Indoor Scene Generation from a Collection of Semantic-Segmented Depth Images – Supplemental Material

Ming-Jia Yang*[1,2] Yu-Xiao Guo[2] Bin Zhou[1] Xin Tong[2]
[1]Beihang University [2]Microsoft Research Asia
{yangmingjia,zhoubin}@buaa.edu.cn {yuxgu,xtong}@microsoft.com

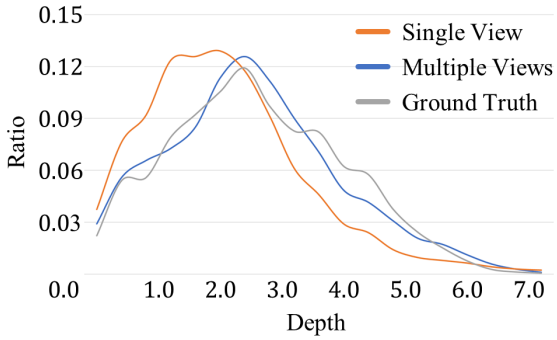

Figure 1. The distribution of the depth of all depth images of the scenes rendered from the ground truth scenes (grey curve) and the scenes generated by GAN with single-view discriminator (orange curve) and our multi-view discriminator (blue curve).

1. The Failure of Single-View Discriminator in 3D Scene Generation

We conduct a preliminary experiment and analysis to investigate the reason behind the failure of the single-view discriminator [3, 2] in our scene generation task.

To this end, we train our volumetric GAN model with the Structured3D-Bedroom dataset. For each scene in the dataset, we set the camera at the room center and 1.5m height to the floor and render four depth images by aligning the camera view to the $+X$, $-X$, $+Y$, $-Y$ directions. We then take these rendered images as training data for learning our volumetric GAN model with a single-view discriminator and our multi-view discriminator separately. After that, we generate 5,000 scenes for each network and render the depth images with the same camera setup as the training images. For depth images of the ground truth scenes and the scenes generated by each network, we plot the distributions of the depth values of all images in Fig. 1. We find that different from the 3D object modeling task in [3, 2] where the objects in the images have similar distances to the viewpoints of images, the objects can be anywhere in a room so that the distances between the viewpoint and objects in the scene have larger variations than the view distances between the camera and 3D objects. For images with such large depth variations, the differentiable ray consistency (DRC) layer used in our network tends to drive the generator to create scenes with more objects closer to the viewpoint with the single-view discriminator (the orange curve in Fig. 1). Although the generated scene is different from the GT, their rendering matches some views in the training dataset and thus could pass the discriminator and results in the model collapse. With the joint-views used in our multi-view discriminator, the model collapse caused by the single-view discriminator could be avoided and the depth distribution of the generated scene (the blue curve in Fig. 1) matches the depth distribution of the GT scenes (the grey curve in Fig. 1) better. One possible reason is that among all random combination of training images, the percentage of combinations in which all images in the combination are close-up views becomes much fewer.

*This work is done when Ming-Jia Yang was an intern at MSRA

Dataset	Selected categories
Structured3D-Bedroom	Cabinet, Bed, Chair, Picture, Desk, Curtain, Television, Night stand, Lamp
Structured3D-Livingroom	Cabinet, Chair, Sofa, Table, Picture, Shelves, Curtain, Lamp, Pillow, Refrigerator, Television
Structured3D-Kitchen	Cabinet, Picture, Curtain, Refrigerator, Lamp,
Matterport3D-Bedroom	Bed, Pillow, Night stand, Chair, Picture, Lamp, Curtain, Table
NYUv2-RGB-Bedroom	Cabinet, Bed, Chair, Desk, Shelves, Curtain, Pillow, Television, Nigh Stand, Lamp

Table 1. The object classes in each training dataset.

2. More Experimental Results

Details of Object Classes Table. 1 lists the list of the object classes in each training dataset.

The Complete Co-Occurrence Maps Fig. 2 illustrates the object co-occurrence maps of the GT scenes, the scenes

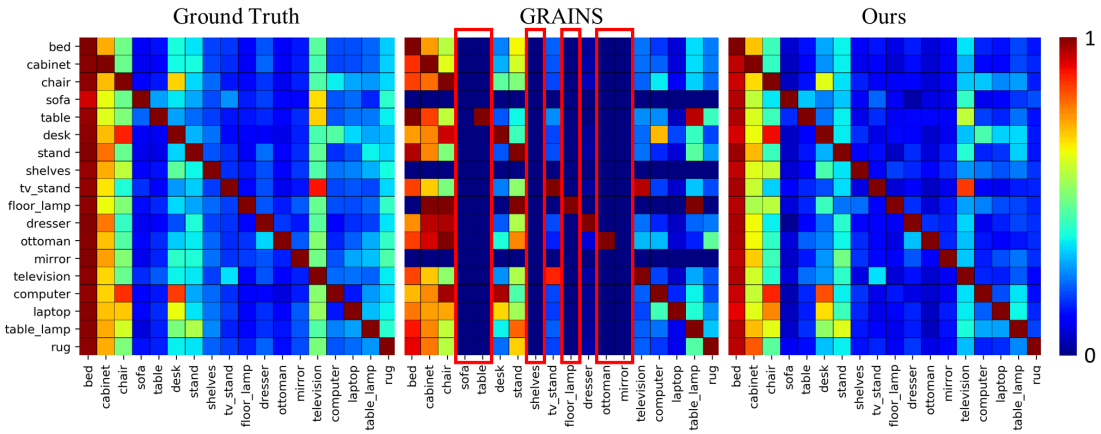

Figure 2. The object co-occurrence maps of all object classes of the ground truth scenes, scenes generated by GRAINS[1], and scenes generated by our method.

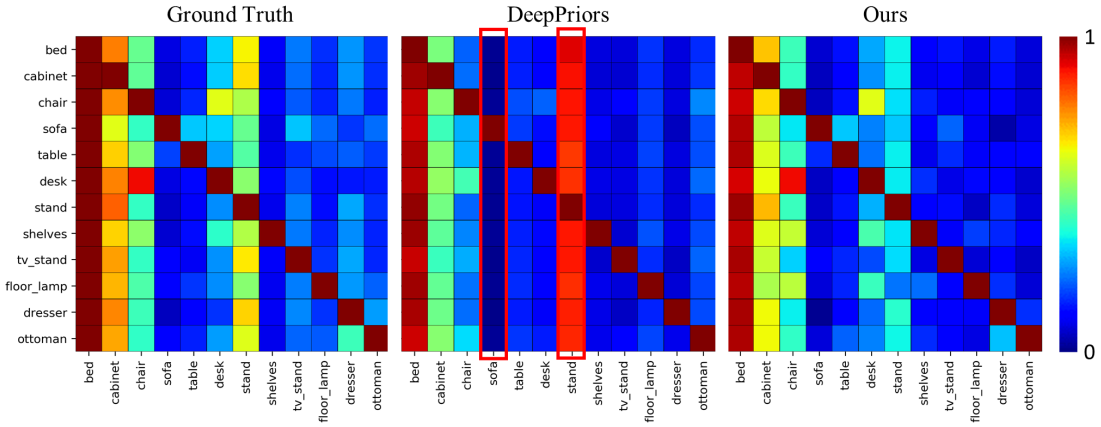

Figure 3. The object co-occurrence maps of all object classes of the ground truth scenes, scenes generated by GRAINS[1], and scenes generated by our method.

generated by our method, and the scenes generated by GRAINS [1] for all object classes in the scene. Fig. 3 visualizes the object co-occurrence maps of the GT scenes, the scenes generated by DeepPrior[4], as well as scenes generated by our method for all object classes in the scene. These two figures are a complete version of the co-occurrence maps shown in the paper.

The User Interface Fig. 4 and Fig. 5 displays user interface used in our user study and the user interface used in our ablation study.

More Visual Results Here we show more images of 3D scenes generated by our method from different training datasets, including Structured3D-Bedroom (Fig. 6), Structured3D-Livingroom and Structured3D-Kitchen (Fig. 7), Matterport3D-Bedroom and NYUv2-RGB-Bedroom (Fig. 8).

References

[1] Manyi Li, Akshay Gadi Patil, Kai Xu, Siddhartha Chaudhuri, Owais Khan, Ariel Shamir, Changhe Tu, Baoquan Chen, Daniel Cohen-Or, and Hao Zhang. Grains: Generative recursive autoencoders for indoor scenes. *ACM Trans. Graph.*, 38(2):1–16, 2019. 2

[2] Xiao Li, Yue Dong, Pieter Peers, and Xin Tong. Synthesizing 3d shapes from silhouette image collections using multi-projection generative adversarial networks. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 5535–5544, 2019. 1

[3] Thu Nguyen-Phuoc, Chuan Li, Lucas Theis, Christian Richardt, and Yong-Liang Yang. Hologan: Unsupervised learning of 3d representations from natural images. In *Int. Conf. Comput. Vis.*, pages 7588–7597, 2019. 1

[4] Kai Wang, Manolis Savva, Angel X Chang, and Daniel Ritchie. Deep convolutional priors for indoor scene synthesis. *ACM Trans. Graph.*, 37(4):1–14, 2018. 2

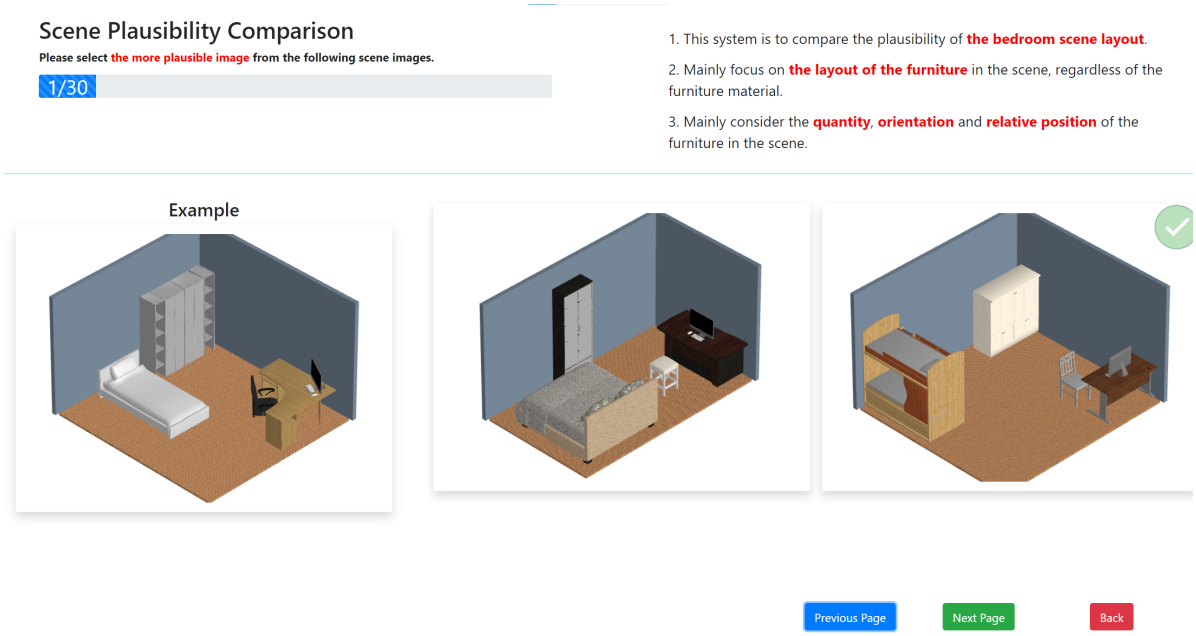

Figure 4. The user interface of our program used in the user study.

Figure 5. The user interface of our program used in the ablation study.

Figure 6. More results generated by our method from the Structured3D-Bedroom dataset.

Figure 7. More results generated by our method from Structured3D-LivingRoom (the first three rows) and Structured3D-Kitchen (the last three rows).

Figure 8. More results generated by our method from Matterport3D-Bedroom (the first three rows) and NYUv2-RGB-Bedroom (the last three rows).