89.26	83.60	53.54	72.76	79.04	82.56	87.31	90.67	86.59	86.98	67.49	66.88	73.29	70.56	69.99	77.37
	RIDet [12]	R-50	√	89.31	80.77	54.07	76.38	**79.81**	81.99	**89.13**	90.72	83.58	87.22	64.42	67.56	78.08	79.17	62.07	77.62
	S^2A-Net [5]	R-101	√	89.28	84.11	56.95	79.21	**80.18**	82.93	**89.21**	90.86	84.66	87.61	71.66	68.23	78.58	78.20	65.55	79.15
	R^3Det-GWD [21]	R-152	√	89.66	84.99	59.26	**82.19**	78.97	84.83	87.70	90.21	86.54	86.85	**73.04**	67.56	76.92	79.22	74.92	80.19
	R^3Det-KLD [23]	R-50	√	89.90	84.91	59.21	78.74	78.82	83.95	87.41	89.89	86.63	86.69	70.47	70.87	76.96	**79.40**	**78.62**	80.17
		R-152	√	89.92	85.13	59.19	**81.33**	78.82	84.38	87.50	89.80	87.33	87.00	72.57	**71.35**	77.12	79.34	**78.68**	**80.63**
Single-stage	PolarDet [25]	R-101	√	89.65	**87.07**	48.14	70.97	78.53	80.34	87.45	90.76	85.63	86.87	61.64	70.32	71.92	73.09	67.15	76.64
	RDD [26]	R-101	√	89.15	83.92	52.51	73.06	77.81	79.00	87.08	90.62	86.72	87.15	63.96	70.29	76.98	75.79	72.15	77.75
	GWD [20]	R-152	√	89.06	84.32	55.33	77.53	76.95	70.28	83.95	89.75	84.51	86.06	**73.47**	67.77	72.60	75.76	74.17	77.43
	KLD [24]	R-50		88.91	83.71	50.10	68.75	78.20	76.05	84.58	89.41	86.15	85.28	63.15	60.90	75.06	71.51	67.45	75.28
		R-50	√	88.91	85.23	53.64	81.23	78.20	76.99	84.58	89.50	86.84	86.38	71.69	68.06	75.95	72.23	75.42	78.32
	Ours (RBox)	R-50		**90.36**	85.31	56.39	76.45	74.55	83.46	87.78	90.86	85.85	85.28	64.52	67.82	77.72	74.32	67.80	77.90
		R-50	√	89.81	85.19	**61.35**	76.18	79.29	84.81	88.26	90.86	**87.55**	87.42	66.89	70.10	78.40	79.28	68.48	79.59

Table 5. Detection results on the DOTA-v1.0 testing set. R-50, R-101, and R-152 denote ResNet-50, ResNet-101, and ResNet-152, respectively. MS indicates that multi-scale testing is used. Red and blue indicate the top two performances.

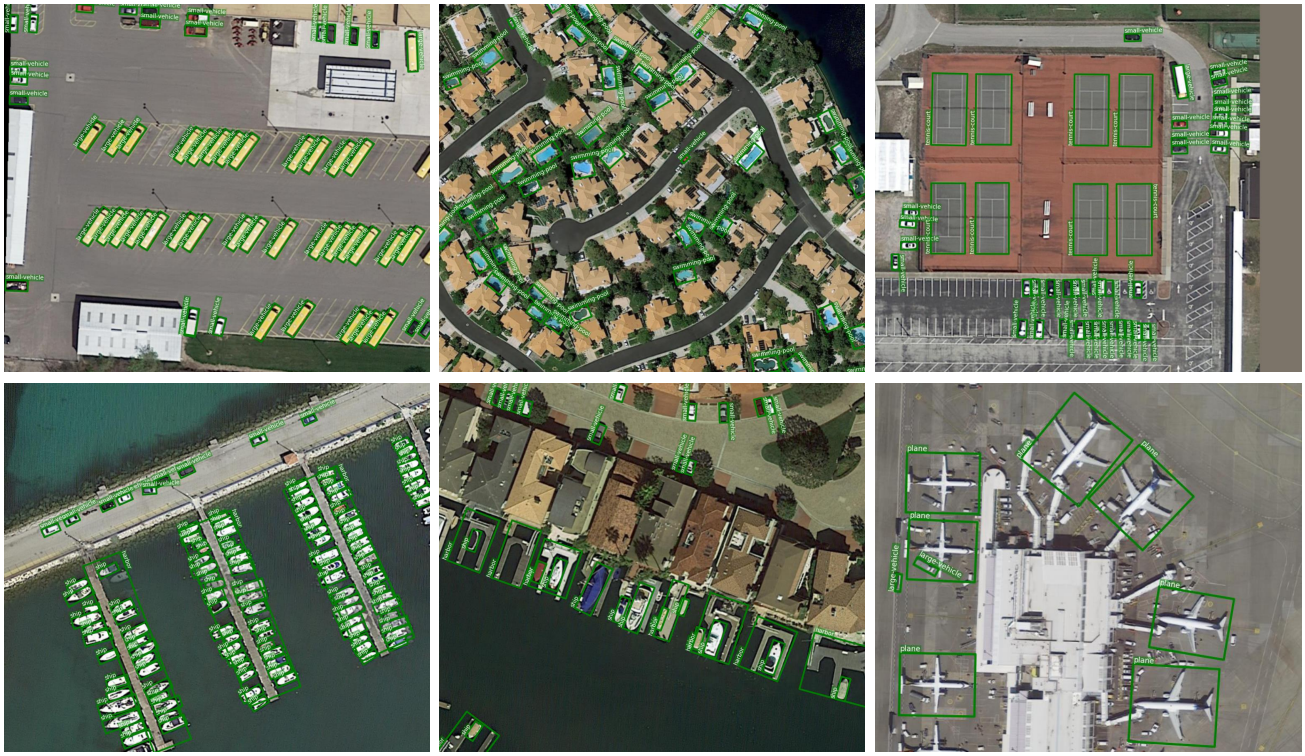

Figure 6. The qualitative results on DOTA-v1.0 testing set. It contains 15 common categories, such as large-vehicle, small-vehicle, plane, swimming-pool, ship, tennis-court, etc.

2.2. Constrained Deformable Pooling

To demonstrate the effectiveness of our constrained deformable pooling, we construct an ablation study on the IC15 and the MLT17 datasets. As shown in Tab. 2, our constrained deformable pooling outperforms average pooling and the original deformable pooling. It achieves 89.1% and 79.5% f-measure on the IC15 and the MLT17 datasets, respectively.

2.3. Loss for Rotated Bounding Boxes

As shown in Tab. 3, we compare the original GWD [20] loss with our proposed scale-invariant form on the IC15 and the MLT17 datasets. Our scale-invariant GWD loss outperforms the original one by 0.7% and 0.5% on the IC15 and the MLT17 datasets.

2.4. Transformer Structure

Despite the state-of-the-art performance achieved by our basic model architecture, we replace the basic transformer encoder layers with those in the modern transformer structure, *i.e.* Swin-Transformer [10], for further improvement. Different from applying Swin-Transformer for images, we only use four swin-transformer blocks for our feature grouping. Since it is designed for 2-D feature maps, we feed the feature map into the swin-transformer stage while masking out the unsampled features. The computation cost would increase to some extent, so we just provide the results in the appendix for reference. Owing to the power of Swin-Transformer layers, our model obtains 0.4% and 0.2% performance gain on the IC15 and the MLT17 datasets as shown in Tab. 4.

2.5. Compared with Non-Transformer Structure

To evaluate sampling and grouping with non-transformer methods, we replace our transformer module with GCN [7] (FPN+GCN) and FC layers (FPN+FC). As shown in Tab. 1, these two settings achieve lower f-measure than ours. This phenomenon validates the effectiveness of our proposed sampling and grouping framework based on transformers.

2.6. Rotated Object Detection

Our proposed method not only achieves state-of-the-art performance on scene text detection, but also performs well on oriented object detection. To prove the effectiveness of our method, we adapt it to oriented object detection and evaluate it on a popular dataset for oriented object detection in aerial images, *i.e.,* DOTA-v1.0 [15]. DOTA-v1.0 is one of the largest dataset for oriented object detection in aerial images, and it contains 15 common categories, 2806 images and 188282 instances.

In the training, we use the same loss function as the loss for multi-oriented text detection. The feature sampling scheme is consistent with the configuration #5. Following the pre-processing in previous methods [20, 24], we split the training images of DOTA-v1.0 into 1024 × 1024 sub-images with an overlap of 200 pixels. We train our model for 100 epochs with an initial learning rate $1e^{-4}$, and decay it at 50th and 80th epoch, respectively.

As shown in Tab. 5, we compare our model with previous oriented object detection approaches in both single-scale and multi-scale testing manners. For a fair comparison, our method achieves the best performance among the single-stage approaches, and outperform KLD [24] by 1.32 AP_{50}. By multi-scale testing, our model also achieves the competitive result 79.59 in terms of AP_{50} with refine-stage and two-stage approaches.

References

[1] Seyed Majid Azimi, Eleonora Vig, Reza Bahmanyar, Marco Körner, and Peter Reinartz. Towards multi-class object detection in unconstrained remote sensing imagery. In *ACCV*, 2018. 4

[2] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *ECCV*, 2020. 2

[3] Jifeng Dai, Haozhi Qi, Yuwen Xiong, Yi Li, Guodong Zhang, Han Hu, and Yichen Wei. Deformable convolutional networks. In *ICCV*, 2017. 1

[4] Jian Ding, Nan Xue, Yang Long, Gui-Song Xia, and Qikai Lu. Learning roi transformer for oriented object detection in aerial images. In *CVPR*, 2019. 4

[5] Jiaming Han, Jian Ding, Jie Li, and Gui-Song Xia. Align deep features for oriented object detection. *TGARS*, 2021. 4

[6] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 1

[7] Guohao Li, Matthias Muller, Ali Thabet, and Bernard Ghanem. Deepgcns: Can gcns go as deep as cnns? In *ICCV*, 2019. 5

[8] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *CVPR*, 2017. 1

[9] Rosanne Liu, Joel Lehman, Piero Molino, Felipe Petroski Such, Eric Frank, Alex Sergeev, and Jason Yosinski. An intriguing failing of convolutional neural networks and the coordconv solution. *NeurIPS*, 2018. 1

[10] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *ICCV*, 2021. 5

[11] Qi Ming, Lingjuan Miao, Zhiqiang Zhou, and Yunpeng Dong. Cfc-net: A critical feature capturing network for arbitrary-oriented object detection in remote-sensing images. *IEEE Transactions on Geoscience and Remote SensingF*, 2021. 4

[12] Qi Ming, Lingjuan Miao, Zhiqiang Zhou, Xue Yang, and Yunpeng Dong. Optimization for arbitrary-oriented object

detection via representation invariance loss. *IEEE Geoscience and Remote Sensing Letters*, 19:1–5, 2021. 4

[13] Wen Qian, Xue Yang, Silong Peng, Yue Guo, and Junchi Yan. Learning modulated loss for rotated object detection. *arXiv preprint arXiv:1911.08299*, 2019. 4

[14] Jinwang Wang, Wen Yang, Heng-Chao Li, Haijian Zhang, and Gui-Song Xia. Learning center probability map for detecting objects in aerial images. *TGARS*, 59(5):4307–4323, 2020. 4

[15] Gui-Song Xia, Xiang Bai, Jian Ding, Zhen Zhu, Serge Belongie, Jiebo Luo, Mihai Datcu, Marcello Pelillo, and Liangpei Zhang. Dota: A large-scale dataset for object detection in aerial images. In *CVPR*, 2018. 5

[16] Xingxing Xie, Gong Cheng, Jiabao Wang, Xiwen Yao, and Junwei Han. Oriented r-cnn for object detection. In *ICCV*, 2021. 4

[17] Yongchao Xu, Mingtao Fu, Qimeng Wang, Yukang Wang, Kai Chen, Gui-Song Xia, and Xiang Bai. Gliding vertex on the horizontal bounding box for multi-oriented object detection. *TPAMI*, 43(4):1452–1459, 2020. 4

[18] Xue Yang, Liping Hou, Yue Zhou, Wentao Wang, and Junchi Yan. Dense label encoding for boundary discontinuity free rotation detection. In *CVPR*, 2021. 4

[19] Xue Yang and Junchi Yan. Arbitrary-oriented object detection with circular smooth label. In *ECCV*, 2020. 4

[20] Xue Yang, Junchi Yan, Qi Ming, Wentao Wang, Xiaopeng Zhang, and Qi Tian. Rethinking rotated object detection with gaussian wasserstein distance loss. In *ICML*, 2021. 2, 4, 5

[21] Xue Yang, Junchi Yan, Qi Ming, Wentao Wang, Xiaopeng Zhang, and Qi Tian. Rethinking rotated object detection with gaussian wasserstein distance loss. In *ICML*, 2021. 4

[22] Xue Yang, Jirui Yang, Junchi Yan, Yue Zhang, Tengfei Zhang, Zhi Guo, Xian Sun, and Kun Fu. Scrdet: Towards more robust detection for small, cluttered and rotated objects. In *CVPR*, 2019. 4

[23] 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. *NeurIPS*, 2021. 4

[24] 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. *NeurIPS*, 2021. 4, 5

[25] Pengbo Zhao, Zhenshen Qu, Yingjia Bu, Wenming Tan, and Qiuyu Guan. Polardet: A fast, more precise detector for rotated target in aerial images. *International Journal of Remote Sensing*, 42(15):5821–5851, 2021. 4

[26] Bo Zhong and Kai Ao. Single-stage rotation-decoupled detector for oriented object. *Remote Sensing*, 12(19):3262, 2020. 4