Cost-effective annotation of fisheye images for object detection

Kai Zhang[1,*], Ahmad Elalailyi[1], Luca Perfetti[2], Francesco Fassi[1]

[1] 3D Survey Group, ABC Department, Politecnico di Milano, Via Ponzio 31, 20133 Milano, Italy – (kai.zhang, ahmad.elalailyi, francesco.fassi)@polimi.it
[2] Department of Civil, Architectural, Environmental Engineering and Mathematics (DICATAM), Università degli Studi di Brescia, 25123 Brescia, Italy – luca.perfetti@unibs.it

Keywords: Fisheye, Barrel distortion, Deep Learning, Object Detection, Classification, YOLO, Artificial Intelligence.

Abstract:
Nowadays, fisheye image has become commonly used in the 3D reality capturing field. Although AI integration for image recognition has become mature with normal images, providing available annotated dataset and pre-trained models, its application for fisheye images is rarely seen. While the object detection models have generalization ability, dealing with barrel distortion requires specific data for fine-tuning. This paper seeks to acquire prior knowledge from normal image and transfer it to the application that deal with fisheye images. This research is devoted to test the annotation shape that could possibly improve the accuracy when representing the shape of objects. It also seeks a way to prove that the annotation can be converted to fisheye images, resulted into a pre-process, which will facilitate the data preparation process. The tests involve annotations with standard box and quadrilateral polygon, the later turned out to be preserving most of the wanted image content after the conversion. The test result shows that the model trained on converted annotations using quadrilateral polygons, compared to detection model trained on non-converted ones, improves the mean average precision by 8%.

1. Introduction

Fisheye cameras are widely utilized in many specialized applications, capitalizing on essential hardware advantages. i.e., fisheye lenses can be produced cheaper, smaller and brings wider field of view compared to non-fisheye alternatives. The wide field of view enables fisheye cameras to capture photos of enhanced spatial coverage, which makes them widely adopted for surveillance, monitoring, navigation. Many are tests and applications of photogrammetry using fisheye lenses. The utilization of fisheye lenses is especially beneficial in applications necessitating extensive spatial coverage, including terrain mapping, documenting of edifices or constructions, and topographical studies. These lenses enhance the efficiency of the acquisition process by capturing a substantial percentage of the scene in a single picture, hence minimizing the number of photographs needed to cover a region. These lenses are especially advantageous in aerial photogrammetry or UAV (Unmanned Aerial Vehicles) operations, as their wide field of view enables the recording of several locations in a single flight, hence optimizing cost and time. Moreover, in applications like metropolitan mapping or complicated infrastructure monitoring, the capacity to acquire a panoramic image without several exposures is essential. Fisheye lenses are also used in panoramic cameras, widely used to color the “lidar data” both static and mobile and nowadays deeply tested to quick 3D modelling of interiors, narrow places and urban environments (Barazzetti et al., 2018; Javadi et al., 2024; Previtali et al., 2024; Zhang et al., 2024).

On the other side, most monitoring applications or vision automatic applications that deal with big volume of data are empowered by AI. However, though AI processing on “normal" rectilinear images have shown promising results (famous examples are YOLO (Redmon et al., 2016), DETR (Carion et al., 2020) and SAM (Kirillov et al., 2023) etc), such processing on highly distorted images is rarely discussed, as it is the case of object detection algorithms. Learning-based object detection models trained on rectilinear images have generalization ability on distorted ones but limited. For this reason, specific training sets made of distorted images must be prepared to fine-tune the model and improve its performance. Considering that, nowadays, manual annotation can be a significantly time-demanding step in the whole application process.

Transforming the manual annotation of an ad hoc created classical training set onto distorted photos would considerably expedite the creation of a new training set tailored for fisheye images or even skip completely the process in case of the use of already existing training sets on rectilinear images.

This work aims to present an effective method for object detection in fisheye photos, utilizing trained information from a rectilinear dataset. The methods for converting annotations from rectilinear to distorted images are examined to provide cost-efficient data preparation and object representation. We collected image data using fisheye camera and cell phone, and deliberately designed tests including testing the conversion of two annotation shapes (standard box and quadrilateral polygon) and validating the behaviour of object detection models trained on them.

2. Related Work

2.1 Deep learning for image processing

Image processing has been a key topic and was developed for a long time, it typically involves three tasks: classification, object detection and segmentation. The classification model makes predictions based on predefined categories, the object detection model maps objects and generates corresponding category predictions. Segmentation models make binary masks for each predefined category.

It started from the most famous LeNet model (LeCun et al., 1989) that is enabled to recognise handwritten digits. Afterwards, typical man-crafted networks with limited depth of layer were developed, like VGG (Simonyan and Zisserman, 2015), inception network (Szegedy et al., 2014) etc. Residual networks such as ResNet were introduced by He et al., 2015 which introduced the concept of residual connection, solved the

problem of gradient vanishing, allowing layer depth increase in the latter model.
Deep learning models, like Faster RCNN (Ren et al., 2016) and YOLO etc, were favoured to be used for detection. They stand for two typical approaches for 2D object mapping. The RCNN, as a two-stage approach, starts with region proposals and then determines if objects are contained in each proposal. The latter one, as a one-stage approach, directly asks the model to output box location and classes. A comparison of the primitive models suggests that the one-stage is less sensitive towards small objects and less accurate, while tested to be fast in referencing.
A famous segmentation model is as a continuation of Fast RCNN, the Mask RCNN (He et al., 2018). The pixel-wise prediction later gained much interest from the field. The SAM model came with expectation of being a foundational model that can be used as a off-the-shelf component for other big models.

2.2 Datasets in deep learning

The capability of the deep learning model is not merely decided by the algorithmic feature, but also greatly affected by the training set. The dataset curation has been discussed for a long time (Deng et al., 2009; Everingham et al., 2010; Lin et al., 2015), including topics like the scale of the dataset (referring to number of categories and instances), the semantic hierarchy of the classes, accuracy (reliability of the annotation), and diversity (appearance, positions, viewpoints and so on) etc.
Available fisheye datasets are comparably rare. VOC-360 and Wider-360 (Fu et al., 2019) provide fisheye dataset for multiple computer vision tasks including object detection and segmentation. Both datasets are obtained from the existing public datasets VOC 2012 and Wider Face. For the object detection tasks, they used the corner points and edge midpoints to map the normal box annotation on the fisheye coordinate system. FRIDA (Cokbas et al., 2022) provides overhead fisheye datasets for people counting tasks, annotated with rotated boxes. WoodScape (Yogamani et al., 2019) provides a multi-camera fisheye dataset for road driving scene. The 4-camera system (also equipped with LiDAR and GNSS sensors) provides an abundant resource for tasks including segmentation, depth estimation, comprising 100,000+ image samples and 40 categories. A continuous development can be seen in SynWoodScape (Sekkat et al., 2022). Fisheye8K (Gochoo et al., 2023) represents as a benchmark for road object detection tasks, providing 157K normal bounding boxes across 5 categories.

2.3 Fisheye image processing

The great distortion of fisheye images makes their use challenging. One typical solution is to process the undistorted images with the consequence that the resulting images are resampled, which brings extra difficulty in object detection, or leads to information reduction at the image border. Using segmentation model (Siam et al., 2017) can be a solution. However, as used in a different task, segmentation model is computationally much less efficient, compared to object detection model.
SphereNet (Coors et al., 2018) is a neural network designed to handle spherical data, such as panoramic images or 360-degree camera outputs. Unlike traditional CNNs that struggle with distortion when projecting spherical data onto a flat surface, SphereNet processes data directly on the spherical geometry, preserving spatial features effectively. It achieves this by:

1. **Spherical sampling**: Adapting convolutional filter sampling positions to the spherical surface.
2. **Distortion invariance**: Maintaining spatial accuracy without planar projection distortions.
3. **Spherical convolution**: Applying filters that align with spherical coordinates.

The limitation of the SphereNet method is related to the fact that not all the fisheye images (equidistant and stereographic) follow exactly the spherical projection models, thus require further development.
FisheyeDet (Li et al., 2020) is an end-to-end object detection network also designed for fisheye images. It use an adaptive representation method, “distortion shape matching”, that accounts for fisheye distortion characteristics and may involve transformations or models that normalize distortion, making object features more recognizable and the so called “no-prior representation method” that allows the model to automatically adapt to the characteristics of the images during training without the need for predefined projection models, enabling more flexible detection across different types of fisheye images. . The two methods combined allow the network adaptively extract distortion features without prior knowledge of the lenses and corresponding calibration, in addition, match the quadrilateral bounding boxes to the distorted contour of objects. This study addresses the lack of public fisheye datasets; thus, they created a dataset from Pascal VOC dataset.
FisheyeYOLO (Rashed et al., 2022) further explored the annotation shapes, including standard box, distorted box, ellipse, polygon and curved box. The proposed model and method greatly improved relative accuracy.

3. Methodology

The experiments were conducted in the university department's office (building 15, Leonardo Campus, Politecnico di Milano, Milan, Italy) facility with the main goal to evaluate the efficacy of object detection models from fisheye photos in identifying common office amenities.

3.1 Data acquisition

The images dataset was acquired using ATOM-ANT3D (Perfetti et al., 2024; Elalailyi et al., 2024).It is a fisheye multi-camera visual mobile mapping system that houses five Megapixels (2448 × 2048) global shutter cameras equipped with ultra-wide fisheye lenses with a 190° field of view (FOV) and 2.7 mm focal length. This always guarantees a quasi-360° field of view. The acquisition was performed by the operator acquiring 6008 grey scale images per camera. To ensure proper format consistency with used open-source libraries (e.g. mapping functions in OpenCv), the cameras were calibrated using a wide 2D plane calibration with a of size 84.1 x 118.9 cm checkerboard that accommodates for the wide FOV. The OpenCV calibration model is a relaxed Kannala-Brandt model specifically used for fisheye images where the camera intrinsics parameters are represented by the principal points (cx, cy) and the focal lengths (fx, fy) and radial distortion coefficients (k1, k2, k3, k4).
A minor phone images dataset is collected with iPhone 13 mini. Uses camera setting $f/2.4$ aperture and 120° field of view, focal length at 13mm. The camera application automatically applies lens correction to the images (L. Perfetti et al., 2024). The phone dataset comprises 470 images with colour. These phone photos are distorted and reshaped to match the size of the Ant3D camera fisheye images.

3.2 The goal and method

The raw fisheye images are undistorted into rectilinear distortion-free images (see Figure 3) using the calibrated distortion parameters of each camera. We therefore obtained two datasets.

The first is composed of the original images with fisheye projection, and the second of the undistorted images that have a calculated rectilinear projection.
The original Fisheye images are manually annotated using both normal boxes and multi-polygon segmentation. The object detection models, trained and evaluated on them, serves for the comparation with the object detection process on the converted image.
The undistorted images are manually annotated using two different object representation forms: normal boxes and 4-sided polygons. The annotation done on them images will be re-converted to fisheye version to make a comparation with the ground truth annotation. The objective is to verify whether the listed annotation method accurately maps the objects, ensuring they are fully contained while minimizing the inclusion of excessive irrelevant information. The annotation shapes in fact are described by the vertex points. They change position if applies transformation for the distortion. Hence the converted annotation could be less efficient in representing the object than before. The experiment examines the effectiveness of geometric shape for converted fisheye annotation.
Moreover, if the performance of the tested model, trained on the dataset with the least efficient annotation, will result to be comparable to that of the standard model trained on dataset F1 (annotated using the traditional method), it would validate the viability of the suggested technique.
In the end, we obtained 8 versions of annotation (as listed in Table 1). In the **F set** the annotations are directly made on fisheye images both using rectangular standard box (**F1**) and 4-sided polygon shape (**F2**). They are trained and used as control group; **R** is the set of undistorted images; fisheye images that were undistorted and used as rectilinear ones. The image belonging to the subsets of **R** are annotated both with standard box (**R1**) and the quadrilateral polygon (**R2**) and then re-converted to their fisheye versions (from **R1** with box annotation to **R3** with box and **R4** with polygon, and from **R2** with polygon to **R5** with polygon). The dataset **N** contains collected photos from mobile phone, annotated with quadrilateral polygon, these phone images as well as their annotations were converted to simulated fisheye camera dataset. (see Figure 1).

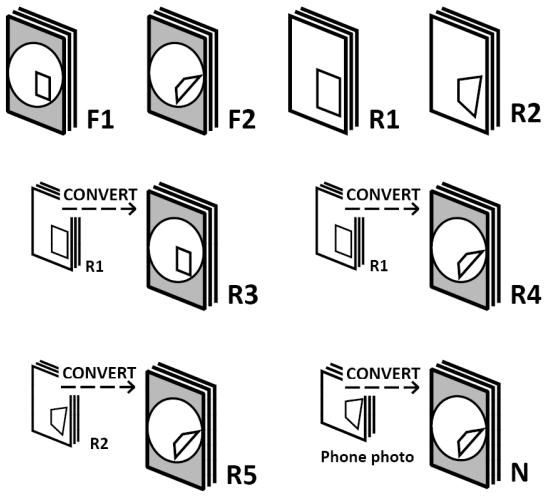

Figure 1. Scheme of elaborated datasets.

Annotation dataset	Ser.	Repres. shape	Convert to fisheye vers	Test set
Fisheye	**F1**	Std. box	-	F1
	F2	Qdl. poly	-	F2
Rectilinear	**R1**	Std. box	-	R1
	R2	Qdl. poly	-	R2, F2
	R3	Std. box	Std. box	R3, F1
	R4	Std. box	Qdl. poly	R4, F2
	R5	Qdl. poly	Qdl. poly	R5, F2
Phone Photo	**N**	Qdl. poly	Qdl. poly	N, F2

*with Std as standard, Qdl as Quadrilateral

Table 1. Datasets and annotation type

Figure 2. Images examples used in the tests: up, photo shoot with Ant3D and its undistorted version; down, image shoot with phone camera and its distorted version.

The experiment includes following steps:

1. Firstly, the image samples processing, include fisheye image data undistortion, phone image distortion using a Kannala-Brandt radial distortion model with $k_1=2$,$k_2=1$, $k_3=1$ to simulate the effect of the used fisheye photos (see Figure 2).
2. Then, images are annotated with standard box and 4-sided polygon. Annotations in R1, R2 and N datasets will be converted to fisheye version, examining the coherency, proving the feasibility of annotation conversion.
3. The detection models will be trained on all datasets, with datasets F1 and F2 used for ground truth comparison. The models will be trained and tested on the R datasets to evaluate performances, training time to convergence, and inference speeds. This step aims to assess the effectiveness of annotating rectilinear images and converting them to fisheye datasets, compared to training directly on the original fisheye images.

4. Last, the model will be trained upon dataset **N** with phone images, testing its behaviour on F2 dataset, as a final evaluation of the annotation conversion approach.

3.3 Dataset and category

The annotation categories were derived from the BIM model of the building resulting in 4 classes and corresponding 7 subclasses (see Table 2).

Class	Sub-Classes	Quantity
Fire Alarm Devices	Smoke Detector	86
	Alarm Button	25
Antifire Protection	Fire Extinguisher	39
Illumination	Celling Light	424
Electric Equipment	FM Power Point (Plug)	572
	Command Point (Switch)	274
Security Equipment	Security Camera	44

Table 2. Classes, sub-classes and instance quantities

3.4 Representation shape

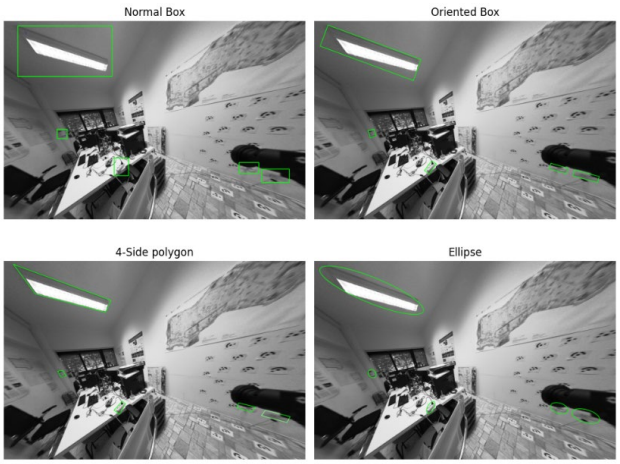

Figure 3. Different annotation shapes are mapped on the same undistorted image to understand which shape preserves most information after the conversion to fisheye

FisheyeYOLO and WoodScape dataset have provided analysis of different annotation shapes: including normal boxes (represented by 4 parameters: x, y, w, h), oriented Box (x, y, w, h, ɵ), Ellipse (x, y, w, h, ɵ), curved boxes (x, y, r_1, r_2, $ɵ_1$, $ɵ_2$), polygons (x_1, y_1, x_2, y_2, x_3, y_3, x_4, y_4 or r_1, α_1, $ɵ_1$, r_2, α_2, $ɵ_2$, …) as annotated on our example in Figure 3. Here the mean Intersection over Union (mIoU) is used to evaluate how accurately different annotation shapes capture object contours, considering instance segmentation masks as ground truth. In these studies, focused on the vehicles category, the standard box (4 parameters) achieves a peak performance of approximately 51.35% mIoU. Alternative shapes, such as curved boxes, oriented boxes, and ellipses, improve mIoU by only about 4.1%. A 4-sided polygon reaches 70.2%, while a 24-sided polygon achieves roughly 85%. Training the model on these shapes shows incremental improvements over the baseline YOLOv3 model, with curved boxes, oriented boxes, ellipses, and 24-sided polygons yielding performance gains of 1.8%, 2.0%, 3.8%, and 13.5%, respectively.
In our tests, the categories are shifted to university department's office facilities. The interior scenes and office categories have significant differences with the street scenario, where tests demand recognition at closer distance and of smaller objects. The variety of objects causes an essential change to the representation capacity of annotation shape. As an initial test for the conversion of annotation, normal box and 4-sided polygon are selected for image annotation, as they typically represent the difficulties in the data processing.

3.5 Dataset elaboration: annotation and conversion

The test utilizes the OpenCV to deal with the conversion of images and annotations. It involves the Kannala-Brandt model.
to generate the transformation maps based on the known intrinsic coefficients of the camera). The transformation maps (map_x and map_y) are two 2-dimensional matrixes that indicate a geometrical transformation of 2D images, pixel by pixel (see Figure 4), enabling swift and accurate conversion of images and annotations by defining how pixel coordinates in one image are mapped to new locations.
These maps are used both to un-distort fisheye images and to do the reverse process to re-apply the distortion to the image, as demonstrated in Figure 4. It is to notice that the undistortion process cause a loss of information along the image border and the reverse process which transforms the undistorted image back to its original view, at the contrary (row 1, line 3 in the Figure 5), highlights a loss in resolution in the image centre. This is important to be taken in consideration positioning and converting the annotation polygons.

Figure 4. Transformation maps plotted on the fisheye images. The mapped green points are subsampled pixel index, the undistorted image will be generated by aligning them in a regular orthogonal grid.

In fact, the conversion of annotation also uses the transformation maps. The representation of oriented boxes and 4-sided polygons follows the same conversion logic, which involves identifying and repositioning all four vertices. Because of the information reduction in the image centre, occasionally happens that the coordinates of the vertexes are not indexed in the maps, therefore required to be relocated to the closest neighbour in the maps.
Moreover, the conversion of the bounding box leads to a reduction of efficiency of the representation shapes when these annotations are located at the periphery. In the case from fisheye to rectilinear ones, vertexes located outside the vision boarder can be problematic. This is because the conversion cropped only the central part of the image, when vertexes of annotations are located outside the image frame, it becomes difficult to preserve

the original annotation. In the case of rectilinear to fisheye ones, on the other hand, rectangular shapes can be compressed and lowered the validity of annotation. Since this paper is aimed at leveraging the annotations from rectilinear images, only the deformation of annotation shape is relevant.

A noteworthy image preprocess involved in this paper is related to the annotations converted from rectilinear images to be used in fisheye object detection: the rectilinear images have no distortion and result as cropped respect to the original fisheye images. Consequently, their borders have different shapes and content.

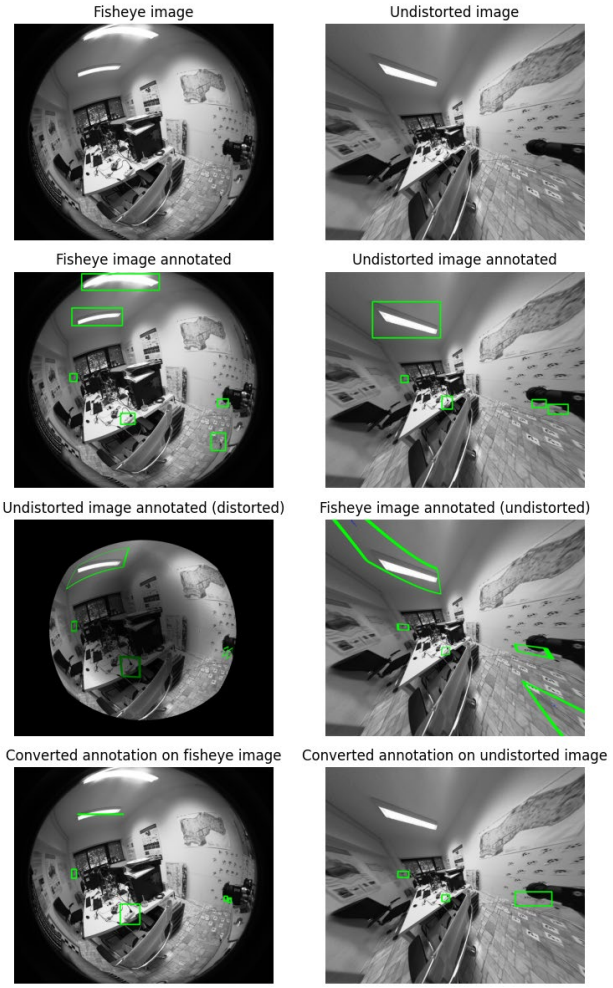

Figure 5. Images comparation of normal box annotation and conversion. The green boxes mark out original annotation. By finding the corresponding location of the up-left and down-right points from the original annotation boxes, the red ones (converted) are mapped.

Using the original fisheye images and the converted annotations to train the model, the statistics will be confused where extended part of the object emerges outside the annotation areas. Given that using transformation maps to double convert the images (undistort to rectilinear and distort them back to fisheye) will leads to unnecessary pixel information loss. In this test is utilized two additional transformation maps to mask the original fisheye images to match the boundary that a back-distorted undistorted image will have (see Figure 6).

Figure 6. Example of mask for the converted annotation. Noticeable is that without the mask the non-annotated ceiling light will cause confusion.

3.6 Model used

Considering inference time, accuracy, and model complexity, this study employs YOLOv5 for standard box detection and its customized variant, Yolo-ArbV2, for polygon detection. YOLO architectures are recognized for their superior speed and convergence time compared to the "two-step" RCNN approach, albeit with a trade-off in precision.

The model architecture consists of two primary components: the backbone and the detection head. The backbone, functioning as a feature extractor, can vary depending on the application but typically employs CSP-Darknet53, an optimized version of Darknet. This backbone includes four C3 blocks (CSP bottleneck blocks with three convolutional layers), eventually output through an SPPF (Spatial Pyramid Pooling – Fast) layer. The outputs from the first two C3 blocks are concatenated and fed into the detection head.

The detection head in YOLOv5 processes outputs from the last three C3 blocks. It uses a concatenation mechanism to produce bounding boxes (x, y, w, h), confidence scores, and class probabilities. This design enhances feature utilization across multiple scales, improving detection performance.

Yolo-ArbV2 (RhineAI-lab, 2021) is a customized version added extra output for the vertexes (x_1, y_1, x_2, y_2, x_3, y_3, x_4, y_4) in addition to boxes, confidence and probabilities of each class. This model calculates Euclidean distances of all vertexes for the loss without using the IoU for polygons.

To evaluate the model behaviour, 10% samples are randomly selected from the overall dataset for the evaluation, using precision, recall, mIoU.

These results will suggest how well the model is fitted to the training set and, for similar scenarios, how the performances are in terms of accuracy. The representativity of the training and evaluating sets is namely their similarity with reality and it has also to be considered in the examination phase.

4. Experiment

4.1 Standard box annotation in fisheye object detection

The initial test is conducted with the easiest but most popular annotation representation shape, the standard box. After the annotation, the YOLOm5 model were trained upon subsets **F1**, **R1** and **R3**. Then the model which was trained on undistorted images is tested on the evaluation part of F1 dataset (see Table 3).

The detection models reached convergence at around 20 epoch and achieved optimal results on the 10% sample evaluation set (with total amount of samples of 1500 images). It's expectable that the model for fisheye images with standard bounding box (**F1**) has relatively lower value of evaluation. It can be reasoned that objects at periphery are largely deformed, and the standard box has limited efficiency for mapping the object. Thus, the redundant neighboring information confuses the model.

Although the model trained on undistorted images (R1) produced satisfactory results on its evaluation set, its performance declined

when applied to the fisheye image evaluation set (R1-F1, see Figure 7, left).
The **R1** dataset consists of image samples which were undistorted from **F1**, which facilitates the prediction of objects that are not significantly deformed. Therefore, the objects located close to the optical center of the image naturally have higher possibility to be recognized. However, in the **F1** dataset, objects located far from the center are highly deformed, the image of objects is curved and compressed, which might not be easily learned by the detection model. In addition, at the periphery of the fisheye images, the illumination is lowered due to vignetting and optical falloff. Thus, the peripherical objects are less possible to be recognized by the model learned from undistorted images where most of the objects are better illuminated.
In the case of the model that is trained on **R3** (see Figure 7 ,right) although the training is based on image samples with barrel distortion, the annotation representations are largely deformed. The training result appears to be satisfying. During the training process, the model is fed with much less redundant image information. However, the deformed annotations boxes also exclude essential information, hence the performance upon full fisheye images is witnessed dropping, especially related to the mean precision with criteria of IoU.

Train-val set	Convert.	Prec.	Rec.	mAP_IoU	
				0.5	0.5:0.95
F1-F1	-	0.894	0.932	0.925	0.436
R1-R1	-	0.934	0.953	0.955	0.543
R1-F1	-	0.601	0.274	0.282	0.124
R3-R3	Box to box	0.859	0.946	0.967	0.665
R3-F1	Box to box	0.486	0.185	0.149	0.049

Table 3. Evaluation matrix of standard box test

Figure 7. Examples of R1-F1 and R3-F1 detection results

4.2 4-sided polygon annotation in fisheye object detection

The following tests are conducted with an efficient annotation representation shape, the quadrilateral polygon. After the annotation, the Yolo-ArbV2 model was trained upon subset **F2**, all other subsets of **R** and dataset **N**. Then the model which was trained on undistorted images is tested on the evaluation part of **F2** dataset (see Table 4).
The detection models reached convergence at around 40 epoch and achieved optimal results on the 10% sample evaluation set (with total amount of samples at 1600 images). The model for fisheye images with quadrilateral polygon (**F2**) has slightly higher value of evaluation than that for rectilinear images. The objects at periphery are deformed but the straight lines of polygon allow mapping the object following the radial direction in certain degree. Thus, it limited the neighboring information that confuses the model in rectilinear cases.
As expected, the performances of models trained on undistorted images on fisheye image evaluation set **F2** decrease. Among all the methods, the model trained on **R5** dataset presented the most satisfying results, outperformed the model that is trained on **R2**, the polygon annotated on undistorted images. Whilst dataset **N** contributed to the results almost comparable with that of **R2**.
From the results dataset **R2** contributes to (see Figure 9 column 2) most objects can be recognized well. In most cases, the polygon includes the object well, but the shape matches the object boundary in a limited degree. To locate correctly the vertexes of the polygon to the best fitting place turns out to be a challenge. The recognition still goes wrong at the peripheries because of the lack of corresponding training samples.
The model trained on dataset **R4** was evaluated on dataset **R2** (see Figure 9 column 3). In the prediction results, most objects can be recognized well. The polygon detections map out the objects, but not efficiently. Their shapes appear to be pressed in the direction of optical center, pushing outwards 2 other vertexes away from the object. This detection behavior can be reasoned from the fisheye conversion of box annotation.

Train-val set	Convert.	Prec.	Rec.	mAP_	
				0.5	0.5:0.95
F2-F2	-	0.930	0.889	0.948	0.558
R2-R2	-	0.836	0.928	0.928	0.629
R2-F2	-	0.673	0.472	0.545	0.242
R4-R4	Box to poly	0.890	0.936	0.952	0.561
R4-F2	Box to poly	0.369	0.146	0.196	0.068
R5-R5	Poly to poly	0.884	0.894	0.912	0.558
R5-F2	Poly to poly	0.693	0.569	0.629	0.319
N-N	Poly to poly	0.811	0.819	0.858	0.548
N-F2	Poly to poly	0.444	0.335	0.312	0.127

Table 4. Evaluation matrix of standard box test

The model trained on dataset **R5** obtains satisfying results on dataset **R2** (see Figure 9 column 4). Not only are the objects recognized well, but the polygons also match the object profiles properly even with the distortion effect. Although it shares common defects with the others, it still cannot overcome the recognition difficulties between the wires and the fire distinguisher. The recognition can go wrong at the round periphery, the coverage of the object is not optimal.
The test on phone dataset **N** (see Figure 9 column 5) provides a promising conclusion to the proposed approach, especially considering that it only contains 1/6 sample amount of the others. The polygon-to-polygon conversion approach largely avoids the distortion effects from the optical center of the image. The detector recognizes well the objects, while limited on locating the corresponding vertexes properly. Providing a comparable number of samples, this approach should be feasible and efficient in practical cases.

4.3 Discussion

The challenges of object detection on fisheye images primarily stem from distortion effects, which intensify as image content moves away from the optical center toward the periphery. This distortion complicates object representation, particularly when using annotation shapes with geometric limits compared to segmentation. Another non-neglectable fact is the vignetting and optical falloff. The cropping procedure for the rectilinear images

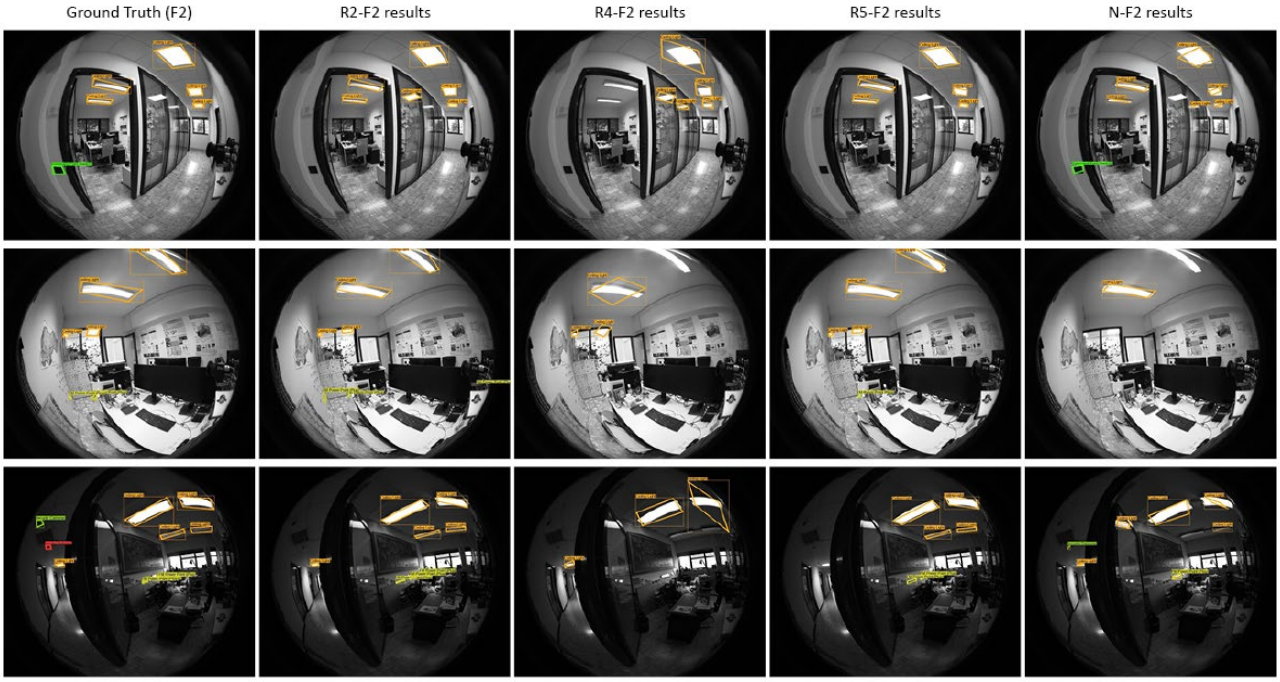

Figure 8. Example of detection results using quadrilateral polygon

and the lens correction canceled the affected image content, making it more challenging to leverage rectilinear images for the detection. Standard box annotations, while simple, introduce significant redundancies but fail to accurately represent objects in distorted regions. The conversion of standard boxes to neither box nor polygon improves the accuracy of detection model. Computationally efficient as it is, the training process of the model takes relatively short time. Specifically, the model requires 14.8 ms for inference and 1.7 ms for Non-Maximum Suppression (NMS) per image when using CUDA on a Quadro P4000.

Conversely, polygon-based annotations, which provide a more flexible shape representation, help mitigate distortion effects. This method is reliable on representing regular objects in rectilinear images. The conversion of the polygons is slightly affected by the distortion, maintaining satisfying representational efficiency. The computational cost for training the model comes at least 5 times compared to that of a standard box detector. It takes 51.8ms for inference, 1.5ms for NMS per image using the same device.

Evaluating the optimal annotation shape is inherently complex due to the trade-offs between geometric precision and computational cost. Annotation shapes with more parameters, such as 4-sided polygons (8 parameters), offer better object containment and improved performance in distortion-corrected models. This was confirmed in the results, which showed that polygon-based annotations outperform standard boxes, particularly in regions near the optical center, while maintaining robustness in highly distorted peripheral areas.

Ultimately, while standard box annotations remain popular due to their simplicity and low computational cost, 4-sided polygon annotations strike a balance between efficiency and accuracy. They provide superior shape representation without excessive computational overhead, making them a reliable choice for handling distortion in fisheye image object detection.

5. Conclusion

This paper discussed the annotation shape representation and its efficiency in accurately including objects, pursuing an approach to convert the knowledge from rectilinear domain to fisheye images domain. The challenge of fisheye object detection lies in controlling the annotation representativeness of peripheric image content in the training set preparation procedure, considering the exponentially increased distortion affect. The vignetting and optical falloff further complicate the detection.

Standard bounding boxes, while computationally simple, are inefficient in representing distorted objects, especially near the image periphery. Quadrilateral polygons, on the other hand, provide a better fit due to their adaptability, though they involve increased computational complexity.

For the standard box detection, the test shows that rectilinear training set can already produce satisfying results upon the fisheye test set. The conversion from standard boxes cannot contribute to better detection results. Regarding the polygon detection, however, the test results demonstrate that converting quadrilateral polygons improve the performance upon fisheye test set, by 8% of mean average precision. The phone photo tests provide promising results of fisheye object detection task, using the model trained with converted annotation on limited number of normal images. It proves the feasibility and efficiency of the proposed approach. However, the reliance on YOLOv5 and its modified version tailored for quadrilateral polygons limits the generalizability of these findings.

Future work should explore alternative annotation shapes such as oriented bounding boxes and ellipses, which, despite their more complex conversion processes, could offer a practical trade-off between operational simplicity and computational efficiency. Additionally, testing the approach on more advanced object detection models specifically designed for polygon annotations would provide deeper insights into the scalability and robustness of the proposed annotation strategy. This could lead to improved detection performance across a wider range of fisheye image applications.

Acknowledgements

This research was partially funded by "Boostech Valorization Program 2022" funded by the Italian "Piano Nazionale di Ripresa e Resillienza—NextGenerationEU with the goal of industrializing the Ant 3D prototype, which is already the subject of the patent proposal n° 102021000000812. (24 January 2023) Financial support from the program of the China Scholarships Council (grant number: 202208520007) is also acknowledged. The authors extend their gratitude to the BIMGroup of ABCLab (Department of ABC, Politecnico di Milano, Italy) research team, for generously providing reference BIM data and models, which were instrumental in this study.

Reference

Barazzetti, L., Previtali, M., Roncoroni, F., 2018. Can We Use Low-Cost 360 Degree Cameras to Create Accurate 3D Models? Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci. XLII–2, 69–75. doi.org/10.5194/isprs-archives-XLII-2-69-2018

Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., Zagoruyko, S., 2020. End-to-End Object Detection with Transformers. doi.org/10.48550/arXiv.2005.12872

Cokbas, M., Bolognino, J., Konrad, J., Ishwar, P., 2022. FRIDA: Fisheye Re-Identification Dataset with Annotations. doi.org/10.48550/arXiv.2210.01582

Coors, B., Condurache, A.P., Geiger, A., 2018. SphereNet: Learning Spherical Representations for Detection and Classification in Omnidirectional Images, in: Ferrari, V., Hebert, M., Sminchisescu, C., Weiss, Y. (Eds.), Computer Vision – ECCV 2018. Springer International Publishing, Cham, pp. 525–541. doi.org/10.1007/978-3-030-01240-3_32

Fu, J., Bajić, I.V., Vaughan, R.G., 2019. Datasets for face and object detection in fisheye images. Data Brief 27, 104752. doi.org/10.1016/j.dib.2019.104752

Gochoo, M., Otgonbold, M.-E., Ganbold, E., Hsieh, J.-W., Chang, M.-C., Chen, P.-Y., Dorj, B., Al Jassmi, H., Batnasan, G., Alnajjar, F., Abduljabbar, M., Lin, F.-P., 2023. FishEye8K: A Benchmark and Dataset for Fisheye Camera Object Detection. Presented at the Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 5305–5313.

He, K., Gkioxari, G., Dollár, P., Girshick, R., 2018. Mask R-CNN. doi.org/10.48550/arXiv.1703.06870

Javadi, P., García-Asenjo, L., Luján, R., Lerma, J.L., 2024. Assessment of Panorama Photogrammetry as a Tool for Long-Range Deformation Monitoring. Sensors 24, 3298. doi.org/10.3390/s24113298

Kirillov, A., Mintun, E., Ravi, N., Mao, H., Rolland, C., Gustafson, L., Xiao, T., Whitehead, S., Berg, A.C., Lo, W.-Y., Dollár, P., Girshick, R., 2023. Segment Anything. doi.org/10.48550/arXiv.2304.02643

LeCun, Y., Boser, B., Denker, J., Henderson, D., Howard, R., Hubbard, W., Jackel, L., 1989. Handwritten Digit Recognition with a Back-Propagation Network, in: NIPS.

Li, T., Tong, G., Tang, H., Li, B., Chen, B., 2020. FisheyeDet: A Self-Study and Contour-Based Object Detector in Fisheye Images. IEEE Access 8, 71739–71751. doi.org/10.1109/ACCESS.2020.2987868

Perfetti, Luca, Fassi, F., Vassena, G., 2024. Ant3D—A Fisheye Multi-Camera System to Survey Narrow Spaces. Sensors 24, 4177. doi.org/10.3390/s24134177

Perfetti, L., Fassi, F., Vassena, G.P.M., 2024. Built-In Lens Correction Profiles in Low-Cost Cameras: an Issue for Photogrammetric Applications? Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci. XLVIII-2-W4-2024, 349–356. doi.org/10.5194/isprs-archives-XLVIII-2-W4-2024-349-2024

Perfetti, L., Polari, C., Fassi, F., 2017. Fisheye Photogrammetry: Tests And Methodologies for The Survey of Narrow Spaces, in: The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences. Presented at the TC II & CIPA 3D Virtual Reconstruction and Visualization of Complex Architectures (Volume XLII-2/W3) - 13 March 2017, Nafplio, Greece, Copernicus GmbH, pp. 573–580. doi.org/10.5194/isprs-archives-XLII-2-W3-573-2017

Previtali, M., Barazzetti, L., Roncoroni, F., 2024. Gnss Assisted Photogrammetric Reconstruction from Combined 360°; Videos And Uav Images. Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci. XLVIII-2-W4-2024, 365–372. doi.org/10.5194/isprs-archives-XLVIII-2-W4-2024-365-2024

Rashed, H., Mohamed, E., Sistu, G., Kumar, V.R., Eising, C., El-Sallab, A., Yogamani, S., 2022. Generalized Object Detection on Fisheye Cameras for Autonomous Driving: Dataset, Representations and Baseline. doi.org/10.48550/arXiv.2012.02124

Redmon, J., Divvala, S., Girshick, R., Farhadi, A., 2016. You Only Look Once: Unified, Real-Time Object Detection. doi.org/10.48550/arXiv.1506.02640

Sekkat, A.R., Dupuis, Y., Kumar, V.R., Rashed, H., Yogamani, S., Vasseur, P., Honeine, P., 2022. SynWoodScape: Synthetic Surround-View Fisheye Camera Dataset for Autonomous Driving. IEEE Robot. Autom. Lett. 7, 8502–8509. doi.org/10.1109/LRA.2022.3188106

Siam, M., Elkerdawy, S., Jagersand, M., Yogamani, S., 2017. Deep semantic segmentation for automated driving: Taxonomy, roadmap and challenges, in: 2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC). Presented at the 2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC), pp. 1–8. doi.org/10.1109/ITSC.2017.8317714

Yogamani, S., Hughes, C., Horgan, J., Sistu, G., Varley, P., O'Dea, D., Uricar, M., Milz, S., Simon, M., Amende, K., Witt, C., Rashed, H., Chennupati, S., Nayak, S., Mansoor, S., Perrotton, X., Perez, P., 2019. WoodScape: A Multi-Task, Multi-Camera Fisheye Dataset for Autonomous Driving. Presented at the Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 9308–9318.

Zhang, X., Hu, Z., Hu, Q., Zhao, J., Ai, M., Zhao, P., Li, J., Zhou, X., Chen, Z., 2024. A 3D urban scene reconstruction enhancement approach based on adaptive viewpoint selection of panoramic videos. Photogramm. Rec. 39, 7–35. doi.org/10.1111/phor.12467