Distortion-Aware Room Layout Estimation from A Single Fisheye Image

Ming Meng[1] Likai Xiao[1] Yi Zhou[2] Zhaoxin Li[3] Zhong Zhou[1] *

[1]State Key Laboratory of Virtual Reality Technology and Systems, Beihang University, Beijing, China
[2]Beijing BigView Technology Co. Ltd, China
[3]Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China

ABSTRACT

Omnidirectional images of 180° or 360° field of view provide the entire visual content around the capture cameras, giving rise to more sophisticated scene understanding and reasoning and bringing broad application prospects for VR/AR/MR. As a result, researches on omnidirectional image layout estimation have sprung up in recent years. However, existing layout estimation methods designed for panorama images cannot perform well on fisheye images, mainly due to lack of public fisheye dataset as well as the significantly differences in the positions and degree of distortions caused by different projection models. To fill theses gaps, in this work we first reuse the released large-scale panorama datasets and reproduce them to fisheye images via projection conversion, thereby circumventing the challenge of obtaining high-quality fisheye datasets with ground truth layout annotations. Then, we propose a distortion-aware module according to the distortion of the orthographic projection (i.e., OrthConv) to perform effective features extraction from fisheye images. Additionally, we exploit bidirectional LSTM with two-dimensional step mode for horizontal and vertical prediction to capture the long-range geometric pattern of the object for the global coherent predictions even with occlusion and cluttered scenes. We extensively evaluate our deformable convolution for room layout estimation task. In comparison with state-of-the-art approaches, our approach produces considerable performance gains in real-world dataset as well as in synthetic dataset. This technology provides high-efficiency and low-cost technical implementations for VR house viewing and MR video surveillance. We present an MR-based building video surveillance scene equipped with nine fisheye lens can achieve an immersive hybrid display experience, which can be used for intelligent building management in the future.

Index Terms: Layout estimation; Deformable convolution; Fisheye image dataset; Orthographic projection

1 INTRODUCTION

Estimating a high-quality 3D room layout from a single image plays an increasingly important role in holistic scene understanding and would benefit numerous applications, e.g., entertainment, marketing productions, surveillance and robotics. Recently, omnidirectional capture is very appealing, which is supported by a wide variety of professional and consumer capture devices such as the panoramic camera or fisheye camera. It can provide a complete coverage of view, 180° or 360°, compared to the narrow field of view (FoV) of perspective images. Compared to the panoramic camera, fisheye camera has three advantages. First, fisheye camera has a relatively lower acquiring cost than the panoramic camera. Second, a single fisheye camera can cover a wide field of view close to 180°, maintaining the completeness of a large FoV. Moreover, a panoramic camera usually captures images with one rotating sensor or several sensors, and then stitches them into a panorama image, where strong chromatic aberration may harm the image quality. Third, compared to the overlapping objects on the left and right ends of the panorama image, the fisheye image can maintain the integrity of the object. Moreover, the information learned from feature extraction does not flow from one side of the image to the other side of the image. This makes the divided objects difficult to learn and affects the result of layout estimation. In this work, we focus on room layout estimation from a single RGB fisheye image.

The automatic layout estimation methods from indoor omnidirectional image using geometry and deep learning techniques have been developed rapidly. A more recent option to recover room layout is instead using the latter methods. Some prominent works are: LayoutNet [42] generates layout from corner and edge map trained by an FCN from panoramas. DuLa-Net [34] estimates Manhattan-world layouts using a perspective ceiling-view from E2P transformation. HorizonNet [29] represents the corners by a 1-dimensional encoding of the whole-room layout for a panoramic scene. AtlantaNet [19] is a novel data-driven method for estimating 3D room layout from a single RGB panorama without Manhattan world assumption constraints. However, the equirectangular projection of panorama image introduces significant distortions, in which existing convolutional neural networks (CNNs) architecture cannot be directly applied. To deal with this problem, several approaches [4, 28, 39] designed for deforming the shape of convolutional are introduced. CFL [6] as a novel end-to-end neural network recovers the 3D layout from a single 360° image. It first attempts to deform the kernel to compensate for the distortion of equirectangular projection.

The aforementioned methods designed for panorama images do not perform well on fisheye images, mainly due to two important and correlated issues: (1) the radically differences between the panorama and fisheye camera models; (2) the lack of suitable fisheye datasets for training and validation with precise annotations. To address the above issues,we propose a novel implementation of the deformable convolution for fisheye images in the orthographic projection (OrthConv), a special case of a fisheye projection. Then, we adopt the deformable convolution to learn offsets for improving the feature map accuracy. We further leverage Recurrent Neural Networks (RNNs) to capture long-range geometric pattern for layout estimation. Then, we construct a top-view fisheye image dataset, the real dataset (PanoContext-F and Stanford2D3D-F) and the synthetic dataset (Structured3D-F), by re-using existing panorama datasets and capturing the fisheye images from a surveillance system. It includes the transformation of three public datasets, which are PanoContext [36], Stanford2D3D [1] and Structure3D [37]. We use the collected dataset to train a more sophisticated model to automatically estimate room layout from a single fisheye image. We also show that the layout models pre-trained on the synthetic dataset and then fine-tuned on the real dataset outperform the models trained only on the real dataset.

Following this direction, we present a distortion-aware omnidirectional convolutional network for fisheye images. We first exploit distortion-aware-based CNNs feature extraction block to handle distortion introduced by orthographic projection. Specially, we adopt deformable convolution to learn offsets for improving the feature

*Corresponding Author: zz@buaa.edu.cn

map accuracy. Second, we further leverage RNNs to capture long-range geometric pattern for layout estimation. Third, we also show that the layout models pre-trained on the synthetic dataset and then fine-tuned on the real dataset outperform the models trained only on the real dataset. After that, we demonstrate the possible application prospect of our method in MR video surveillance through the visualization of synchronized multiple video streams.

Our contributions can be summarized as follows:

- We present a distortion-aware module to handle the distortion in fisheye images by adopting deformable convolution based on the orthographic projection of the fisheye images (i.e., OrthConv). It makes the network focus on informative areas to achieve fast convergence and promising performance.

- We introduce an encoder–decoder strategy for the layout estimation from a single fisheye image. The feature map is extracted by ResNet50 with OrthConv in encoder. In decoder, we exploit bidirectional LSTM with two-dimensional step mode for horizontal and vertical prediction to capture the long-range geometric pattern of the object for the global coherent predictions even with occlusion and cluttered scenes.

- We construct a top-view fisheye image dataset, the real dataset and the synthetic dataset, that contains 22,583 indoor fisheye color images paired with the corresponding corner ground truth.

2 Related Work

Layout estimation from a single image has been extensively studied for a long time. It provides a strong prior for visual tasks in holistic scene understanding, such as depth estimation, indoor object recognition, human pose estimation, and pedestrian detection. Early works for this area focused on conventional perspective images have progressed rapidly with geometric reasoning [5, 12] and data-driven [11]. However, these methods are limited by the restricted FoV of perspective images, which only record the small geometric context of the 3D scene. With the advent of the consumer-level omnidirectional cameras, such as panorama or fisheye cameras, it can capture all the visuals of the scene surrounding the viewpoint. Therefore, a noticeable series of works concentrating on the layout estimation from omnidirectional images is flourishing. These approaches can be divided into three categories: geometric-based methods, data-driven methods, and distortion-aware methods.

Geometric-based methods. The seminal approaches to room layout estimation from a single omnidirectional image were [9, 32, 33, 36]. Zhang et al. proposed PanoContext [36] to construct a whole-room 3D context model, which is the first work that extended the solution designed for perspective images to panoramas. It recovers the room layout with salient objects for panoramic scene understanding. Yang et al. [33] inferred the 3D room shape from panorama based on geometric context and line segments supplemented by superpixel facets, and embedded as vertices in constraint graph. Xu et al. [32] estimated the geometry of the room and the 3D pose of objects from a single panorama image for holistic indoor scene recovery. However, both methods relied on leveraging the existing frameworks for single perspective images transformed from the input panorama. Jia et al. [9] introduced the symmetrical rule describing geometric constraints in indoor fisheye images and performed layout retrieval only through a collection of line segments extracted from them. Perez-Yus et al. [21] used RGB-D and fisheye cameras to obtain a scaled 3D model with wide scene reconstruction. Since it needs to calibrate the two cameras together, which takes extra effort and reduces its feasibility.

Inspired by the recent significant performance of CNNs for learning image cues, researchers began to study hybrid-driven methods to improve performance. It combined geometry prior with depth or semantics to generate the optimal layout [7, 13, 35]. Fernandez et al. [7] presented a novel procedure for indoor 3D layout recovery from 360° panoramic images. They combined the accuracy achieved by geometric reasoning with an edge map extracted by deep learning techniques to improve performance. Yang et al. [35] inferred the room 3D structure from a single panorama, and the layout is recovered using geometric cues and the object mask is estimated by semantic cues. Immediately afterward, Li et al. [13] proposed a rapid and accurate approach to improve the results of layout reconstruction by combining geometric and semantic information. It could effectively solve the problem of object occlusions and clutters. For these methods, the quality of the extracted features determines the effectiveness of these methods.

Data-driven methods. Recent methods make use of deep networks to improve the result of layout estimation. The pioneering work of Zou et al. [42] designed an encoder-decoder architecture (e.g., LayoutNet also denoted as LayoutNetV1) to train an FCN from panoramas and vanishing lines, generating edge and corner maps for layout recovery. Moreover, they also extended the annotations of the Stanford2D3D dataset with a carefully labeled 3D shape layout, providing 571 RGB panoramas for layout estimation. Subsequently, the author introduced an improved version of LayoutNet called LayoutNetV2 [43]. Yang et al. [34] proposed a deep learning framework (e.g., DuLa-Net), which exploits features by combining the original panoramic view with the perspective ceiling-view to predict a more accurate Manhattan-world 3D room layouts. The further development of this research content could be found in [29], in which the whole-room layout of the panoramic scene was represented as 1D vectors encoding at each image column to reduce the parameters and time consumption. Additionally, it further leveraged RNNs in layout estimation task, to learn the long-range geometric pattern of room layout for improving the accuracy (e.g., HorizonNet). Pintore et al. [19] introduced a novel end-to-end neural network architecture (e.g., AtlantaNet) to predict 3D room layout from panoramic image without being restricted by Manhattan World.

Distortion-aware methods. All above methods directly apply standard convolution based conventional CNNs on panoramas, and the geometric structure is fixed in the modules it uses, so the ability of geometric transformation modeling is limited. Recently, a novel line of research focuses on model adaptation based on the shape transformation of the convolution operator to enhance CNN's modeling ability. Dai et al. [4] introduced a new module namely deformable convolution (DCNv1) which can further adjust the spatial sampling locations with additional offsets. The offsets can be learned in the target task without additional supervision. While the visualization result of DCNv1 shows that the coverage of the receptive field over an object is inexact, which interferes with feature extraction and reduces the algorithm performance. Therefore, Zhu et al. [39] presented a new version of deformable convolution (DCNv2), adding the weight of each sampling point based on DCNv1. The aforementioned affordable distortion-aware deformable convolution has achieved significant success in visual recognition tasks, among which the more prominent ones are semantic segmentation [14, 16, 20], object detection [2, 3, 27, 31, 40], depth estimation [30, 41], and layout estimation. In the follow-up work, Fernandez et al. [6] presented a novel end-to-end neural network that recovers the 3D layout by the corners prediction from a single 360° image (e.g., CFL). This network introduces a convolution defined in equirectangular projection (e.g., EquiConv) to compensate for the distortion of panorama. In the case of fisheye images, the serious distortions introduced by orthographic projection are variable at the same horizontal and vertical location, so it is hard to achieve promising performance while directly using the panoramic method. With our work, we aim to explore distortion-aware convolution kernel specialized for orthographic projection to aggregate more geometric information of fisheye image for layout estimation.

Although our method, like many of the recent ones, shares the encoder-decoder concept with HorizonNet [29] and CFL [6], we introduce important cues in the network according to the fisheye projection model. In particular, the distinct difference from HorizonNet is that it completely works on the 1D domain for equirectangular projection, while we work on the 2D domain for orthographic projection of the fisheye image. Moreover, in contrast to CFL, we use OrthConv on Res5 of ResNet50 for high-level feature extraction, while CFL applies EquiConvs directly on the entire ResNet50.

3 Approach

Our goal is to design a network architecture for layout estimation from fisheye images. Before introducing our network, we first formulate the transformation from panorama images to fisheye images (P2F) for building the fisheye dataset in Sec.3.1. Then in Sec.3.2, we introduce the proposed distortion-aware convolution operator for orthographic projection of the fisheye image. Subsequently, in Sec.3.3 we describe the architecture of our layout estimation network with distortion-aware convolution.

3.1 P2F Conversion

Currently, there is no publicly available dataset for room fisheye layout estimation, thus we build the PanoContext-F dataset, Stanford2D3D-F dataset, and Structured3D-F dataset by transforming images from the PanoContext [36], Stanford2D3D [1], and Structured3D [37] via the fisheye orthographic projection model. The size of the collected dataset is the same as the original dataset, except for Structured3D-F. We exclude images in the Structured3D dataset with (1) incorrect ground truth annotation like redundant corners and missing corners, (2) incorrect floor background, and (3) outdoor scenes. Finally, the Structured3D-F dataset contains 18111 training images, 1739 valid images and 1671 test images. Next, we explain the projection model of fisheye and the formulation of P2F conversion that transforms an equirectangular projection of panorama to the orthographic projection of fisheye.

The distortion in the image collected by the fisheye camera system, a typical non-linear system, increases from image's center to the edge. The design of fisheye lenses takes into account a series of factors, such as size, focal length, and geometry, etc. Thus, fisheye systems designed by different manufacturers have different fisheye projection models. According to different expressions of image radius r and incident angle φ, fisheye projection models are classified as [24]: equidistant projection, stereographic projection, equisolid-angle projection and orthographic projection. Specially, orthographic projection (abbreviated as Orth) is mostly used in real monitoring scenarios, and expressed as follows:

$$\mathrm{r} = \mathrm{f} \cdot \sin\varphi. \tag{1}$$

The basic idea of P2F conversion is to find the coordinate system mapping between fisheye and panorama, that is, to calculate the mapping position of each pixel on the fisheye image to the panorama image. For an RGB panorama image I_p with the resolution of $W * H$, W and H are the width and height of panorama image in pixels and $W:H$=2:1. Each pixel coordinate in the fisheye image I_f is converted to its corresponding pixel coordinate in the panoramic image by polar coordinate conversion. That is, the pixels in the fisheye image domain $\Omega_f : (\mathrm{m},\mathrm{n}) \in [0,\ \mathrm{W}] \times [0,\ \mathrm{W}]$ are mapped to the angular domain $\mathscr{A} : [\theta,\varphi] \in [-\pi,\pi] \times \left[-\frac{\pi}{2},\frac{\pi}{2}\right]$, and then are projected to the panoramic image domain $\Omega_p : (\mathrm{m},\mathrm{n}) \in [0,\ \mathrm{W}] \times [0,\mathrm{H}]$. For each pixel in Ω_f at position $\mathrm{p} = \mathrm{m}(\mathrm{p}),\mathrm{n}(\mathrm{p}))$, we derive the corresponding pixel position $\mathrm{p}' = (m(\mathrm{p}'), n(\mathrm{p}'))$ in the panorama by the longitude and latitude in the spherical coordinate system, as shown in Fig. 1. First, the focal length f and the distance r from the center of the pixel to p are defined as:

$$\mathrm{r} = \sqrt{\mathrm{m}(\mathrm{p})^2 + \mathrm{n}(\mathrm{p})^2}, \mathrm{f} = \mathrm{H}/\pi, \tag{2}$$

where m and n are operators that return the cartesian coordinates of the location p. To project it onto a unit sphere, we adopt the following relation:

$$\theta = \mathrm{arsin}(\mathrm{n}(\mathrm{p})/r), \varphi = \mathrm{arsin}(r/f), \tag{3}$$

then we apply the following formula:

$$\begin{aligned} m(\mathrm{p}') &= H - (\varphi * f)/6 \\ n(\mathrm{p}') &= (\varphi * f)/8, \end{aligned} \tag{4}$$

to calculate the corresponding position in equirectangular panorama space. Finally, the pixel value is interpolated from the panorama. For end-to-end training, the ground truth annotations of corners provided on the panoramic image are converted in the same way.

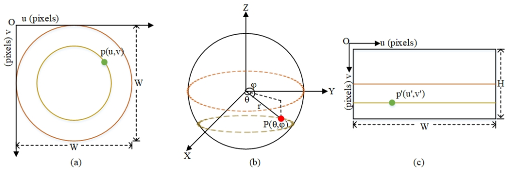


Figure 1: The geometric process of transforming a panoramic image into a fisheye image. (a) The orthographic projected image I_f with $w * w$ pixels. (b) is a sphere model that can be represented by orthographic projection or equirectangular projection. Given a 3D point $P(\theta,\varphi)$ (red point) on the sphere, its corresponding image coordinates can be found on I_f and I_p (green points). (c) The equirectangular projected image I_p with $w * h$ pixels.

3.2 Distortion-aware Convolution Module

In [4,6,39], they put forward the distortion-aware deformable convolution by learning additional offsets of the regular kernel to realize the free-form deformation of the kernel. Inspired by these works, we propose OrthConv, a deformable convolution according to the orthographic projection, which compensates for serious distortion of fisheye image introduced by the transformation from a non-Euclidean space to a Euclidean space. The central idea of OrthConv is to conduct convolution operation of CNNs in the spherical domain instead of the regular image domain from the preceding feature maps. It denotes the convolution kernel as a small patch tangent in the sphere surface where fisheye images are represented without distortions. Next, we explain how to calculate the distorted pixel positions from the original ones.

The standard convolution contains two steps: first is sampling a set of locations on the regular grid R (represented as receptive field size and scale) over the input feature map f_l at layer l. Here, the grid R is defined as a 3×3 convolution kernel with a dilation of 1 and $\mathrm{R} = \{(-1,-1),(-1,0)\ldots,(0,1),(1,1)\}$. Then the summation of a neighborhood of sampled values weighted by w is calculated. For each location $p_0 = (\mathrm{u}(p_0), \mathrm{v}(p_0))$, the operation result from regular grid structure is assigned to the corresponding element of output feature map f_{l+1} at layer $l+1$ as:

$$f_{l+1}(p_0) = \sum_{p_n \in R} w(p_n) \cdot f_l(p_0 + p_n), \tag{5}$$

where $p_0 + p_n$ represents the sampling location. p_n enumerates the relative location of pixels in the convolution region R. However,

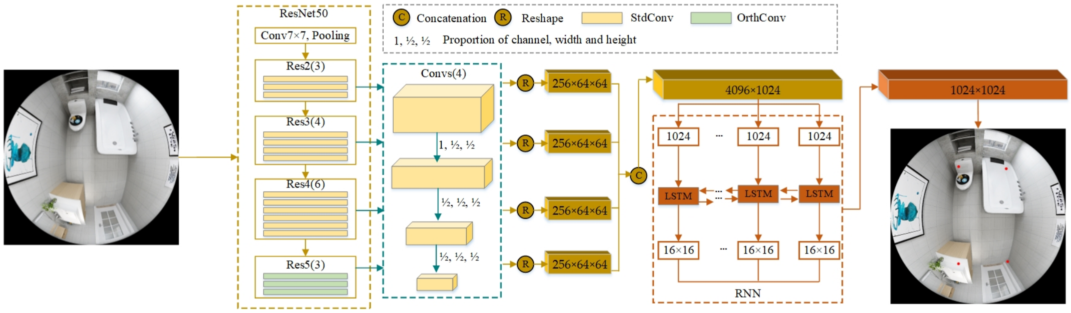


Figure 2: Overview of our network framework for room layout estimation from a single RGB fisheye image. The network takes a high-resolution fisheye as input and first processes it by the encoder backbone (e.g., ResNet50), with the OrthConv for solving the distortion of the orthographic projection (detailed in Sec.3.2). Then, the last four feature maps of the encoder are retained for simultaneous capture of low-level and high-level features through a sequence of convolutional layers (e.g., Convs). The concatenated sequential feature maps are fed to the decoder backbone (e.g., LSTM) to yield the final corners estimation (e.g., red dot).

this sampling strategy from the regular grid cannot be directly applied to fisheye images due to the varying distortions introduced by orthographic projection.

Following [4], we transform the regular grid according to the image distortion model. To preserve the consistency of context, we first extract the valid region of the input image. If p_0 is located in the invalid region, the offset is set as (0, 0). Based on Equation (1), we can sample a non-regular grid from fisheye image and perform distortion-aware convolution for layout estimation by Equation (2):

$$f_{l+1}(p_0) = \sum_{p_n \in R} w(p_n) \cdot f_l\left(p_0 + p_n + \Delta p'_n\right), \tag{6}$$

where $p_n + \Delta p'_n$ represents the irregular sampling process on the non-regular grid. $\Delta p'_n$ is the offset calculated according to the geometric relationship of orthographic projection. We first calculate the longitude and latitude of p_0 in the spherical coordinate system $p_{\theta-\varphi}(\varphi(p_0), \theta(p_0))$ as:

$$\varphi(p_0) = \arcsin\left(\frac{2r_0}{W}\right), \theta(p_0) = \arctan\left(\frac{\mathrm{v}(p_0)}{\mathrm{u}(p_0)}\right), \tag{7}$$

where $r_0 = \sqrt{\mathrm{u}(p_0)^2 + \mathrm{v}(p_0)^2}$. Then the rotation matrix $\mathrm{T}\left(R_y(\theta(p_0)), R_x((\pi/2) - \varphi(p_0))\right)$ is generated using the Euler-Rodrigues formula associated with counterclockwise rotation. Subsequently, any point p_n on the convolution kernel (resolution is $\mathrm{k_w}$*$\mathrm{k_h}$) is rotated by T as follows:

$$p'_n = T \times \frac{p_n}{|p_n|}, \tag{8}$$

where $p_n = [i, j, d]$i,j $\in [-k_w/2, k_h/2]$. d is the distance from R to the center of the unit sphere and defined as:

$$d = \frac{k_w}{2\tan\left(\frac{2\pi k_w}{W}\right)}. \tag{9}$$

Next, p'_n is converted to the longitude and latitude coordinates as follows:

$$\varphi(\mathrm{p}'_n) = \arctan\left(\frac{\sqrt{x(\mathrm{p}'_n)^2 + y(\mathrm{p}'_n)^2}}{z(\mathrm{p}'_n)}\right), \theta(\mathrm{p}'_n) = \arctan\left(\frac{y(\mathrm{p}'_n)}{x(\mathrm{p}'_n)}\right). \tag{10}$$

Finally, p'_n is back-projected to the fisheye image domain using the longitude and latitude coordinates:

$$\begin{aligned} u(\mathrm{p}'_n) &= \mathrm{W}/2 * \sin\varphi(\mathrm{p}'_n) * \cos\theta(\mathrm{p}'_n) \\ v(\mathrm{p}'_n) &= \mathrm{W}/2 * \sin\varphi(\mathrm{p}'_n) * \sin\theta(\mathrm{p}'_n), \end{aligned} \tag{11}$$

and the relative offset coordinates are calculated as:

$$\begin{aligned} u(\Delta p'_n) &= u(\mathrm{p}'_n) - u(p_n) \\ v(\Delta p'_n) &= v(\mathrm{p}'_n) - v(p_n). \end{aligned} \tag{12}$$

3.3 Network Architecture for Layout Estimation

An overview of our network architecture for layout estimation is depicted in Fig. 2. We use the orthographic projection (Orth) for the fisheye images. The input Orth fisheye is C×H×W (for channel, height, width). The network output is a binary segmentation mask of 1×H×W, describing the shape of the floor. Given HorizonNet's simplicity, efficiency and state-of-the-art performance on room layout estimation, we followed its encoder-decoder strategy for learning floor layout from a single fisheye image. We introduce the deformable convolution according to the distortion of the orthographic projection into the encoder to improve modeling ability of CNNs for geometric transformations. Considering the distribution characteristics of the floor corners in the fisheye image, we restore the 2D representation [C, H, W] from the 1D vector [C, W]. Additionally, the "time step" of RNNs is designed as two parameters: row and column, to capture long-range geometric pattern of the entire indoor scene.

Encoder: We leverage ResNet [8] as a feature extractor, which has proved to be one of the most effective encoders for both perspective and omnidirectional images [29]. Considering that the low-level convolution layer can learn low-level features such as edge and color, the high-level convolution layer can learn key distinguishing features. We follow the design strategy of [4] and replace the standard convolution (of 3 × 3 filter) of the last block in ResNet-50 with our deformed convolution, OrthConv. The last four feature maps of the encoder are retained for simultaneous capture of low-level and high-level features. Res2- Res5 of feature maps are then reduced to 32 × 64 × 64, 64 × 64 × 64, 128 × 64 × 64 and 256 × 64 × 64, respectively, through a sequence of convolutional layers (Convs in Fig. 2). Then we reshape these features maps to the same size, 256×64×64. Finally, the reshaped features maps are concatenated to obtain a single sequential feature map of 4096×1024 (i.e., 4096 layers for a sequence having a length 4096).

Decoder: We apply the bi-directional LSTM [23] as the core of the decoder. The sequential feature map is fed to capture the long-range geometric pattern of the object for the global coherent predictions even ambiguous situations such as occlusion and cluttered scenes [29]. The output of the decoder is a 1× 1024 × 1024 feature map, which collects all the time steps of the LSTM layers to obtain the prediction mask of floor shape.

4 Experiments and Results

In this section, we present a large corpus of experiments aimed at assessing the effectiveness of our proposed distortion-aware room layout estimation method from a single fisheye image. We first describe the collection of fisheye dataset. Then explain the implementation details of experiment, including evaluation metrics and training strategy. Next, the performance of our OrthConv is evaluated on a real dataset and synthetic dataset for layout reconstruction by quantitative and qualitative comparison. Extensive experiments demonstrated that the proposed OrthConv generally outperforms DCNv1 and DCNv2. Finally, we compare our approach with other state-of-the-art approaches [6, 29, 42, 43] on layout estimation of our collected fisheye dataset, and find that our approach can achieve remarkably better performance than LayoutNetV1 [42], LayoutNetV2 [43], CFL [6] and HorizonNet [29].

4.1 Fisheye Dataset

Collecting a high-quality fisheye dataset with sufficient number of images and the corresponding layout groundtruth is crucial for training more sophisticated models. Unfortunately, existing public indoor omnidirectional datasets, such as the real dataset, consisting PanoContext dataset [36] and Stanford2D3D dataset [1], and the synthetic Structured3D dataset [37], are all panorama images. To fill this gap, we construct a top-view fisheye image dataset through transforming panorama images from [1, 36, 37], the real fisheye dataset (e.g., PanoContext-F, Stanford2D3D-F) and the synthetic fisheye dataset (Structured3D-F), with a total size of 22,583.

The PanoContext-F dataset consists of 512 real indoor fisheye images such as bedrooms and living rooms. The Stanford2D3D-F dataset with 550 real fisheye images is collected from six large-scale indoor scene such as cluttered classrooms and office. Note that these two datasets are too small, thus, following [5, 43], we combine their training and validation data for the training stage and the verification stage respectively to prevent overfitting. The split strategy of training/validation/test for the real dataset is similar to [29], with numbers at 817, 79 and 166, respectively. The Structured3D-F dataset contains 21521 synthetic fisheye images, all of which are excellent data after eliminating mislabeling. For this dataset, we select 18111, 1739 and 1671 fisheye images for training, validation and test, respectively.

4.2 Implementation Details

Evaluation metrics. Following the quantitative benchmark of [29, 34, 42], our approach is evaluated on three standard metrics: i) 2D Intersection over Union (2D IoU), calculates the pixel-wise intersection over union by projecting our predicted floor corners to the ground-truth and averaged across all fisheye images. Higher values indicate better performance. ii) Corner error (CE), defined as the average Euclidean distance between the predicted floor corners and the ground truth corners across all fisheye images. It is normalized by the image diagonal length and smaller is better. iii) Pixel error (PE), is the pixel-wise error between the semantic layout prediction (wall and floor) and the ground-truth. This error is averaged across all fisheye images and smaller is better.

Training strategy. We implement our approach using PyTorch and test on a single NVIDIA Titan X GPU. All input RGB images and the corresponding ground-truths are 1024×1024. We employ Adam optimizer [10] to train the network for 100 epochs with batch size 4 and learning rate 0.0001. The Binary Cross-Entropy Loss is applied for the floor corners. We train our network on the real dataset of PanoContext-F dataset and Stanford2D3D-F dataset and test them separately. Moreover, we further pre-train on Structured3D-F dataset for 100 epochs, then fine-tune the model on the real dataset.

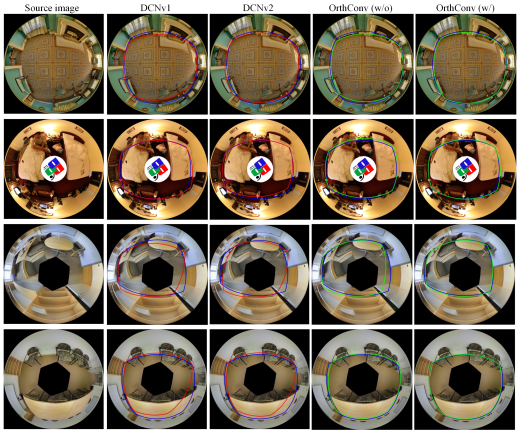


Figure 3: Qualitative comparison results of layout estimation on the real dataset. The first two rows are PanoContext-F and the following two rows are Stanford2D3D-F dataset. In each result, we display the source fisheye image, the ground truth (blue) and the predicted layout (red and green). Notice that the green emphasizes our plausible result and the pre-training on Structured3D-F dataset achieves remarkable improvement, denoted as w/. The black mask is due to the corresponding panorama from the Stanford dataset that does not cover full view vertically.

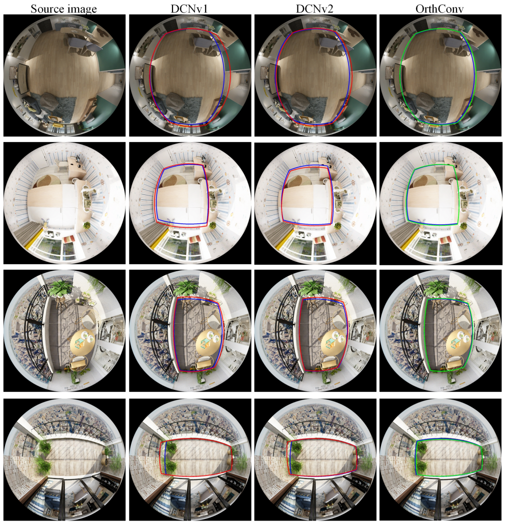


Figure 4: Qualitative comparison of layout estimation on the synthetic dataset. In each result, we display the source fisheye image, the ground truth (blue) and the predicted layout (red and green). Notice that the green emphasizes our plausible result.

4.3 Results of OrthConv

Evaluation on the real dataset. A quantitative comparison for layout estimation between the proposed deformable convolution and other convolutions on the real dataset are summarized in the first (2-nd to 4-th columns for PanoContext-F) and second (5-th to 7-th columns for Stanford2D3D-F) block of Table 1. It can be observed that OrthConv achieves more satisfactory results than DCNv1 and DCNv2, which validates the effectiveness of our deformable convolutions (OrthConv) for the fisheye image. We visually compare layout estimation results by different convolution methods on PanoContext-F and Stanford2D3D-F dataset as shown in Fig. 3. Compared to DCNv1 and DCNv2, our network with deformable

Table 1: Quantitative comparison for layout estimation on our collected fisheye dataset with the distortion-aware network using different deformable convolutions (DCNv1, DCNv2 and OrthConv). The accuracy is shown in % and bold numbers indicate the best performance. w/o and w/ indicate whether to use Structured3D-F for pre-training. Evaluation metrics with (↓), smaller is better; while for evaluation metrics with (↑), bigger is better.

Convolution Type	PanoContext-F			Stanford2D3D-F			Structured3D-F		
	CE(%)↓	PE(%)↓	2D IoU(%)↑	CE(%)↓	PE(%)↓	2D IoU(%)↑	CE(%)↓	PE(%)↓	2D IoU(%)↑
DCNv1[18]	4.14	1.47	79.55	5.14	1.69	74.47	0.77	0.61	94.29
DCNv2[19]	3.9	1.45	79.81	4.97	1.66	75.21	0.75	0.6	94.5
OrthConv (w/o)	3.77	1.44	80.59	4.81	1.79	75.72	**0.68**	**0.55**	**94.93**
OrthConv (w/)	**2.45**	**1.09**	**86.53**	**2.99**	**1.17**	**83.46**	-	-	-

convolutions (OrthConv) can better handle the fisheye distortion caused by orthographic projection and produces high-quality layout results.

We further verify the generalization of our method to room layout estimation. Our network is initially trained on the synthetic Structured3D-F dataset, generating model parameters for 100 epochs suffixed with 'SF100'. We then fine-tune this model on the real dataset, containing PanoContext-F and Stanford2D3D-F, respectively. As illustrated in the last row of Table 1, it makes a remarkable improvement on the PanoContext-F and Stanford2D3D-F dataset, with an overall performance gain of CE (1.32%, 1.82%), PE (0.35%, 0.62%) and 2D IoU (5.94%, 7.74%) respectively. Furthermore, qualitatively our results (w/ Structured3D-F pre-training) are significantly better than the results (w/o Structured3D-F pre-training) as shown in Fig. 3. This indicates that the generalization capability and the efficiency of the proposed model.

Evaluation on the synthetic dataset. The quantitative results for layout estimation between the proposed deformable convolution and other convolutions on the synthetic dataset are shown in the third (8-th to 10-th columns for Structured3D-F) block of Table 1. The proposed OrthConv obtains much higher estimation accuracy on 2D IoU and much lower error on CE compared with DCNv1 and DCNv2. The visual results on the Structured3D-F dataset are shown in Fig. 4. We can see that DCNv2 performs better than DCNv1, while our OrthConv is less error-prone and generally produces a more plausible layout estimation than others.

4.4 Comparison with State-of-The-Art Layout Estimation Methods

We compare our approach with the state-of-the-art data-driven methods: LayoutNetV1 [42], LayoutNetV2 [43], CFL [6] and HorizonNet [29]. In particular, we make a comparison with the results of CFL for standard convolutions (denoted as CFL$_{Std}$) and equirectangular convolutions (denoted as CFL$_{Equi}$) to further verify the effectiveness of our OrthConv. Additionally, we modify LayoutNetV2, replacing the last block in ResNet50 with OrthConv (denoted as LayoutNetV2$_{OrthConv}$), to verify the effectiveness of using RNN as a decoding strategy. Though all four methods follow the same encoder-decoder strategy, they take a single RGB panorama as the network input, and the general framework exists differently in the details. A direct comparison of these four methods may confuse the impact of contributions and obtain the implausible layout estimation. To carry out a fair comparison, we first unify some of the training details consisting of the input of a single RGB fisheye and the parameters of the network. Then we modify the PyTorch source code of LayoutNetV1 [42], LayoutNetV2 [43], CFL [6] and HorizonNet [29] available for retraining on the fisheye dataset. Finally, we make a quantitative and qualitative comparisons of four methods on estimating Manhattan layout using the PanoContext-F, Stanford2D3D-F and Structured3D dataset.

Qualitative results. The qualitative comparisons of the four methods on three datasets are shown in Fig. 5. The first three rows are the comparison results of all data-driven methods on PanoContext-F dataset. The middle two rows demonstrate the comparison results on Stanford2D3D-F dataset. Our approach is superior to the other methods on both datasets, and has better robustness to many situations, such as open corridors and severe occlusion as shown in Fig. 5 (2nd row, 3rd-5th rows). Additionally, the qualitative results of the four methods on the PanoContext-F dataset are better than the Stanford2D3D-F dataset, which is mainly related for two reasons. The first one is the fisheye image which contains a black mask converted from Stanford2D3D dataset, which generated from the original panorama does not cover the full view vertically. This mask influences the accuracy of feature extraction. The second one is Stanford2D3D-F dataset which shows more challenging scenarios such as cluttered laboratories or corridors. PanoContext-F dataset contains simpler indoor scene like bedrooms and living rooms.

The last three rows of Fig. 5 illustrate the qualitative comparisons of the four methods on Structured3D dataset, where the scenarios are less complex than the two real datasets mentioned above. The results show that our approach can achieve more accurate estimation of localization of layout corners, especially for a better estimation of the cluttered scenes and the open spaces, such as the balcony and kitchen (7th row and 8th row).

Quantitative Evaluation. Table 2 demonstrates the performance of these four methods on layout corners estimation using PanoContext-F dataset, Stanford2D3D-F dataset and Structured3D dataset, respectively. We consider CE, PE and 2DIoU for the performance evaluation. The results display that our approach obtains state-of-the-art layout estimation from a single fisheye image on three datasets. For PanoContext-F, Stanford2D3D-F dataset, LayoutNetV1 and CFL$_{Std}$ have similar performance on CE and 2DIoU while LayoutNetV2 boosts the overall performance. Especially, LayoutNetV2$_{OrthConv}$ with a large margin (∼3.5% in CE, ∼18% in 2DIoU). Observably, the performance of CFL $_{Equi}$ has a large drop compared with CFL$_{Std}$ on these two real datasets (∼4.5% in CE, ∼15% in 2DIoU), while it is comparable on Structured3D-F dataset. This shows that the equirectangular convolutions cannot solve the distortion in the fisheye that is different from the panoramic distortion. Compared with LayoutNetV2$_{OrthConv}$, our method has a further improvement, especially on PanoContext-F dataset and Structured3D-F dataset. This is due to the fact that LSTM, a type of RNN architecture, stores its prediction information for other regions in the cell state, and thus it can accurately predict the occlusion region based on the geometric pattern of the entire scene.

5 Applications

The use of digital images and videos captured by traditional and omnidirectional cameras has grown explosively in such fields as surveillance and social networking. The quantity of media places a cognitive burden on users, particularly in tasks such as monitoring videos captured from massive camera networks. The most widely used but also low efficient way to display massive surveillance videos is to arrange them in a monitor matrix. Researchers have studied how to use 3D graphics and mixed reality techniques to effectively

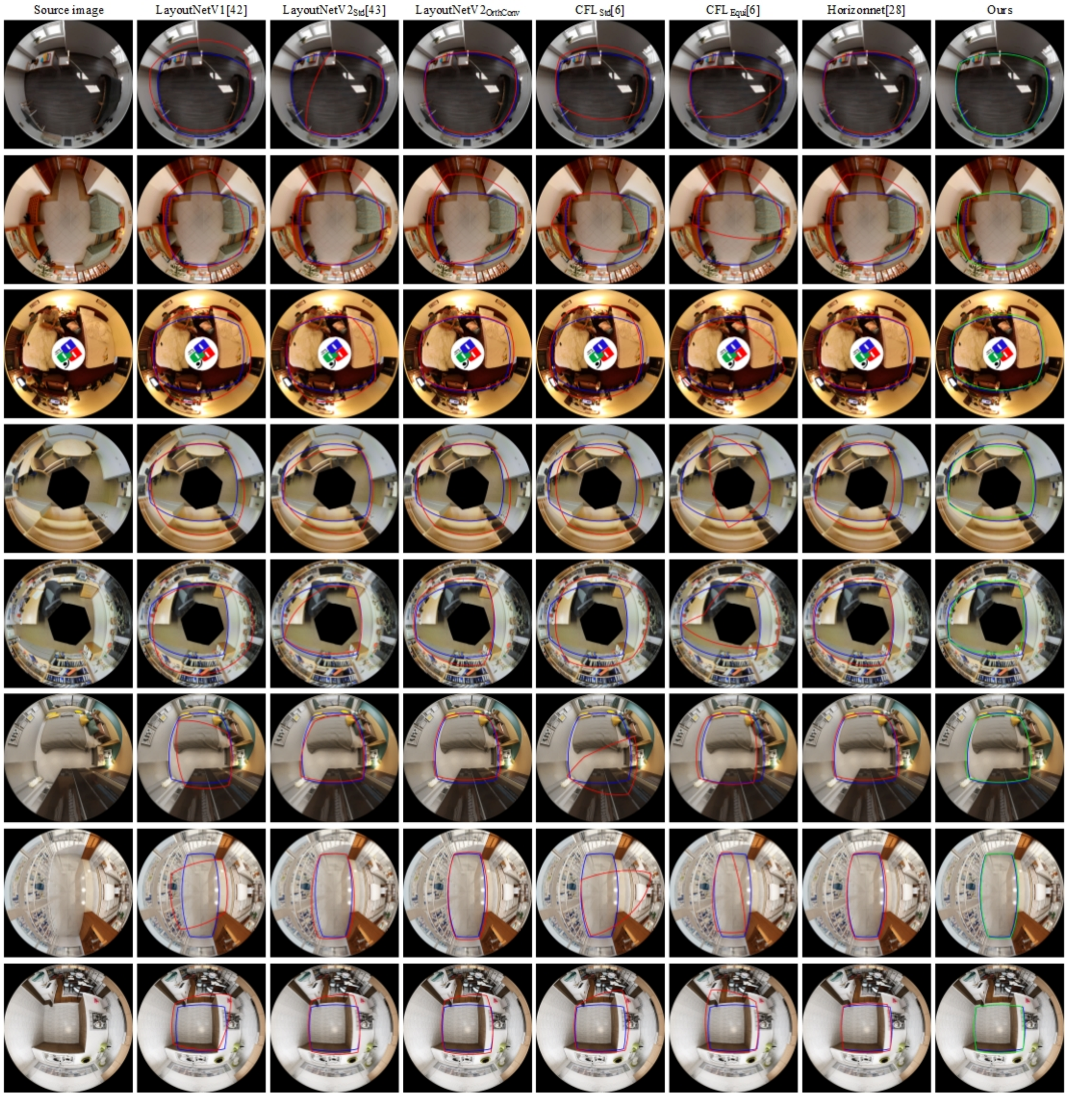


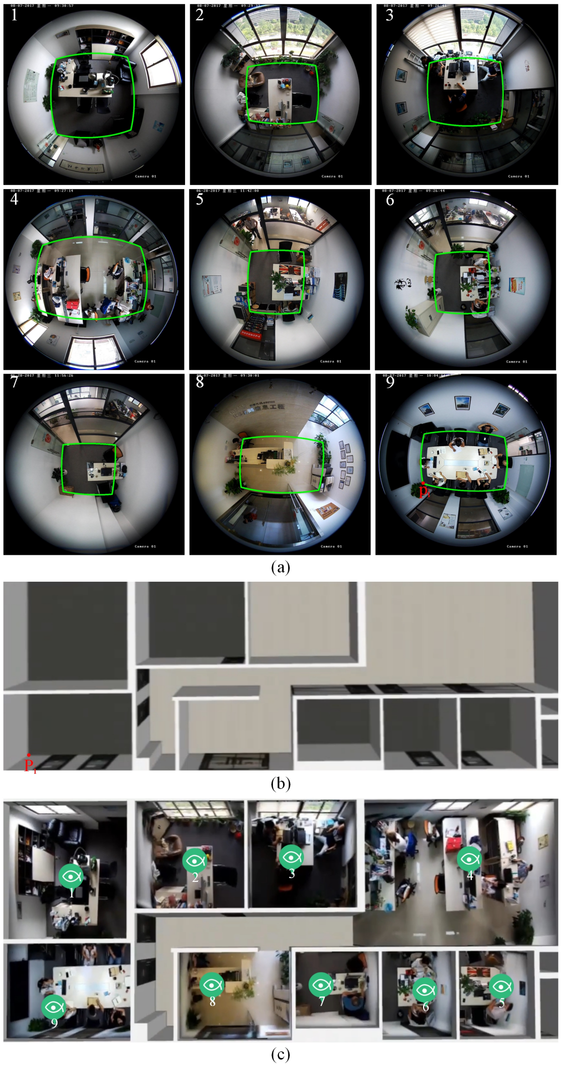
Figure 5: Comparison with state-of-the-art methods on three datasets. The 1st to 3rd rows are PanoContext-F, the 4th and 5th rows are Stanford2D3D-F dataset and the last three rows are Structured3D dataset. Left to right: results of LayoutNetV1, LayoutNetV2$_{Std}$, LayoutNetV2$_{OrthConv}$, CFL$_{Std}$, CFL$_{Equi}$, HorizonNet and our approach. In each result, we display the source fisheye image, the ground truth (blue) and the predicted layout (red and green). Note that the green emphasizes our plausible result.

Table 2: Quantitative comparison of [6, 29, 42, 43] and our approach on our collected fisheye dataset. The accuracy is shown in % and bold numbers indicate the best performance. Evaluation metrics with (↓), smaller is better; while for evaluation metrics with (↑), bigger is better.

Methods	PanoContext-F			Stanford2D3D-F			Structured3D-F		
	CE(%)↓	PE(%)↓	2D IoU(%)↑	CE(%)↓	PE(%)↓	2D IoU(%)↑	CE(%)↓	PE(%)↓	2D IoU(%)↑
LayoutNetV1 [42]	7.62	7.20	60.75	8.65	6.71	56.72	5.9	6.23	58.21
LayoutNetV2$_{Std}$ [43]	5.73	5.96	75.49	6.64	5.91	70.96	1.33	2.00	90.67
LayoutNetV2$_{OrthConv}$	4.40	4.61	78.91	4.89	4.90	75.46	1.15	1.88	91.58
CFL$_{Std}$ [6]	8.53	2.19	62.71	9.89	2.78	57.78	5.96	1.34	68.95
CFL$_{Equi}$ [6]	13.85	3.29	46.21	14.18	3.43	44.82	6.11	1.46	66.79
HorizonNet [29]	4.14	1.48	79.42	5.19	1.71	73.91	0.89	0.69	93.33
Ours	**3.77**	**1.44**	**80.59**	**4.81**	**1.79**	**75.72**	**0.68**	**0.55**	**94.93**

organize and visualize videos captured from such networks, also known as immersive video [15, 18, 22, 25, 38]. It could produce immersive, detailed, informative, spatio-temporally consistent visual experience.

For a single ceiling mounted camera equipped with 180° circular fisheye lenses, thanks to its low cost and omnidirectional FoV, it can obtain image information of the entire scene to the maximum and efficiently. This type of image has great advantages in structure recovery, content visualization and overall context understanding. DeCamp et al. [18] proposed an immersive system for browsing and



(a)

(b)

(c)

Figure 6: MR video surveillance: an real office environment, captured from a building video surveillance system, constructed from 9 fisheye lens cameras. (a) The layout estimation result of our approach for each fisheye image (marked by green lines). (b) The existing 3D models of the office environment. (c) The texture model recovered from each fisheye image consists of one floor and four walls.

visualizing surveillance video, HouseFly. It uses fisheye camera as a video capture tool and regular indoor CAD model a visual carrier of video. However, it uses an interactive calibration method to project the video onto the 3D model to display multiple streams, which is time-consuming and labor-intensive.

The essence of camera calibration is to calculate the camera intrinsic and extrinsic parameters. We perform the calibration by the pairs of corresponding points (p_i, P_i), i is the number of wall-floor corners. p_i is the image-coordinate point of corner generated from our layout estimation approach, as shown in Fig. 6(a). P_i denotes the corresponding world-coordinate point on the corner of the existing 3D model, a manually built CAD model, as shown in Fig. 6(b). This model as one of the primary display elements, provides an overall space for MR video surveillance. We also need manual operations to annotate 4 matching points for the upper corners of fisheye image and the 3D model. With help of them, since the correct corner maps have been found for each fisheye image, rich corresponding points could be provided. Given these corresponding points, we define an error function for the calculation of camera parameters (e.g., rotation and translation matrix) through the Levenberg Marquardt solver [17]. Then the projective texture mapping [26] is conducted to calculate the corresponding world-coordinate point for image-coordinate point on the fisheye image, generating the textured model as shown in Fig. 6(c). Finally, we perform texture update for real-time video to achieve immersive video surveillance.

The method of automatic or semi-automatic room layout estimation based on fisheye image can provide high-efficiency and low-cost technical implementations for the mapping process. This technology bringing broad application prospects in VR house viewing and MR video surveillance. Fig. 6 We display an MR-based real building video surveillance scene, equipped with nine fisheye lens cameras, with a deployment height of 2.8 meters, as shown in Fig. 6. It achieves the visualization of the synchronized multiple video streams, which can be used for intelligent building management in the future. However, it also have some limitations: (i) due to the lack of non-cuboid images, one assumption of our method is the cuboid geometry. It is difficult for our method to obtain plausible layout estimation from noncuboid image, such as the 4th and 5th in Fig. 5; and (ii) the lack of texture in the fisheye image will bring a poor visual experience, such as the black mask of the 4th and 5th in Fig. 5.

6 Conclusion

In this work, we have presented a distortion-aware learning network to estimate room layout from a single fisheye image. Our network architecture is trained under the complete supervision of ground truth corners. To achieve this, we collect the first fisheye dataset by re-using public available panorama images with both real-world and synthetic datasets. Since distortion is a major challenge for layout estimation from fisheye images, we introduce deformable convolution (i.e., OrthConv) to overcome it caused by the orthographic projection. Experiments show that it outperforms the state-of-the-art convolutions for omnidirectional image processing, including DCNv1 and DCNv2. Additionally, extensive comparative experiments with the state-of-the-art methods show that our method can achieve superior performance, where directly training using our fisheye dataset is the key to achieve appreciable accuracy. What's more, we present an MR-based building video surveillance scene, achieving an immersive hybrid display experience and demonstrating the high-efficiency of our approach for MR video surveillance. Finally, our fisheye dataset can contribute to development of future applications (e.g., surveillance, navigation and entertainment) requiring layout estimation, depth estimation and object detection in fisheye images.

As our approach is the first data-driven work for fisheye layout estimation, there are many challenges needed to overcome. One is that the prediction performance of our method may be affected by the large occluding objects in the fisheye image of the real scene. The other is the layout estimation results of our approach are restricted to the Manhattan world. We will explore the following research directions in future work. First, introducing instance segmentation or object depth estimation branch to the network architecture, which ignores the large occluding objects to potentially improve the accuracy of layout estimation. Second, designing a more general network for omnidirectional image processing that is not limited to and Manhattan, including panorama and fisheye images with serious distortions.

Acknowledgments

This work was supported by the National Key Research and Development Program of China (Grant No. 2018YFB2100601, 2018YFB2100602), and the National Natural Science Foundation of China (Grant No. 61872023).

References

[1] I. Armeni, S. Sax, A. R. Zamir, and S. Savarese. Joint 2D-3D-semantic data for indoor scene understanding, 2017.

[2] Z. Cai and N. Vasconcelos. Cascade r-cnn: Delving into high quality object detection. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6154–6162, 2018.

[3] X. Chen, J. Yu, S. Kong, Z. Wu, and L. Wen. Joint anchor-feature refinement for real-time accurate object detection in images and videos. *IEEE Transactions on Circuits and Systems for Video Technology*, 31(2):594–607, 2021.

[4] J. Dai, H. Qi, Y. Xiong, Y. Li, G. Zhang, H. Hu, and Y. Wei. Deformable convolutional networks. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 764–773, 2017.

[5] E. Delage, H. Lee, and A. Ng. A dynamic bayesian network model for autonomous 3d reconstruction from a single indoor image. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, volume 2, pages 2418–2428, 2006.

[6] C. Fernandez-Labrador, J. M. Facil, A. Perez-Yus, C. Demonceaux, J. Civera, and J. J. Guerrero. Corners for layout: End-to-end layout recovery from 360 images. In *IEEE Robotics and Automation Letters*, volume 5, pages 1255–1262, 2020.

[7] C. Fernandez-Labrador, A. Perez-Yus, G. Lopez-Nicolas, and J. J. Guerrero. Layouts from panoramic images with geometry and deep learning. In *IEEE Robotics and Automation Letters*, volume 3, pages 3153–3160, 2018.

[8] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016.

[9] H. Jia and S. Li. Estimating the structure of rooms from a single fisheye image. In *IEEE Computer Society*, 2013.

[10] D. P. Kingma and J. L. Ba. Adam: A method for stochastic optimization. In *ICLR 2015 : International Conference on Learning Representations 2015*, 2015.

[11] C.-Y. Lee, V. Badrinarayanan, T. Malisiewicz, and A. Rabinovich. Roomnet: End-to-end room layout estimation. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 4875–4884, 2017.

[12] D. C. Lee, M. Hebert, and T. Kanade. Geometric reasoning for single image structure recovery. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 2136–2143, 2009.

[13] M. Li, Y. Zhou, M. Meng, Y. Wang, and Z. Zhou. 3d room reconstruction from a single fisheye image. In *2019 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8, 2019.

[14] S. Liu, L. Qi, H. Qin, J. Shi, and J. Jia. Path aggregation network for instance segmentation. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8759–8768, 2018.

[15] S. Moezzi, A. Katkere, D. Y. Kuramura, and R. Jain. Reality modeling and visualization from multiple video sequences : Virtual reality. *IEEE Computer Graphics and Applications*, 16(6):58–63, 1996.

[16] R. Mohan and A. Valada. Efficientps: Efficient panoptic segmentation. *International Journal of Computer Vision*, 129(5):1–29, 2021.

[17] J. J. More, B. S. Garbow, and K. E. Hillstrom. User guide for minpack-1. *Report ANL-80-74*, 1980.

[18] D. Philip, S. George, K. Rony, and R. Deb. An immersive system for browsing and visualizing surveillance video. *ACM International Conference on Multimedia*, pages 371–380, 2010.

[19] G. Pintore, M. Agus, and E. Gobbetti. Atlantanet: Inferring the 3d indoor layout from a single 360° image beyond the manhattan world assumption. In *European Conference on Computer Vision*, pages 432–448, 2020.

[20] C. Playout, O. Ahmad, F. Lécué, and F. Cheriet. Adaptable deformable convolutions for semantic segmentation of fisheye images in autonomous driving systems. *arXiv preprint arXiv:2102.10191*, 2021.

[21] A. Pérez-Yus, G. López-Nicolás, and J. J. Guerrero. Peripheral expansion of depth information via layout estimation with fisheye camera. In *European Conference on Computer Vision*, pages 396–412, 2016.

[22] H. Sawhney, A. Arpa, R. Kumar, S. Samarasekera, M. Aggarwal, S. Hsu, D. Nister, and K. J. Hanna. Video flashlights: real time rendering of multiple videos for immersive model visualization. *In Eurographics Workshop on Rendering*, pages 157–168, 2002.

[23] M. Schuster and K. Paliwal. Bidirectional recurrent neural networks. *IEEE Transactions on Signal Processing*, 45(11):2673–2681, 1997.

[24] E. Schwalbe. Geometric modelling and calibration of fisheye lens camera systems. In *Proc Isprs*, 2005.

[25] I. O. Sebe, J. Hu, S. You, and U. Neumann. 3D video surveillance with Augmented Virtual Environments. *ACM SIGMM Workshop on Video Surveillance*, pages 107–112, 2003.

[26] M. Segal, C. Korobkin, and R. V. Widenfelt. Fast shadows and lighting effects using texture mapping. *ACM International Conference on Computer Graphics and Interactive Techniques*, pages 249–252, 1992.

[27] B. Singh, M. Najibi, A. Sharma, and L. S. Davis. Scale normalized image pyramids with autofocus for object detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1–1, 2021.

[28] Y.-C. Su and K. Grauman. Learning spherical convolution for fast features from 360° imagery. In *Advances in Neural Information Processing Systems*, volume 30, pages 529–539, 2017.

[29] C. Sun, C.-W. Hsiao, M. Sun, and H.-T. Chen. Horizonnet: Learning room layout with 1d representation and pano stretch data augmentation. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1047–1056, 2019.

[30] K. Tateno, N. Navab, and F. Tombari. Distortion-aware convolutional filters for dense prediction in panoramic images. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 732–750, 2018.

[31] W. Xiongwei, S. Hoi, and D. Sahoo. Polarnet: Learning to optimize polar keypoints for keypoint based object detection. In *ICLR 2021: The Ninth International Conference on Learning Representations*, 2021.

[32] J. Xu, B. Stenger, T. Kerola, and T. Tung. Pano2cad: Room layout from a single panorama image. In *2017 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 354–362, 2017.

[33] H. Yang and H. Zhang. Efficient 3d room shape recovery from a single panorama. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5422–5430, 2016.

[34] S.-T. Yang, F.-E. Wang, C.-H. Peng, P. Wonka, M. Sun, and H.-K. Chu. Dula-net: A dual-projection network for estimating room layouts from a single rgb panorama. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3358–3367, 2019.

[35] Y. Yang, S. Jin, R. Liu, S. B. Kang, and J. Yu. Automatic 3d indoor scene modeling from single panorama. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3926–3934, 2018.

[36] Y. Zhang, S. Song, P. Tan, and J. Xiao. Panocontext: A whole-room 3d context model for panoramic scene understanding. In *European Conference on Computer Vision*, pages 668–686, 2014.

[37] J. Zheng, J. Zhang, J. Li, R. Tang, S. Gao, and Z. Zhou. Structured3d: A large photo-realistic dataset for structured 3d modeling. In *ECCV (9)*, pages 519–535, 2019.

[38] Y. Zhou, M. Cao, J. You, M. Meng, Y. Wang, and Z. Zhou. MR video fusion: interactive 3D modeling and stitching on wide-baseline videos. *ACM Symposium on Virtual Reality Software and Technology*, page 17, 2018.

[39] X. Zhu, H. Hu, S. Lin, and J. Dai. Deformable convnets v2: More deformable, better results. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9308–9316, 2019.

[40] X. Zhu, W. Su, L. Lu, B. Li, X. Wang, and J. Dai. Deformable detr: Deformable transformers for end-to-end object detection. In *ICLR 2021: The Ninth International Conference on Learning Representations*, 2021.

[41] N. Zioulis, A. Karakottas, D. Zarpalas, and P. Daras. Omnidepth: Dense depth estimation for indoors spherical panoramas. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 453–471, 2018.

[42] C. Zou, A. Colburn, Q. Shan, and D. Hoiem. Layoutnet: Reconstructing the 3d room layout from a single rgb image. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2051–2059, 2018.

[43] C. Zou, J. W. Su, C. H. Peng, A. Colburn, Q. Shan, P. Wonka, H. K. Chu, and D. Hoiem. Manhattan room layout reconstruction from a single 360° image: A comparative study of state-of-the-art methods. *International Journal of Computer Vision*, pages 1–22, 2021.