PanelNet: Understanding 360 Indoor Environment via Panel Representation

Haozheng Yu Lu He Bing Jian Weiwei Feng Shan Liu
Tencent America
{haozhengyu, lhluhe, bingjian, wfeng, shanl}@tencent.com

Abstract

Indoor 360 panoramas have two essential properties. (1) The panoramas are continuous and seamless in the horizontal direction. (2) Gravity plays an important role in indoor environment design. By leveraging these properties, we present PanelNet, a framework that understands indoor environments using a novel panel representation of 360 images. We represent an equirectangular projection (ERP) as consecutive vertical panels with corresponding 3D panel geometry. To reduce the negative impact of panoramic distortion, we incorporate a panel geometry embedding network that encodes both the local and global geometric features of a panel. To capture the geometric context in room design, we introduce Local2Global Transformer, which aggregates local information within a panel and panel-wise global context. It greatly improves the model performance with low training overhead. Our method outperforms existing methods on indoor 360 depth estimation and shows competitive results against state-of-the-art approaches on the task of indoor layout estimation and semantic segmentation.

1. Introduction

Understanding indoor environments is an important topic in computer vision as it is crucial for multiple practical applications such as room reconstruction, robot navigation, and virtual reality applications. Early methods focus on modeling indoor scenes using perspective images [9, 10, 19]. With the development of CNNs and omnidirectional photography, many works turn to understand indoor scenes using panorama images. Compared to the perspective images, panorama images have a larger field-of-view (FoV) [43] and provide the geometric context of the indoor environment in a continuous way [24].

There are several 360 input formats used in indoor scene understanding. One of the most commonly-used formats is the equirectangular projection (ERP). Modeling the holistic scene from an ERP is challenging. The ERP distortion increases when pixels are close to the zenith or nadir of the image, which may decrease the power of the convolutional

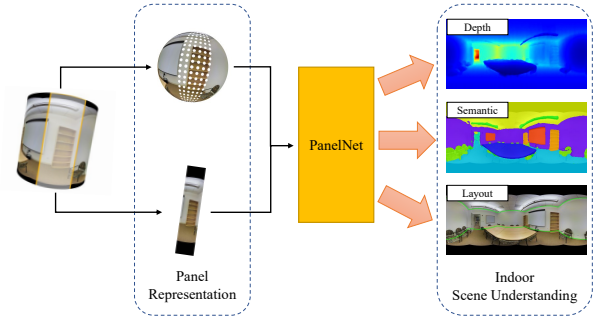


Figure 1. An overview of the proposed system. We present PanelNet, a network that learns the indoor environment using a novel panel representation of ERP. We formulate the panel representation as consecutive ERP panels with corresponding global and local geometry. By slightly modifying the network structure, PanelNet is capable of tackling major 360 indoor understanding tasks such as depth estimation, semantic segmentation and layout prediction.

structures designed for distortion-free perspective images. To eliminate the negative effects of ERP distortion, recent works [8, 21, 28] focus on decomposing the whole panorama into perspective patches, i.e., tangent images. However, partitioning a panorama into discontinuous patches breaks the local continuity of gravity-aligned scenes and objects which limits the performance of these works. To reduce the impact of distortion while preserving the local continuity, we present PanelNet, a novel network to understand the indoor scene from equirectangular projection.

We design our PanelNet based on two essential properties of equirectangular projection. (1) The ERP is continuous and seamless in the horizontal direction. (2) Gravity plays an important role in indoor environment design, which makes the gravity-aligned features crucial for indoor 360 understanding [24, 31]. Following these two properties, we tackle the challenges above through a novel panel representation of ERP. We represent an ERP as consecutive panels with corresponding global and local 3D geometry, which preserves the gravity-aligned features within a panel and maintains the global continuity of the indoor structure across panels. Inspired by Omnifusion [21], we design a geometry embed-

ding network for panel representations that encodes both local and global features of panels to reduce the negative effects of ERP distortion without adding further explicit distortion fixing modules. We further introduce Local2Global Transformer as a feature processor. Considering the nature of panel representation, we design this Transformer with Window Blocks for local information aggregation and Panel Blocks for panel-wise context capturing. The main contributions of our work are:

- We represent the ERP as consecutive vertical panels with corresponding 3D geometry. We introduce PanelNet, a novel indoor panorama understanding pipeline using the panel representation. Following the essential geometric properties of the indoor equirectangular projection, our framework outperforms existing methods on the task of indoor 360 depth estimation and shows competitive results on other indoor scene understanding tasks such as semantic segmentation and layout prediction.
- We propose a panel geometry embedding network that encodes both local and global geometric features of panels and reduces the negative impact of ERP distortion implicitly while preserving the geometric continuity.
- We design Local2Global Transformer as a feature processor, which greatly enhances the continuity of geometric features and improves the model performance by successfully aggregating the local information within a panel and capturing panel-wise context accurately.

2. Related Work

We aim to design a general framework to tackle the major tasks of indoor scene understanding using 360 images. We briefly review the related works.

2.1. Monocular depth estimation

Estimating the depth from an image is an essential problem in computer vision. Early works tackle this problem via stereo matching [29] and motion clues in a video [18]. With the flourishment of deep learning, researchers develop monocular depth estimation methods via deep neural networks. Eigen et al. [10] first develop a multi-scale deep neural network to regress depth from a single image. Their later work [9] introduces a more general multi-scale network with a VGG encoder for predicting depth, surface normal and semantic labels. Laina et al. [19] design a fully convolutional residual network with upsampling layers. They also introduce Berhu Loss for network training. Cao et al. [2] formulate the depth regression task as a classification task and apply fully-connected Conditional Random Fields (CRF) to obtain the final depth prediction. Other works address this problem with different strategies. Fu et al. [11] introduce dilated convolutions to enlarge receptive fields and utilize an ordinary regression loss for network optimization. Geometric constraints are also often exploited to enhance potential geometry relationships [26, 38].

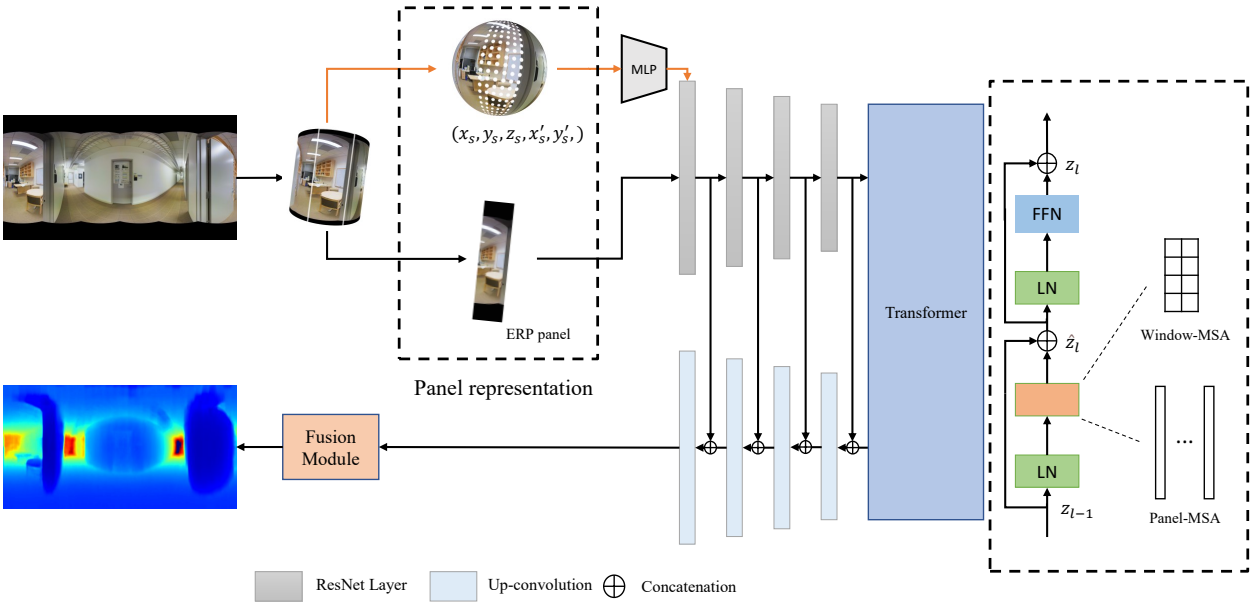
2.2. Panorama depth estimation

One key limitation for understanding the scene via a perspective image is the lack of geometric context due to the small FoV. Recently, the development of 360 imagery and the popularity of 360 cameras encourage researchers to address the scene understanding tasks directly on panoramas. Compared to perspective images, panoramas preserve the structural context of the room while introducing distortion. Recent works estimate the depth from panoramas by jointly learning the room structure [17, 40], planes and normals [6, 13]. By leveraging the gravity-aligned features in indoor panoramas, Pintore et al. [25] and Sun et al. [31] design networks that directly work on the equirectangular projections. However, directly applying convolution-based structures designed for distortion-free perspective images on panoramas may lead to sub-optimal results [45]. To reduce the negative impact of panorama distortion, several works design distortion-aware CNNs [4, 7, 32, 44] based on the nature of ERP distortion. Other methods handle this problem via less-distortion representations instead of directly modeling the distortion. Wang et al. [34] and Jiang et al. [15] fuse the cubemap projection with ERP to mimic peripheral and foveal vision as the human eye. Recently, Eder et al. [8] propose to handle panoramic distortion with tangent representation, which inspires further studies on tangent-based depth estimation such as Omnifusion [21] and 360MonoDepth [28]. However, these methods introduce discrepancies between the patches that are hard to be removed by the fusion modules. We tackle these challenges by introducing the panel representation of ERP. The panels are directly extracted from the original panorama via a sliding window mechanism similar to CNNs, which preserves the gravity-aligned information of indoor scenes within panels. Rather than fixing panoramic distortion in an explicit way, we add a panel geometry embedding network to learn the distortion for panels and reduce the negative impact of distortion with minimal computation cost.

2.3. Other indoor understanding tasks

360 semantic segmentation is another important dense prediction task for indoor understanding. Similar to 360 depth estimation, most of the recent panorama semantic segmentation works focus on reducing the negative impact of ERP distortion [14, 20, 32]. Other approaches try different strategies such as joint learning the semantic labels with layout [41] and unsupervised transfer learning [42].

For layout estimation, previous approaches model this task as a dense prediction task. Zou et al. [46] design a network in U-Net structure to jointly learn the layout boundaries



(a) Architecture (b) Architecture of Local2Global Transformer

Figure 2. The architecture of the network. Given the stride and interval, an ERP is first partitioned into consecutive panels by a sliding window. Meanwhile, for each pixel on every panel, the corresponding global coordinates are represented as its absolute 3D coordinates (x_s, y_s, z_s). Its local coordinates are represented as its relative 3D coordinates to the panel (x'_s, y'_s). The local and global coordinates are used as input of an MLP to generate geometric features. The Local2Global Transformer is applied to aggregate local information (Window Blocks) and capture global dependencies (Panel Blocks). In each Transformer block, we stack LayerNorm (LN), multi-head self-attention module (W/P-MSA) and Forward Feed Network (FFN) with skip connection as shown in (b).

and corner positions from the input RGB image and Manhattan line map. Yang et al. [37] introduce a two-branch network that incorporates both equirectangular projection and perspective ceiling view to learn different layout clues. Sun et al. [30] simplify the layout estimation task from dense prediction to three 1-D boundary predictions. They also propose a panorama stretch method that can diversify the panorama data as data augmentation. Wang et al. [35] transform layout estimation to depth prediction on the horizon line of a panorama. They design a layout-to-depth transformation to convert the layout into horizon-depth via ray casting. Jiang et al. [16] represent the room layout as the floor boundary and height. We follow this representation when predicting room layouts using our modified PanelNet.

2.4. Vision Transformer

Transformers are originally proposed in the field of natural language processing [33] and soon become very popular due to their superior performance on NLP tasks. ViT [5] and its following works [22, 23, 27, 39] demonstrate that Transformers are suitable for capturing long-range dependencies for vision tasks by achieving superior results against CNN based models on image classification, image segmentation and dense prediction. Transformers are also used for 360 indoor understanding tasks such as depth estimation [21], layout estimation [16] and semantic segmentation [42]. We design a Local2Global Transformer as a feature processor, which contains Window Blocks to aggregate local information within a panel and Panel Blocks to capture the long-range relationships among the panels. Our proposed Local2Global Transformer greatly improves the model performance on 360 indoor understanding tasks.

3. Method

3.1. Network architecture

As illustrated in Figure 2, we implement our network in an encoder-decoder fashion. We incorporate a panel geometry embedding network to reduce the negative impact of panorama distortion and Local2Global Transformer to aggregate local and global information.

Panel Representation of ERP Given the stride S and the interval I of the panels, an input RGB ERP in resolution $H_e \times W_e$ is divided into N consecutive panels by a vertical sliding window in size $H_e \times I$. Since an ERP is continuous and seamless in the horizontal direction, we extract the panels across the left and right edges of the ERP. So $N = \frac{W_e}{S}$. The corresponding local and global geometric features are generated together given I and S, details discussed in Section 3.3.

Backbone We use a ResNet-34 [12] based architecture as the feature extractor of our model. It takes the ERP panels as input and generates the feature maps of each panel in 4 different scales. We apply a 1×1 convolution layer to reduce the dimensions of the final feature map of each panel to $f_b \in \mathbb{R}^{C_b \times H_b \times W_b}$, where $H_b = \frac{H_e}{32}$, $W_b = \frac{I}{32}$, $C_b = \frac{D}{H_b \times W_b}$ and $D = 512$ for any interval and stride. The feature maps are then used as input of the Local2Global Transformer for information aggregation, discussed in Section 3.2.

Decoder As illustrated in Figure 2. For each decoder layer, we concatenate the feature map with the feature map generated by the corresponding encoder layer. We apply up-convolutions to gradually recover the feature map to the RGB input resolution. Similar to Omnifusion [21], we predict a learnable confidence map by the decoder to improve the final merging result. For the final merge, we take the average of the prediction of all panels. By slightly modifying the network structure, our model is capable of other indoor 360 dense prediction tasks such as semantic segmentation. For 360 layout estimation, we follow the layout representation of LGT-Net [16] and represent the room layouts as floor boundary and room height. We add one linear layer to generate floor boundary after the last decoder layer and two linear layers to generate room height. The default length of the output 1-D floor boundary is 1024.

3.2. Local2Global Transformer

Although partitioning the ERP into consecutive panels via a sliding window preserves the continuity of indoor structure, capturing the long-range dependencies is still crucial. Since the ERP is seamless in the horizontal direction, two distant panels on a panorama have a closer realistic distance. To address this problem and further improve local information aggregation, we present Local2Global Transformer, which consists of two major important components. Window Blocks to enhance the geometry relations within a local panel and Panel Blocks for capturing long-range context.

In Window Blocks, we compute the window multi-head self-attention similar to ViT [5]. For each panel, we reshape the input feature map $f_b \in \mathbb{R}^{C_b \times H_b \times W_b}$ into a sequence of flattened 2D feature patches $f_w \in \mathbb{R}^{N_w \times (P^2 \cdot C_b)}$, where $(P \times P)$ is the size of the feature patch and $N_w = \frac{H_b W_b}{P^2}$ is the number of feature patches in current Window Block. In our experiment, $P = 1, 2, 4$ for Window Blocks in different resolutions. Similar to ViT [5], we apply a learnable position embedding $E_w \in \mathbb{R}^{N_w \times (P^2 \cdot C_b)}$ to maintain the positional information of feature patches.

In Panel Blocks, we aim to aggregate global information via panel-wise multi-head self-attention. The feature maps of all panels are compressed to N 1-D feature vectors $f_p \in \mathbb{R}^{N \times D}$ and then used as tokens in the Panel Blocks. Similar to Window Blocks, we add a learnable positional embedding $E_p \in \mathbb{R}^{N \times D}$ to the tokens to retain patch-wise positional information. See more discussion of positional embedding in Section 4.6.

Following the standard Transformer block architecture of ViT [5], we stack multi-head self-attention module (MSA) and Feed-Forward Network (FFN). We apply a LayerNorm (LN) before each MSA and FFN. A Local2Global Transformer block is computed as

$$\begin{aligned} \hat{z}_l &= \text{(W/P)-MSA}(\text{LN}(z_{l-1})) + z_{l-1} \\ z_l &= \text{FFN}(\text{LN}(\hat{z}_l)) + \hat{z}_l \end{aligned} \tag{1}$$

where l is the block number of each stage. $\hat{z}_l$ and z_l is the output feature map of the Window/Panel - MSA and FFN. To aggregate the features from local to global, we stack the Window Blocks according to the window size from small to high successively. The Panel Blocks are stacked after the Window Blocks. For the best performance, we use 12 Transformer blocks and place them in this order: Low-Res W-Blocks(2), Mid-Res W-Blocks(2), High-Res W-blocks(2), Panel Blocks(6). We observe a performance drop when shuffling this order because the compress operation in Panel Blocks reduces the impact of local information aggregation performed by Window Blocks.

3.3. Panel geometry embedding

Inspired by the geometry fusion pipeline of Omnifusion [21], we develop a panel geometry embedding module to combine the geometry feature with the image feature together and reduce the negative impact of the ERP distortion. For a pixel $P_e(x_e, y_e)$ located on an ERP, the absolute 3D world position of its counterpart located on a unit sphere $P_s(\varphi, \theta)$ can be calculated as:

$$\begin{cases} x_s = \sin\theta \cos\varphi \\ y_s = \sin\theta \sin\varphi \\ z_s = \cos\theta \end{cases} \tag{2}$$

where φ and θ are the azimuth angle and the polar angle of the point on a sphere, respectively. The 3D world coordinates $P_s(x_s, y_s, z_s)$ are then used to generate global features. Since each ERP panel has the same distortion, the relative position of each pixel to the panel where it is located is also important. Similar to the absolute 3D position, we assign a relative 3D local position $P_s(x'_s, y'_s, z'_s)$ for each pixel per panel. We use the global 3D world coordinates of a randomly selected panel to represent the relative 3D position of all panels, which is unchanged during the entire experiment. Note that $z_s = z'_s$. So the final input of a point on a panel to the geometry embedding network is the combination of its local and global coordinates $(x_s, y_s, z_s, x'_s, y'_s)$.

We generate global and local geometric features via a two-layer MLP. The generated geometry features are added to the first layer of the backbone to make the network aware of ERP distortion.

Dataset	Method	MRE ↓	MAE ↓	RMSE ↓	RMSE (log) ↓	δ^1 ↑	δ^2 ↑	δ^3 ↑
Stanford2D3D	FCRN [19]	0.1837	0.3428	0.5774	0.1100	0.7230	0.9207	0.9731
	OmniDepth [45]	0.1996	0.3743	0.6152	0.1212	0.6877	0.8891	0.9578
	Bifuse [34]	0.1209	0.2343	0.4142	0.0787	0.8660	0.9580	0.9860
	HoHoNet [31]	0.1014	0.2027	0.3834	0.0668	0.9054	0.9693	0.9886
	SliceNet [24]	0.1043	0.1838	0.3689	0.0771	0.9034	0.9645	0.9864
	Omnifusion [21]	0.1031	0.1958	0.3521	0.0698	0.8913	0.9702	0.9875
	Ours	**0.0829**	**0.1520**	**0.2933**	**0.0579**	**0.9242**	**0.9796**	**0.9915**
Matterport3D	FCRN [19]	0.2409	0.4008	0.6704	0.1244	0.7703	0.9174	0.9617
	OmniDepth [45]	0.2901	0.4838	0.7643	0.1450	0.6830	0.8794	0.9429
	Bifuse [34]	0.2048	0.3470	0.6259	0.1143	0.8452	0.9319	0.9632
	HoHoNet [31]	0.1488	0.2862	0.5138	0.0871	0.8786	0.9519	0.9771
	SliceNet [24]	0.1764	0.3296	0.6133	0.1045	0.8716	0.9483	0.9716
	Omnifusion [21]	0.1387	0.2724	0.5009	0.0893	0.8789	0.9617	0.9818
	Ours	**0.1150**	**0.2205**	**0.4528**	**0.0814**	**0.9123**	**0.9703**	**0.9856**

Table 1. Quantitative results on real-world indoor panorama depth estimation datasets-Stanford2D3D [1] and Matterport3D [3]. Our model outperforms existing methods on all metrics.

3.4. Loss function

For depth estimation, we follow the previous works and train the network by minimizing the *Reverse Huber Loss (BerHu)* [19] in a fully supervised way.

$$B(e) = \begin{cases} |e| & |e| \leq c, \\ \frac{e^2+c^2}{2c} & |e| > c. \end{cases} \tag{3}$$

where e is the error term and the threshold c determines where the switch from L1 to L2 occurs. For semantic segmentation, we use Cross-Entropy Loss with class-wise weights to balance the examples. For layout estimation, we strictly follow LGT-Net [16] and use the combination of L1 loss for horizon depth and room height, normal loss and normal gradient loss to train our network.

4. Experiments

4.1. Datasets

Stanford2D3D [1] is a real-world dataset consisting of 1,413 panoramas collected in 6 large-scale indoor areas. For depth estimation, we follow the official split and use area1, area2, area3, area4, area6 for training and area5 for testing. For semantic segmentation, we follow the previous works and use the official 3-fold split for training and evaluation. The resolution used for depth estimation is 512×1024 and 256×512 for semantic segmentation.

PanoContext [43] and the extended Stanford2D3D [46] are two cuboid room layout datasets. PanoContext [43] contains 514 annotated cuboid room layouts collected from SunCG [36] dataset. Zou et al. [46] collected 571 panoramas from Stanford2D3D [1] and annotated them with room layouts. The input resolution of both datasets is 512×1024. We follow the same splits of previous works [16, 46] for training and testing.

Matterport3D [3] is a large-scale RGB-D dataset that contains 10,800 panoramic images collected in 90 scenes. We use this dataset for our depth estimation experiment. We follow the official split that takes 7829 panoramas from 61 houses for training and the rest for testing. The resolution used in our experiment is 512×1024.

4.2. Implementation details

For depth estimation, we evaluate the performance of our model using the standard depth estimation metrics, including Mean Relative Error (MRE), Mean Absolute Error (MAE), Root Mean Square Error (RMSE), log-based Root Mean Square Error (RMSE(log)) and threshold-based precision δ^1, δ^2 and δ^3. For semantic segmentation, we evaluate the performance using the standard semantic segmentation metrics class-wise mIoU and class-wise mAcc. For layout prediction, we use 3D Intersection over Union (3DIoU%) to evaluate the performance.

We implement our model using Pytorch and train it on eight NVIDIA GTX 1080 Ti GPUs with a batch size of 16. We train the network using Adam optimizer, and the initial learning rate is set to 0.0001. For the depth estimation, we train our model on Stanford2D3D [1] for 100 epochs and Matterport3D [3] for 60 epochs. We train our model 200 epochs on semantic segmentation datasets, and 1000 epochs on layout prediction datasets. We adopt random flipping, random horizontal rotation and random gamma augmentation for data augmentation. The default stride and interval for depth estimation are 32 and 128 while the stride is set to 16 for semantic segmentation.

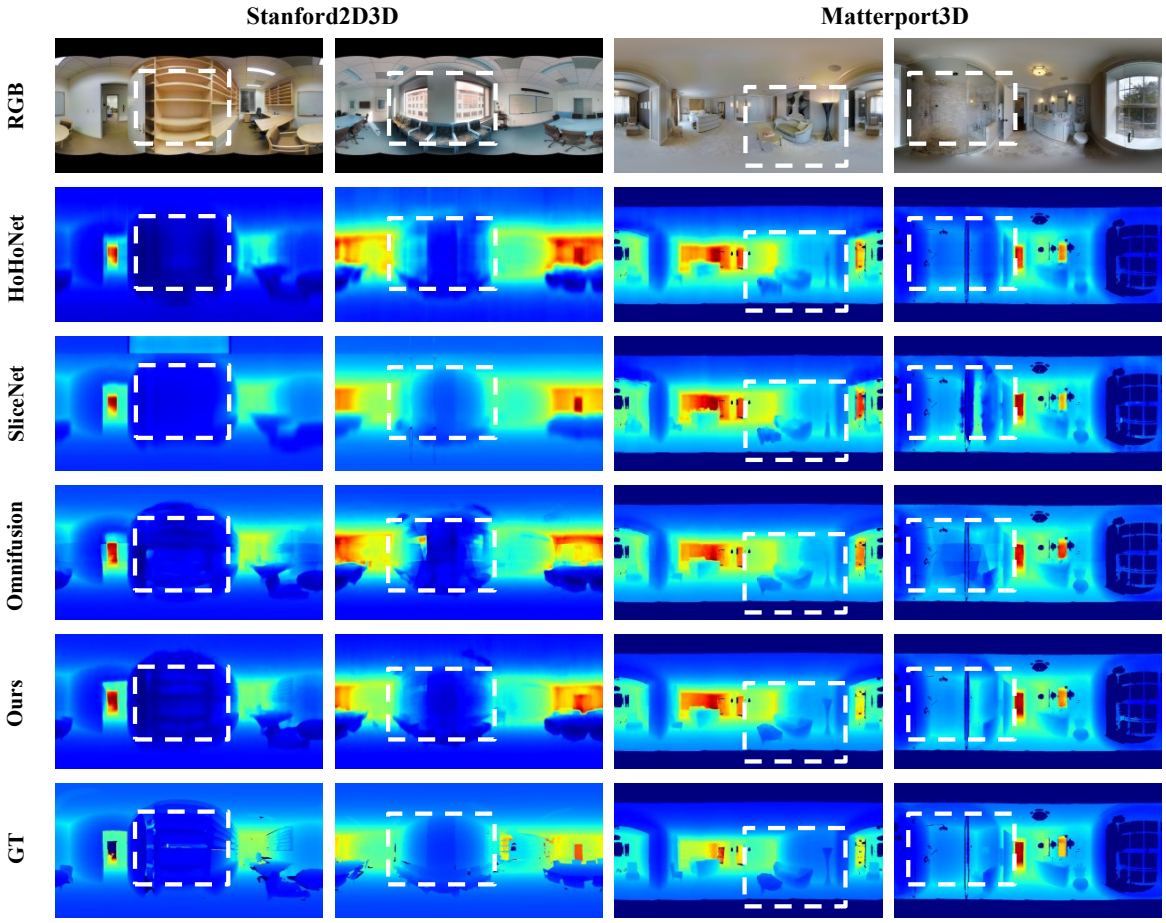


Figure 3. Qualitative results on Stanford2D3D [1] and Matterport3D [3]. Our method generates sharp object edges and shows consistent indoor structure depth prediction in different scenes. The black spots stand for the invalid depth values.

4.3. Evaluation on depth estimation datasets

We evaluate our method against state-of-the-art panorama depth estimation algorithms in Table 1. The results are averaged by the best results from three training sessions. Note that the results of SliceNet [24] on Stanford2D3D [1] were reproduced by the fixed metrics and we retrain and re-evaluate a 2-iteration Omnifusion [21] model on Matterport3D [3] dataset. Our model outperforms existing models on all metrics on both datasets. Figure 3 shows the qualitative results of our model on Stanford2D3D [1] and Matterport3D [3]. For the method that directly works on the panoramas [24, 31], they predict continuous background while lacking object details. Fusion-based method [21] generates sharp depth boundaries while the strange artifacts caused by the patch-wise discrepancy lead to inconsistent depth prediction, e.g. the bookshelf in column 1 and the shower glass in column 4. It is not removable with its patch fusion module or iteration mechanism. With the help of our proposed Local2Global Transformer, our model preserves the geometric continuity of the room structure and shows superior performance even for some challenging scenarios, e.g. the windows in column 2. Our model also generates sharp object depth edges, e.g. the floor lamp and sofa in column 3.

4.4. Evaluation on semantic segmentation datasets

We evaluate our method against state-of-the-art panorama semantic segmentation methods in Table 2. Our method improves the mIoU by 6.9% and mAcc by 8.9% against HoHoNet [31]. Note that we only use RGB panoramas as input. Figure 4 shows the qualitative results of our semantic segmentation model on Stanford2D3D [1]. Our model shows a strong ability to segment out the objects with a smooth surface, e.g. the whiteboards and windows. The segmentation edges are natural and continuous. This is because the Local2Global Transformer successfully captures the geometric context of the object. Our model is also good at segmenting out the small objects from the background, e.g. the computer in both columns. Note that the segmentation boundaries of

Method	Input	mIoU ↑	mAcc ↑
TangentImg [8]	RGB-D	41.8	54.9
HoHoNet [31]	RGB-D	43.3	53.9
Ours	RGB	**46.3**	**58.7**

Table 2. Quantitative results of semantic segmentation on Stanford2D3D [1] dataset.

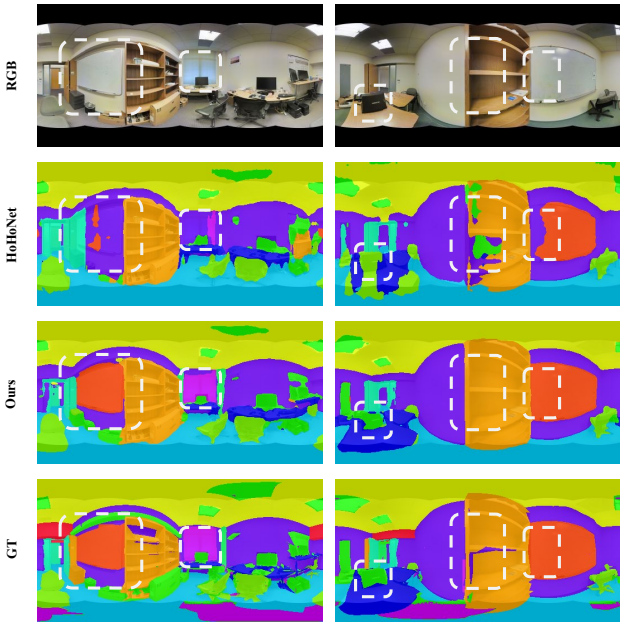


Figure 4. Qualitative results of semantic segmentation on Stanford2D3D [1] dataset.

the ceiling and the walls generated by our model are very smooth against the previous work [31], which shows the power of our panel geometry embedding network to learn the ERP distortion. Zoom in to view more details.

4.5. Evaluation on layout estimation datasets

We evaluate our method against state-of-the-art panorama layout estimation methods in Table 3. By adding linear layers at the end of our depth estimation network, our model achieves competitive performance against state-of-the-art methods designed specifically for layout estimation. Since our model is initially designed for dense prediction, it suffers an information loss in the process of upsampling and channel compression. Our layout prediction model shares the same structure with the depth estimation model before the linear layers. We can activate this model with the weights pretrained on depth estimation datasets to reduce the training overhead. We find that our layout prediction model has the best performance when the stride is 64 and the interval is 128 so we use this setup for experiments on both datasets.

Method	PanoContext	Stanford2D3D
LayoutNet v2 [47]	85.02	82.66
DuLa-Net v2 [47]	83.77	**86.60**
HorizonNet [30]	84.23	83.51
AtlantaNet [25]	-	83.94
LGT-Net [16]	**85.16**	85.76
Ours	84.52	85.91

Table 3. Quantitative results of layout estimation on PanoContext dataset and Stanford2D3D [1] dataset in 3DIoU (%). Following the previous works [16, 46], we use the combination of PanoContext [43] and Stanford2D3D [46] for training.

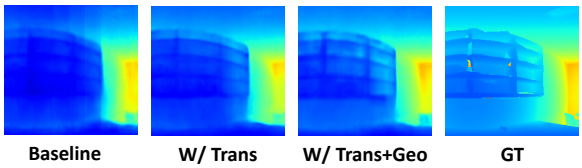


Figure 5. Qualitative effects of each network element. The baseline model is the pure CNN model listed in the first row of Table 4.

4.6. Ablation study

In this section, we conduct ablation studies to evaluate the impact of the elements and hyper-parameters of our model on Stanford2D3D [1] dataset for depth estimation.

Effects of individual network components We conduct an ablation study to evaluate the impact of each component in our model, presented in Table 4. The stride is set to 32 and the interval is 128 for all networks. We conduct our baseline model with a ResNet-34 [12] encoder and a depth decoder as illustrated in Section 3.1. Since partitioning the entire panorama into panels with overlaps greatly increase the computational complexity, we use ResNet-34 rather than vision Transformers as backbones. As we observed in Table 4, the performance improvement of adding the panel geometry embedding network to the pure CNN structure is small since the network's ability to aggregate distortion information with image features is low. By applying the Local2Global Transformer as a feature processor, our baseline network gains a significant performance improvement on all evaluation metrics. Benefiting from the information aggregation ability of our proposed Local2Global Transformer, the panel geometry embedding network fully performs its ability on distortion perception and improves the performance both quantitatively and qualitatively. As shown in Figure 5, the combination of Local2Global Transformer and panel geometry embedding network leads to the clearest object edges. We further test panel-wise relative position embedding similar to LGT-Net [16] for Panel Blocks. However, it brings minimal performance improvements on depth estimation while increasing the computational complexity.

We conduct an ablation study to further validate the use-

Method	Train Mem.	MRE	MAE	RMSE	δ^1	δ^2
Baseline	10231	0.1033	0.1859	0.3212	0.8976	0.9741
Baseline + Geo(G)	10371	0.1029	0.1861	0.3205	0.8980	0.9765
Baseline + Geo(G+L)	10509	0.1000	0.1815	0.3149	0.9012	0.9775
Baseline + Transformer(P)	10359	0.0904	0.1652	0.3058	0.9123	0.9776
Baseline + Transformer(P+W)	10379	0.0854	0.1610	0.3016	0.9164	0.9785
Baseline + Geo(G+L) + Transformer(P)	10639	0.0851	0.1572	0.2954	0.9218	0.9789
Baseline + Geo(G+L) + Transformer(P+W)	10659	**0.0829**	**0.1520**	**0.2933**	**0.9242**	**0.9796**

Table 4. Ablation study about the impact of each PanelNet component. "P" and "W" stands for the Panel Blocks and Window Blocks in Local2Global Transformer. "G" and "L" stands for the global and local feature embedded by the panel geometry embedding network. "Train Mem." stands for the GPU memory (MB) overhead of training our model on a single GTX-1080Ti GPU, the batch size is 2.

Method	MRE	MAE	RMSE	δ^1	δ^2
Omnifusion w/o Trans.	0.1132	0.1932	0.3248	0.8728	0.9690
PanelNet w/o Trans.	0.1000	0.1815	0.3149	0.9012	0.9775
Omnifusion w/ L2G	0.1054	0.1918	0.3351	0.8870	0.9754
PanelNet w/ L2G	0.0829	0.1520	0.2933	0.9242	0.9796

Table 5. Ablation study on the usefulness of panel representation against tangent images.

fulness of panel representation against tangent images. We use Omnifusion [21] as a comparison since it has a similar input format and can be trained via the same encoder-decoder CNN architecture with our model. As shown in Table 5, the panel representation with pure CNN architecture outperforms the original Omnifusion [21], which demonstrates the superiority of panel representation. We replace the default transformer of Omnifusion [21] with Local2Global Transformer. However, the Local2Global Transformer doesn't bring a huge performance improvement for tangent images since the discontinuous tangent patches lower the ability of the Window Blocks to aggregate local information in the vertical direction which reduces the continuity of depth estimation for gravity-aligned objects and scenes. On the contrary, the vertical continuity is preserved within the vertical panels. With the panel representation, the Local2Global Transformer exerts its greatest information aggregation ability.

Effects of panel size and stride We further study the effect of panel size on the performance and speed, as displayed in Table 6. The FPSs are obtained by measuring the average inference time on a single NVIDIA GTX 1080Ti GPU. We observe that for the models that have the same panel interval, i.e. width, a smaller stride enhances the performance. For the same stride, the models with larger panels have better performance. Theoretically, smaller strides improve performance because horizontal consistency is preserved by the more overlapping area of consecutive panels. Larger panels also lead to better performance because larger panels provide larger FoV, which contains more geometric context within a panel. However, we observe that keep increasing the interval may have a negative impact on performance. The larger

I	S	#Panel	FPS	MRE	RMSE	δ^1
64	16	128	6.4	0.0866	0.3040	0.9181
64	32	64	12.4	0.0909	0.3207	0.9102
64	64	32	24.4	0.0952	0.3319	0.9041
128	32	32	6.9	**0.0829**	**0.2933**	**0.9242**
128	64	16	13.5	0.0892	0.3109	0.9172
128	128	8	25.7	0.0920	0.3181	0.9103
256	64	16	7.5	0.0894	0.3047	0.9132
256	128	8	13.9	0.0908	0.3069	0.9182
256	256	4	**26.4**	0.0986	0.3248	0.8991

Table 6. Ablation study on the impact of panel size and stride. "I" and "S" stand for the panel interval and stride mentioned in Section 3.1. "#Panel" stands for the number of panels.

panel brings higher computational complexity, which forces the stride to increase to reduce the computational overhead. This makes the performance gain brought by the larger FoV be wiped out by the consistency loss due to fewer overlaps. To gain the best performance, we set the interval to 128 and the stride to 32 for most of our experiments.

5. Conclusion

We present PanelNet, a framework that understands indoor environments from 360 images. Based on the essential properties of indoor equirectangular projection (ERP), we introduce a novel panel representation to model the indoor scene. We design a panel geometry embedding network to encode both local and global geometric features which reduces the negative impact of ERP distortion implicitly. We introduce Local2Global Transformer for information aggregation, which greatly improves the performance of our model by successfully aggregating the local information within a panel and capturing panel-wise global context. Our model outperforms existing panorama depth estimation approaches on all evaluation metrics and achieves competitive results on 360 indoor semantic segmentation and layout estimation.

References

[1] Iro Armeni, Sasha Sax, Amir R Zamir, and Silvio Savarese. Joint 2d-3d-semantic data for indoor scene understanding. *arXiv*, 2017. 5, 6, 7

[2] Yuanzhouhan Cao, Zifeng Wu, and Chunhua Shen. Estimating depth from monocular images as classification using deep fully convolutional residual networks. *IEEE Transactions on Circuits and Systems for Video Technology*, 2017. 2

[3] Angel Chang, Angela Dai, Thomas Funkhouser, Maciej Halber, Matthias Niessner, Manolis Savva, Shuran Song, Andy Zeng, and Yinda Zhang. Matterport3D: Learning from RGB-D data in indoor environments. In *3DV*, 2017. 5, 6

[4] Benjamin Coors, Alexandru Paul Condurache, and Andreas Geiger. Spherenet: Learning spherical representations for detection and classification in omnidirectional images. In *ECCV*, 2018. 2

[5] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv*, 2020. 3, 4

[6] Marc Eder, Pierre Moulon, and Li Guan. Pano popups: Indoor 3d reconstruction with a plane-aware network. In *3DV*, 2019. 2

[7] Marc Eder, True Price, Thanh Vu, Akash Bapat, and Jan-Michael Frahm. Mapped convolutions. *arXiv*, 2019. 2

[8] Marc Eder, Mykhailo Shvets, John Lim, and Jan-Michael Frahm. Tangent images for mitigating spherical distortion. In *CVPR*, 2020. 1, 2, 7

[9] David Eigen and Rob Fergus. Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture. In *ICCV*, 2015. 1, 2

[10] David Eigen, Christian Puhrsch, and Rob Fergus. Depth map prediction from a single image using a multi-scale deep network. *NIPS*, 2014. 1, 2

[11] Huan Fu, Mingming Gong, Chaohui Wang, Kayhan Batmanghelich, and Dacheng Tao. Deep ordinal regression network for monocular depth estimation. In *CVPR*, 2018. 2

[12] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 4, 7

[13] Lu He, Bing Jian, Yangming Wen, Haichao Zhu, Kelin Liu, Weiwei Feng, and Shan Liu. Rethinking supervised depth estimation for 360deg panoramic imagery. In *CVPR Workshops*, 2022. 2

[14] Chiyu Max Jiang, Jingwei Huang, Karthik Kashinath, Prabhat, Philip Marcus, and Matthias Niessner. Spherical CNNs on unstructured grids. In *ICLR*, 2019. 2

[15] Hualie Jiang, Zhe Sheng, Siyu Zhu, Zilong Dong, and Rui Huang. Unifuse: Unidirectional fusion for 360 panorama depth estimation. *RAL*, 2021. 2

[16] Zhigang Jiang, Zhongzheng Xiang, Jinhua Xu, and Ming Zhao. Lgt-net: Indoor panoramic room layout estimation with geometry-aware transformer network. In *CVPR*, 2022. 3, 4, 5, 7

[17] Lei Jin, Yanyu Xu, Jia Zheng, Junfei Zhang, Rui Tang, Shugong Xu, Jingyi Yu, and Shenghua Gao. Geometric structure based and regularized depth estimation from 360 indoor imagery. In *CVPR*, 2020. 2

[18] Kevin Karsch, Ce Liu, and Sing Bing Kang. Depth transfer: Depth extraction from video using non-parametric sampling. *TPAMI*, 2014. 2

[19] Iro Laina, Christian Rupprecht, Vasileios Belagiannis, Federico Tombari, and Nassir Navab. Deeper depth prediction with fully convolutional residual networks. In *3DV*, 2016. 1, 2, 5

[20] Yeonkun Lee, Jaeseok Jeong, Jongseob Yun, Wonjune Cho, and Kuk-Jin Yoon. Spherephd: Applying cnns on a spherical polyhedron representation of 360deg images. In *CVPR*, 2019. 2

[21] Yuyan Li, Yuliang Guo, Zhixin Yan, Xinyu Huang, Ye Duan, and Liu Ren. Omnifusion: 360 monocular depth estimation via geometry-aware fusion. In *CVPR*, 2022. 1, 2, 3, 4, 5, 6, 8

[22] Ze Liu, Han Hu, Yutong Lin, Zhuliang Yao, Zhenda Xie, Yixuan Wei, Jia Ning, Yue Cao, Zheng Zhang, Li Dong, et al. Swin transformer v2: Scaling up capacity and resolution. In *CVPR*, 2022. 3

[23] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *ICCV*, 2021. 3

[24] Giovanni Pintore, Marco Agus, Eva Almansa, Jens Schneider, and Enrico Gobbetti. SliceNet: deep dense depth estimation from a single indoor panorama using a slice-based representation. In *CVPR*, 2021. 1, 5, 6

[25] Giovanni Pintore, Marco Agus, and Enrico Gobbetti. Atlantanet: Inferring the 3d indoor layout from a single 360° image beyond the manhattan world assumption. In *ECCV*, 2020. 2, 7

[26] Xiaojuan Qi, Renjie Liao, Zhengzhe Liu, Raquel Urtasun, and Jiaya Jia. Geonet: Geometric neural network for joint depth and surface normal estimation. In *CVPR*, 2018. 2

[27] René Ranftl, Alexey Bochkovskiy, and Vladlen Koltun. Vision transformers for dense prediction. In *ICCV*, 2021. 3

[28] Manuel Rey-Area, Mingze Yuan, and Christian Richardt. 360monodepth: High-resolution 360deg monocular depth estimation. In *CVPR*, 2022. 1, 2

[29] Daniel Scharstein and Richard Szeliski. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *IJCV*, 2002. 2

[30] Cheng Sun, Chi-Wei Hsiao, Min Sun, and Hwann-Tzong Chen. Horizonnet: Learning room layout with 1d representation and pano stretch data augmentation. In *CVPR*, 2019. 3, 7

[31] Cheng Sun, Min Sun, and Hwann-Tzong Chen. Hohonet: 360 indoor holistic understanding with latent horizontal features. In *CVPR*, 2021. 1, 2, 5, 6, 7

[32] Keisuke Tateno, Nassir Navab, and Federico Tombari. Distortion-aware convolutional filters for dense prediction in panoramic images. In *ECCV*, 2018. 2

[33] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *NIPS*, 2017. 3

[34] Fu-En Wang, Yu-Hsuan Yeh, Min Sun, Wei-Chen Chiu, and Yi-Hsuan Tsai. Bifuse: Monocular 360 depth estimation via bi-projection fusion. In *CVPR*, 2020. 2, 5

[35] Fu-En Wang, Yu-Hsuan Yeh, Min Sun, Wei-Chen Chiu, and Yi-Hsuan Tsai. Led 2-net: Monocular 360° layout estimation via differentiable depth rendering. In *CVPR*, 2021. 3

[36] Jianxiong Xiao, Krista A Ehinger, Aude Oliva, and Antonio Torralba. Recognizing scene viewpoint using panoramic place representation. In *CVPR*, 2012. 5

[37] Shang-Ta Yang, Fu-En Wang, Chi-Han Peng, Peter Wonka, Min Sun, and Hung-Kuo Chu. Dula-net: A dual-projection network for estimating room layouts from a single rgb panorama. In *CVPR*, 2019. 3

[38] Wei Yin, Yifan Liu, Chunhua Shen, and Youliang Yan. Enforcing geometric constraints of virtual normal for depth prediction. In *ICCV*, 2019. 2

[39] Li Yuan, Yunpeng Chen, Tao Wang, Weihao Yu, Yujun Shi, Zi-Hang Jiang, Francis EH Tay, Jiashi Feng, and Shuicheng Yan. Tokens-to-token vit: Training vision transformers from scratch on imagenet. In *ICCV*, 2021. 3

[40] Wei Zeng, Sezer Karaoglu, and Theo Gevers. Joint 3d layout and depth prediction from a single indoor panorama image. In *ECCV*, 2020. 2

[41] Cheng Zhang, Zhaopeng Cui, Cai Chen, Shuaicheng Liu, Bing Zeng, Hujun Bao, and Yinda Zhang. Deeppanocontext: Panoramic 3d scene understanding with holistic scene context graph and relation-based optimization. In *ICCV*, 2021. 2

[42] Jiaming Zhang, Kailun Yang, Chaoxiang Ma, Simon Reiß, Kunyu Peng, and Rainer Stiefelhagen. Bending reality: Distortion-aware transformers for adapting to panoramic semantic segmentation. In *CVPR*, 2022. 2, 3

[43] Yinda Zhang, Shuran Song, Ping Tan, and Jianxiong Xiao. Panocontext: A whole-room 3d context model for panoramic scene understanding. In *ECCV*, 2014. 1, 5, 7

[44] Chuanqing Zhuang, Zhengda Lu, Yiqun Wang, Jun Xiao, and Ying Wang. Acdnet: Adaptively combined dilated convolution for monocular panorama depth estimation. In *AAAI*, 2022. 2

[45] Nikolaos Zioulis, Antonis Karakottas, Dimitrios Zarpalas, and Petros Daras. Omnidepth: Dense depth estimation for indoors spherical panoramas. In *ECCV*, 2018. 2, 5

[46] Chuhang Zou, Alex Colburn, Qi Shan, and Derek Hoiem. Layoutnet: Reconstructing the 3d room layout from a single rgb image. In *CVPR*, 2018. 2, 5, 7

[47] Chuhang Zou, Jheng-Wei Su, Chi-Han Peng, Alex Colburn, Qi Shan, Peter Wonka, Hung-Kuo Chu, and Derek Hoiem. Manhattan room layout reconstruction from a single 360° image: A comparative study of state-of-the-art methods. *IJCV*, 2021. 7