OBJECTS DETECTION FOR REMOTE SENSING IMAGES BASED ON POLAR COORDINATES

Lin Zhou[a,b,c†], Haoran Wei[a,b,c†], Hao Li[a,c], Zhang Yue[a,c], Xian Sun[a,c], Wenzhe Zhao[a,c,*]

[a] Aerospace Information Research Institute, Chinese Academy of Sciences, Beijing 100190, China
[b] School of Electronic, Electrical and Communication Engineering,
University of Chinese Academy of Sciences, Beijing 100190, China
[c] Key Laboratory of Network Information System Technology (NIST),
Aerospace Information Research Institute, Chinese Academy of Sciences, Beijing 100190, China
[†] Equal contribution
(zhoulin18, weihaoran18)@mails.ucas.ac.cn

KEY WORDS: Remote sensing images, Object detection, Polar coordinates, Anchor-free, Oriented object

ABSTRACT:

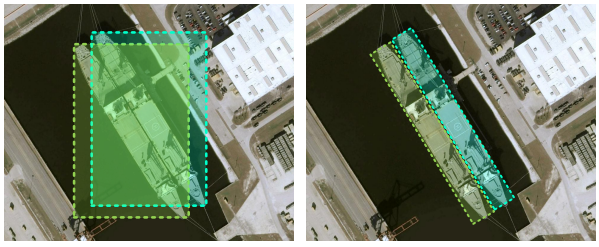
Oriented and horizontal bounding box are two typical output forms in the field of remote sensing object detection. In this filed, most present state-of-the-art detectors belong to anchor-based method and perform regression tasks in Cartesian coordinates, which cause the design of oriented detectors is much more complicated than the horizontal ones, because the former usually needs to devise more complex rotated anchors, rotated Intersection-over-Union (IOU) and rotated Non Maximum Supression (NMS). In this paper, we propose a novel anchor-free detector modeled in polar coordinates to detect objects for remote sensing images, which makes the acquisition of oriented output form be as simple as the horizontal one. Our model, named Polar Remote Sensing Object Detector (P-RSDet), takes the center point of each object as the pole point and the horizontal positive direction as the polar axis to establish the polar coordinate system. The detection of one object can be regarded as predictions of one polar radius and two polar angles for both horizontal and oriented bounding box by our model. P-RSDet realizes the combination of two output forms with minimum cost. Experiments show that our P-RSDet achieves competitive performances on DOTA, UCAS-AOD and NWPU VHR-10 datasets on both horizontal and oreinted detection fileds.

1. INTRODUCTION

In recent years, object detection in remote sensing images has made extraordinary progress driven by the applications of deep convolution neural network (DCNN). Present DCNN-based detectors in the remote sensing field can be divided into two research branches according to the different output forms: horizontal and oriented bounding box. And these two types of models have their own advantages in practical applications.

Most horizontal detectors(Wang et al., 2019, Deng et al., 2018, Ding et al., 2018, Zhang et al., 2018) are designed based on anchor mechanism which was first proposed in Faster RCNN (Ren et al., 2015). They set up anchor boxes with different size and aspect ratio intensively in feature maps to guide the regressions of the position as well as size of each object. These type of detectors in remote sensing are easy to design and simple to implement relatively, and sometimes can obtain satisfactory results without nearly any changes in the original baselines (Ren et al., 2015, He et al., 2016, Lin et al., 2017a, Redmon and Farhadi, 2018, Liu et al., 2016, Lin et al., 2017b, Cai and Vasconcelos, 2018). However, it is limited to locate objects which have large aspect ratios with the output form of horizontal bounding box. As shown in Figure 1(a), when the aspect ratio of an object is too large, the horizontal bounding box will bring a lot of redundant pixels that do not belong to the object actually, which will make the final locating results inaccurate. In addition, when two large aspect ratio objects park side by side, their horizontal bounding boxes may have a large IOU, which will cause one of them to be filtered out by NMS in anchor-based network. It will result in missed detection uncontrollable.

The problems faced with horizontal detectors aforementioned can be solved by oriented detectors (Ma et al., 2018, Yang et al., 2018, Yang et al., 2019). As shown in Figure 1(b), the output form of this type of detectors is oriented bounding box which can provide more precise location for objects with large aspect ratio. However, the design of these models is more complicated than that of horizontal ones. In order to achieve the aim of getting oriented bounding box, more anchors with different angles will be set. In addition, both the IOU and NMS in the horizontal models should be replaced by more complex ones with oriented form. Although this type of detectors is effective in the detection of specific objects, the cost of devising them is usually large.



(a) (b)

Figure 1: When the aspect ratio of an object (such as warship) is too large, the horizontal bounding box is not a good representation of object as shown in Figure (a). At present, oriented bounding boxes as shown in Figure (b) become more popular.

Naturally, *how to design a novel remote sensing detector which can integrate the two types of output form in a simple and effective way* is the starting point of our research. In this paper, we propose a new model named Polar Remote Sensing Object Detector (P-RSDet) to achieve this aim. Our P-RSDet abandons two

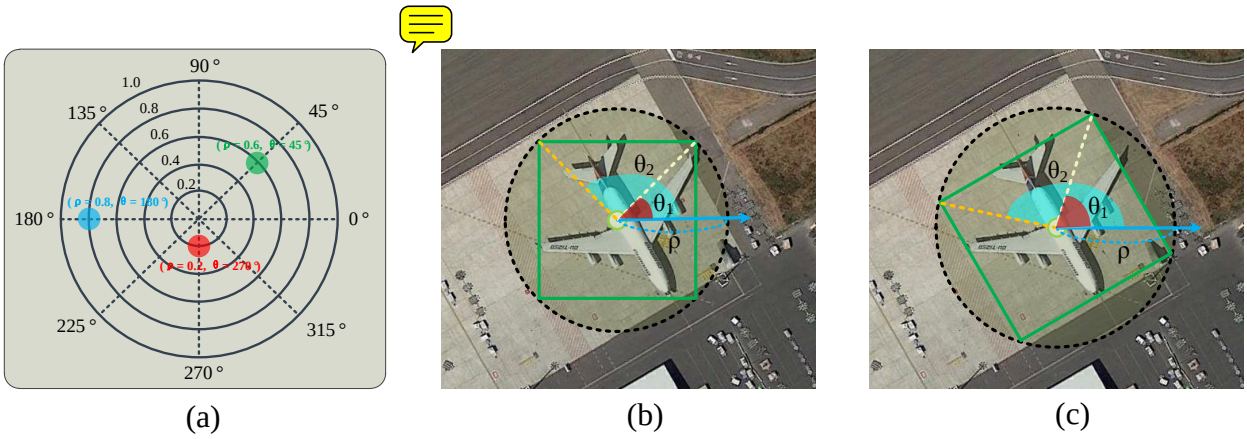


Figure 2: In polar coordinates, a point can be represented by polar radius ρ and polar angle θ as shown in Figure (a). For both horizontal and oriented bounding box of one object, they can be represented through a unified definition which is (ρ, θ_1, θ_2) as shown in Figure (b) and (c).

inherent modes of most present remote sensing detectors: anchor-based and Cartesian coordinates modeling. Instead, we design P-RSDet which is an anchor-free detector modeled in polar coordinates. As shown in Figure 2(a), a point in polar coordinates can be represented by (ρ, θ), where ρ is the polar radius and θ is the polar angle of this point. In this way, both horizontal and oriented bounding box can be denoted by one polar radius and two polar angles with the form of (ρ, θ_1, θ_2) as shown in Figure 2(b) and (c). Our P-RSDet takes the center point of corresponding bounding box of each object as the pole point and the horizontal-right direction as the polar axis with degree of 0°. Then it can output these two types of bounding box according to the different annotations via regressing three values: ρ, θ_1 and θ_2. We select the method of keypoint detection to predict the locations of the pole points, and perform polar radius and angles regression tasks at these positions simultaneously. P-RSDet cleverly combines anchor-free with polar coordinates to realize the combination of horizontal detectors and oriented detectors in the simplest way. It is worth mentioning that our P-RSDet achieves satisfactory results on multiple remote sensing public datasets, which proves its excellent performance.

Our innovations and contributions are as follows:
(1). We propose a novel object detector named P-RSDet for remote sensing images via the combination of anchor-free and polar coordinates.
(2). By introducing polar coordinates, our P-RSDet can detect objects with the annotation forms of both horizontal and oriented bounding box in a simple and efficient way via regressing three values at the corresponding pole points.
(3). In order to make the predicted bounding box more accurate, we design a new method of extracting pole points and new loss functions for our model.

The rest of this paper is organized as follows: We introduce the related works done by researchers before and basic principle in our method in Section 2. The details of P-RSDet are shown in Section 3. We place our experiment results and analyses in Section 4. At last, our work is summarized and concluded in Section 5.

2. RELATED WORKS

2.1 Horizontal Object Detectors

Horizontal bounding box is a general annotation form in the field of natural scene as well as remote sensing object detection. At present, horizontal object detection models can be divided into two types: anchor-based models(Ren et al., 2015, Liu et al., 2016, Cai and Vasconcelos, 2018) and anchor-free models(Law and Deng, 2018, Zhou et al., 2019b, Zhou et al., 2019a, Tian et al., 2019) according to whether the anchor mechanism is used or not.

For anchor-based models, according to whether Region Proposal Network(RPN) exists in the networks, they can also be divided into two-stage detectors and one-stage detectors, which are represented by Faster RCNN(Ren et al., 2015) and SSD(Liu et al., 2016) respectively. In this kind of model, anchor boxes can be regarded as fixed reference regions with different scales and ratios. After calculating their IoU with ground-truth, boxes which are classified as positive will be regarded as proposals. Then accurate prediction bounding boxes are obtained by regression on the basis of these proposals. However, the existence of anchor boxes brings about some limitations such as too many hype-parameters and complex post-processing.

Recently, anchor-free models represented by CornerNet(Law and Deng, 2018) etc, appeared in succession in order to get rid of the shortcomings of anchor-based detectors. CornerNet inspired by keypoint detection outputs some heatmaps which act as predicting and grouping left-top points as well as right-bottom points of bounding boxes. Compared with CornerNet, CenterNet(Zhou et al., 2019a) regards the detetcion task as the prediction of center point as well as the regression of width and height of each object at corresponding center, which greatly improve the efficiency of networks by removing the process of keypoints grouping.

Compared with objects in natural scene, objects in remote sensing images have properties such as dense parking and large aspect ratios, which causes the object representation only by horizontal bounding box is not enough. As a result, oriented bounding box is introduced in this field and gets more and more attention recently.

2.2 Oriented Object Detectors

Two algorithms of oriented object detection, R2CNN (Jiang et al., 2017) and RRPN(Ma et al., 2018), come from the field of scene

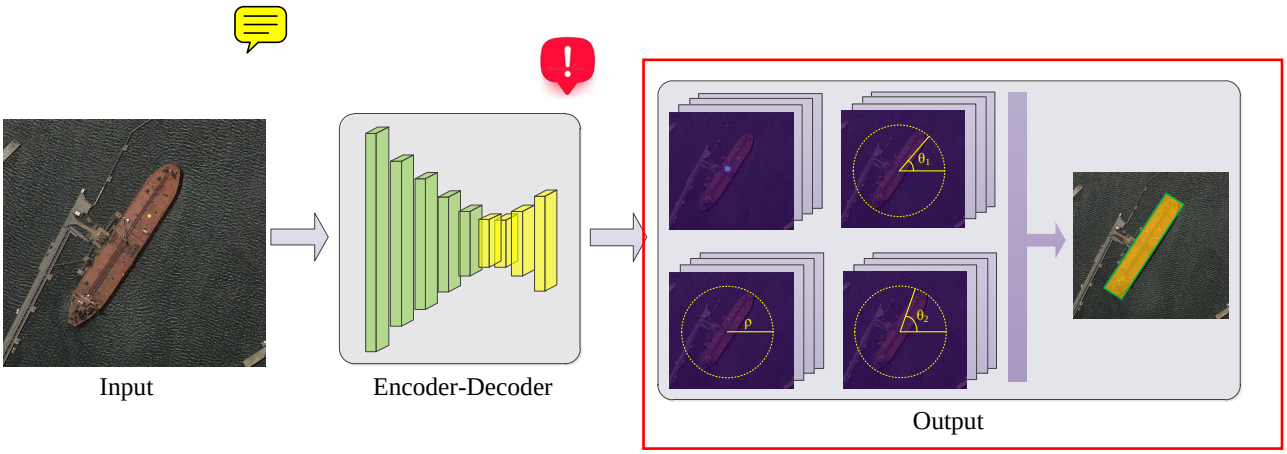


Figure 3: Architecture of P-RSDet. When an image inputs our model, it will output four maps in parallel. One is the heatmap which used to predict the pole point, and the other three are used to regress ρ, θ_1, θ_2 respectively. These 4 maps have c channels which represent c classes.

text detection. R2CNN and RRPN all base on Faster RCNN. R2CNN adds two pooled sizes and a branch to regress inclined box coordinates. RRPN improves RPN in Faster RCNN by adding rotation proposals. In the field of remote sensing, R-DFPN(Yang et al., 2018) improves RRPN to obtain precise oriented bounding box to solve the problem of ship rotation and dense parking. However, these anchor-based oriented detectors not only face the disadvantages brought by anchor, but also greatly increase the computation complexity due to the introduction of anchors with different angles.

Due to the simplicity, anchor-free detectors have also been improved to realize oriented object detection. O^2-DNet(Wei et al., 2019b) which abandons anchor mechanism and detects oriented objects in aerial images by predicting a pair of middle lines inside each object. Nevertheless, it needs regressing eight offsets to predict two middle lines of each object in Cartesian coordinate system, which lead to too many degrees of freedom and require more complex loss functions to control them. Our P-RSDet, which directly models objects in polar coordinate system and only needs to regress three degrees of freedom, pursues detecting oriented objects in a simple yet efficient way,

3. P-RSDET

In this section, we first briefly introduce the framework of our proposed P-RSDet. Then, we show how we model oriented objects in remote sensing images based on polar coordinates with minimum degrees of freedom. Finally, we elaborate the details of our model, including the design of specific loss functions and the optimization of keypoints extraction method.

3.1 Framework

Figure 3 illustrates the overall framework of our P-RSDet. A modified higher-resolution ResNet-101(Zhou et al., 2019a) with 4 output stride is selected as the Encoder-Decoder of P-RSDet. Suppose the size of one input image is $W \times H$, P-RSDet will output four maps with $C \times \frac{W}{d} \times \frac{H}{d}$ size, where C is the number of categories and d represents the output stride which is 4 as aforementioned. In these four output maps, one is in the form of heatmap to predict the pole points, and the other three are to regress the corresponding polar radius along with the polar angles of each object. As mentioned above and shown in Figure 3, our model is very simple to design.

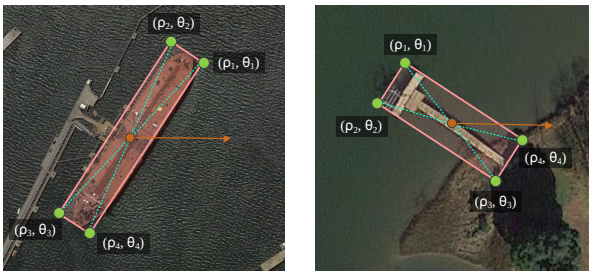


Figure 4: Objects in Polar Coordinates. We make the center point of each object be the pole point of polar coordinate system, and take the right direction and the counterclockwise as the positive direction of the polar axis and polar angle respectively. Then the bounding boxes of objects can be represented with (ρ_1, θ_1), (ρ_2, θ_2), (ρ_3, θ_3), (ρ_4, θ_4).

3.2 Objects in Polar Coordinates

The four corners of the oriented bounding box are usually represented by (x_1, y_1), (x_2, y_2), (x_3, y_3), (x_4, y_4) in Cartesian coordinates. In order to model it in polar coordinates, for an object, we first make its center point be the pole point of polar coordinate system, then we take the horizontal-right direction and the counterclockwise as the positive direction of the polar axis and polar angle in radians respectively. As shown in Figure 4, the four corners can be represented in sequence as (ρ_1, θ_1), (ρ_2, θ_2), (ρ_3, θ_3), (ρ_4, θ_4). According to the properties of rectangle, we can get the following relations:

$$\rho_1 = \rho_2 = \rho_3 = \rho_4 \tag{1}$$

$$\theta_3 = \theta_1 + \pi, \theta_4 = \theta_2 + \pi \tag{2}$$

Therefore, let $\rho = \frac{(\rho_1+\rho_2+\rho_3+\rho_4)}{4}$, only three variables, ρ, θ_1 and θ_2 are needed to represent a bounding box of object in Polar Coordinates.

Due to the process of evaluating one detector's performance is only carried in Cartesian coordinates at present, we need to transform the points in polar coordinates to Cartesian one in the test stage. We first extract the position of pole points (x_p^i, y_p^i) of objects in heatmap, where i denotes the number of targets. Then according to the pole points, we obtain the polar radius and angles ρ^i, θ_1^i and θ_2^i in other three output maps. Finally, the final

bounding boxes in the form of $[(x_1^i, y_1^i), (x_2^i, y_2^i), (x_3^i, y_3^i), (x_4^i, y_4^i)]$ can be obtained through the transformation calculation formulas as follows:

$$x_n^i = x_p^i + \rho^i \cdot \cos(\theta_n^i) \qquad y_n^i = y_p^i + \rho^i \cdot \sin(\theta_n^i) \tag{3}$$

where n represents $1, 2, 3$ and 4.

3.3 Pole Point

Accurate pole point prediction is very important for getting accurate bounding box. In our model, the detection of pole points follows CornerNet(Law and Deng, 2018) for its excellent performance in the detection of keypoints of objects.

As mentioned in Section 3.1, P-RSDet output a heatmap with size of $C \times \frac{W}{d} \times \frac{H}{d}$ for predicting pole points. The heatmap is actually a confidence map with the value of each pixel $p \in [0,1]^{\frac{W}{d} \times \frac{H}{d}}$. In the training stage, let h and w represent the height and weight of one bounding box, and we give the ground-truth of each point (x, y) in the heatmap in form of Gauss kernel as $e^{-\frac{(x-x_p)^2+(y-y_p)^2}{2(min(h,w)/3)^2}}$, where (x_p, y_p) is the pole point of this bounding box. We use a modified Focal Loss(Lin et al., 2017b) follows CornerNet to guide the regression of pole points:

$$\mathcal{L}_{pole} = -\frac{1}{N}\sum_{cij}\begin{cases}(1-p_{cij})^{\alpha} log(p_{cij}), & p^*_{cij} = 1\\ (1-y_{cij})^{\beta}(p_{cij})^{\alpha} & \\ \quad log(1-p_{cij}), & p^*_{cij} = others\end{cases} \tag{4}$$

where N is the number of objects in the input image, α and β are the hyper-parameters which we set α to 2 and β to 4 in experiments to control the contribution of positive and negative points . p^*_{cij} is the ground-truth and p_{cij} is the confidence with which a point at location (i, j) be regarded as a pole point for class c in the predicted heatmap.

We follow the method of keypoints detection in CornerNet during training stage. But in the test stage, the method of keypoint extraction in CornerNet is not suitable for us. Different from the natural images, a remote sensing image may contain hundreds of targets in the same class. CornerNet keeps 80 keypoints with top scores, which may cause missed detection in remote sensing field.To address this problem, O^2-DNet binaries the heatmap via a threshold, and then finds the center point of each connected domain in the binary image as the keypoint of extraction. Obviously, this method is not accurate which may cause the drift of the extracted keypoint in corresponding connected domain.

In P-RSDet, we optimized the extraction method in O^2-DNet. We first find connected domains in the heatmap following O^2-DNet. Then we take the extreme point in each connected domain as pole point, which is more accurate than the method of extracting center point proposed in O^2-DNet.

3.4 Polar Radius & Polar Angle

Our P-RSDet only need to regress polar radius ρ and the first two angles θ_1, θ_2. For one bounding box, according to the original annotation $(x_1, y_1), (x_2, y_2), (x_3, y_3), (x_4, y_4)$, let $(\sum_{i=1}^4 x_i/4, \sum_{i=1}^4 y_i/4)$ be the pole point (x_p, y_p), and the corresponding polar radius is computed as follows:

$$\rho = \frac{\sum_{i=1}^4 [(x_i - x_p)^2 + (y_i - y_p)^2]^{1/2}}{4} \tag{5}$$

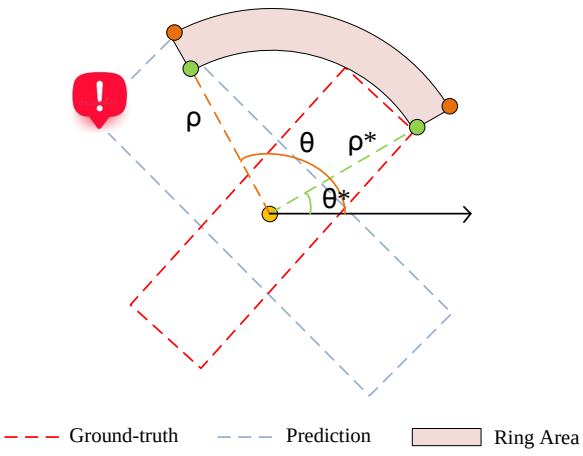


Figure 5: Ring Area in Polar Representation

For θ_1 and θ_2, we first compute the polar angles of four corners and turn them between 0 and 2π, then choose the minimum two in the counterclockwise direction as θ_1 and θ_2. The angles are calculated as follows:

$$\theta_i = \begin{cases} 2\pi - arctan(y_i - y_p, x_i - x_p), & 0 <= \theta_i <= \pi \\ -arctan(y_i - y_p, x_i - x_p), & otherwise \end{cases} \tag{6}$$

So far, we have obtained all the regression targets, polar radius ρ and the first two polar angles θ_1, θ_2. We select Smooth-L1(Girshick, 2015) Loss to regress these three values in corresponding three output maps. Considering the correlation between radius and angles of the same object, we also design a new loss named Polar-RingLoss for our model to control the relationship. The details of Polar-RingLoss are as follows:

As shown in Figure 5, let ρ, θ be the prediction results and ρ^*, θ^* be the ground-truth. The area of the Ring Area shows the deviation between the prediction results with the ground-truth. The area can be calculated according to the following formula:

$$S = \frac{1}{2}|[\rho^2 - (\rho^*)^2](\theta - \theta^*)| \tag{7}$$

Depends on the area formula above, we define Polar-RingLoss as follows:

$$\mathcal{L}_{pr}(\rho, \theta) = e^{|[\rho^2 - (\rho^*)^2](\theta - \theta^*)|} - 1 \tag{8}$$

The total regression loss of P-RSDet is:

$$\mathcal{L}_{reg} = \sum_{u=\rho,\theta_1,\theta_2} Smooth\mathcal{L}1(u, u^*) + \sum_{\theta=\theta_1,\theta_2} \mathcal{L}_{pr}(\rho, \theta) \tag{9}$$

4. EXPERIMENTS

4.1 Datasets

In the stage of experiments, we verify the performance of our model on three popular remote sensing public datasets: DOTA, UCAS-AOD and NWPU VHR-10. All the experiments are performed on two V100 GPUs with PyTorch 1.0 (Paszke et al., 2017). The details of these three datasets are as follows.

DOTA: DOTA (Xia et al., 2018) consists of 2806 aerial images which includes 15 categories objects annotated with horizontal

Table 1: Comparisons on DOTA with the form of oriented bounding boxes. The short names are defined as: Pl: Plane, Bd: Baseball diamond, Br: Bridge, Gft: Ground field track, Sv: Small vehicle, Lv: Large vehicle, Sh:Ship, Tc: Tennis court, Bc: Basketball court, St: Storage tank, Sbf: Soccer-ball field, Ra: Roundabout, Ha: Harbor, Sp: Swimming pool, and He: Helicopter.

Models	Pl	Bd	Br	Gft	Sv	Lv	Sh	Tc	Bc	St	Sbf	Ra	Ha	Sp	He	mAP
R^2CNN (Jiang et al., 2017)	80.94	65.67	35.34	67.44	59.92	50.91	55.81	90.67	66.92	72.39	55.06	52.23	55.14	53.35	48.22	60.67
RRPN (Ma et al., 2018)	88.52	71.20	31.66	59.30	51.85	56.19	57.25	90.81	72.84	67.38	56.69	52.84	53.08	51.94	53.58	61.01
R-DFPN (Yang et al., 2018)	80.92	65.82	33.77	58.94	55.77	50.94	54.78	90.33	66.34	68.66	48.73	51.76	55.10	51.32	35.88	57.94
ICN (Azimi et al., 2018)	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-Transformer (Jian Ding, 2019)	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
P-RSDet	**89.02**	73.65	47.33	72.03	**70.58**	**73.71**	72.76	**90.82**	**80.12**	81.32	**59.45**	57.87	60.79	**65.21**	**52.59**	**69.82**

Table 2: Comparisons on UCAS-AOD with both oriented and horizontal bounding boxes. We choose the default parameters in *PASCAL VOC*with IoU(Intersection over Union) which is 0.5 during calculating AP.

Oriented bounding box				**Horizontal bounding box**			
Models	plane	car	mAP	Models	plane	car	mAP
RRPN(Ma et al., 2018)	88.04	74.36	81.20	YOLO9000(Redmon and Farhadi, 2017)	87.62	70.13	78.87
R2CNN(Jiang et al., 2017)	89.76	78.89	84.32	SSD(Liu et al., 2016)	89.12	81.37	85.24
R-DFPN(Yang et al., 2018)	88.91	81.27	85.09	RetinaNet(Lin et al., 2017b)	89.95	83.22	86.58
X-LineNet(Wei et al., 2019a)	91.3	-	-	Faster R-CNN+FPN(Lin et al., 2017a)	90.83	86.79	88.81
O^2-DNet(Wei et al., 2019b)	**93.21**	86.72	89.96	ConerNet(Law and Deng, 2018)	77.43	64.80	71.11
P-RSDet	92.69	**87.38**	**90.03**	P-RSDet	**93.13**	**87.36**	**90.24**

Table 3: Comparisons on NWPU VHR-10 with the form of horizontal bounding boxes. The abbreviations of the names are defined as: ap-airplane, sh-ship, st-storage tank, bd-baseballdiamond, tc- tennis court, bc-basketball court, gtf-ground track field, hb-harbor, br-bridge and ve-vehicle.

Models	ap	sh	st	bd	tc	bc	gtf	hb	br	ve	mAP
SSD(Liu et al., 2016)	90.40	60.90	79.80	89.90	82.60	80.60	98.30	73.40	76.70	53.10	78.40
DSSD(Fu et al., 2017)	82.70	62.80	**89.20**	90.10	87.80	80.90	79.80	**82.10**	81.20	61.30	79.80
RetinaNet(Lin et al., 2017b)	87.50	83.80	88.60	91.40	86.20	81.70	92.30	79.30	71.10	77.90	83.90
Faster R-CNN+FPN(Lin et al., 2017a)	96.40	87.80	84.10	93.60	**89.60**	92.50	**95.70**	81.20	79.20	83.90	88.60
P-RSDet	**97.90**	**92.40**	88.30	**95.80**	89.30	**96.20**	94.90	81.90	**83.30**	88.70	**90.80**

and oriented bounding boxes. In this dataset, the proportions of training, validation and test images are $1/2$, $1/6$ and $1/3$ respectively. The size of each image is in the range of 800×800 to 4000×4000 pixels. In experiments, we only use the annotations of oriented bounding boxes, and the size of our crop images are multiple which are 512×512, 800×800 and 1024×1024 with 0.2 overlap.

UCAS-AOD: In UCAS-AOD (Zhu et al., 2015), there are two catagories: airplane and small car. The number of plane images is 1000 which contain 7482 objects and the number of car images is 510 with 7114 objects. All objects in UCAS-AOD are labeled with both oriented and horizontal bounding boxes. In our experiments, we randomly divide the training and test set by $8:2$, and train P-RSDet on both two type of annotations on UCAS-AOD to verify its performance better.

NWPU VHR-10: There are 650 images of objects and 150 of the background total in NWPU VHR-10 (Cheng et al., 2016) dataset. It includes 10 categories such as plane, ship, oil tank and baseball field. Similarly, we divide the training set and test set by 8:2 in experiments. Unlike the first two datasets, annotations of NWPU VHR-10 has only horizontal bounding box.

4.2 Training and Testing Details

In the training stage, the input resolution of P-RSDet is set to 511×511. For DOTA, we resize the crop images to the input size directly, and use some simple methods to enhance the data, including random horizontal and vertical flipping as well as color dithering. Adam(Kingma and Ba, 2014) is selected as the optimizer for our model. We train our model from scratch to $300k$ iterations with the batch size setting to 32. The learning rate starts from 0.0025 and 10 times lower for every third iterations. The total loss of P-RSDet is as follows:

$$\mathcal{L}oss = \mathcal{L}_{pole} + \alpha \mathcal{L}_{reg} \tag{10}$$

where α is set to 0.1 in all experiments.

For the other two datasets, because the size of the original image is not fixed, in order not to deform the object, we randomly cut the original image into the networks with windows of $(511 \times 511) \times scale$. The $scale$ is set to $0.6, 0.8, 1.0, 1.2$ during training. Compared with DOTA, the data volume of these two datasets is smaller, so we only trained 30000 iterations for them. Other settings are the same as DOTA.

During the testing phase, we keep the input image in its original resolution to P-RSDet. The threshold value of transforming the

heatmap of pole points into a binary image is 0.3. For UCAS-AOD and NWPU VHR-10, we choose the default IoU in *PASCAL VOC*(Everingham et al., 2010) which is 0.5 during calculating AP.

4.3 Comparisons with State-of-the-art detectors

In this section, we first prove the excellent performance of P-RSDet in the detection of oriented bounding box on DOTA and UCAS-AOD. Then, in order to verify the generality of our model, we also do experiments on UCAS-AOD and NWPU VHR-10 with the annotations form of horizontal bounding box.

Oriented Bounding Boxes: As shown in Tale 1 and 2, our P-RSDet achieve satisfactory 69.82% mAP on DOTA, and 90.03% mAP on UCAS-AOD with the output form of oriented bounding boxes. Compared with the anchor-based detectors modeled in Cartesian coordinate system, our model is more competitive in the task of detecting oriented objects for remote sening images with the simpler design and higher accuracy.

Horizontal Bounding Boxes: In order to verify the excellent general capability of our model, we do the experiments on UCAS-AOD and NWPU VHR-10 datasets with the annotations of horizontal bounding box. As shown in Table 2 and 3, P-RSDet gets 90.24% mAP and 90.80% mAP on these two datasets respectively. Experimental results show that our model has excellent performance in both horizontal and oriented detection tasks. P-RSDet successfully integrates the two type of detectors in the remote sensing field with minimum consumption via the combination of anchor-free and polar coordinates.

4.4 Ablation Studies

4.4.1 Different Encoder-Decoder In P-RSDet, we use a high resolution ResNet-101 modified in (Zhou et al., 2019a) as the Encoder-Decoder. For the sake of testing the influences of different Encoder-Decoders on our model, we replace the ResNet-101 with DLA-34 (Dai et al., 2017, Zhou et al., 2019a) and 104-Hourglass(Newell et al., 2016, Law and Deng, 2018). DLA-34 and 104-Hourglass are two backbone networks smaller and larger than ResNet-101 respectively. We do the experiments on UCAS-AOD with oriented bounding box.

Table 4: Comparisons of Different Encoder-Decoders.

Encoder-Decoders	plane	car	mAP
ResNet-101	92.69	87.38	90.03
DLA-34	91.02	85.24	88.13
104-Hourglass	94.15	89.29	91.72

As shown in Table 4, our model can still achieve satisfactory results of 88.1% mAP when using small DLA-34 as the Encoder-Decoder. It is noteworthy that the performance of our model can be further improved when we choose the stronger 104-Hourglass. Experiments show that our model is effective with different Encoder-Decoders.

4.4.2 Polar-RingLoss As mentioned in Section 3.4, we design a new Polar-RingLoss for our P-RSDet. In order to verify its effectiveness, we designed this comparative experiment on UCAS-AOD with oriented bounding box.

As shown in Table 5, our model with Polar RingLoss outperforms one without Polar RingLoss by 1.85% mAP. Therefore, the design of Polar RingLoss is effective for P-RSDet.

Table 5: Effects of Polar RingLoss.

With vs. Without Polar RingLoss	plane	car	mAP
With Polar RingLoss	92.69	87.38	90.03
Without Polar RingLoss	91.21	85.16	88.18

4.4.3 Different Method of Extracting Polar Points We optimized the keypoints extraction methods of CornerNet and O^2-DNet, and compare the effect of our new method with theirs. We also do this experiments on UCAS-AOD with the detection of oriented objects. As shown in Table 6, the method which keep 80 top scores points in Cornernet only achieves 85.93% mAP because there are more than 80 targets in many remote sensing images. Our method gives decent mAP improvements of 1.18% than the method in O^2-DNet , which proves that our optimization is more effective.

Table 6: Comparisons of Different Polar Points Extraction Methods.

Extraction Methods	plane	car	mAP
P-RSDet(ours)	92.69	87.38	90.03
P-RSDet(CornerNet)	90.02	81.85	85.93
P-RSDet(O^2-DNet)	91.62	86.09	88.85

4.5 Application Extension

In the field of remote sensing object detection, to get more accurate object information, some datasets are labeled in the form of keypoints such as Aircraft-KP(Wei et al., 2019a) which marks five keypoints of each aircraft. It is worth mentioning that our P-RSDet can be migrated to a more accurate keypoints dataset by simply increasing the number of regression values in polar coordinates. As shown in Figure 6 , our modeling process on keypoints datasets by regressing five polar radii and five polar angles.

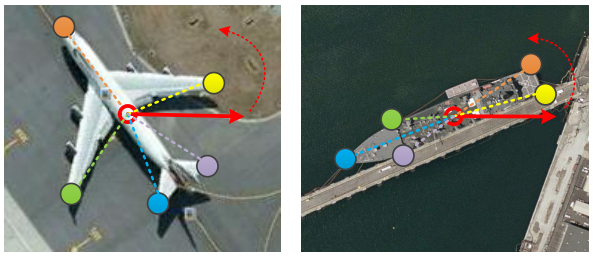


Figure 6: In polar coordinates, P-RSDet can be transplanted to more precise keypoint detection tasks by simply increasing the number of polar radius ρ and polar angles θ in regression.

5. CONCLUSION

We propose a novel object detector named P-RSDet for remote sensing images via the combination of anchor-free and polar coordinates. By introducing polar coordinates, our model can detect objects with the annotations forms of both the horizontal and oriented bounding boxes in a simple and efficient way. In order to make the output results more accurate, we also optimize a new method of extracting pole points and design a special Polar RingLoss for our model. Experimental results on multiple datasets show that the detector modeling in polar coordinates is effective.

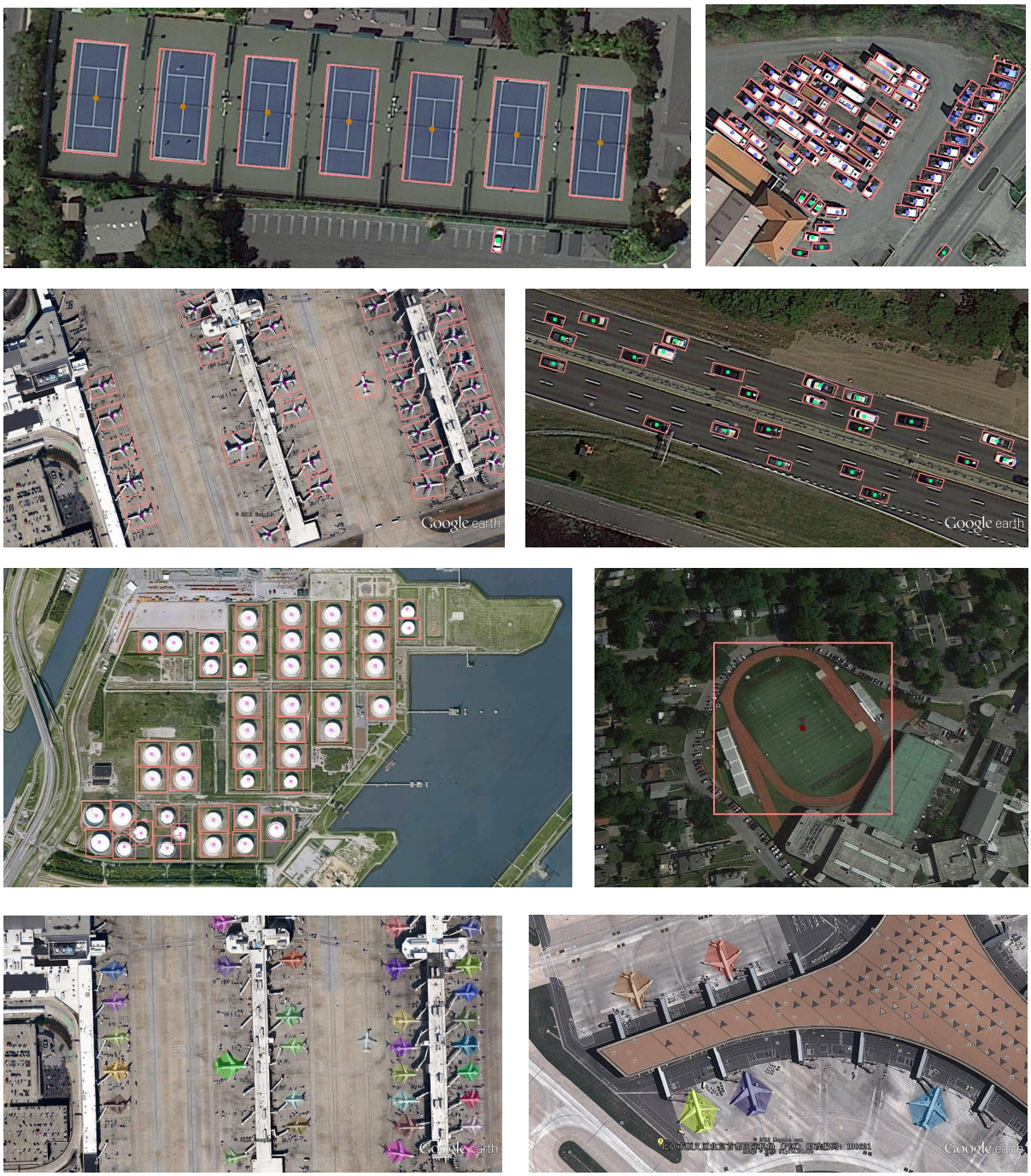


Figure 7: High quality outputs of P-RSDet on DOTA, UCAS-AOD, NWPU VHR-10 and Aircraft-KP.

REFERENCES

Azimi, S. M., Vig, E., Bahmanyar, R., Körner, M. and Reinartz, P., 2018. Towards multi-class object detection in unconstrained remote sensing imagery. arXiv preprint arXiv:1807.02700.

Cai, Z. and Vasconcelos, N., 2018. Cascade r-cnn: Delving into high quality object detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 6154–6162.

Cheng, G., Zhou, P. and Han, J., 2016. Learning rotation-invariant convolutional neural networks for object detection in vhr optical remote sensing images. IEEE Transactions on Geoscience and Remote Sensing 54(12), pp. 7405–7415.

Dai, J., Qi, H., Xiong, Y., Li, Y., Zhang, G., Hu, H. and Wei, Y., 2017. Deformable convolutional networks. In: Proceedings of the IEEE international conference on computer vision, pp. 764–773.

Deng, Z., Sun, H., Zhou, S., Zhao, J., Lei, L. and Zou, H., 2018. Multi-scale object detection in remote sensing imagery with convolutional neural networks. ISPRS journal of photogrammetry and remote sensing 145, pp. 3–22.

Ding, P., Zhang, Y., Deng, W.-J., Jia, P. and Kuijper, A., 2018. A light and faster regional convolutional neural network for object

detection in optical remote sensing images. ISPRS journal of photogrammetry and remote sensing 141, pp. 208–218.

Everingham, M., Van Gool, L., Williams, C. K., Winn, J. and Zisserman, A., 2010. The pascal visual object classes (voc) challenge. International journal of computer vision 88(2), pp. 303–338.

Fu, C.-Y., Liu, W., Ranga, A., Tyagi, A. and Berg, A. C., 2017. Dssd: Deconvolutional single shot detector. arXiv preprint arXiv:1701.06659.

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

He, K., Zhang, X., Ren, S. and Sun, J., 2016. Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 770–778.

Jian Ding, Nan Xue, Y. L. G.-S. X. Q. L., 2019. Learning roi transformer for detecting oriented objects in aerial images. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR).

Jiang, Y., Zhu, X., Wang, X., Yang, S., Li, W., Wang, H., Fu, P. and Luo, Z., 2017. R2cnn: Rotational region cnn for orientation robust scene text detection. arXiv preprint arXiv:1706.09579.

Kingma, D. P. and Ba, J., 2014. Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980.

Law, H. and Deng, J., 2018. Cornernet: Detecting objects as paired keypoints. In: Proceedings of the European Conference on Computer Vision (ECCV), pp. 734–750.

Lin, T.-Y., Dollár, P., Girshick, R., He, K., Hariharan, B. and Belongie, S., 2017a. Feature pyramid networks for object detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2117–2125.

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

Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.-Y. and Berg, A. C., 2016. Ssd: Single shot multibox detector. In: European conference on computer vision, Springer, pp. 21–37.

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

Newell, A., Yang, K. and Deng, J., 2016. Stacked hourglass networks for human pose estimation. In: European Conference on Computer Vision, Springer, pp. 483–499.

Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E., DeVito, Z., Lin, Z., Desmaison, A., Antiga, L. and Lerer, A., 2017. Automatic differentiation in pytorch.

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

Redmon, J. and Farhadi, A., 2018. Yolov3: An incremental improvement. arXiv preprint arXiv:1804.02767.

Ren, S., He, K., Girshick, R. and Sun, J., 2015. Faster r-cnn: Towards real-time object detection with region proposal networks. In: Advances in neural information processing systems, pp. 91–99.

Tian, Z., Shen, C., Chen, H. and He, T., 2019. Fcos: Fully convolutional one-stage object detection. arXiv preprint arXiv:1904.01355.

Wang, P., Sun, X., Diao, W. and Fu, K., 2019. Mergenet: Feature-merged network for multi-scale object detection in remote sensing images. In: IGARSS 2019-2019 IEEE International Geoscience and Remote Sensing Symposium, IEEE, pp. 238–241.

Wei, H., Bing, W. and Yue, Z., 2019a. X-linenet: Detecting aircraft in remote sensing images by a pair of intersecting line segments. arXiv preprint arXiv:1907.12474.

Wei, H., Zhou, L., Zhang, Y., Li, H., Guo, R. and Wang, H., 2019b. Oriented objects as pairs of middle lines. arXiv preprint arXiv:1912.10694.

Xia, G.-S., Bai, X., Ding, J., Zhu, Z., Belongie, S., Luo, J., Datcu, M., Pelillo, M. and Zhang, L., 2018. Dota: A large-scale dataset for object detection in aerial images. In: Proc. CVPR.

Yang, X., Sun, H., Fu, K., Yang, J., Sun, X., Yan, M. and Guo, Z., 2018. Automatic ship detection in remote sensing images from google earth of complex scenes based on multiscale rotation dense feature pyramid networks. Remote Sensing 10(1), pp. 132.

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

Zhang, S., Wen, L., Bian, X., Lei, Z. and Li, S. Z., 2018. Single-shot refinement neural network for object detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 4203–4212.

Zhou, X., Wang, D. and Krähenbühl, P., 2019a. Objects as points. arXiv preprint arXiv:1904.07850.

Zhou, X., Zhuo, J. and Krahenbuhl, P., 2019b. Bottom-up object detection by grouping extreme and center points. pp. 850–859.

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