RAPiD: Rotation-Aware People Detection in Overhead Fisheye Images

Zhihao Duan, M. Ozan Tezcan, Hayato Nakamura, Prakash Ishwar, Janusz Konrad *
Boston University
{duanzh, mtezcan, nhayato, pi, jkonrad}@bu.edu

Abstract

Recent methods for people detection in overhead, fisheye images either use radially-aligned bounding boxes to represent people, assuming people always appear along image radius or require significant pre-/post-processing which radically increases computational complexity. In this work, we develop an end-to-end rotation-aware people detection method, named RAPiD, that detects people using arbitrarily-oriented bounding boxes. Our fully-convolutional neural network directly regresses the angle of each bounding box using a periodic loss function, which accounts for angle periodicities. We have also created a new dataset[1] with spatio-temporal annotations of rotated bounding boxes, for people detection as well as other vision tasks in overhead fisheye videos. We show that our simple, yet effective method outperforms state-of-the-art results on three fisheye-image datasets. The source code for RAPiD is publicly available[2].

1. Introduction

Occupancy sensing is an enabling technology for smart buildings of the future; knowing where and how many people are in a building is key for saving energy, space management and security (e.g., fire, active shooter). Various approaches to counting people have been developed to date, from virtual door tripwires to WiFi signal monitoring. Among those, video cameras combined with computer vision algorithms have proven most successful [6, 18, 1]. Typically, a wide-angle, standard-lens camera is side-mounted above the scene; multiple such cameras are used for large spaces. An alternative is to use a single overhead, fisheye camera with a 360° field of view (FOV). However, people detection algorithms developed for side-view, standard-lens images do not perform well on overhead, fisheye images due to their unique radial geometry and barrel distortions.

*This work was supported in part by ARPA-E under agreement DE-AR0000944 and by the donation of Titan GPUs from NVIDIA Corp.

[1] vip.bu.edu/cepdof

[2] vip.bu.edu/rapid

(a) Axis-aligned (b) Radius-aligned (c) Human-aligned

Figure 1: Illustration of *typical* people-detection results on overhead, fisheye images (one quarter shown) for algorithms using various bounding-box orientation constraints; the human-aligned bounding boxes fit bodies most accurately. These are not outputs from any algorithms. See the text for discussion.

In standard images, standing people usually appear in an upright position and algorithms that detect bounding boxes aligned with image axes, such as YOLO [21], SSD [15] and R-CNN [24], work well. However, these algorithms perform poorly on overhead, fisheye images [12], usually missing non-upright bodies (Fig. 1a). In such images, standing people appear along image radius, due to the overhead placement of the camera, and rotated bounding boxes are needed. To accommodate this rotation, several people-detection algorithms, mostly YOLO-based, have been recently proposed [2, 30, 11, 12, 27, 34], each dealing differently with the radial geometry. For example, in one of the top-performing algorithms [12], the image is rotated in 15° steps and YOLO is applied to the top–center part of the image (where people usually appear upright) followed by post-processing. However, this requires 24-fold application of YOLO. Another recent algorithm [27] requires that bounding boxes be aligned with image radius, but often fails to detect non-standing poses (Fig. 1b).

In this paper, we introduce Rotation-Aware People Detection (RAPiD), a novel end-to-end people-detection algorithm for overhead, fisheye images. RAPiD is a single-stage convolutional neural network that predicts arbitrarily-rotated bounding boxes (Fig. 1c) of people in a fisheye image. It extends the model proposed in YOLO [21, 22, 23], one of the most successful object detection algorithms for standard images. In addition to predicting the center and

size of a bounding box, RAPiD also predicts its angle. This is accomplished by a periodic loss function based on an extension of a common regression loss. This allows us to predict the exact rotation of each bounding box in an image without any assumptions and additional computational complexity. Since RAPiD is an end-to-end algorithm, we can train or fine-tune its weights on annotated fisheye images. Indeed, we show that such fine-tuning of a model trained on standard images significantly increases the performance. An additional aspect of this work, motivated by its focus on people detection, is the replacement of the common regression-based loss function used in multi-class object detection algorithms [21, 15, 8, 24] with single-class object detection. The inference speed of RAPiD is nearly identical to that of YOLO since it is applied to each image only once without the need for pre-/post-processing.

We evaluate the performance of RAPiD on two publicly-available, people-detection datasets captured by overhead fisheye cameras, Mirror Worlds (MW)[3] and HABBOF [12]. Although these datasets cover a range of scenarios, they lack challenging cases such as unusual body poses, wearing a hoodie or hat, holding an object, carrying a backpack, strong occlusions, or low light. Therefore, we introduce a new dataset named Challenging Events for Person Detection from Overhead Fisheye images (CEPDOF) that includes such scenarios. In our evaluations, RAPiD outperforms state-of-the-art algorithms on all three datasets.

The main contributions of this work can be summarized as follows:

- We propose an end-to-end neural network, which extends YOLO v3, for rotation-aware people detection in overhead fisheye images and demonstrate that our simple, yet effective approach, outperforms the state-of-the-art methods.
- We propose a continuous, periodic loss function for bounding-box angle that, unlike in previous methods, facilitates arbitrarily-oriented bounding boxes capable of handling a wide range of human-body poses.
- We introduce a new dataset for people detection from overhead, fisheye cameras that includes a range of challenges; it can be also useful for other tasks, such as people tracking and re-identification.

2. Related work

People detection using side-view standard-lens cameras: Among traditional people-detection algorithms for standard cameras, the most popular ones are based on the histogram of oriented gradients (HOG) [3] and aggregate channel features (ACF) [5]. Recently, deep learning algorithms have demonstrated outstanding performance in object and people detection [21, 15, 7, 8, 24, 10]. These algorithms can be divided into two categories: two-stage methods and one-stage methods. Two-stage methods, such as R-CNN and its variants [8, 24, 10], consist of a Region Proposal Network (RPN) which predicts the Region of Interest (ROI) and a network head refines the bounding boxes. One-stage methods, such as variants of SSD [15, 7] and YOLO [21, 22, 23], could be viewed as independent RPNs. Given an input image, one-stage methods directly regress bounding boxes through CNNs. Recently, attention has focused on fast one-stage detectors [33, 28] and anchor-free detectors [29, 32].

Object detection using rotated bounding boxes: Detection of rotated bounding boxes has been widely studied in text detection and aerial image analysis [16, 4, 31, 20]. RRPN [16] is a two-stage object detection algorithm which uses rotated anchor boxes and a rotated region-of-interest (RRoI) layer. RoI-Transformer [4] extended this idea by first computing a horizontal region of interest (HRoI) and then learning the warping from HRoI to RRoI. R^3Det [31] proposed a single-stage rotated bounding box detector by using a feature refinement layer to solve feature misalignment occurring between the region of interest and the feature, a common problem of single-stage methods. In an alternative approach, Nosaka *et al.* [19] used orientation-aware convolutional layers [34] to handle the bounding box orientation and a smooth $L1$ loss for angle regression. All of these methods use a 5-component vector for rotated bounding boxes (coordinates of the center, width, height and rotation angle) with the angle defined in $[\frac{-\pi}{2}, 0]$ range and a traditional regression loss. Due to symmetry, a rectangular bounding box having width b_w, height b_h and angle θ is indistinguishable from one having width b_h, height b_w and angle $(\theta - \pi/2)$. Hence a standard regression loss, which does not account for this, may incur a large cost even when the prediction is close to the ground truth, e.g., if the ground-truth annotation is $(b_x, b_y, b_h, b_w, -4\pi/10)$, a prediction $(b_x, b_y, b_w, b_h, 0)$ may seem far from the ground truth, but is not so since the ground truth is equivalent to $(b_x, b_y, b_w, b_h, \pi/10)$. RSDet [20] addresses this by introducing a modulated rotation loss.

People detection in overhead, fisheye images: People detection using overhead, fisheye cameras is an emerging area with sparse literature. In some approaches, traditional people-detection algorithms such as HOG and LBP have been applied to fisheye images with slight modifications to account for fisheye geometry [30, 2, 25, 11]. For example, Chiang and Wang [2] rotated each fisheye image in small angular steps and extracted HOG features from the top-center part of the image. Subsequently, they applied SVM classifier to detect people. In another algorithm, Krams and Kiryati [11] trained an ACF classifer on side-view images and dewarped the ACF features extracted from the fisheye

[3] www2.icat.vt.edu/mirrorworlds/challenge/index.html

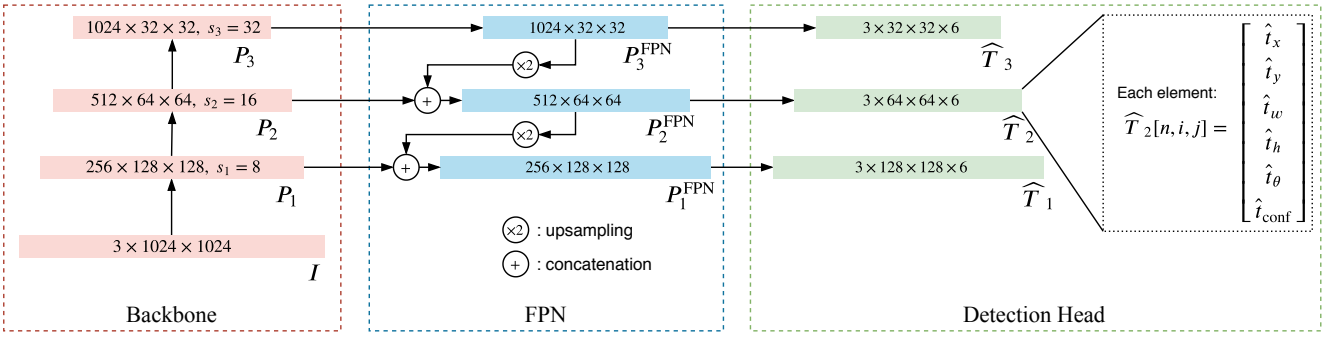


Figure 2: RAPiD architecture. Following the paradigm of one-stage detectors, our model contains a backbone, FPN, and detection head (bounding-box regression network). In the diagram, each arrow represents multiple convolutional layers and the colored rectangles represent multi-dimensional matrices, i.e., feature maps, whose dimensions correspond to input image of size $h \times w = 1,024 \times 1,024$.

image for person detection.

Recently, CNN-based algorithms have been applied to this problem as well. Tamura *et al.* introduced a rotation-invariant version of YOLO [21] by training the network on a rotated version of the COCO dataset [14]. The inference stage in their method relies on the assumption that bounding boxes in a fisheye image are aligned with the image radius. Another YOLO-based algorithm [26] applies YOLO to dewarped versions of overlapping windows extracted from a fisheye image. Li *et al.* [12] rotate each fisheye image in 15° steps and apply YOLO only to the upper-center part of the image where people usually appear upright. Subsequently, they apply post-processing to remove multiple detections of the same person. Although their algorithm is very accurate, it is computationally complex as it applies YOLO 24 times to each image.

In this work, we introduce an angle-aware loss function to predict the exact angle of bounding boxes without any additional assumptions. We also change the commonly-used representation of rotated bounding boxes to overcome the symmetry problem (Section 3.2.2).

3. Rotation-Aware People Detection (RAPiD)

We propose RAPiD, a new CNN that, in addition to the location and size, also estimates the angle of each bounding box in an overhead, fisheye image. During training, RAPiD includes a rotation-aware regression loss to account for these angles. RAPiD's design has been largely motivated by YOLO. Below, we explain this design in detail and we highlight the concepts we borrowed from YOLO as well as novel ideas that we proposed.

Notation: We use $\boldsymbol{b} = (b_x, b_y, b_w, b_h, b_\theta) \in \mathbb{R}^5$ to denote a ground-truth bounding box, where b_x, b_y are the coordinates of the bounding box center; b_w, b_h are the width and height and b_θ is the angle by which the bounding box is rotated clockwise. Similarly $\hat{\boldsymbol{b}} = (\widehat{b}_x, \widehat{b}_y, \widehat{b}_w, \widehat{b}_h, \widehat{b}_\theta, \widehat{b}_{\text{conf}}) \in \mathbb{R}^6$ denotes a predicted bounding box, where the additional element $\widehat{b}_{\text{conf}}$ denotes the confidence score of the prediction. All the angles used in the paper are in radians.

3.1. Network Architecture

Our object-detection network can be divided into three stages: backbone network, feature pyramid network (FPN) [13], and bounding box regression network, also known as the detection head:

$$\begin{aligned} P_1, P_2, P_3 &= \text{Backbone}(I) \\ P_1^{\text{fpn}}, P_2^{\text{fpn}}, P_3^{\text{fpn}} &= \text{FPN}(P_1, P_2, P_3) \\ \widehat{T}_k &= \text{Head}_k(P_k^{\text{fpn}}) \quad \forall k = 1, 2, 3 \end{aligned} \tag{1}$$

where $I \in [0,1]^{3 \times h \times w}$ is the input image, $\{P_k\}_{k=1}^3$ denotes a multi-dimensional feature matrix and $\{\widehat{T}_k\}_{k=1}^3$ denotes a list of predicted bounding boxes in transformed notation (the relationship between $\widehat{T}$ and $\hat{\boldsymbol{b}}$ will be defined soon – see equation (2)) at three levels of resolution. Fig. 2 shows the overall RAPiD architecture, while below we describe each stage in some depth. For more details, interested readers are referred to [23].

Backbone: The backbone network, also known as the feature extractor, takes an input image I and outputs a list of features (P_1, P_2, P_3) from different parts of the network. The main goal is to extract features at different spatial resolutions (P_1 being the highest and P_3 being the lowest). By using this multi-resolution pyramid, we expect to leverage both the low-level and high-level information extracted from the image.

Feature Pyramid Network (FPN): The multi-resolution features computed by the backbone are fed into FPN in order to extract features related to object detection, denoted $(P_1^{\text{fpn}}, P_2^{\text{fpn}}, P_3^{\text{fpn}})$. We expect P_1^{fpn} to contain information about small objects and P_3^{fpn} – about large objects.

Detection Head: After FPN, a separate CNN is applied to each feature vector $P_k^{\text{FPN}}, k \in \{1,2,3\}$ to produce a transformed version of bounding-box predictions, denoted $\widehat{T}_k$ – a 4-dimensional matrix with $\langle 3, h/s_k, w/s_k, 6\rangle$ dimensions. The first dimension indicates that there are three anchor boxes being used in $\widehat{T}_k$, the second and third dimensions denote the prediction grid, where $h \times w$ is the resolution of the input image and s_k is the stride at resolution level k as shown in Fig. 2, and the last dimension denotes a transformed version of the predicted bounding box for each grid cell. We denote the n^{th} transformed bounding box prediction of $Head_k$ in grid cell (i,j) as $\widehat{T}_k[n,i,j] = (\widehat{t}_x, \widehat{t}_y, \widehat{t}_w, \widehat{t}_h, \widehat{t}_\theta, \widehat{t}_{\text{conf}})$ from which a bounding-box prediction can be computed as follows:

$$\begin{aligned} \widehat{b}_x &= s_k\left(j + \text{Sig}(\widehat{t}_x)\right), & \widehat{b}_w &= w_{k,n}^{\text{anchor}} e^{\widehat{t}_w} \\ \widehat{b}_y &= s_k\left(i + \text{Sig}(\widehat{t}_y)\right), & \widehat{b}_h &= h_{k,n}^{\text{anchor}} e^{\widehat{t}_h} \\ \widehat{b}_\theta &= \alpha\,\text{Sig}(\widehat{t}_\theta) - \beta, & \widehat{b}_{\text{conf}} &= \text{Sig}(\widehat{t}_{\text{conf}}) \end{aligned} \tag{2}$$

where $\text{Sig}(\cdot)$ is the logistic (sigmoid) activation function and $w_{k,n}^{\text{anchor}}$ and $h_{k,n}^{\text{anchor}}$ are the width and height of the n^{th} anchor box for $Head_k$. Note, that angle prediction $\widehat{b}_\theta$ is limited to range $[-\beta, \alpha-\beta]$ (2). In Section 3.2.2 below, we discuss the selection of α and β values.

3.2. Angle-Aware Loss Function

Our loss function is inspired by that used in YOLOv3 [23], with an additional bounding-box rotation-angle loss:

$$\begin{aligned} \mathfrak{L} = &\sum_{\hat{\boldsymbol{t}}\in\widehat{T}^{\text{pos}}} \text{BCE}(\text{Sig}(\widehat{t}_x), t_x) + \text{BCE}(\text{Sig}(\widehat{t}_y), t_y) \\ &+ \sum_{\hat{\boldsymbol{t}}\in\widehat{T}^{\text{pos}}} (\text{Sig}(\widehat{t}_w) - t_w)^2 + (\text{Sig}(\widehat{t}_h) - t_h)^2 \\ &+ \sum_{\hat{\boldsymbol{t}}\in\widehat{T}^{\text{pos}}} \ell_{\text{angle}}(\widehat{b}_\theta, b_\theta) \\ &+ \sum_{\hat{\boldsymbol{t}}\in\widehat{T}^{\text{pos}}} \text{BCE}(\text{Sig}(\widehat{t}_{\text{conf}}), 1) + \sum_{\hat{\boldsymbol{t}}\in\widehat{T}^{\text{neg}}} \text{BCE}(\text{Sig}(\widehat{t}_{\text{conf}}), 0) \end{aligned} \tag{3}$$

where BCE denotes binary cross-entropy, ℓ_{angle} is a new angle loss function that we propose in the next section, $\widehat{T}^{\text{pos}}$ and $\widehat{T}^{\text{neg}}$ are positive and negative samples from the predictions, respectively, as described in YOLOv3, $\widehat{b}_\theta$ is calculated in equation (2) and t_x, t_y, t_w, t_h are calculated from the ground truth as follows:

$$\begin{aligned} t_x &= \frac{b_x}{s_k} - \left\lfloor \frac{b_x}{s_k} \right\rfloor, & t_w &= \ln\left(\frac{b_w}{w_{k,n}^{\text{anchor}}}\right) \\ t_y &= \frac{b_y}{s_k} - \left\lfloor \frac{b_y}{s_k} \right\rfloor, & t_h &= \ln\left(\frac{b_h}{h_{k,n}^{\text{anchor}}}\right) \end{aligned} \tag{4}$$

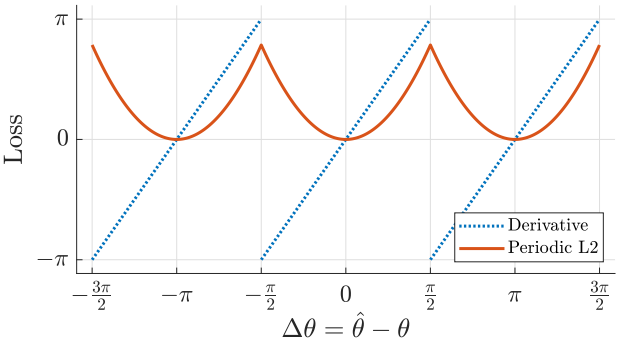


Figure 3: Periodic loss function with $L2$ norm as regressor and its derivative

Note, that we do not use the category-classification loss since we use only one class (person) in our problem.

Traditionally, regression functions based on $L1$ or $L2$ distance are used for angle prediction [16, 4, 31]. However, these metrics do not consider the periodicity of the angle and might result in misleading cost values due to symmetry in the parameterization of rotated bounding boxes. We solve these issues by using a periodic loss function and changing the parameterization, respectively.

3.2.1 Periodic Loss for Angle Prediction

Since a bounding box remains identical after rotation by π, the angle loss function must satisfy $\ell_{\text{angle}}(\widehat{\theta}, \theta) = \ell_{\text{angle}}(\widehat{\theta} + \pi, \theta)$, i.e., must be a π-periodic function with respect to $\widehat{\theta}$.

We propose a new, periodic angle loss function:

$$\ell_{\text{angle}}(\hat{\theta}, \theta) = f(\text{mod}(\hat{\theta} - \theta - \frac{\pi}{2}, \pi) - \frac{\pi}{2}) \tag{5}$$

where $\text{mod}(\cdot)$ denotes the modulo operation and f is any symmetric regression function such as $L1$ or $L2$ norm. Since $\frac{\partial}{\partial x}\text{mod}(x, \cdot) = 1$, the derivative of this loss function with respect to $\hat{\theta}$ can be calculated as follows,

$$\ell'_{\text{angle}}(\hat{\theta}, \theta) = f'(\text{mod}(\hat{\theta} - \theta - \frac{\pi}{2}, \pi) - \frac{\pi}{2}) \tag{6}$$

except for angles such that $\hat{\theta} - \theta = (k\pi + \pi/2)$ for integer k, where ℓ_{angle} is non-differentiable. However, we can ignore these angles during backpropagation as is commonly done for other non-smooth functions, such as $L1$ distance. Fig. 3 shows an example plot of $\ell_{\text{angle}}(\hat{\theta}, \theta)$ with $L2$ distance as well as its derivative with respect to $\Delta\theta = \hat{\theta} - \theta$.

3.2.2 Parameterization of Rotated Bounding Boxes

In most of the previous work on rotated bounding-box (RBB) detection, $[-\frac{\pi}{2}, 0]$ range is used for angle representation. This ensures that all RBBs can be uniquely expressed

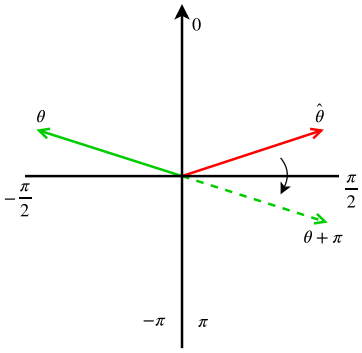


Figure 4: Illustration of the necessity to expand the predicted-angle value range. Gradient descent applied to the predicted angle $\hat{\theta}$ (red arrow) may rotate it clockwise and away from the ground truth angle θ (green arrow). Since a bounding box at angle $\theta + \pi$ is the same as the one at θ, we need to extend the angle range to include $\theta + \pi$ (dashed green arrow) otherwise $\hat{\theta}$, pushed by the gradient, will stop at $\pi/2$.

as $(b_x, b_y, b_w, b_h, b_\theta)$ where $b_\theta \in [-\frac{\pi}{2}, 0]$. However, as discussed in Section 2 and also in [20], this approach might lead to a large cost even when the prediction is close to the ground truth due to the symmetry of the representation, i.e., $(b_x, b_y, b_w, b_h, b_\theta) = (b_x, b_y, b_h, b_w, b_\theta - \pi/2)$. We address this by enforcing the following rule in our ground-truth annotations: $b_w < b_h$ and extending the ground-truth angle range to $[-\frac{\pi}{2}, \frac{\pi}{2})$ to be able represent all possible RBBs. For bounding boxes that are exact squares, a rare situation, we simply decrease a random side by 1 pixel. Under this rule, each bounding box will correspond to a unique 5-D vector representation.

Given the fact that the ground-truth angle θ is defined in $[-\frac{\pi}{2}, \frac{\pi}{2})$ range, it seems logical to force the predicted angle $\hat{\theta}$ to be in the same range by assigning $(\alpha, \beta) = (\pi, \pi/2)$ in equation (2). However, this creates a problem for gradient descent when $\pi/2 < \hat{\theta} - \theta < \pi$ since the derivative of angle loss (6) will be negative (Fig. 3). In this case, gradient descent will tend to increase $\hat{\theta}$ which will move it further away from the actual angle θ. Clearly, the network should learn to estimate the angle as $\theta + \pi$ instead of θ (Fig. 4). To allow this kind of behavior, we extend the range of allowed angle predictions to $[-\pi, \pi)$ by assigning $(\alpha, \beta) = (2\pi, \pi)$.

Note that our new RBB parameterization will not have the symmetry problem explained above if the network eventually learns to predict the parametrization rule, $\widehat{b}_w \leq \widehat{b}_h$, which is very likely considering the fact that all ground-truth RBBs satisfy $b_w \leq b_h$. Indeed, based on our experiments in Section 4.4.3 we show that nearly all RBBs predicted by RAPiD satisfy $\widehat{b}_w \leq \widehat{b}_h$.

In summary, by 1) defining $[-\frac{\pi}{2}, \frac{\pi}{2})$ as the ground truth angle range and forcing ground truth $b_w < b_h$, 2) using our proposed periodic angle loss function, and 3) setting predicted angle range to be $(-\pi, \pi)$, our network can learn to predict arbitrarily-oriented RBBs without problems experienced by previous RBB methods. Based on the experimental results in Section 4.4.2, we choose periodic L1 to be our angle loss function ℓ_{angle}.

3.3. Inference

During inference, an image $I \in \mathbb{R}^{3\times h\times w}$ is fed into the network, and three groups of bounding boxes (from three feature resolutions) are obtained. A confidence threshold is applied to select the best bounding box predictions. After that, non-maximum suppression (NMS) is applied to remove redundant detections of the same person.

4. Experimental Results

4.1. Dataset

Although there are several existing datasets for people detection from overhead, fisheye images, either they are not annotated with rotated bounding boxes [17], or the number of frames and people are limited [12]. Therefore, we collected and labeled a new dataset named Challenging Events for Person Detection from Overhead Fisheye images (CEPDOF), and made it publicly available[4]. We also manually annotated a subset of the MW dataset with rotated bounding-box labels, that we refer to as MW-R. We use MW-R, HABBOF, and CEPDOF to evaluate our method and compare it to previous state-of-the-art methods. Table 1 shows various statistics of these three datasets. Clearly, the new CEPDOF dataset contains many more frames and human objects, and also includes challenging scenarios such as crowded room, various body poses, and low-light scenarios, which do not exist in the other two datasets. Furthermore, CEPDOF is annotated spatio-temporally, i.e., bounding boxes of the same person carry the same ID in consecutive frames, and thus can be also used for additional vision tasks using overhead, fisheye images, such as video-object tracking and human re-identification.

4.2. Performance Metrics

Following the MS COCO challenge [14], we adopt Average Precision (AP), i.e., the area under the Precision-Recall curve, as one of our evaluation metrics. However, we only consider AP at IoU = 0.5 (AP_{50}) since even a perfect people-detection algorithm could have a relatively low IoU due to the non-uniqeness of ground truth: for the same person there could be multiple equally good bounding boxes at different angles, but only one of them will be selected by a human annotator to be labeled as the ground truth. In addition to AP, we also adopt F-measure at a fixed confidence

[4] vip.bu.edu/cepdof

Table 1: Statistics of our new CEPDOF dataset in comparison with existing overhead fisheye image datasets. Since all fisheye images have a field of view with 1:1 aspect ratio, we only list one dimension (i.e., "$1,056$ to $1,488$" means frame resolution for different videos is between $1,056 \times 1,056$ and $1,488 \times 1,488$). Note that the MW-R dataset in this table is a subset of the original MW dataset that we annotated with bounding-box rotation angles.

Dataset	# of videos	Avg. # of people	Max # of people	# of frames	Resolution	FPS
MW-R	19	2.6	6	8,752	1,056 to 1,488	15
HABBOF	4	3.5	5	5,837	2,048	30
CEPDOF	8	6.8	13	25,504	1,080 to 2,048	1-10

Scenarios	CEPDOF details					
Common activities	2	6.0	11	2,101	2,048	1
Walking activities	1	6.0	9	7,202	1,080	10
Crowded scene	1	10.8	13	3,000	2,048	10
Edge cases	1	5.5	8	4,201	2,048	10
Low-light scenarios	3	6.8	8	9,000	1,080	10

Table 2: Performance comparison of RAPiD and previous state-of-the-art methods. P, R, and F denote Precision, Recall, and F-measure, respectively. All metrics are averaged over all the videos in each dataset. Therefore, the F-measure in the table is not equal to the harmonic mean of Precision and Recall results in the table. The inference speed (FPS) is estimated from a single run on the *Edge cases* video in CEPDOF at confidence threshold $\widehat{b}_{\text{conf}} = 0.3$, using Nvidia GTX 1650 GPU.

		MW-R				HABBOF				CEPDOF			
	FPS	AP_{50}	P	R	F	AP_{50}	P	R	F	AP_{50}	P	R	F
Tamura *et al.* [27] (608)	6.8	78.2	0.863	0.759	0.807	87.3	0.970	0.827	0.892	61.0	0.884	0.526	0.634
Li *et al.* AA [12] (1,024)	0.3	88.4	0.939	0.819	0.874	87.7	0.922	0.867	0.892	73.9	0.896	0.638	0.683
Li *et al.* AB [12] (1,024)	0.2	95.6	0.895	0.902	0.898	93.7	0.881	0.935	0.907	76.9	0.884	0.694	0.743
RAPiD (608)	**7.0**	96.6	**0.951**	0.931	**0.941**	97.3	**0.984**	0.935	0.958	82.4	**0.921**	0.719	0.793
RAPiD (1,024)	3.7	**96.7**	0.919	**0.951**	0.935	**98.1**	0.975	**0.963**	**0.969**	**85.8**	0.902	**0.795**	**0.836**

threshold $\widehat{b}_{\text{conf}} = 0.3$ as another performance metric. Note that the F-measure for a given value of $\widehat{b}_{\text{conf}}$ corresponds to a particular point on the Precision-Recall curve.

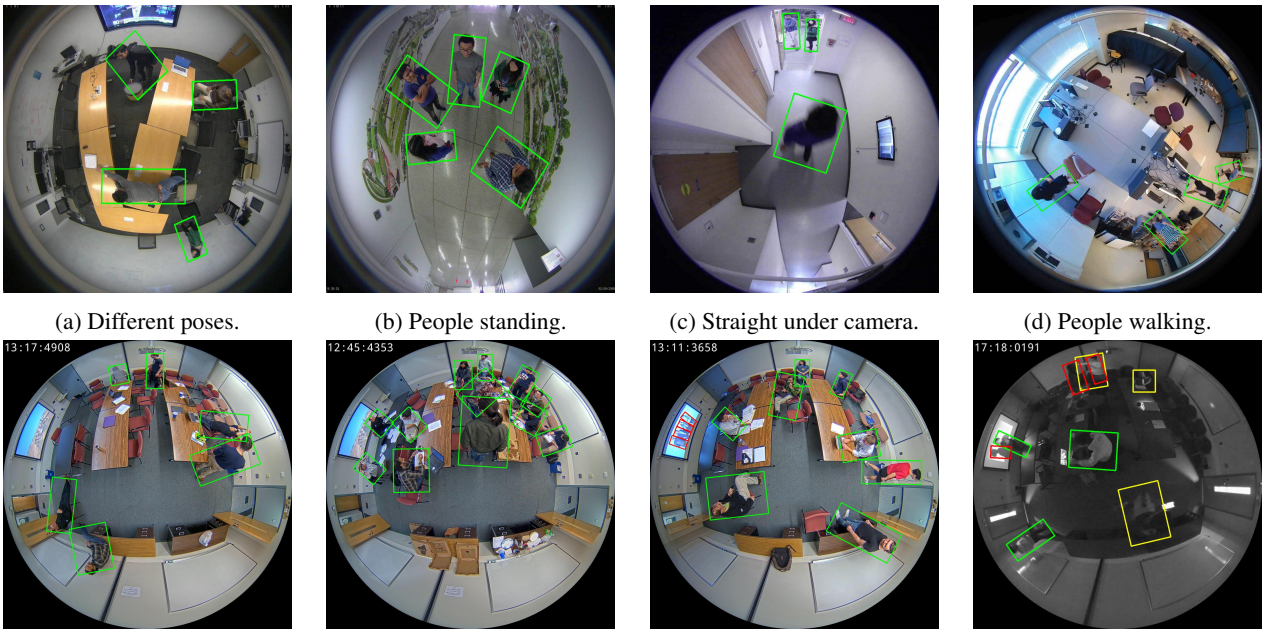
4.3. Main Results

Implementation details: Unless otherwise specified, we first train our network on the MS COCO 2017 [14] training images for 100,000 iterations and fine-tune the network on single or multiple datasets from Table 1 for 6,000 iterations (one iteration contains 128 images). On COCO images, the network weights are updated by Stochastic Gradient Descent (SGD) with the following parameters: step size 0.001, momentum 0.9, and weight decay 0.0005. For datasets in Table 1, we use standard SGD with a step size of 0.0001. Rotation, flipping, resizing, and color augmentation are used in both training stages. All results have been computed based on a single run of training and inference.

Table 2 compares RAPiD with other competing algorithms. In order to evaluate AA and AB algorithms from Li *et al.* [12], we used the authors' publicly-available implementation.[5] Since the code of Tamura *et al.* [27] is not publicly available, we implemented their algorithm based on our best understanding. Since there is no predefined train-test split in these three datasets, we cross-validate RAPiD on these datasets, i.e., two datasets are used for training and the remaining one for testing, and this is repeated so that each dataset is used once as the test set. For example, RAPiD is trained on MW-R + HABBOF, and tested on CEPDOF, and similarly for other permutations. We use only one *Low-light* video (with infra-red illumination) during training, as other videos have extremely low contrast, but we use all of them in testing. Since neither Li *et al.* [12] nor Tamura *et al.* [27] are designed to be trained on rotated bounding boxes, we just trained them on COCO as described in their papers. Tamura *et al.* used a top-view standard-lens image dataset called DPI-T [9] for training in addition to COCO, however currently this dataset is not accessible. In the ablation study (Section 4.4.1), we show the effect of fine-tuning Tamura *et al.* with overhead, fisheye frames as well. We use 0.3 as the confidence threshold for all the methods to calculate Precision, Recall, and F-measure. All methods are tested without test-time augmentation.

Results in Table 2 show that RAPiD at 608×608 resolu-

[5] vip.bu.edu/projects/vsns/cossy/fisheye



(a) Different poses. (b) People standing. (c) Straight under camera. (d) People walking.

(e) Various angles. (f) Crowded scene + occlusions. (g) People on the screen. (h) Low-light scenario.

Figure 5: Qualitative results of RAPiD on videos from MW-R (a–c), HABBOF (d) and CEPDOF (e–h). Green boxes are true positives, red boxes are false positives, and yellow boxes are false negatives. Images (a–d) are for relatively easy cases, (e–f) are for challenging cases, and (g–h) are failure examples. As shown in (a–f), RAPiD works very well in most scenarios, including various poses, orientations, occupancy levels, and background scenes. However, it produces false positives in (g) on a projection screen (images of people who should not be counted) and in (h). It also misses people in low-light conditions, such as in (h).

tion achieves the best performance and the fastest execution speed among all the methods tested. Our method is tens of times faster than Li *et al.*'s method and slightly faster than the method of Tamura *et al.*. We note that RAPiD's performance is slightly better, in terms of AP, than that of Li *et al.*'s AB algorithm on the MW-R dataset in which most human objects appear in an upright pose (walking). This is encouraging since people walking or standing appear radially oriented in overhead, fisheye images, a scenario for which Tamura *et al.*'s and Li *et al.*'s algorithms have been designed. However, RAPiD outperforms the other algorithms by a large margin on both HABBOF, which is relatively easy, and CEPDOF, which includes challenging scenarios, such as various body poses and occlusions. We conclude that RAPiD works well in both simple and challenging cases while maintaining high computational efficiency. Furthermore, it achieves even better performance when the input image resolution is raised to $1,024 \times 1,024$ but at the cost of a doubled inference time. Fig. 5 shows sample results of RAPiD applied to the three datasets; the detections are nearly perfect in a range of scenarios, such as various body poses, orientations, and diverse background scenes. However, some scenarios, such as people's images on a projection screen (Fig. 5g), low light, and hard shadows, remain challenging.

4.4. Design Evaluation

We conducted several experiments to analyze the effects of the novel elements we introduced in RAPiD. Specifically, we conducted an ablation study and compared different angle loss functions. Due to the limited amount of GPU resources we have, we did not run a full cross-validation for these experiments. Instead, we trained all of these algorithms on COCO and then fine-tuned them on MW-R using the same optimization parameters as reported in Section 4.3. Then, we tested each algorithm on every video in the HABBOF and CEPDOF datasets at $1,024 \times 1,024$ resolution. The resulting AP was averaged over all videos.

4.4.1 Ablation Experiments

In this section, we present various ablation experiments to analyze how each part of RAPiD individually contributes to the overall performance. As the baseline, we use Tamura *et al.* [27] with NMS and analyze the differences between this baseline and RAPiD one-by-one. Tamura *et al.* use standard YOLO [23] trained on 80-classes of COCO with rotation-invariant training [27] in which the object's angle

Table 3: Ablation study of RAPiD. Fine-tuning is applied using the MW-R dataset.

No. of classes	Angle prediction	Fine-tuning	AP_{50}	FPS
80	Rotation-invariant		81.4	3.7
1	Rotation-invariant		81.2	3.8
1	Rotation-invariant	✓	85.9	3.8
1	Rotation-aware	✓	**88.9**	3.7

Table 4: Comparison of RAPiD's performance for different angle ranges and loss functions.

Prediction range	Angle loss	AP_{50}
$(-\infty, \infty)$	L1	86.0
$(-\pi, \pi)$	L1	87.0
$(-\pi, \pi)$	Periodic L1	**88.9**
$(-\infty, \infty)$	L2	86.1
$(-\pi, \pi)$	L2	86.1
$(-\pi, \pi)$	Periodic L2	88.1

is uniquely determined by its location. The first row of Table 3 shows the result of this baseline algorithm. Note that, the baseline algorithm is not trained or fine-tuned on overhead, fisheye frames.

Multi-class vs. single-class: In RAPiD, we remove the category classification part of YOLO since we are dealing with a single object category, namely, person (see Section 3.2). As can be seen from the second row of Table 3, this results in a slight performance drop, which is to be expected since training on 80 classes of objects can benefit from multi-task learning. However, removing the category-classification branch reduces the number of parameters by 0.5M and slightly increases the inference speed (FPS in Table 2 and Table 3).

Fine-tuning with overhead, fisheye images: To analyze this effect, we fine-tuned the single-class algorithm trained on COCO with images from MW-R. As shown in the third row of Table 3, this results in a significant performance increase. Recall that the test set used in Table 3 does not include any frames from the MW-R dataset.

Rotation-aware people detection: As discussed in Section 3.2, we introduced a novel loss function to make RAPiD *rotation-aware*. Instead of setting the object's angle to be along the FOV radius, we add a parameter, $\widehat{b}_\theta$, to each predicted bounding box and train the network using periodic L1 loss. As shown in the last row of Table 3, the angle prediction further improves the performance of RAPiD.

4.4.2 Comparison of Different Angle Loss Functions

To analyze the impact of the loss functions on angle prediction, we ablate the angle value range and angle loss in RAPiD while keeping the other parts unchanged. We compare our proposed periodic loss with two baselines: standard unbounded regression loss and bounded regression loss. We perform the same experiment for both L1 and L2 loss. As can be seen in Table 4, the periodic L1 loss achieves the best performance, and both the periodic L1 and periodic L2 losses outperform their non-periodic counterparts.

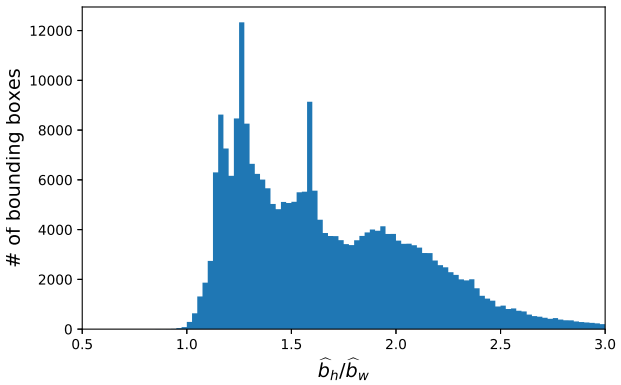


Figure 6: Histogram of the height-width ratio of the predicted bounding boxes.

4.4.3 Analysis of the Prediction Aspect Ratio

As discussed in Section 3.2.2, we relax the angle range to be inside $[-\pi/2, \pi/2)$ and force $b_w < b_h$ in ground-truth annotations so that every bounding box corresponds to a unique representation. In the same section, in order to handle the bounding-box symmetry problem we assumed that the network can learn to predict bounding boxes such that $\widehat{b}_w < \widehat{b}_h$. To demonstrate that this is indeed the case, we analyze the output of our network on both HABBOF and CEPDOF datasets. Fig. 6 shows the histogram of $\widehat{b}_h/\widehat{b}_w$. We observe that nearly all predicted bounding boxes satisfy $\widehat{b}_w < \widehat{b}_h$ (i.e., $\widehat{b}_h/\widehat{b}_w > 1$), which validates our assumption.

5. Conclusions

In this paper, we proposed RAPiD, a novel people detection algorithm for overhead, fisheye images. Our algorithm extends object-detection algorithms which use axis-aligned bounding boxes, such as YOLO, to the case of person detection using human-aligned bounding boxes. We show that our proposed periodic loss function outperforms traditional regression loss functions in angle prediction. With rotation-aware bounding box prediction, RAPiD outperforms previous state-of-the-art methods by a large margin without introducing additional computational complexity. We also introduced a new dataset, that consists of 25K frames and 173K people annotations. We believe both our method and dataset will be beneficial for various real-world applications and research using overhead, fisheye images and videos.

References

[1] Antonio Brunetti, Domenico Buongiorno, Gianpaolo Francesco Trotta, and Vitoantonio Bevilacqua. Computer vision and deep learning techniques for pedestrian detection and tracking: A survey. *Neurocomputing*, 300:17–33, 2018. 1

[2] An-Ti Chiang and Yao Wang. Human detection in fisheye images using hog-based detectors over rotated windows. *Proc. IEEE Intern. Conf. on Multimedia and Expo Workshops*, pages 1–6, 2014. 1, 2

[3] Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. In *Proc. IEEE Conf. Computer Vision Pattern Recognition*, volume 1, pages 886–893. IEEE, 2005. 2

[4] Jian Ding, Nan Xue, Yang Long, Gui-Song Xia, and Qikai Lu. Learning roi transformer for oriented object detection in aerial images. In *Proc. IEEE Conf. Computer Vision Pattern Recognition*, pages 2849–2858, 2019. 2, 4

[5] Piotr Dollár, Ron Appel, Serge Belongie, and Pietro Perona. Fast feature pyramids for object detection. *IEEE Trans. Pattern Anal. Machine Intell.*, 36(8):1532–1545, 2014. 2

[6] Markus Enzweiler and Dariu M Gavrila. Monocular pedestrian detection: Survey and experiments. *IEEE Trans. Pattern Anal. Machine Intell.*, 31(12):2179–2195, 2008. 1

[7] Cheng-Yang Fu, Wei Liu, Ananth Ranga, Ambrish Tyagi, and Alexander C. Berg. DSSD : Deconvolutional Single Shot Detector. *arXiv e-prints*, page arXiv:1701.06659, Jan 2017. 2

[8] Ross Girshick. Fast r-cnn. In *Proc. IEEE Int. Conf. Computer Vision*, December 2015. 2

[9] Albert Haque, Alexandre Alahi, and Li Fei-Fei. Recurrent attention models for depth-based person identification. In *Proc. IEEE Conf. Computer Vision Pattern Recognition*, pages 1229–1238, 2016. 6

[10] Kaiming He, Georgia Gkioxari, Piotr Dollar, and Ross Girshick. Mask r-cnn. In *Proc. IEEE Int. Conf. Computer Vision*, Oct 2017. 2

[11] O. Krams and N. Kiryati. People detection in top-view fisheye imaging. In *Proc. IEEE Int. Conf. Advanced Video and Signal-Based Surveillance*, pages 1–6, Aug 2017. 1, 2

[12] S. Li, M. O. Tezcan, P. Ishwar, and J. Konrad. Supervised people counting using an overhead fisheye camera. In *Proc. IEEE Int. Conf. Advanced Video and Signal-Based Surveillance*, pages 1–8, Sep. 2019. 1, 2, 3, 5, 6

[13] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *Proc. IEEE Conf. Computer Vision Pattern Recognition*, pages 2117–2125, 2017. 3

[14] Tsung-Yi Lin, Michael Maire, Serge Belongie, Lubomir Bourdev, Ross Girshick, James Hays, Pietro Perona, Deva Ramanan, C. Lawrence Zitnick, and Piotr Dollár. Microsoft COCO: Common Objects in Context. *arXiv e-prints*, page arXiv:1405.0312, May 2014. 3, 5, 6

[15] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C. Berg. SSD: Single Shot MultiBox Detector. *arXiv e-prints*, page arXiv:1512.02325, Dec 2015. 1, 2

[16] Jianqi Ma, Weiyuan Shao, Hao Ye, Li Wang, Hong Wang, Yingbin Zheng, and Xiangyang Xue. Arbitrary-oriented scene text detection via rotation proposals. *IEEE Trans. Multimedia*, 20(11):3111–3122, 2018. 2, 4

[17] Nuo Ma. Mirror worlds challenge, 2018. 5

[18] Duc Thanh Nguyen, Wanqing Li, and Philip O Ogunbona. Human detection from images and videos: A survey. *Pattern Recognition*, 51:148–175, 2016. 1

[19] Ryusuke Nosaka, Hidenori Ujiie, and Takaharu Kurokawa. Orientation-aware regression for oriented bounding box estimation. In *Proc. IEEE Int. Conf. Advanced Video and Signal-Based Surveillance*, pages 1–6. IEEE, 2018. 2

[20] Wen Qian, Xue Yang, Silong Peng, Yue Guo, and Chijun Yan. Learning modulated loss for rotated object detection. *arXiv preprint arXiv:1911.08299*, 2019. 2, 5

[21] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *Proc. IEEE Conf. Computer Vision Pattern Recognition*, June 2016. 1, 2, 3

[22] Joseph Redmon and Ali Farhadi. Yolo9000: Better, faster, stronger. In *Proc. IEEE Conf. Computer Vision Pattern Recognition*, July 2017. 1, 2

[23] Joseph Redmon and Ali Farhadi. Yolov3: An incremental improvement. *arXiv*, 2018. 1, 2, 3, 4, 7

[24] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, editors, *Proc. Conf. Neural Inf. Proc. Systems*, pages 91–99. Curran Associates, Inc., 2015. 1, 2

[25] Mamoru Saito, Katsuhisa Kitaguchi, Gun Kimura, and Masafumi Hashimoto. People detection and tracking from fish-eye image based on probabilistic appearance model. In *SICE Annual Conference 2011*, pages 435–440. IEEE, 2011. 2

[26] Roman Seidel, André Apitzsch, and Gangolf Hirtz. Improved person detection on omnidirectional images with non-maxima suppression. *arXiv preprint arXiv:1805.08503*, 2018. 3

[27] Masato Tamura, Shota Horiguchi, and Tomokazu Murakami. Omnidirectional pedestrian detection by rotation invariant training. In *Proc. IEEE Winter Conf. on Appl. of Computer Vision*, pages 1989–1998. IEEE, 2019. 1, 6, 7

[28] Mingxing Tan, Ruoming Pang, and Quoc V. Le. EfficientDet: Scalable and Efficient Object Detection. *arXiv e-prints*, page arXiv:1911.09070, Nov 2019. 2

[29] Zhi Tian, Chunhua Shen, Hao Chen, and Tong He. Fcos: Fully convolutional one-stage object detection. In *Proc. IEEE Int. Conf. Computer Vision*, October 2019. 2

[30] T. Wang, C. Chang, and Y. Wu. Template-based people detection using a single downward-viewing fisheye camera. In *Intern. Symp. on Intell. Signal Process. and Comm. Systems*, pages 719–723, Nov 2017. 1, 2

[31] Xue Yang, Qingqing Liu, Junchi Yan, and Ang Li. R3det: Refined single-stage detector with feature refinement for rotating object. *arXiv preprint arXiv:1908.05612*, 2019. 2, 4

[32] Shifeng Zhang, Cheng Chi, Yongqiang Yao, Zhen Lei, and Stan Z. Li. Bridging the Gap Between Anchor-based and Anchor-free Detection via Adaptive Training Sample Selection. *arXiv e-prints*, page arXiv:1912.02424, Dec 2019. 2

[33] Qijie Zhao, Tao Sheng, Yongtao Wang, Zhi Tang, Ying Chen, Ling Cai, and Haibin Ling. M2Det: A Single-Shot Object Detector based on Multi-Level Feature Pyramid Network. *arXiv e-prints*, page arXiv:1811.04533, Nov 2018. 2

[34] Yanzhao Zhou, Qixiang Ye, Qiang Qiu, and Jianbin Jiao. Oriented response networks. In *Proc. IEEE Conf. Computer Vision Pattern Recognition*, pages 519–528, 2017. 1, 2