Total3DUnderstanding: Joint Layout, Object Pose and Mesh Reconstruction for Indoor Scenes from a Single Image Supplementary Material

Yinyu Nie[1,2,3,†], Xiaoguang Han[2,3,*], Shihui Guo[4], Yujian Zheng[2,3], Jian Chang[1], Jian Jun Zhang[1]
[1]National Centre for Computer Animation, Bournemouth University
[2]The Chinese University of Hong Kong, Shenzhen
[3]Shenzhen Research Institute of Big Data [4]Xiamen University

The supplementary material contains:

- Camera and world system configuration.
- Network architecture, parameter setting and training strategies.
- 3D detection results on SUN RGB-D.
- Object class mapping from NYU-37 to Pix3D.
- More qualitative comparisons on Pix3D.
- More reconstruction samples on SUN RGB-D.

A. Camera and World System Setting

We build the world and the camera systems in this paper as Figure 1 shows. The two systems share the same center. The y-axis indicates the vertical direction perpendicular to the floor. We rotate the world system around its y-axis to align the x-axis toward the forward direction of the camera, such that the camera's yaw angle can be removed. Then the camera pose relative to the world system can be expressed by the angles of pitch β and roll γ:

$$\mathbf{R}(\beta,\gamma) = \begin{bmatrix} \cos(\beta) & -\cos(\gamma)\sin(\beta) & \sin(\beta)\sin(\gamma) \\ \sin(\beta) & \cos(\beta)\cos(\gamma) & -\cos(\beta)\sin(\gamma) \\ 0 & \sin(\gamma) & \cos(\gamma) \end{bmatrix}.$$

B. Network Architecture

Architecture. We present the architecture of our Object Detection Network (ODN), Layout Estimation Network (LEN) and Mesh Generation Network (MGN) in Table 1-3.

Training strategy. Our training involves two phases. We first train the three networks individually. ODN and LEN are trained on SUN RGB-D [6], while MGN is trained on Pix3D [7] with their specific loss ($\sum \lambda_x \mathcal{L}_x$, $\sum \lambda_y \mathcal{L}_y$ and $\sum \lambda_z \mathcal{L}_z$ respectively) (see Line 455, Page 5). All of them are with the batch size of 32 and learning rate at 1e-3 (scaled by 0.5 for every 20 epochs, 100 epochs in total). The MGN is trained with a progressive manner following [5]. Afterwards, we fine-tune them with the joint losses $\lambda_{co}\mathcal{L}_{co}$ and $\lambda_g \mathcal{L}_g$ (see Equation 4) together on SUN RGB-D. Specifically, in the joint training, we randomly blend a few Pix3D samples into each batch of SUN RGB-D data to supervise the mesh generation network (i.e. to optimize the mesh loss $\sum \lambda_z \mathcal{L}_z$). We do so to regularize the mesh generation network because not like Pix3D, SUN RGB-D provides only a partial point-cloud scan of objects, which is not sufficient to supervise full mesh generation. For joint training, we input the full network with a hierarchical batch, where the scene image (from SUN RGB-D) is inputted to LEN, and the object images (from SUN RGB-D and Pix3D) are fed into ODN and MGN for object detection and mesh prediction. We set the hierarchical batch size at 1, and the learning rate at 1e-4 (scaled by 0.5 for every 5 epochs, 20 epochs in total). All the training tasks are implemented on 6x Nvidia 2080Ti GPUs. During testing, our network requires 1.2 seconds on average to predict a scene mesh on a single GPU.

Parameters. We set the threshold in our MGN at 0.2.

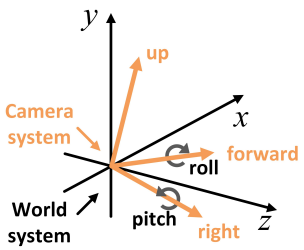

Figure 1: Camera and world systems

† Work done during visiting Shenzhen Research Institute of Big Data.
* Corresponding author: hanxiaoguang@cuhk.edu.cn

Edges with the classification score below it are removed. In joint training (Section 3.3), we let $\lambda_r = 10$, $\lambda_x = 1, \forall x \in \{\boldsymbol{\delta}, d, \boldsymbol{s}, \theta\}$, $\lambda_y = 1, \forall y \in \{\beta, \gamma, \boldsymbol{C}, \boldsymbol{s}^l, \theta^l\}$, $\lambda_c = 100$, $\lambda_e = 10$, $\lambda_b = 50$, $\lambda_{ce} = 0.01$, $\lambda_{co} = 10$, $\lambda_g = 100$.

Index	Inputs	Operation	Output shape
(1)	Input	Object images in a scene	Nx3x256x256
(2)	Input	Geometry features [3, 8]	N x N x 64
(3)	(1)	ResNet-34 [2]	Nx2048
(4)	(2), (3)	Relation Module [3]	Nx2048
(5)	(3), (4)	Element-wise sum	Nx2048
(6)	(5)	FC(128-d)+ReLU+Dropout+FC	$\boldsymbol{\delta}$
(7)	(5)	FC(128-d)+ReLU+Dropout+FC	d
(8)	(5)	FC(128-d)+ReLU+Dropout+FC	θ
(9)	(5)	FC(128-d)+ReLU+Dropout+FC	$\boldsymbol{s}$

Table 1: Architecture of Object Detection Network. It takes all object detections in a scene as input and outputs their projection offset $\boldsymbol{\delta}$, distance d, orientation θ and size $\boldsymbol{s}$. N is the number of objects in a scene.

Index	Inputs	Operation	Output shape
(1)	Input	Scene image	3x256x256
(2)	(1)	ResNet-34 [2]	2048
(3)	(2)	FC(1024-d)+ReLU+Dropout+FC	β
(4)	(2)	FC(1024-d)+ReLU+Dropout+FC	γ
(5)	(2)	FC+ReLU+Dropout	2048
(6)	(5)	FC(1024-d)+ReLU+Dropout+FC	$\boldsymbol{C}$
(7)	(5)	FC(1024-d)+ReLU+Dropout+FC	$\boldsymbol{s}^l$
(8)	(5)	FC(1024-d)+ReLU+Dropout+FC	θ^l

Table 2: Architecture of Layout Estimation Network. LEN takes the full scene image as input and produces the camera pitch β and roll γ angles, the 3D layout center $\boldsymbol{C}$, size $\boldsymbol{s}$ and orientation θ in the world system.

C. 3D Detection on SUN RGB-D

We report the full results of 3D object detection on SUN RGB-D in Table 5.

D. Object Class Mapping

Pix3D has nine object categories for mesh reconstruction, which contains: 1. bed, 2. bookcase, 3. chair, 4. desk, 5. sofa, 6. table, 7. tool, 8. wardrobe, 9. miscellaneous. In 3D object detection, we obtain object bounding boxes with NYU-37 labels in SUN RGB-D. As our MGN is pretrained on Pix3D, and the object class code is required as an input for mesh deformation, we manually map the NYU-37 labels to Pix3D labels based on topology similarity for scene reconstruction (see Table 4).

Index	Inputs	Operation	Output shape
(1)	Input	Object image	3x256x256
(2)	Input	Object class code	d_c
(3)	Input	Template Sphere	3x2562
(4)	(1)	ResNet-18 [2]	1024
(5)	(2),(4)	Concatenate	1024+d_c
(6)	(5)	Repeat	(1024+d_c)x2562
(7)	(3),(6)	Concatenate	(1024+d_c+3)x2562
(8)	(7)	AtlasNet decoder [1]	3x2562
(9)	(3),(8)	Element-wise sum	3x2562
(10)	(9)	Sample points	3xN_e
(11)	(5)	Repeat	(1024+d_c)xN_e
(12)	(10),(11)	Concatenate	(1024+d_c+3)xN_e
(13)	(12)	Edge classifier	1xN_e
(14)	(13)	Threshold	1xN_e (Mesh topology)
(15)	(6),(9)	Concatenate	(1024+d_c+3)x2562
(16)	(15)	AtlasNet decoder [1]	3x2562
(17)	(9),(16)	Element-wise sum	3x2562 (Mesh points)

Table 3: Architecture of Mesh Generation Network. Note that d_c denotes the number of object categories, and N_e represents the number of points sampled on edges. The edge classifier has the same architecture with AtlasNet decoder, where the last layer is replaced with a fully connected layer for classification.

cabinet	bed	chair	sofa	table	door	window
8	1	3	5	6	8	9
bookshelf	picture	counter	blinds	desk	shelves	curtain
2	9	9	9	4	2	9
dresser	pillow	mirror	floor mat	clothes	books	fridge
8	9	9	9	9	9	8
tv	paper	towel	shower curtain	box	whiteboard	person
8	9	9	9	8	8	9
nightstand	toilet	sink	lamp	bathtub	bag	wall
8	9	9	9	9	8	-
floor	ceiling	-	-	-	-	-
-	-	-	-	-	-	-

Table 4: Object class mapping from NYU-37 to Pix3D

E. More Comparisons of Object Mesh Reconstruction on Pix3D

More qualitative comparisons with Topology Modification Network (TMN) [5] are shown in Figure 2. The threshold τ in TMN is set at 0.1 to be consistent with their paper.

F. More Samples of Scene Reconstruction on SUN RGB-D

We list more reconstruction samples from the testing set of SUN RGB-D in Figure 3.

Method	cabinet	bed	chair	sofa	table	door	window	bookshelf	picture	counter
CooP [4]**	10.47	57.71	15.21	36.67	31.16	0.14	0.00	3.81	0.00	27.67
Ours (w/o. joint)	11.39	59.03	15.98	43.95	35.28	0.36	0.16	**5.26**	0.24	**33.51**
Ours (joint)	**14.51**	**60.65**	**17.55**	**44.90**	**36.48**	**0.69**	**0.62**	4.93	**0.37**	32.08
Method	blinds	desk	shelves	curtain	dresser	pillow	mirror	floor mat	clothes	books
CooP [4]**	**2.27**	19.90	2.96	1.35	15.98	2.53	**0.47**	-	0.00	**3.19**
Ours (w/o. joint)	0.00	23.65	**4.96**	2.68	19.20	2.99	0.19	-	0.00	1.30
Ours (joint)	0.00	**27.93**	3.70	**3.04**	**21.19**	**4.46**	0.29	-	0.00	2.02
Method	fridge	tv	paper	towel	shower curtain	box	whiteboard	person	nightstand	toilet
CooP [4]**	21.50	5.20	0.20	2.14	20.00	**2.59**	0.16	20.96	11.36	42.53
Ours (w/o. joint)	20.68	4.44	0.41	**2.20**	20.00	2.25	0.43	23.36	6.87	**48.37**
Ours (joint)	**24.42**	**5.60**	**0.97**	2.07	20.00	2.46	**0.61**	**31.29**	**17.01**	44.24
Method	sink	lamp	bathtub	bag	wall	floor	ceiling			
CooP [4]**	15.95	3.28	24.71	1.53	-	-	-			
Ours (w/o. joint)	14.40	3.46	**27.85**	2.27	-	-	-			
Ours (joint)	**18.50**	**5.04**	21.15	**2.47**	-	-	-			

Table 5: Comparison of 3D object detection. We compare the average precision (AP) of detected objects on SUN RGB-D (higher is better). CooP [4]** presents the model trained on the NYU-37 object labels for a fair comparison.

Figure 2: Qualitative comparisons between the proposed method and TMN [5] on object mesh reconstruction. From left to right: input images, results from TMN, and our results.

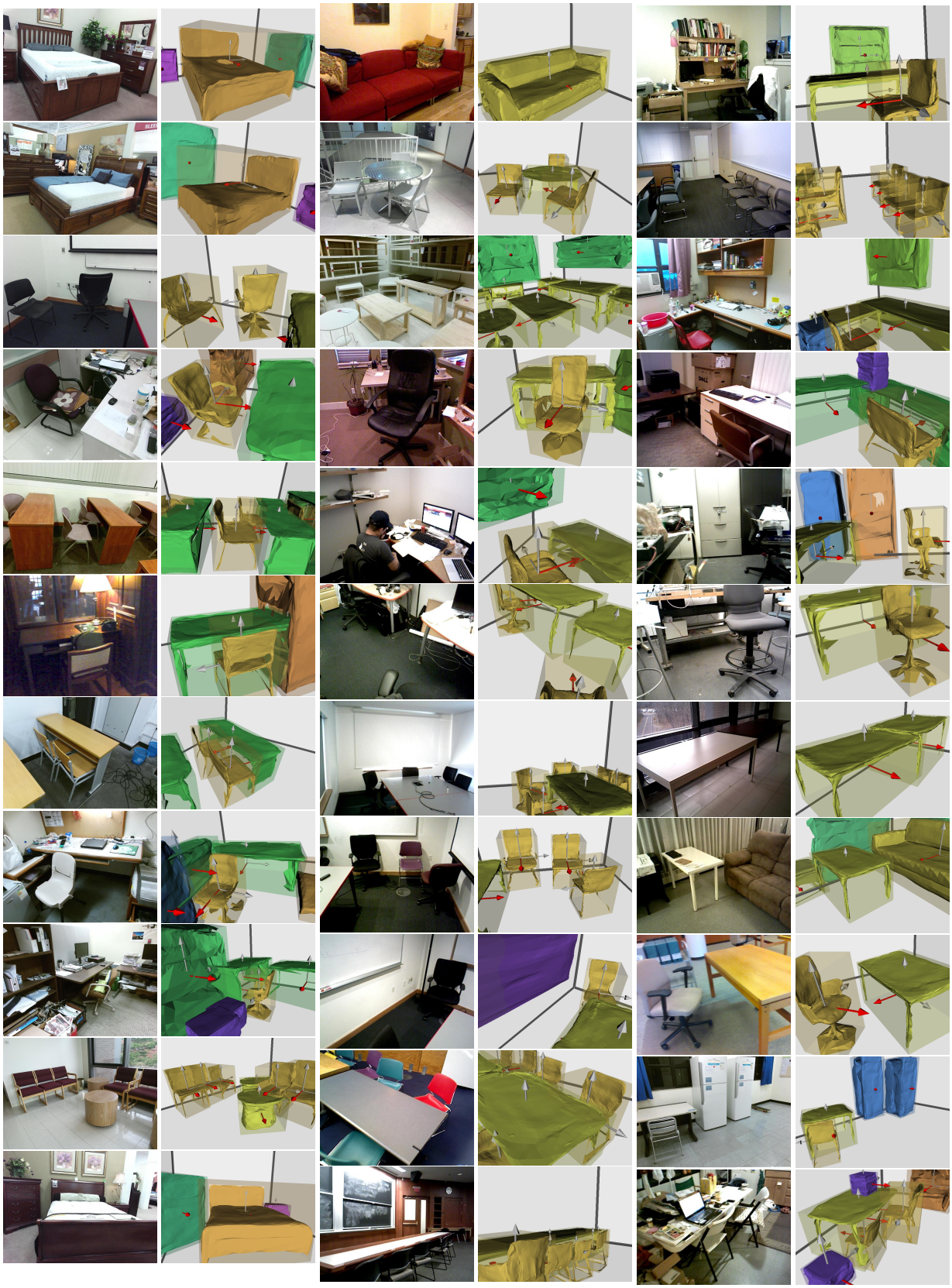

Figure 3: Reconstruction results of test samples on SUN RGB-D

References

[1] Thibault Groueix, Matthew Fisher, Vladimir G. Kim, Bryan Russell, and Mathieu Aubry. AtlasNet: A Papier-Mâché Approach to Learning 3D Surface Generation. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 2

[2] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 2

[3] Han Hu, Jiayuan Gu, Zheng Zhang, Jifeng Dai, and Yichen Wei. Relation networks for object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3588–3597, 2018. 2

[4] Siyuan Huang, Siyuan Qi, Yinxue Xiao, Yixin Zhu, Ying Nian Wu, and Song-Chun Zhu. Cooperative holistic scene understanding: Unifying 3d object, layout, and camera pose estimation. In *Advances in Neural Information Processing Systems*, pages 207–218, 2018. 3

[5] Junyi Pan, Xiaoguang Han, Weikai Chen, Jiapeng Tang, and Kui Jia. Deep mesh reconstruction from single rgb images via topology modification networks. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 9964–9973, 2019. 1, 2, 3

[6] Shuran Song, Samuel P Lichtenberg, and Jianxiong Xiao. Sun rgb-d: A rgb-d scene understanding benchmark suite. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 567–576, 2015. 1

[7] Xingyuan Sun, Jiajun Wu, Xiuming Zhang, Zhoutong Zhang, Chengkai Zhang, Tianfan Xue, Joshua B Tenenbaum, and William T Freeman. Pix3d: Dataset and methods for single-image 3d shape modeling. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2974–2983, 2018. 1

[8] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017. 2