PanoViT: Vision Transformer for Room Layout Estimation from a Single Panoramic Image

Weichao Shen, Yuan Dong, Zonghao Chen, Zhengyi Zhao, Yang Gao, and Zhu Liu

Alibaba Group

Abstract. In this paper, we propose PanoViT, a panorama vision transformer to estimate the room layout from a single panoramic image. Compared to CNN models, our PanoViT is more proficient in learning global information from the panoramic image for the estimation of complex room layouts. Considering the difference between a perspective image and an equirectangular image, we design a novel recurrent position embedding and a patch sampling method for the processing of panoramic images. In addition to extracting global information, PanoViT also includes a frequency-domain edge enhancement module and a 3D loss to extract local geometric features in a panoramic image. Experimental results on several datasets demonstrate that our method outperforms state-of-the-art solutions in room layout prediction accuracy.

Keywords: Room layout estimation, Panorama vision transformer, Panoramic image

1 Introduction

Estimating 3D room layout from images plays an important role in many applications, such as virtual/augmented reality and robot vision. Compared with a perspective image with a narrow field of view (FOV), a 360-degree panoramic image contains the content of the entire room and provides more information for room layout estimation. With the popularity of 360 cameras, estimating the 3D room layout from a single panoramic image has become an active research topic [24,10].

Recent works utilize the convolutional neural network (CNN)[23,12,9,13,15] to estimate the 3D room layout. Although many methods have achieved good reconstruction accuracy, CNN-based methods still struggled in estimating complex room layouts. One reason is that convolutional filters are more suitable at extracting local features, but estimating complex room layouts also requires global information in a panoramic image to deal with the occluded regions. In recent years, transformer [14] is very popular in many natural language processing (NLP) tasks [3,11,1] and also shows strong performance on many computer vision tasks[4,2,8]. Compared with CNNs, transformer is skilled at extracting the global relationship in an image, which is very useful for the estimation of

complex room layouts. Therefore, in this article, we focus on introducing a vision transformer in estimating the room layout from a single panoramic image.

There are several issues that need to be addressed when designing a vision transformer for panoramic images. First of all, as shown in many existing works, visual transformers need to be pre-trained on large data sets in order to work well. However, the spherical geometric information in panoramic images makes them different from perspective images, so that visual transformers pre-trained on perspective image datasets cannot be used to obtain satisfactory room layout estimation results from panoramic images. At the same time, the data in panoramic image datasets is insufficient for adequate pre-training of a vision transformer. The second problem is that the position attribute of the patch on a panoramic image is different from that on a perspective image, so that the position embedding methods for perspective images are not suitable for panoramic images.

In this article, we propose PanoViT, a panorama vision transformer for estimating the room layout from a single RGB panoramic image. Since the visual transformer pre-trained on perspective images does not perform well on panoramic images, we introduce the CNN features in PanoViT as a bridge between the panoramic image and the visual transformer. Specifically, our model contains a CNN backbone to extract multi-scale features, then PanoViT takes the original panoramic image and the multi-scale features as the inputs. The CNN pre-trained on perspective image datasets performs well in processing panoramic images [13.15], so that the PanoViT can better process panoramic images by learning the global relationship among the panoramic image patches and the multi-scale CNN features. Considering the different position attributes between the patches sampled from panoramic images and the patches sampled from perspective images, we design a novel patch sampling method and a recurrent position embedding in PanoViT for processing panoramic images. The recurrent position embedding can ignore the absolute starting position of the patches in the horizontal direction, thereby emphasizing the translation invariance of a panoramic image in the horizontal axis.

In addition to extracting global information, we also emphasize the ability of PanoViT to extract local features in panoramic images. Therefore, we design a frequency-domain edge enhancement module and 3D loss for PanoViT by exploiting the geometric information of panoramic images. On the one hand, the edge information in a panoramic image is important for room layout estimation[23]. We analyze the Fourier transform of panoramic images and propose an efficient edge enhancement method in the frequency domain. On the other hand, we design a 3D loss based on the geometric information of a panoramic image to measure the reconstruction error in 3D space, thereby alleviating the error imbalance problem caused by the calculation of the loss function in the image coordinates.

We characterize our contributions as follows:

 We propose the PanoViT model to introduce vision transformers into the room layout estimation from a single RGB panoramic image. The experimental results on several datasets demonstrate that our method outperforms state-of-the-art solutions in room layout prediction accuracy.

- We designed a new patch sampling method and a recurrent position embedding for the vision transformer to process panoramic images.
- We further design an edge enhancement method in the frequency domain and propose a 3D loss to use the geometric information in a panoramic image to improve the overall performance.

2 Related works

Estimating room layout from images has been a hot research topic for many years. In this section, we only discuss the literature that focuses on the room layout estimation from a single panorama. We also analyze the works about vision transformers that are related to our method.

Room layout estimation from panoramic images Compared with a perspective image, a panoramic image can capture a wide indoor environment that is useful for room layout estimation. Therefore, many room layout estimation methods rely on panoramic images. Zhang et al. [20] proposed to estimate the 3D bounding box of a room from an input panorama and constructed an annotated PanoContect dataset. Some extended works focused on improving the model with more geometry metric [16,17] and semantic features [19]. Recently, many methods utilized deep neural networks to improve room layout estimation. Zou et al. [23] designed the LayoutNet to directly predict the corner and boundary map from a panorama. Yang et al. [18] proposed the DuLa-Net that can leverage different clues in both the equirectangular panorama-view image and the perspective ceiling-view image to predict the room layout. Unlike the dense prediction methods for layout estimation, Sun et al. [12] proposed the HorizonNet to leverage the property of aligned panoramic image to predict the positions of floor-wall, ceiling-wall boundaries, and wall-wall boundaries. Sun et al. [13] further extended the HorizonNet with a latent horizontal features and proposed the HohoNet with much better speed and accuracy. Pintore et al. [9] proposed the AtlantaNet to reconstruct rooms that are not following Manhattan world constraints. Wang et al. [15] formulated the task of 360 layout estimation as a problem of predicting the depth on the horizon line of a panorama. Unlike all the existing methods based on CNN neural network, we propose a room layout estimation method based on vision transformers. Compared with CNN, our PanoViT can leverage the transformer to exploit the global vision dependencies in a panoramic image, which is helpful to the inference of the occluded room layout.

Vision transformer Transformer models have demonstrated excellent performance on a broad range of language tasks. The breakthroughs from transformer networks in NLP motivate a lot of works to adapt the transformer into vision tasks. Carion et al. [2] proposed a detect transformer (DETR) that firstly introduced the transformer into the visual detection task. Based on DETR, Zhu et



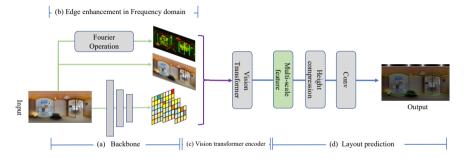


Fig. 1. The overview of PanoViT. Our model can be divided into four modules, including (a) a backbone, (b) an edge enhancement module, (c) a vision transformer encoder, and (d) a layout prediction module.

al. [22] proposed the deformable DETR to combine the best of the sparse spatial sampling of deformable convolution and the relation modeling capability of transformers. Recently, transformers were further introduced in different vision tasks, such as image classification [4], semantic segmentation [21], and image generation [5]. All the existing vision transformer methods are designed to deal with the perspective images. While the 360 degree panoramic image processing becomes more popular, we attempt to introduce transformers in this area.

Approach 3

3.1 Overview

The goal of the PanoViT model is to predict the room layout based on 360-degree panoramic images. According to HorizonNet [12], we represent the room layout as a one-dimensional representation. Specifically, our model takes a 360-degree panoramic image as input and outputs a $C \times 1 \times W$ tensor, where C = 3 is the number of channels, and W represents the width of the panoramic image. The first two channels represent the ceiling-wall and floor-wall boundaries, and the last one represents the probability of existence of the wall-wall boundary. Based on this representation, the framework of PanoViT's network can be divided into four modules, including a backbone, a panoramic visual transformer encoder, an edge enhancement module and a layout prediction module. A panoramic image is sent to the backbone to extract multi-scale feature maps, and sent to the edge enhancement module to obtain an edge enhancement map. The panoramic vision transformer encoder takes the original image, edge enhancement maps and multi-scale feature maps as input and outputs a feature vector for the layout prediction module to estimate the one-dimensional room layout representation. The pipeline of our network is shown in Fig. 1. We will introduce each module in detail in the next section.

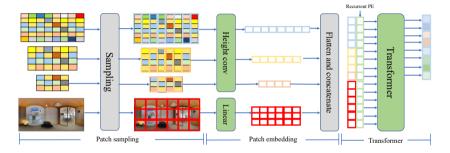


Fig. 2. The overview of our panorama vision transformer. Our model contains a patching sampling module, a patch embedding module, and a vision transformer backbone.

3.2 Backbone

The backbone is a convolutional network used to extract multi-scale features from panoramic images. In this article, we use ResNet34 as the backbone, and combine intermediate feature maps from different ResNet blocks as multi-scale features to capture low-level and high-level visual patterns. At the end of each block in ResNet34, we add a strip pooling layer [6] to make multi-scale features more robust.

3.3 Vision transformer encoder

The vision transformer encoder takes the original image and multi-scale feature maps as input and outputs a room layout feature vector. Fig. 2 shows the overview of the vision transformer encoder. The encoder first samples patches from the original image and the multi-scale feature maps. Then different patches are projected into different patch embeddings through different embedding operations. All the patch embeddings are attached with a recurrent position embedding to predict the room layout feature vector with a vision transformer backbone.

Patch sampling We designed two different sampling methods for the multi-scale feature map and the original image. For multi-scale feature maps, we sample vertical patches along its horizontal axis. Specifically, given a feature map with a dimension of $C \times W \times H$, we sample it as W patches of size $C \times 1 \times H$, where H is the height of the original feature map, W is the width of the original feature map, and C is the number of channels. Feature maps of other scales are sampled in the same way. For panoramas, we sample square patches uniformly across the entire image. Given a panoramic image of size $C_p \times W_p \times H_p$, we reshape it into N patches of $C_p \times P \times P$, where P is the size of the patch, and $N = W_p H_p/P^2$ is the number of patches.

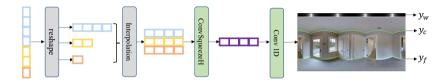


Fig. 3. The overview of the layout prediction module.

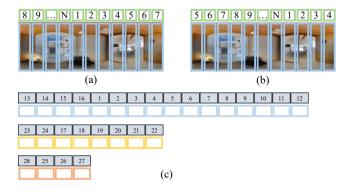


Fig. 4. The presentation of the recurrent position embedding. In each image, the bottom row shows patches and the top row shows the corresponding position embedding. Figure (a) and (b) present two position embedding with different start positions. Figure (c) shows the recurrent position embedding for multi-scale features. The colored rectangles in different rows in figure (c) represent patches sampled from feature maps of different scales.

Patch embedding We project different types of patches into patch embeddings through different embedding operations. Inspired by the one-dimensional features used in [13], we utilize a height compression module to calculate the patch embeddings of patches sampled from multi-scale feature maps. The height compression module is a convolutional layer containing C_{emb} filters with kernel size $1 \times H$ to cover the height of feature maps. For a patch with the dimension of $C \times 1 \times H$, the height compression module project it into a one-dimensional patch embedding with the size of $C_{emb} \times 1 \times 1$. Notice that C_{emb} does not change with the number of patch channels C, so that the one-dimensional patch embeddings have the same feature dimension C_{emb} for patches sampled from feature maps with different scales. For patches sampled from the panoramic image, we use a simple multilayer perceptron to calculate their patch embeddings. All patch embeddings are concatenated together as the input of the vision transformer encoder.

Recurrent position embedding Existing position embeddings are all designed for perspective images. In this section, we design a recurrent position embedding for panoramic images. A major difference between a perspective image and a

panoramic image is that the panoramic image is rolling stable, which means that the panoramic image rolling with any pixels along the horizon axis results in the same 3D layout compared to the original image. This property indicates that there is no need to emphasize the absolute starting position of patches along the horizontal axis. For instance, as shown in Fig. 4, the position embedding shown in (a) should have the same result compared with (b). To emphasize this attribute, the recurrent position embedding is equipped with a starting position randomly sampled along the horizontal axis. In particular, our recurrent position embedding is calculated as

$$RPE(pos, 2i) = sin\left(\frac{pos + randinit}{10000^{2i/d_{model}}}\right),$$

$$RPE(pos, 2i + 1) = cos\left(\frac{pos + randinit}{10000^{2i/d_{model}}}\right),$$
(1)

where d_{model} is the dimension of the position embedding, pos is the absolute position that starts from the left and ends at the right, and randinit is a random initial position.

For the multi-scale feature outputted from the ResNet, we compute the recurrent position embedding as following,

$$RPE^{k}(pos, 2i) = sin\left(\frac{pos + randinit + \sum_{k} s_{k}}{10000^{2i/d_{model}}}\right),$$

$$RPE^{k}(pos, 2i + 1) = cos\left(\frac{pos + randinit + \sum_{k} s_{k}}{10000^{2i/d_{model}}}\right),$$
(2)

where the RPE^k is the k-th position embedding of the patches sampled from the feature map of the k-th ResNet block, and s_k is the number of the patches sampled from the k-th feature map (according to our sampling method, s_k equals to the width of the feature map from the k-th ResNet block).

Vision transformer backbone We use the ViT[4] pre-trained on the ImageNet as our backbone of the vision transformer. Compared to ViT, two changes in the transformer encoder are that we delete the extend patch in ViT designed for object classification, and we combine the feature vector of all the patches sampled from the multi-scale feature maps as the outputted room layout feature vector, whose length equals to $\sum_{k=1}^{N} (k)$ where N is the number of the scales.

3.4 Layout prediction

The structure of the layout prediction module is illustrated in Fig. 3. We first reshape the room layout feature vector back to multi-scale features following the scale of the corresponding feature maps. The features with different scales are resized to a same length with the interpolation operation, and are combined together in the vertical dimension. Then we utilize the ConvSqueezeH layer introduced in [13] to compress the multi-scale features in the vertical dimension. Finally, we apply three Conv1D layers of kernel size 3, 3, and 1 respectively with BN, ReLU to output the probability y_w of existence of the wall-wall boundary, the ceiling-wall y_c and the floor-wall boundaries y_f .



Fig. 5. The overview of edge enhancement in frequency domain. We first compute the frequency transformation of a panoramic image with FFT. Then we design a filter according to the direction of the basis function to sample the frequency transformation and reconstruct the edge enhancement map with the iFFT.

3.5 Edge enhancement

The edge information is useful for the wall line detection, as shown in [23]. In this paper, we extract the edge information in the frequency domain. Compared to the LSD used in [23], our method is more efficient and can achieve better accuracy.

The edge enhancement module can be represented by

$$E = \mathcal{F}^{-1}(M\mathcal{F}(I)) \tag{3}$$

where I is a panorama, E is the edge enhanced image, M is a binary mask, \mathscr{F} is the Fast Fourier transformation, and \mathscr{F}^{-1} is the inverse fast Fourier transformation. Fig. 5 shows the pipeline of the edge enhancement in the frequency domain. We first compute the frequency transformation of a panoramic image P with the fast Fourier transformation \mathscr{F} . Then we sample a part of the frequency components with a mask M. Finally, we reconstruct the edge enhance map E with the inverse Fourier transformation \mathscr{F}^{-1} .

The key to edge enhancement is sampling different parts of the frequency transformation with the mask M. As is known, the high-frequency part contains the edge information while the low-frequency contains the mean content of the image. Therefore, a straightforward design for the mask M is a high pass filter, and its reconstruction result is shown in Fig. 6 (a). As shown in [23], separately extracting the edge along different axes (i.e. x,y,z) helps to improve the accuracy. To extract the edge along different axes in the frequency domain, we propose to design the mask following the direction of the trigonometric function in the frequency transformation. The direction of the trigonometric function in the frequency transformation at (u_i, v_i) can be calculated by

$$D_i = \arctan(u_i - u_0/v_i - v_0) \tag{4}$$

where (u_0, v_0) is the center of the frequency transformation. Trigonometric functions in different directions highlight the edges of the panorama in the same direction. Therefore, in order to obtain edges in different directions, we design two masks for edge enhancement in the horizontal and vertical directions. The calculation formulas are respectively

$$M_v(u_i, v_i) = \begin{cases} 1, & |D_i| < \alpha \& \sqrt{u_i^2 + v_i^2} > \theta \\ 0, & else \end{cases}$$
 (5)

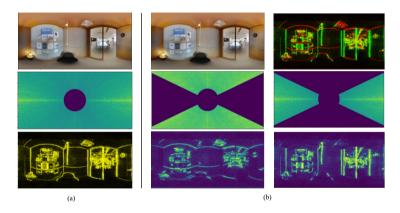


Fig. 6. The presentation of different kinds of sampling strategies. From the top to the bottom, figure (a) shows the panoramic image, the high-pass filter, and the reconstructed edge enhancement map. From the left to the right and the top to the bottom, figure (b) presents the panoramic image, the reconstructed edge enhancement map, the horizon mask M_h , the vertical mask M_v , the corresponding reconstructed edge enhancement map of M_v .

and

$$M_h(u_i, v_i) = \begin{cases} 1, & |D_i| > = \beta \& \sqrt{u_i^2 + v_i^2} > \theta \\ 0, & else \end{cases}$$
 (6)

where $\sqrt{u_i^2 + v_i^2} > \theta$ works as a high pass filter, and both α and β are hyperparameters. M_v is the mask to extract the horizon edge enhanced image E_v , so we only maintain the trigonometric function whose direction angle is less than α . The detailed information is shown in Fig. 6 (b). Finally, we concatenate two edge enhancement maps E_v and E_h with the original panoramic image P in the channel dimension.

3.6 Loss functions

The loss function consists of \mathcal{L}_{cor} and \mathcal{L}_{bon} , where \mathcal{L}_{cor} measures the error between the predicted position of the corners and the ground truths. We use the binary cross-entropy to define \mathcal{L}_{cor} .

 \mathcal{L}_{bon} can be directly defined upon the errors of our two predicted ceiling and floor boundaries (i.e., y_f and y_c) with respect to the ground truth. Previous works [12,13] always define this error in the equirectangular image space, which does not consider the geometry information in the panorama image. In this paper, we design a panoramic geometry-aware loss in 3D space (i.e. 3D loss) inspired by RenderLoss in L2DNet [15]. In particular, we represent a pixel q on

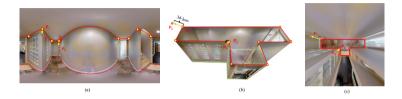


Fig. 7. An illustration of the 3D loss. Figure (a) shows the ground truth boundary in red lines, a couple of the points \overline{P}_1 and \overline{P}_2 on the ground truth boundary, and the corresponding predicted points P_1 and P_2 . In figure (a), points P_1 and P_2 have the same L1 error in the equirectangular image space, i.e., $\Delta(\overline{P}_1 - P_1) = \Delta(\overline{P}_2 - P_2)$. However, as illustrated figure (b) and figure (c), in the 3D space, $\Delta(\overline{P}_1 - P_1) \neq \Delta(\overline{P}_2 - P_2)$. Compared to L1 loss in the equirectangular image space, our 3D loss can deal with the unbalance errors caused by the panoramic geometry.

the equirectangular image with its longitude δ and latitude γ , i.e., $q = (\delta, \gamma)$. Then we convert the pixel q to a 3D point $P(p_x, p_y, p_z)$ in the 3D space by

$$P(p_x, p_y, p_z) = \mathcal{F}(\delta, \gamma),$$

$$p_x = s \cdot \cos(\gamma) \cdot \sin(\delta),$$

$$p_y = s \cdot \sin(\gamma),$$

$$p_z = s \cdot \cos(\delta) \cdot \cos(\gamma),$$
(7)

where \mathcal{F} denotes the convert function, $\delta \in [-\pi,\pi]$, $\gamma \in [-\pi/2,\pi/2]$, and s denotes the scale which can be calculated by the predefined camera height. The 3D loss can be expressed as the L_1 error between the 3D point P sampled from the boundary (i.e., y_f and y_c) and the corresponding 3D point \hat{P} sampled from the ground truth. An illustration of 3D loss is shown in Fig. 7.

4 Experiments

In this section, we test the performance of our method on different datasets and compare it with the state-of-the-art baselines. We also provide several ablation studies to analyze the effectiveness of different modules in our model.

4.1 Dataset

We conduct the experiments on two different datasets, including the PanoContext and the Matterport3D (Mp3D) dataset. PanoContext dataset contains around 500 panoramas along with ground truth layout annotations. The Matterport3D dataset contains 2,295 panoramas of the complex cases with different numbers of layout corners. We adopt the official train/val/test split in [24].

4.2 Training Setup

Our model is trained end-to-end with the Adam [7] optimizer, where the learning rate is 0.0001. The hyperparameters α , β and θ are set to 20, 25 and 100. The batch size is set to 8 and the epoch is set to 300. During the training stage, we apply the same data augmentations following [23], including flip, rotation, gamma transformation, and stretch. In addition, we design a new data augmentation method, adding 50 random masks to the panoramic image, each of which is a 50×50 rectangle with a value of 0.

Table 1. The results on the MP3D dataset. Our method achieves best results on both 2D IoU and 3D IoU.

Dataset	Method	Metric	# of corners					
	Method		overall	4	6	8	10+	
	AtlantaNet[9]	2D IoU	82.09%	84.42%	83.85%	76.97%	73.18%	
	HorizonNet [12]		81.71%	84.67%	84.82%	73.91%	70.58%	
	HoHoNet [13]		82.52%	85.57%	84.89%	75.72%	70.88%	
	LED2 [15]		83.91%	86.91%	85.53%	78.72%	71.79%	
Mp3D	Ours		$\pmb{84.25\%}$	$\pmb{86.92\%}$	86.71%	78.07%	73.88%	
Mp3D	AtlantaNet[9]		80.02%	82.09%	82.08%	75.19%	71.61%	
	HorizonNet [12]	3D IoU	79.11%	81.88%	82.26%	71.78%	68.32%	
	HoHoNet [13]		80.08%	82.90%	82.28%	73.85%	69.17%	
	LED2 [15]		81.52%	84.22%	83.22%	$\pmb{76.89\%}$	70.09%	
	Ours		$\boldsymbol{82.04\%}$	84.40%	$\pmb{84.67\%}$	76.34%	$\textcolor{red}{\mathbf{72.32\%}}$	

4.3 Results

Evaluation metrics We evaluate our model with two standard metrics: 3D IoU and 2D IoU. The 3D IoU measures the intersection over union between the predicted 3D layout and the ground truth. The 2D IoU measures the intersection over the union between the predicted 2D layout and the ground truth.

Table 2. The results on the PanoContext. Our method achieves the best result on PanoContext dataset.

Dataset	Model	3D IoU (%)		
	AtlantaNet[9]	78.76		
	DuLaNet [18]	77.42		
	LayoutNet [23]	74.48		
PanoContext	HorizonNet [12]	82.17		
	HoHoNet [13]	82.82		
	LED2 [15]	82.75		
	Ours	83.88		



Fig. 8. The visualization of the reconstruction result on the Matterport3D dataset.

Quantitative results We exam our model on two datasets and compare it with popular 360 room layout estimation methods, including HorizonNet [12], Ho-HoNet [13], AtlantaNet [9], and LED2Net [15]. Table 1 and Table 2 separately present the reconstruction results on two datasets. Overall, our method achieves the best results on Mp3D and PanoContext datasets. Compared with the HorizonNet and HoHoNet methods that also use one-dimensional layout representation, PanoViT achieves an improvement of 2% and 1% on the Mp3D and PanoContext datasets. This proves the effectiveness of the visual transformer on the room layout estimation task. Compared with PanoContext, the Mp3D dataset contains more complex room layouts. PanoViT has achieved the best results on the Mp3D dataset (especially in the 10+ corner room layout, PanoViT is 2% higher than LED2).

Quantitative results To better compare the performance of different methods, we also visualize some reconstruction results in Fig. 8. The results illustrate that PanoViT achieves a better result on the local details.

Table 3. The results of the PanoViT with different inputs. PanoViT w/4-scale uses feature maps of all four blocks from ResNet34, while PanoViT w/1-scale only uses feature maps of the final blocks and PanoViT w/0-scale ignore the feature maps from the ResNet.

Dataset	Model	Metric			
Dataset	Model	2D IoU	3D IoU		
	PanoViT w/4-scale				
	PanoViT w/1-scale				
	PanoViT w/0-scale	69.14%	65.32%		



Fig. 9. The visualization of reconstruction result of PanoViT with and without CNN multi-scale features. The top row presents the reconstruction result of PanoViT without CNN Multi-scale features. The bottom row presents the reconstruction result with CNN Multi-scale features. The PanoViT with CNN Multi-scale features achieves a better reconstruction result

4.4 Ablation Study

Multi-scale features Besides the patches sampled on the original panoramic image, the PanoViT also receives the patches sampled from the multi-scale feature maps. To exam the effect of the multi-scale feature maps, we test our model with patches sampled from the original image, the patches sampled from both the original image and the feature maps of the last blocks in ResNet34, and the patches sampled from both the original image and feature maps of all four blocks. All the experiments are conducted on the PanoContext dataset. Table 3 shows the results. We can see that using the multi-scale feature map can significantly improve the estimation accuracy.

3D loss, recurrent PE and edge enhancement Other ablation experiments are presented in Table 4. We report the result of our model equipped with different modules, including the 3D loss, the recurrent position embedding, and the edge enhancement module. All the experiments are conducted on the Mp3D dataset. Overall, Table 4 presents that all the modules are helpful for the estimation of the room layout.

Table 4. The ablation experiments of the PanoViT. The block with \checkmark means that the PanoViT contains the corresponding module.

ID	Posiumont DI	Frequency	3D Loss	metric	of corners (%)				
	rtecurrent i E				overall	4	6	8	10+
1					81.49	84.34	84.09	75.23	69.88
2			✓		81.53	84.01	83.94	76.14	71.01
3	✓		√	3D IoU	81.70	84.70	84.62	76.03	69.86
4		✓	√		81.77	84.21	84.11	76.16	71.93
5	√	√	√		82.04	84.40	84.67	76.34	72.32

Experiments 1 and 2 test the effectiveness of the 3D loss. The model in experiment 1 is equipped with 3D loss while the model in experiment 2 is equipped with L_1 loss. The results in Table 4 show that the model with 3D loss is 0.11% better than the model with L_1 loss.

Experiments 2 and 3, and experiments 4 and 5 show the effectiveness of the recurrent position embedding. The model in experiment 2 is equipped with the learnable position embedding same as [4], while the model in experiment 3 is equipped with recurrent position embedding. We observe that the model with recurrent position embedding achieves a better 3D IoU result. Experiments 4 and 5 further present the results about the recurrent position embedding when the model is equipped with the edge enhancement module. The model with recurrent position embedding achieves a better result.

Experiments 4 and 5 show the effectiveness of the recurrent position embedding. We test the model with Fourier edge enhancement and the model without Fourier edge enhancement. The results in Table 4 show that the model with edge enhancement shows better performance in all types of room layouts.

5 Conclusions

In this paper, we have proposed PanoViT to estimate the room layout from a single panoramic image. PanoViT successfully applied the transformer pretrained on perspective images to panoramic image processing. The patch sampling method and the recurrent position embedding in PanoViT are proved to be effective for panoramic image processing. The proposed frequency-domain edge enhancement further improves the prediction accuracy. The experimental results on two datasets validate that the proposed PanoViT can achieve better results.

References

- Brown, T.B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., et al.: Language models are few-shot learners. arXiv preprint arXiv:2005.14165 (2020)
- Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., Zagoruyko, S.: Endto-end object detection with transformers. In: European Conference on Computer Vision. pp. 213–229. Springer (2020)
- Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805 (2018)
- Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al.: An image is worth 16x16 words: Transformers for image recognition at scale. In: International Conference on Learning Representations (2020)
- Esser, P., Rombach, R., Ommer, B.: Taming transformers for high-resolution image synthesis. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 12873–12883 (2021)

- Hou, Q., Zhang, L., Cheng, M.M., Feng, J.: Strip pooling: Rethinking spatial pooling for scene parsing. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 4003–4012 (2020)
- 7. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)
- 8. Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., Guo, B.: Swin transformer: Hierarchical vision transformer using shifted windows. arXiv preprint arXiv:2103.14030 (2021)
- 9. Pintore, G., Agus, M., Gobbetti, E.: Atlantanet: Inferring the 3d indoor layout from a single 360 image beyond the manhattan world assumption. In: European Conference on Computer Vision. pp. 432–448. Springer (2020)
- Pintore, G., Mura, C., Ganovelli, F., Fuentes-Perez, L., Pajarola, R., Gobbetti, E.: State-of-the-art in automatic 3d reconstruction of structured indoor environments. In: Computer Graphics Forum. vol. 39, pp. 667–699. Wiley Online Library (2020)
- Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., Sutskever, I., et al.: Language models are unsupervised multitask learners. OpenAI blog 1(8), 9 (2019)
- Sun, C., Hsiao, C.W., Sun, M., Chen, H.T.: Horizonnet: Learning room layout with 1d representation and pano stretch data augmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 1047– 1056 (2019)
- 13. Sun, C., Sun, M., Chen, H.T.: Hohonet: 360 indoor holistic understanding with latent horizontal features. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 2573–2582 (2021)
- Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. In: Advances in neural information processing systems. pp. 5998–6008 (2017)
- 15. Wang, F.E., Yeh, Y.H., Sun, M., Chiu, W.C., Tsai, Y.H.: Led2-net: Monocular 360deg layout estimation via differentiable depth rendering. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 12956–12965 (2021)
- 16. Xu, J., Stenger, B., Kerola, T., Tung, T.: Pano2cad: Room layout from a single panorama image. In: 2017 IEEE winter conference on applications of computer vision (WACV). pp. 354–362. IEEE (2017)
- 17. Yang, H., Zhang, H.: Efficient 3d room shape recovery from a single panorama. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 5422–5430 (2016)
- Yang, S.T., Wang, F.E., Peng, C.H., Wonka, P., Sun, M., Chu, H.K.: Dula-net: A dual-projection network for estimating room layouts from a single rgb panorama. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 3363–3372 (2019)
- Yang, Y., Jin, S., Liu, R., Kang, S.B., Yu, J.: Automatic 3d indoor scene modeling from single panorama. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 3926–3934 (2018)
- Zhang, Y., Song, S., Tan, P., Xiao, J.: Panocontext: A whole-room 3d context model for panoramic scene understanding. In: European conference on computer vision. pp. 668–686. Springer (2014)
- Zheng, S., Lu, J., Zhao, H., Zhu, X., Luo, Z., Wang, Y., Fu, Y., Feng, J., Xiang, T., Torr, P.H., Zhang, L.: Rethinking semantic segmentation from a sequence-tosequence perspective with transformers. In: CVPR (2021)

- 22. Zhu, X., Su, W., Lu, L., Li, B., Wang, X., Dai, J.: Deformable detr: Deformable transformers for end-to-end object detection. In: International Conference on Learning Representations (2020)
- 23. Zou, C., Colburn, A., Shan, Q., Hoiem, D.: Layoutnet: Reconstructing the 3d room layout from a single rgb image. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 2051–2059 (2018)
- 24. Zou, C., Su, J.W., Peng, C.H., Colburn, A., Shan, Q., Wonka, P., Chu, H.K., Hoiem, D.: 3d manhattan room layout reconstruction from a single 360 image. arXiv preprint arXiv:1910.04099 (2019)