PANDORA: A Panoramic Detection Dataset for Object with Orientation

Hang Xu[1,2*], Qiang Zhao[2*†], Yike Ma[2], Xiaodong Li[3], Peng Yuan[3], Bailan Feng[3], Chenggang Yan[1,4], and Feng Dai[2†]

[1] Hangzhou Dianzi University, Hangzhou, China
[2] Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China
[3] Huawei Noah's Ark Lab, Beijing, China
[4] State Key Laboratory of Media Convergence Production Technology and Systems, Beijing, China
{hxu, cgyan}@hdu.edu.cn, {zhaoqiang, ykma, fdai}@ict.ac.cn, {lixiaodong33, yuanpeng126, fengbailan}@huawei.com

Abstract. Panoramic images have become increasingly popular as omnidirectional panoramic technology has advanced. Many datasets and works resort to object detection to better understand the content of the panoramic image. These datasets and detectors use a Bounding Field of View (BFoV) as a bounding box in panoramic images. However, we observe that the object instances in panoramic images often appear with arbitrary orientations. It indicates that BFoV as a bounding box is inappropriate, limiting the performance of detectors. This paper proposes a new bounding box representation, Rotated Bounding Field of View (RBFoV), for the panoramic image object detection task. Then, based on the RBFoV, we present a PANoramic Detection dataset for Object with oRientAtion (PANDORA). Finally, based on PANDORA, we evaluate the current state-of-the-art panoramic image object detection methods and design an anchor-free object detector called R-CenterNet for panoramic images. Compared with these baselines, our R-CenterNet shows its advantages in terms of detection performance. Our PANDORA dataset and source code are available at `https://github.com/tdsuper/SphericalObjectDetection`.

Keywords: PANDORA, panoramic, object detection, RBFoV

1 Introduction

In the past few years, with the numerous development of panoramic cameras with omnidirectional vision, the applications of panoramic images are also becoming more and more extensive, such as virtual reality [9], robotics [8], street view

* This work was done when Hang Xu and Qiang Zhao were at ICT.
† Corresponding author.

(a) BFoV (b) RBFoV

Fig. 1. Visualization of two annotation methods (i.e., BFoV and RBFoV). (a) is a failure case of the BFoV annotation, which brings high overlap compared to (b). In our PANDORA dataset, we use the RBFoV as the bounding box.

[2,39,38], etc. As these panoramic data increase, the demand for panoramic object detection tasks increases [27,20,34]. Object detection has achieved an excellent performance in planar images, even comparable to human vision [40,12,25,32]. This is mainly attributed to the publication of large-scale planar image object detection datasets such as Pascal VOC [5], COCO [13], etc. However, object detection in panoramic images is still challenging for the following two reasons, as listed below:

Appropriate annotations are lacking. Object detection necessitates the location of objects and the computation of metrics, i.e., bounding box (BB) and intersection-over-union (IoU). Previous works either introduced bias in the BB [26,11] or could not calculate the IoU accurately [1]. Recent works [33,35] use the Bounding Field of View (BFoV) [24] as BB and precisely compute IoU by spherical geometry, making the BFoV the dominant representation of the bounding box in panoramic object detection. Objects without many orientations can be adequately annotated with this method. However, the object instances in panoramic images often appear with arbitrary orientations, depending on the observer's perspective. In an actually common condition as shown in Fig. 1, the overlap between two BFoVs is so large that state-of-the-art (SOTA) object detectors cannot differentiate them. In Section 6.1, we provide the quantifications regarding the overlap issue and show that using RBFoV we proposed enhances the detector's performance.

Labeling objects are complex. First, since the panoramic image has a 360° view, there are many objects of different sizes and categories in a panoramic image. Second, panoramic image is typically represented by equirectangular projection (ERP) [4]. The ERP is generated by polar transformation and thus suffers from distortion in the polar regions and discontinuity on the boundary [36]. Especially, the distortion in the polar regions is severe, which causes the annotator to be unable to identify these objects in the polar regions well. For these reasons, the development of panoramic object detection is greatly limited, resulting in the poor performance of the existing methods.

To address the above challenges, we propose a new bounding box representation, Rotated Bounding Field of View (RBFoV), for the panoramic image object detection task. Then, based on the RBFoV, we develop a new annotation tool to annotate objects at the polar regions and the boundary in panoramic images easily, and we present a PANoramic Detection dataset for Object with oRientAtion (PANDORA) in this work. To our best knowledge, PANDORA is the first dataset to use the RBFoV as the bounding box. It can be used to develop and evaluate object detectors in panoramic images. Finally, based on PANDORA, we evaluate the current SOTA methods and design an anchor-free object detector called R-CenterNet. Compared with these baselines, our R-CenterNet shows its advantages in terms of detection performance.

2 Related Work

2.1 Existing Bounding boxes

The existing bounding box definitions are mainly divided into three representations, i.e., planar rectangle, circle and spherical rectangle. The works in [29], [31] and [26] use the planar rectangle as the bounding box. This bounding box representation does not consider the distortions of panoramic images. Thus it is biased representations and has large errors. The work in [11] exploit the circular as the bouding box in the panoramic object detection. However, circular may exceed panoramic's upper or lower boundaries when the objects are near the pole. The works in [1], [33] and [35] utilizes the Bounding Field of View (BFoV) as the bounding box. This bounding box is called a spherical rectangle, which is an unbiased representation. The BFoV is currently the most dominant bounding box representation in panoramic object detection.

2.2 Panoramic Object Detection Dataset

Till now, existing panoramic image object detection datasets can be roughly divided into two subsets, i.e., synthetic dataset and natural scenes dataset. Because of the difficulty of panoramic image annotation, early methods [1,23,33] for panoramic object detection used synthetic datasets. However, the synthetic dataset cannot adequately reflect the problem complexity in the natural scene. The natural scenes dataset popular benchmarks mainly include OSV [31], ERA [29] and 360-indoor [3]. These datasets are manually annotated on panoramic images of natural scenes. Therefore, they can better validate the performance of the panoramic object detection model compared to the synthetic datasets above. However, the bounding box they use is not the suitable in panoramic object detection task.

As a result, we present a PANoramic Detection dataset for Object with oRientAtion (PANDORA) in this work. Table 1 lists the existed panoramic object detection dataset for comparison.

Dataset	Domain	Annotation	#Category	#Boxes
OSV [31]	Street Scenes	BBoX	5	5,636
FlyingCars [1]	Synthesis Cars	BFoV	1	6,000
ERA [29]	Dynamic Activities	BFoV	10	7,199
360-Indoor [3]	Indoor Scenes	BFoV	37	89,148
PANDORA	Indoor Scenes	**RBFoV**	**47**	**94,353**

Table 1. Existing panoramic object detection dataset comparison. The BBoX is the planar rectangle.

2.3 Panoramic Image Object Detection

Multi-projection YOLO [29] handles projection distortions by making multiple stereographic sub-projections. Then each sub-projection is separately processed by the YOLO detector. Multi-kernel [26] introduces multi-kernel layers for improving accuracy for distorted object detection and adds position information into the model for learning spatial information. Sphere-SSD [1] is the spherical single shot multi-box detector with the RMSProp optimizer to panoramic images. SpherePHD [11] utilizes a spherical polyhedron to represent Omnidirectional views, which minimizes the variance of the spatial resolving power on the sphere surface. Reprojection R-CNN [33] is a two-stage panoramic object detector. The first stage generates coarse proposals, and the second stage refines the proposals to yield precise BFoVs. Sph-CenterNet [35] is an anchor-free object detection algorithm for spherical images. It adds the geometry for spherical images.

3 RBFoV

3.1 RBFoV Representation

The bounding box and IoU are a fundamental part of the object detector, where the positive and negative sample definition, NMS [16] and mAP [6] are all defined on those two elements. Therefore, it is important to establish a reasonable bounding box representation and an accurate and efficient IoU calculation method for the panoramic image object detection task.

We use the RBFoV as the bouding box. The RBFoV is defined by $(\theta, \phi, \alpha, \beta, \gamma)$, where θ and ϕ are the longitude and latitude coordinates of the object center, and α, β denote the up-down and left-right field-of-view angles of the object's occupation, γ represents the angle (clockwise is positive, counterclockwise is negative) of the rotation of the tangent plane of the RBFoV along the axis $\vec{OM}$ (The M is the tangent point (θ, ϕ)), as shown in Fig. 2(a,b). The range of values of γ is $[-90, 90]$.

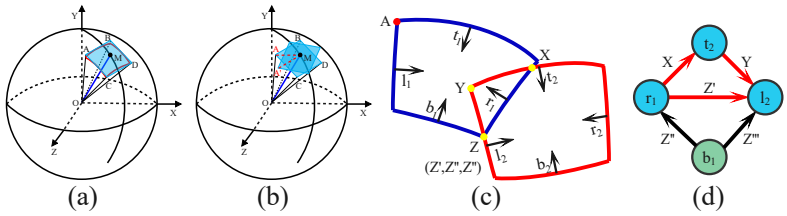


Fig. 2. (a) The RBFoV can be represented by either a spherical rectangle (red) or a tangent plane (blue) with M as the tangent point. (b) The angle γ (i.e., $\angle AMA'$) in the RBFoV is obtained by rotating the tangent plane along the axis $\vec{OM}$. (c) The intersection area of two RBFoVs is determined from the normal vectors $[\vec{t_i}, \vec{b_i}, \vec{l_i}, \vec{r_i}]$ of the planes that the neighboring sides of each RBFoV lie on. (d) We create the directed graph and use the DFS algorithm [19] to remove duplicated points.

3.2 IoU Calculation between two RBFoVs

The shape of a RBFoV $(\theta, \phi, \alpha, \beta, \gamma)$ can be regarded as a spherical rectangle. The work in [35] gives the formula of the area for a spherical rectangle:

$$Area(B) = 4 \arccos(-\sin\frac{\alpha}{2}\sin\frac{\beta}{2}) - 2\pi. \tag{1}$$

In order to compute the intersection area between two RBFoVs, we need to obtain the normal vectors $[\vec{t}, \vec{b}, \vec{l}, \vec{r}]$ of the planes that the neighboring sides of each RBFoV lie on. The normal vector derivation is given in the supplementary material. Next, the intersection points are obtained by normal vectors. The intersection points may contain the vertices of RBFoVs and the intersection points of boundaries. Vertices can be easily calculated by cross multiplication of two normal vector of RBFoV boundary planes, e.g. Vertex A is obtained by $\vec{t_1} \times \vec{l_1}$ shown in Fig. 2(c). The intersection points of boundaries can be computed by cross multiplication of two normal vectors, one is from the first RBFoV and another from the second, e.g., as shown in Fig. 2(c), Point X is computed by $\vec{r_1} \times \vec{t_2}$. In addition, some points that are duplicate or outside the intersection region must be removed. We first conduct dot product of points and normal vectors, and all result values not less than 0 are the inner points. Then we remove duplicated points, such as Point Z shown in Fig. 2(c), by creating the directed graph and using DFS algorithm [19] to find real intersection points in red Fig. 2(d). Finally, normal vectors of boundaries for real intersection points are conducted by dot product to calculate spherical angles of the intersection region. Based on spherical angles, we can find the area of the intersection region using the following formula [28]:

$$A(B_1 \cap B_2) = \sum_{i=1}^{n} \omega_i - (n-2)\pi, \tag{2}$$

where n is the number of intersection points, ω is the spherical angle of the intersection region.

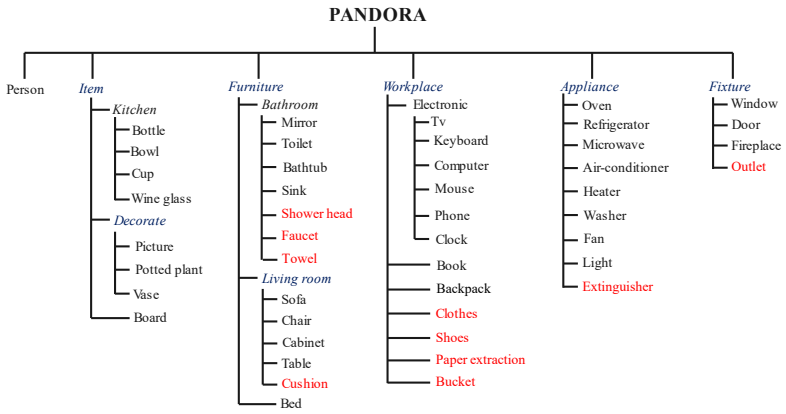


Fig. 3. Categories in our PANDORA dataset. The italic font denotes the super-categories, and the black font denotes the 37 categories in the existing 360-indoor [3] dataset, and the red font denotes the 10 categories added to our PANDORA dataset. There are 47 categories in our PANDORA dataset.

4 PANDORA Dataset

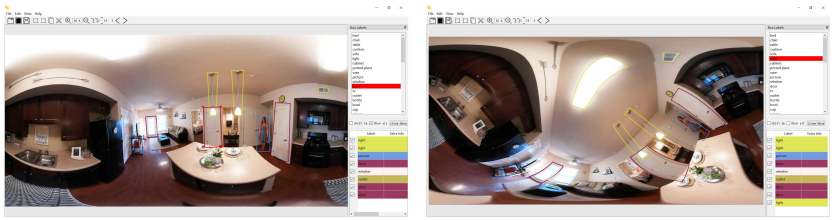
In this section, based on the RBFoV, we present a PANoramic Detection dataset for Object with oRientAtion (PANDORA).

4.1 Image Collection

We aim to cover diverse indoor scenarios in our PANDORA dataset. We selected some popular indoor scenes. Based on these scenes, we collected $3,000$ panoramic images, of which most are from the 360cities and Flickr. Specifically, we consider three main aspects when selecting images, namely, **1)** images of the real world, **2)** many instances per image, and **3)** many different indoor scenes, which make the dataset approach real-world applications. All images are with $1,920 \times 960$ resolution.

4.2 Category Selection

Forty-seven categories are chosen and annotated in our PANDORA dataset. The first 37 categories are in the existing dataset [3], we keep them all. Others are added mainly from the values in real applications. For example, we select *extinguishers* considering that measures for conflagration prevention are of significant importance indoor. We also add some categories which are common in the indoor scenes, such as *shoes*, *clothes*, *cushion*, etc. Next, similar to 360-indoor, we classify the object categories into five super-categories, except *person*. Each super-category represents a kind of scene. Fig. 3 shows the 48 categories selected for annotation and the super categories in the PANDORA.



(a) (b)

Fig. 4. The annotation tool of object detection for panoramic images. For objects with orientation, the annotator can rotate the annotation box to better bounding the object, as shown in (a). For objects of the poles, the annotator can rotate the image to find the appropriate annotation view, as shown in (b).

4.3 Image Annotation

In existing panoramic image annotation tools, such as the tool in [3], annotators are asked first to choose a viewpoint and use the buttons to adjust the bounding box size. Compared with LabelImg [14], which is inefficient. According to the particularity of panoramic images, we find that the planar rectangle in the panoramic image can be converted to the spherical rectangle. Based on this, we designed an annotation tool similar to LabelImg, as shown in Fig. 4 For objects in the polar regions and on the boundary, annotators can rotate the panoramic image to find the appropriate annotation view, as shown in Fig. 4(b).

4.4 Dataset Statistics

Next, we analyze the properties of the PANDORA dataset. Our PANDORA contains 3,000 images, including 94,353 bounding box from 47 categories. We split the dataset into trainning and testing set with 0.7 and 0.3. Firstly, we show the distribution of the top 10 categories and the number of per image instances in Fig. 5(a) and Fig. 5(b). In addition, aspect ratio (AR) is an essential factor for object detection models, such as Faster RCNN [22], SSD [15] and YOLOv2 [21]. We count the AR for all the instances in our PANDORA dataset to provide a reference for better model design. Fig. 5(c) illustrates the distribution of aspect ratio for instances in our PANDORA dataset. We can see that instances varies greatly in aspect ratio. Moreover, there are a large number of instances with a large aspect ratio in our dataset. Finally, we analyze the distribution of latitude coordinates of object center in PANDORA in Fig. 5(d). The majority of objects appear between latitudes of 0° to $\pm 50^\circ$. It is common because objects appear more often at the image center than at the polar regions in indoor scenes. There is less distortion of the object in the panoramic image center, but the distortion becomes more pronounced when the object is near the polar regions. The PANDORA dataset provides data of many objects at the polar regions that can assist the model in recognizing the high latitudes region.

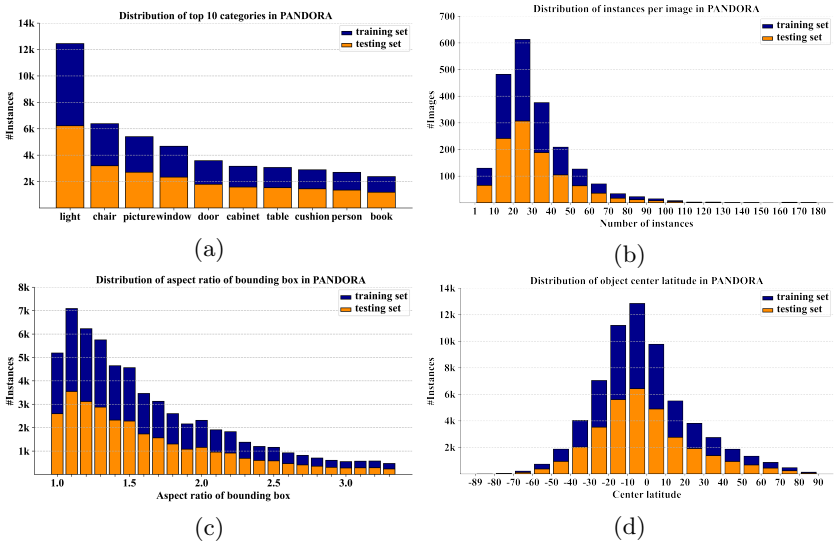


Fig. 5. Statistics of instances in PANDORA. (a) Number of annotated instances per catetory in the top 10 categories for PANDORA. (b) Number of annotated instances per image for PANDORA. (c) The aspect ratio of bounding box. (d) Distribution of latitude coordinates of object center in PANDORA.

5 R-CenterNet

We propose an anchor-free object detection method based on Sph-CenterNet [35], called R-CenterNet, to evaluate our PANDORA dataset better. In addition, we propose a panoramic rotation data augmentation technique that can increase the diversity of training data.

5.1 Network architecture and Loss Definition

We use the anchor-free detection Sph-CenterNet [35] as the baseline. First, we need to clarify that the network has not changed the output of the original regression branch. To predict the RBFoV, we add a branch to regress the rotation angle γ of the RBFoV, as illustrated in Fig. 6. We use direct and indirect two forms for the regression of γ.

First, for direct regression, the model directly predicts the angle $\hat{\gamma}$ to match the ground truth γ:

$$L_{direct} = \frac{1}{N}\sum_{i=1}^{N}\left|\gamma_i - \hat{\gamma}_i\right|, \tag{3}$$

where γ_i and $\hat{\gamma}_i$ are the target and predicted rotation angles for object i; and N is the number of positive samples.

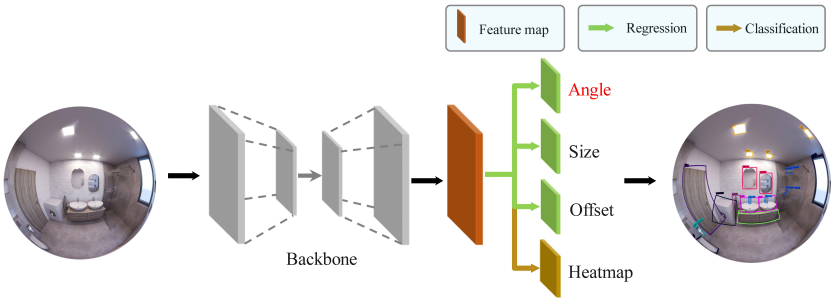


Fig. 6. Overall framework of our R-CenterNet. The network takes panoramic images as input, and predicts heatmaps, offsets, sizes and angles.

Second, for indirect regression, the R-CenterNet predicts two vectors ($\sin\hat{\gamma}$ and $\cos\hat{\gamma}$) to match the two targets from the ground truth ($\sin\gamma$ and $\cos\gamma$):

$$L_{indirect} = \frac{1}{N}\sum_{i=1}^{N} |\sin\gamma_i - \sin\hat{\gamma}_i| + |\cos\gamma_i - \cos\hat{\gamma}_i| \,. \tag{4}$$

We will carry out the normalization processing to make $\sin\hat{\gamma}^2 + \cos\hat{\gamma}^2 = 1$:

$$\sin\hat{\gamma} = \frac{\sin\hat{\gamma}}{\sqrt{\sin^2\hat{\gamma} + \cos^2\hat{\gamma}}}, \quad \cos\hat{\gamma} = \frac{\cos\hat{\gamma}}{\sqrt{\sin^2\hat{\gamma} + \cos^2\hat{\gamma}}}. \tag{5}$$

Thus, the overall training objective of our model is

$$L_{det} = L_{cls} + \lambda_{size}L_{size} + \lambda_{off}L_{off} + \lambda_{ang}(L_{direct} + L_{indirect}), \tag{6}$$

where L_{cls}, L_{size} and L_{off} are the losses of center point recognition, scale regression, and offset regression, which are the same as Sph-CenterNet; and λ_{size}, λ_{off} and λ_{ang} are constant factors, set to 0.1 in our experiments.

5.2 Implementation Details

Generating Ground-truth Heatmaps. When assigning ground-truth information to heatmaps in the Sph-CenterNet, cells around the center point of a bounding box showed an independent Gaussian density, which draws a circle in the sphere regardless of the actual shape and orientation of the object in panoramic images. We propose a new method of assigning ground-truth that can change the shape of the Gaussian according to the shape and orientation of the objects, as illustrated in Fig. 7.

First, for each point (u, v) within the RBFoV, the corresponding coordinates in tangent plane $\mathit{\Pi}[\theta, \phi]$ could be calculated via the gnomonic projection [18,37]:

$$\begin{aligned} x(u,v) &= \frac{\cos u \, \sin(v-\phi)}{\sin\theta \, \sin u + \cos\theta \, \cos u \, \cos(v-\phi)}, \\ y(u,v) &= \frac{\cos\theta \, \sin v - \sin\theta \, \cos u \, \cos(v-\phi)}{\sin\theta \, \sin u + \cos\theta \, \cos u \, \cos(v-\phi)}. \end{aligned} \tag{7}$$

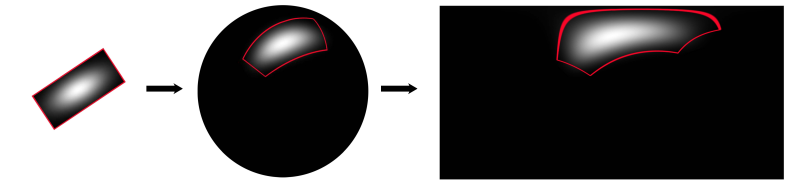


Fig. 7. (a) We convert the tangent plane Π of the RBFoV into a 2-D Gaussian distribution. (b) The tangent plane Π projects back onto the spherical heatmap. (c) We project the spherical heatmap to the ERP heatmap.

The $\Pi[\theta, \phi]$ is an oriented rectangle $B(\theta, \phi, w, h, \gamma)$, where $w = 2\tan(0.5\alpha)$ and $h = 2\tan(0.5\beta)$. As illustrated in Fig. 7(a), we convert the Π into a 2-D Gaussian distribution $\mathcal{N}(\mu, \sigma^2)$ by the following formula [30]:

$$\begin{aligned} \sigma =& \mathbf{R}\mathbf{\Lambda}\mathbf{R}^\top = \begin{pmatrix} \cos\gamma & -\sin\gamma \\ \sin\gamma & \cos\gamma \end{pmatrix} \begin{pmatrix} \frac{w}{2} & 0 \\ 0 & \frac{h}{2} \end{pmatrix} \begin{pmatrix} \cos\gamma & \sin\gamma \\ -\sin\gamma & \cos\gamma \end{pmatrix} \\ \mu =& (\theta, \phi) \end{aligned} \tag{8}$$

where $\mathbf{R}$ represents the rotation matrix, and $\mathbf{\Lambda}$ represents the diagonal matrix of eigenvalues.

Then, as illustrated in Fig. 7(b), the inverse gnomonic projection is used to prject the $\Pi[\theta, \phi]$ back onto the spherical heatmap by the following formula [18]:

$$\begin{aligned} u(x, y) &= \sin^{-1}(\cos\nu \sin\theta + \frac{y \sin\nu \cos\theta}{\rho}), \\ v(x, y) &= \phi + \tan^{-1}\left(\frac{x \sin\nu}{\rho\cos\theta\cos\nu - y\sin\theta\sin\nu}\right). \end{aligned} \tag{9}$$

where $\rho = \sqrt{x^2 + y^2}$ and $\nu = \tan^{-1}\rho$.

Last, we project the spherical heatmap to the ERP heatmap , As illustrated in Fig. 7(c). The ERP heatmap is the ground-truth Y_{xyc}. If two Gaussians of the same class overlap, we take the element-wise maximum.

5.3 Panoramic Rotation Data Augmentation

For a panoramic image, we propose to rotate the panoramic image by η angle along n-axis in 3-D space to augment training data, where η and n are arbitrary values. To achieve this goal, we first represent each pixel under UV space as (u, v) where $u \in [-\pi, \pi], v \in [-\pi/2, \pi/2]$. The coordinate (u, v) can be easily computed as the column and row of an equirectangular image. We project the pixels to 3-D space and multiply their x, y, z by the rotation matrix $\mathcal{T}(n, \eta)$, where n is the axis and η is the angle of rotation along the axis. The equation

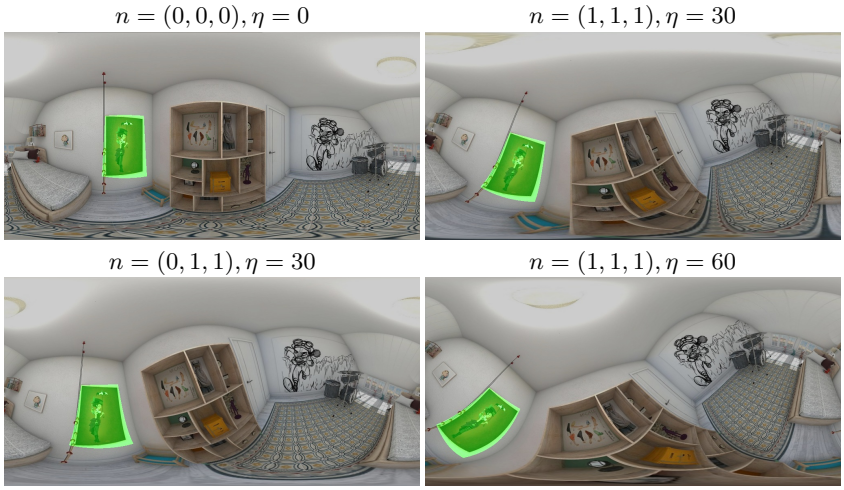


Fig. 8. Visualization of the proposed *Panoramic Rotation Data Augmentation*. We rotate the panoramic image by η angle along n-axis. The green bounding box is RBFoV. This augmentation strategy improves our quantitative results under experiment settings (Table 4).

of rotated x', y', z' are shown in Eq. 10.

$$\begin{pmatrix} x' \\ y' \\ z' \end{pmatrix} = \mathcal{T}(n, \eta) \cdot \begin{pmatrix} x & y & z \end{pmatrix}^\top = \mathcal{T}(n, \eta) \cdot \begin{pmatrix} \cos(v) \cdot \sin(u) \\ \sin(v) \\ \cos(v) \cdot \cos(u) \end{pmatrix} \tag{10}$$

We can then project the rotated points back to the sphere by Eq. 11 for further equirectangular projection. $atan2$ in the equation is 2-argument arctangent.

$$\begin{aligned} u' &= atan2(x', z'), \\ v' &= atan2(y', \sqrt{(x')^2 + (z')^2}). \end{aligned} \tag{11}$$

After that, we need to obtain the parameters in the RBFoV after the panoramic image rotation. As we rotate the panoramic image by η angle along n-axis, the $b(\theta, \phi, \alpha, \beta, \gamma)$ become $b'(\theta', \phi', \alpha, \beta, \gamma')$. The θ' and ϕ' can be obtained from Eq. 10-11. The vertex A^* is obtained from the bounding box $(\theta', \phi', \alpha, \beta, 0)$, which can be found by rotating vertice A^* by γ' angle along the axis $O\vec{M}$ (M is the tangent point (θ', ϕ')) to obtain vertice A'.

$$\gamma' = \arccos(\vec{MA'}, \vec{MA^*}) \tag{12}$$

Fig. 8 is the visualization we proposed Panoramic Rotation Data Augmentation.

Method	Bounding box	$\mathbb{A}$	$\mathbb{B}$	$\mathbb{C}$	AP_{50}
Sph-CenterNet	BFoV	0.23	0.19	0.27	20.3
	RBFoV	0.11	0.07	0.18	**21.4** (+1.1)

Table 2. Quantify the study of BFoV and RBFoV as bounding boxes in panoramic images. Ground truths for BFoV experiments are generated by calculating the minimum bounding BFoVs over original annotated RBFoVs.

6 Experiment

6.1 Quantify BFoV and RBFoV

Fig. 1 is a visual example, which clearly shows how BFoVs lead to the significant overlap in bounding boxes where there should be none. To further explore the effect of the BFoV and RBFoV on the detector's performance, we design three metrics as follows:

$$\mathbb{A} = \frac{affected\ instances}{total\ instances}, \quad \mathbb{B} = \frac{overlap\ misses}{total\ misses}, \quad \mathbb{C} = \frac{overlap\ misses}{overlap\ cases}. \tag{13}$$

When IoU ≥ 0.3, we consider this an *affected instance* or *overlap case*. We consider it an *overlap miss* when the affected instance is incorrectly recognized or location (i.e., the IoU of predicted RBFoV and ground truth RBFoV ≥ 0.5). In addition to presenting Fig. 1, $\mathbb{A}$ is to quantify how common two nearby objects overlapped with BFoVs and RBFoVs representations in our dataset. $\mathbb{B}$ quantifies the percentage of all missed detections due to this overlap, which gives an idea of how different box impacts total performance. $\mathbb{C}$ quanfities the percentage of times this overlap is not detected when it should have been, which gives an idea of how bad SOTA detection methods are at addressing this inappropriate bounding box.

We use Sph-CenterNet [35] as baseline. To input a image, aim of Sph-CenterNet is to predict the BFoV for each object. To predict the RBFoV, we add a branch to regress the rotation angle of the RBFoV and use direct and indirect regression loss for angle γ. As shown in Tab. 2, we provide the quantifications regarding the overlap issue and show that using RBFoV reduces these error percentages. The results in Tab. 2 verify our analysis: compared with BFoV, RBFoV is more reasonable as the bounding box for panoramic image object detection.

6.2 Evaluations

Dataset Splits. The train set and test set of PANDORA contain 2,100 and 900 images, respectively. Considering the limitation of the computation source, we resize all the images in PANDORA into 1024×512 for training and testing.
Metric. We use standard mAP [6] as the evaluation metric for object detection in panoramic images. Please note that as original evaluation metrics used in the baseline methods are biased, we use our IoU method for evaluation.

Bounding box	Methods	Backbone	AP	AP_{50}	AP_{75}
RBFoV	Multi-Kernel [26]	ResNet-101	3.8	13.7	1.0
	Sphere-SSD [1]	ResNet-101	3.2	12	0.6
	Reprojection R-CNN [33]	ResNet-101	4.3	16.6	0.7
	Sph-CenterNet [35]	ResNet-101	5.5	19.9	1.1
	Our R-CenterNet	ResNet-101	**7.3**	**22.7**	**2.6**

Table 3. Numerical results (AP) of baseline models evaluated with RBFoV ground-truths on PANDORA test-dev. We add a branch to the output of these methods for predicting the angle of the RBFoV and use L1 loss.

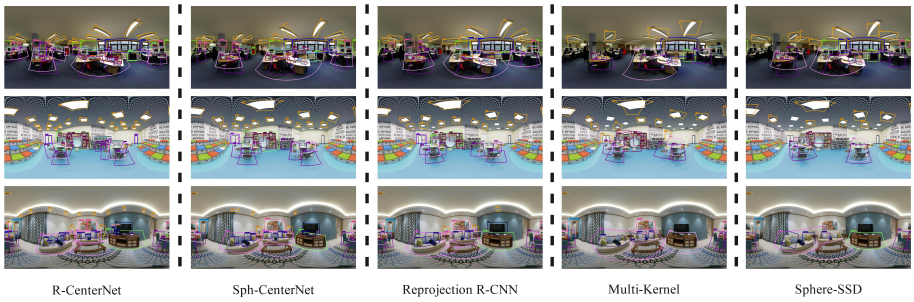


Fig. 9. Visualization results of different methods on the PANDORA dataset.

Training Details. Our approach is implemented in PyTorch [17], and training is done on 8 GeForce RTX 2080Ti GPUs with a batch size of 32. We utilize Adam [10] to optimize the overall parameters objective for 160 epochs with the initial learning rate of 1.25×10^{-4}, and at 90 and 120 epochs, the learning rate is divided by 10.

Evaluation Tasks. We take Multi-Kernel [26], Sphere-SSD [1], Reprojection R-CNN [33] and Sph-CenterNet [35] as our baseline methods. To make it fair, we keep all the experiments' settings and hyper parameters the same as depicted in corresponding papers. All the methods take the ERP image as input, and the backbone networks are all the same ResNet-101 [7] architecture. The SphereNet Kerner [1] instead of the regular kernel in the CNN and the IoU use all we proposed except Multi-Kernel. Since its output bounding boxes are planar rectangles for Multi-Kernel, we still use the original planar IoU calculation method in its first stage. After these planar rectangles are predicted, we convert them to spherical rectangles. We add a branch to the output of these methods for predicting the angle of the RBFoV and use L1 loss.

Quantitative Results. The results of prediction are shown in Tab.3. It is obvious that the two-stage approach Multi-Kernel and Reprojection R-CNN achieve better performance than the one-stage Sphere-SSD. Multi-Kernel uses a planar IoU calculation method in the first stage, resulting in a lower performance than

Method	Our heatmap	PRDA	IDR	DR	AP_{50}
Sph-CenterNet [35]				✓	19.9
	✓			✓	**20.5** (+0.6)
		✓		✓	**21.6** (+1.7)
			✓	✓	**21.4** (+1.5)
	✓	✓	✓	✓	**22.7** (+2.8)

Table 4. Ablation study demonstrates the effectiveness of each component. The PRDA is Panoramic Rotation Data Augmentation. The DR and IDR are direct and indirect regression for angle γ, respectively.

Reprojection R-CNN. Since we change the generating ground-truth heatmaps, and use *Panoramic Rotation Data Augmentation*, our R-CenterNet effect is better than Sph-CenterNet.

Visual Detection Results. As illustrated in Figure 9, we give the results of the visualization of different methods on the PANDORA dataset. As shown, the R-CenterNet has good performance in both dense and small object detection. The visualization results are consistent with the data results in Tab.3.

6.3 Ablation Study

Ablation experiments are presented in Table 4. We choose Sph-CenterNet [35] as the baseline for ablation study. For fairness, all experimental data and parameter settings are strictly consistent. We use AP_{50} as a measure of performance. It also can be evidenced in Table 4 that the detection results have been improved to varying degrees after adding each of the components we propose, and the total AP_{50} increased by 2.8%.

7 Conclusion

In this paper, we propose a new bounding box representation, RBFoV, for the panoramic image object detection task. Then, based on the RBFoV, we present a PANoramic Detection dataset for Object with oRientAtion (PANDORA). To our best knowledge, PANDORA is the first dataset to use the RBFoV as the bounding box. Finally, based on PANDORA, we evaluate the current SOTA methods and design an anchor-free object detector called R-CenterNet for panoramic images. Compared with these baselines, our R-CenterNet shows its advantages in terms of detection performance. By releasing PANDORA, We believe it will promote the development of object detection algorithms in panoramic images.

8 Acknowledgements

This work is supported by the National Key Research and Development Program of China (2020YFB1406604) and the National Natural Science Foundation of China (62072438, U1936110, 61931008, U21B2024).

References

1. AndreasGeiger, BenjaminCoors, AlexandruPaulCondurache, AndreasGeiger, BenjaminCoors, AlexandruPaulCondurache, AndreasGeiger, BenjaminCoors, AlexandruPaulCondurache, and, A.: Spherenet: Learning spherical representations for detection and classification in omnidirectional images. In: ECCV (2018)
2. Anguelov, D., Dulong, C., Filip, D., Frueh, C., Lafon, S., Lyon, R., Ogale, A., Vincent, L., Weaver, J.: Google street view: Capturing the world at street level. Computer (2010)
3. Chou, S.H., Sun, C., Chang, W.Y., Hsu, W.T., Sun, M., Fu, J.: 360-indoor: Towards learning real-world objects in 360deg indoor equirectangular images. In: WACV (2020)
4. Cormack, R.: Flattening the earth: Two thousand years of map projectionsby john p. snyder;two by two: Twenty-two pairs of maps from the newberry library illustrating five hundred years of western cartographic historyby james akerman; robert karrow; david buisseret. Isis **85**(3), 488–489 (1994)
5. Everingham, M., Gool, L.V., Williams, C.K.I., Winn, J., Zisserman, A.: The pascal visual object classes (voc) challenge. IJCV **88**(2), 303–338 (2010)
6. Everingham, M., ·, S.M.A.E., Gool, L.V., Williams, C.K.I., Winn, J., Zisserman, A.: The pascal visual object classes challenge: A retrospective. IJCV **111**(1), 98–136 (2015)
7. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. CVPR (2016)
8. Hu, H.N., Lin, Y.C., Liu, M.Y., Cheng, H.T., Chang, Y.J., Sun, M.: Deep 360 pilot: Learning a deep agent for piloting through 360 sports videos. In: CVPR (2017)
9. Huang, J., Chen, Z., Research, A., Ceylan, U.D., Hailin, U.: 6-dof vr videos with a single 360-camera. In: VR (2017)
10. Kingma, D., Ba, J.: Adam: A method for stochastic optimization. Computer Science (2014)
11. Lee, Y., Jeong, J., Yun, J., Cho, W., Yoon, K.J.: Spherephd: Applying cnns on a spherical polyhedron representation of 360 images (2019)
12. Lin, T.Y., Dollar, P., Girshick, R., He, K., Hariharan, B., Belongie, S.: Feature pyramid networks for object detection. In: CVPR (2017)
13. Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft COCO: Common objects in context. In: ECCV (2014)
14. Lin, T.: Labelimg (2015)
15. Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.Y., Berg, A.C.: Ssd: Single shot multibox detector. In: ECCV (2016)
16. Neubeck, A., Gool, L.: Efficient non-maximum suppression. In: ICPR (2006)
17. Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E., DeVito, Z., Lin, Z., Desmaison, A., Antiga, L., Lerer, A.: Automatic differentiation in pytorch (2017)
18. Pearson, F.: Map projections: Theory and applications. Crc Press (1990)
19. Putri, S.E., Tulus, Napitupulu, N.: Implementation and analysis of depth-first search (dfs) algorithm for finding the longest path. In: InteriOR (2011)
20. Ran, L., Zhang, Y., Zhang, Q., Tao, Y.: Convolutional neural network-based robot navigation using uncalibrated spherical images. Sensors **17**(6), 1341 (2017)
21. Redmon, J., Farhadi, A.: Yolo9000: Better, faster, stronger. In: CVPR (2017)
22. Ren, S., He, K., Girshick, R., Sun, J.: Faster r-cnn: Towards real-time object detection with region proposal networks. In: NIPS (2015)

23. Su, Y.C., Grauman, K.: Learning spherical convolution for fast features from 360 imagery. In: CVPR (2017)
24. Su, Y., Jayaraman, D., Grauman, K.: Pano2vid: automatic cinematography for watching 360° videos. In: ACCV (2016)
25. Tian, Z., Shen, C., Chen, H., He, T.: Fcos: Fully convolutional one-stage object detection. In: ICCV (2020)
26. Wang, K.H., Lai, S.H.: Object detection in curved space for 360-degree camera. In: ICASSP (2019)
27. Wei-Sheng, Lai, Yujia, Huang, Neel, Joshi, Christopher, Buehler, Ming-Hsuan, Yang: Semantic-driven generation of hyperlapse from 360[formula: see text] video. TVCG (2017)
28. Wikipedia contributors: Spherical trigonometry. `https://en.wikipedia.org/w/index.php?title=Spherical_trigonometry&oldid=1016967508` (2021)
29. Yang, W., Qian, Y., Cricri, F., Fan, L., Kamarainen, J.K.: Object detection in equirectangular panorama
30. Yang, X., Yan, J., Qi, M., Wang, W., Xiaopeng, Z., Qi, T.: Rethinking rotated object detection with gaussian wasserstein distance loss. In: International Conference on Machine Learning (2021)
31. Yu, D., Ji, S.: Grid based spherical cnn for object detection from panoramic images. Sensors **19**(11), 2622 (2019)
32. Zhang, S., Chi, C., Yao, Y., Lei, Z., Li, S.Z.: Bridging the gap between anchor-based and anchor-free detection via adaptive training sample selection. In: CVPR (2020)
33. Zhao, P., You, A., Zhang, Y., Liu, J., Tong, Y.: Spherical criteria for fast and accurate 360° object detection. In: AAAI. vol. 34, pp. 12959–12966 (2020)
34. Zhao, Q., Zhu, C., Dai, F., Ma, Y., Zhang, Y.: Distortion-aware cnns for spherical images. In: Twenty-Seventh International Joint Conference on Artificial Intelligence IJCAI-18 (2018)
35. Zhao, Q., Chen, B., Xu, H., Ma, Y., Li, X., Feng, B., Yan, C., Dai, F.: Unbiased iou for spherical image object detection. In: AAAI (2022)
36. Zhao, Q., Feng, W., Wan, L., Zhang, J.: Sphorb: A fast and robust binary feature on the sphere. International Journal of Computer Vision **113**(2), 143–159 (2015)
37. Zhao, Q., Wan, L., Feng, W., Zhang, J., Wong, T.T.: Cube2video: Navigate between cubic panoramas in real-time. IEEE Transactions on Multimedia **15**(8), 1745–1754 (2013)
38. Zheng, J., Liu, X., Gu, X., Sun, Y., Gan, C., Zhang, J., Liu, W., Yan, C.: Gait recognition in the wild with multi-hop temporal switch. In: ACM MM (2022)
39. Zheng, J., Liu, X., Liu, W., He, L., Yan, C., Mei, T.: Gait recognition in the wild with dense 3d representations and a benchmark. In: CVPR. pp. 20228–20237 (2022)
40. Zhou, X., Wang, D., Krhenbühl, P.: Objects as points. In: arXiv (2019)