Supplementary Material
for
Semantic Scene Completion via Integrating Instances and Scene in-the-Loop

1. Introduction

This supplementary material presents: (1) details of three datasets; (2) the detail description of training/inference strategies and network architecture for instance and scene completion; (3) more quantitative and qualitative ablation studies; (4) visualization results.

2. Dataset Details

In this section, we will discuss the detailed information of the three datasets we explored in the main paper.

Volumentic SSC data generation. For the whole scene, we follow the original semantic scene completion dataset preparation [9] to rotate the scene to align with gravity and room orientation based on Manhattan assumption. The size of the whole room is $4.8m$ length, $2.88m$ height and $4.8m$ width and the scene is voxelized into $240 \times 144 \times 240$ volume with a grid size of $0.02m$ and a truncation value of $0.24m$. For the saving of computational cost, we follow [9, 12, 5, 4, 3] to downsample the ground truth by a rate of 4 and the volume size is $60 \times 36 \times 60$, and we also follow [6, 1, 4] to down sample the input volume into the size of $60 \times 36 \times 60$.

Instance-level data generation. For the instances in the scene, as mentioned in the main paper, the goal of proposal generation module in the scene-to-instance completion stage is to provide high quality instance proposals for the follow up instance reconstruction. Thus, 3d bounding box/proposal labels are required as supervision. Instead of using expensive manually annotated, We exploit *max-connected-region* to generate ground truth 3D bounding boxes *automatically*. More specifically, since the three datasets provide voxel-wise semantic labels, the adjacent voxels that have the same semantic usually belong to the same object. Therefore, starting at one voxel, we can get the *max-connected-region* and all the voxels in this region belong to one object. The smallest envelope axis-aligned box that closes voxels of the *max-connected-region* is labeled as ground truth 3D bounding box. Although such gratis labeling of 3d bounding boxes are coarse without precise size and orientation, it could provide sufficient information, including localization and completion space, for helping distinguishing nearby objects and constraining instance's shape in our shape completion module.

3. Implementation Details

In this section, we provide detailed description of the proposed instance completion and scene completion.

3.1. Details of Instance Completion

As mentioned in the main paper, instance completion includes a proposal generation module and a shape completion module. The architecture and parameter details are shown in Table 1 and Table 2.

As illustrated in Table 1, the proposal generation module groups and extracts location and semantic features of input points by $sa1_0$ and $sa1_1$, respectively. Then we element-wise add the two features and sequentially feed them to $sa2$, $sa3$ and $sa4$ that followed by two FP layers, *i.e.*, $fp1$ and $fp2$, to propagate features among different points. With the powerful points features, three offset and proposal blocks are exploited to predict 3d bounding boxes/proposals, which is supervised by our generated ground truth boxes.

To preserve the structural and context of instances as complete as possible, we follow [11] introduce 3D grids as intermediate representations during the reconstruction process, just as mentioned in main paper Section 3.3. As illustrated in Table 2, our semantic encoder utilize two simple fully connected layers to encode instance-level semantic, which provides shape prior for better convergence and complete shape. Furthermore, our geometry encoder extracts geometry feature by exploiting points relationship which is established in $32 \times 32 \times 32$ grid. Consequently, we concatenate geometry and instance semantic features and feed the enhanced features to our decoder which consists of three 3D deconv layers, to obtain our initial results. To further improve local details of instances with complex shapes, such as chairs, we take a concatenation of downsampled point set and corresponding 3D grid features as input to an MLP, consisting of two fully connected layer, to learned accu-

Layer name	Input	Type	Output size
sa_1_0	location of point cloud	SA	(2048, 3+128)
sa_1_1	semantic vector of point cloud	SA	(2048, 3+128)
sa_2	sa_1_0, sa_1_1	SA	(1024,3+256)
sa_3	sa_2	SA	(512, 3+256)
sa_4	sa_3	SA	(256, 3+256)
fp_1	sa_3, sa_4	FP	(512, 3+256)
fp_2	sa_2, sa_3	FP	(1024, 3+256)
offset_1	fp_2	CBR	1024, 256)
offset_2	offset_1	CBR	(1024, 256)
offset_3	offset_2	Conv	(1024, 3+256)
proposal_1	offset_3	CBR	(256, 128)
proposal_2	proposal_1	CBR	(256, 128
proposal_3	proposal_2	Conv	(256, 62)

Table 1. Architecture details of proposal generation. *SA* and *FP* represent the set abstraction layer and feature propagation layer, respectively. *CBR* denotes a convolution block consists of a 1×1 convolution layer followed by a batchnorm and Relu function while *Conv* means a single 1×1 convolution layer.

	Layer name	Input	Output size
Semantic Encoder	FC_1	Semantic Vector	(16,1)
	FC_2	FC_1	(64,1)
	Reshape_1	FC_2	$(1, 4^3)$
Geometry Encoder	Griding	Partial Point Cloud	$(1, 32^3)$
	Conv3D_1	Griding	$(16, 16^3)$
	Conv3D_2	Conv3D_1	$(32, 8^3)$
	Conv3D_3	Conv3D_2	$(32, 4^3)$
	Concat_1	Conv3D_3, FC_2	$(33, 4^3)$
	Conv3D_4	Concat_1	$(32, 4^3)$
Decoder	Deconv3D_1	Conv3D_4	$(32, 8^3)$
	Deconv3D_2	Deconv3D_1, Conv3D_2	$(32, 16^3)$
	Deconv3D_3	Deconv3D_2, Conv3D_1	$(32, 32^3)$
Feature-Point DownSampling(DS)	Rev-Griding	Deconv3D_3	$(1, 32^3)$
	Point_DS	Rev-Griding, Partial Point Cloud	(3,1024)
	Feature_DS_1	Point_DS, Conv3D_2	(128,1024)
	Feature_DS_2	Point_DS, Conv3D_1	(256,1024)
	Concat_2	Feature_DS_1, Feature_DS_2	(384,1024)
MLP	FC_3	Concat_2	(96,1024)
	FC_4	FC_3	(24,1024)
	Reshape_2	FC_4	(3,8192)
	Output	Tile(Point_DS), Reshape_2	(3, 8192)

Table 2. Architecture details of shape completion. Tile denotes a operation that replicate the input $(3, m)$ n times and return a new point set with the size of $(3, m \times n)$. We use the Griding, Rev-Griding and downsampling operations proposed in [11]

rate residual offset between initial shape results and ground truth. Thus we obtain more complete and refined instance shape with fine-grained details. For practice, we set M = 2×10^4, M' = 1024, N_p = 2048, N_R = 8192 and C' = 256 for feature learning of C = 12 classes and K = 16, σ = 0.2, and β = 0.75 for proposal selection while use H = W = D = 32 for the 3D grid.

3.2. Details of Scene Completion

Table 3 illustrates the details of our scene completion. We encode semantic information volume V_S and completion information volume V_T, respectively. Then we combine the semantic and completion features and feed them to the follow-up encoder blocks, *i.e.*, $s1$ and $s2$. Then two 3D deconv are exploited to recover the semantic labels to the

	Layer name	Input	Type	Output size
Semantic Encoder	Conv3D_1_0	Semantic Volume V_S	CBR	(3,60,36,60)
	Conv3D_1_1	Conv3D_1_0	CBR	(64,60,36,60)
	Conv3D_1_2	Conv3D_1_1	CBR	(128,60,36,60)
Geometry Encoder	Conv3D_2_0	TSDF Volume V_T	CBR	(3,60,36,60)
	Conv3D_2_1	Conv3D_2_0	CBR	(64,60,36,60)
	Conv3D_2_2	Conv3D_2_1	CBR	(128,60,36,60)
	Add_1	Conv3D_1_2, Conv3D_2_2	-	(128,60,36,60)
Encoder	DDR_0	Add_1	DDR	(128,30,18,30)
	DDR_1	DDR_0	DDR	(128,30,18,30)
	DDR_2	DDR_1	DDR	(128,30,18,30)
	DDR_3	DDR_2	DDR	(128,30,18,30)
	Add_2	DDR_3, Down(Add_1)	-	(128,30,18,30)
	DDR_4	Add_2	DDR	(256,15,9,15)
	DDR_5	DDR_4	DDR	(256,15,9,15)
	DDR_6	DDR_5	DDR	(256,15,9,15)
	DDR_7	DDR_6	DDR	(256,15,9,15)
Decoder	Deconv3D_1	DDR_7	DBR	(128,30,18,30)
	Add_3	Deconv3D_1, Add_2	-	(128,30,18,30)
	Deconv3D_2	Add_3	DBR	(128,60,36,60)
	Conv3D_3	Deconv3D_2, Up(Add_3)	Conv	(12,60,36,60)

Table 3. Architecture and parameters details of scene completion stage. *Down*, *Up* mean downsampling and upsampling operation, respectively. *CBR* denotes a convolution block consists of a 3×3 convolution layer flowed by a batchnorm layer and Relu function and *DBR* replaces the convolution layer of *CBR* with a 4×4 deconvolution layer. *Conv* means a single 1×1 convolution layer.

original resolution and a final classify to get semantic scene completion results with size $12 \times 60 \times 36 \times 60$.

In addition, the class distribution is imbalanced. For example, the ratio between floor and TVs is about 70:1 in NYU. In order to solve the problem of imbalance distribution of different categories' voxel-wise samples, we count the number of voxels of each category and employ weights $\mathbf{w}$ for each category loss, inspired by the shrink function proposed in [8],

$$w_i = \frac{10}{a + b \cdot \tanh\left(c \cdot (v_i - v_{min})\right)}, i \in \{1, \cdots, C\}. \tag{1}$$

Specifically, v_i is the number of voxel-wise samples of one category and v_{min} is the minimum number among all categories. c stands for the shrinkage rate of weight and the combination of a and b limit the the weight w_i to the range of $(10/(a+b), 10/a)$. In practice, we set $a = 1$, $b = 99$ and $c = 3$. This re-weight ensures that some categories with a small number of samples will not be overwhelmed during the training process.

4. More Quantitative and Qualitative Ablation Studies

The effects of scene completion. In the main paper, we show the effectiveness of scene completion in our method for the overall 3D semantic scene completion. Here, we task a step further to explore and discuss how the scene completion in particular benefit the final result through influencing other component in our framework. We present the performance of the proposal generation module at dif-

Datasets	Iteration	win.	chair	bed	sofa	table	tvs	furn.	objs.	avg
NYU	I	7.5	21.1	59.6	45.7	20.1	18.0	27.6	7.4	25.6
	S0 + Iter I-S	15.8	30.0	66.0	48.8	29.8	31.6	35.9	19.0	34.6
	S0 + 2 Iter I-S	**21.9**	**33.2**	**69.0**	**55.8**	**43.4**	**36.3**	**43.2**	**23.8**	**40.8**
NYUCAD	I	32.7	38.3	64.9	54.4	43.1	35.1	38.9	21.7	41.1
	S0 + Iter I-S	30.1	50.0	66.6	52.3	44.0	36.8	46.3	31.3	44.5
	S0 + 2 Iter I-S	**42.0**	**50.2**	**72.3**	**60.5**	**58.5**	**46.7**	**54.3**	**37.2**	**52.7**
SUNCG-RGBD	I	74.2	65.0	83.9	82.0	67.3	32.7	72.1	43.3	65.0
	S0 + Iter I-S	76.1	75.7	85.2	**84.2**	70.8	**35.6**	82.6	53.3	70.4
	S0 + 2 Iter I-S	**76.8**	**76.9**	**86.6**	83.2	**72.9**	34.0	**86.0**	**56.5**	**71.6**

Table 4. Ablation studies of the effects of the scene completion to the proposal generation module of instance completion on three datasets. The numbers reported are detection mAP (IoU=0.25) for different classes, where I is the instance completion and S is the scene completion.

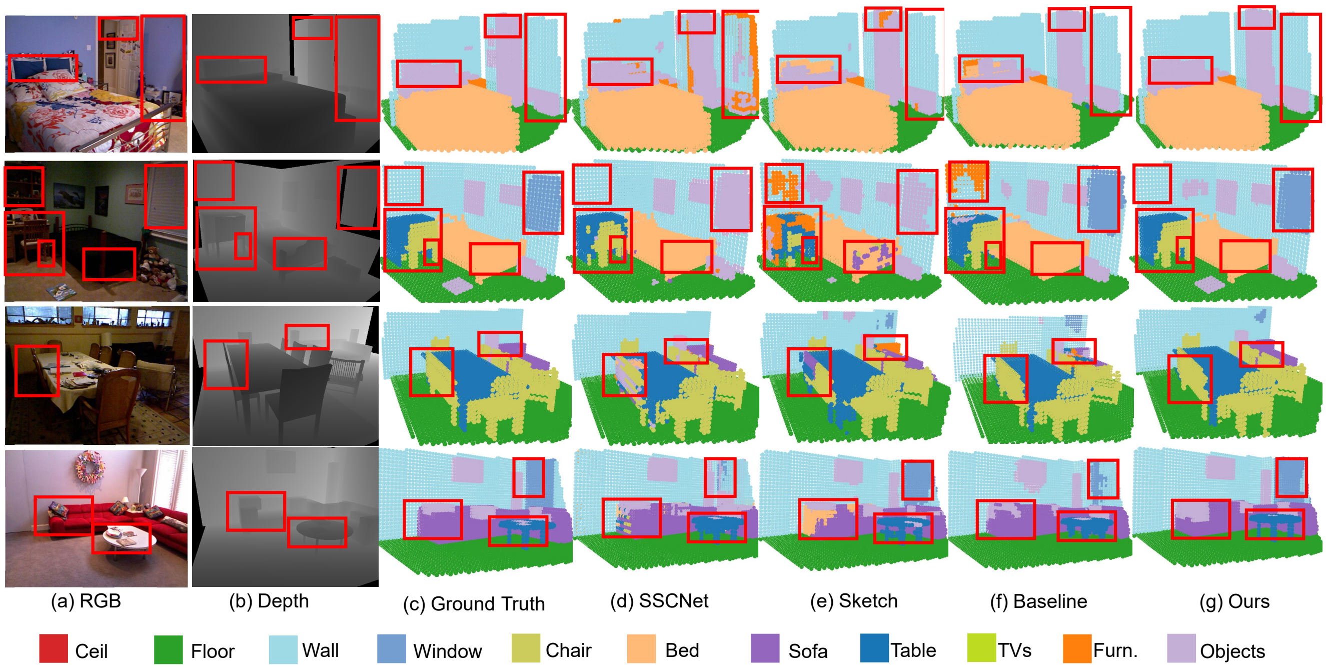

Figure 1. **Semantic Scene Completion results on NYUCAD dataset.** From left to right: (a) RGB input, (b) Depth, (c) ground truth, (d) results of SSCNet [10], (e) results of Sketch [2], (f) baseline (without using instance completion), (g) our results. Our results achieve higher voxel-level accuracy compared with SSCNet [10] and Sketch [2]. Better viewed in color and zoom in.

ferent stages in detail. As shown in Table 4, without S_0, the proposal generation module are not able to obtain an accurate understanding of the whole 3D scene, which means that it can not provide an accurate estimation of objects and easily confuse some close-by objects. However, with the guidance of semantic prior from scene completion, the proposal module can distinguish some objects that are mix-up to each other. For example, objects such as windows and paintings have no obvious structural features so if only the location information of point clouds is used, they are easily mixed with the background wall. However, when the semantic prior in hand, these objects can be easily detected, and the performance increases with the improvement of semantic accuracy two iterations bring more gain than once. Secondly, the proposal generation module can better estimate the size of each object and reduce the proposal overlap between different objects by using the structural information from scene completion in the invisible area. In addition, some small partial objects, which tends to be missed in the overall scene, and are easier to be localized after scene completion in the visible area.

The effects of Iterative Integrating Instances and Scene information. We try to find out what benefits does the network actually get from the iterative refinement scheme. As mentioned in the main paper, the proposed method aims to recover more fine-grained shape details in not only the visible but also invisible areas. To verify the detailed effect, we conduct experiments to explore the improvements on visible and invisible areas, respectively. Results are illustrated in Table 5. We observe that there are relatively uniform increases in both visible and invisible areas, which proves that our novel framework effectively explores the integrated

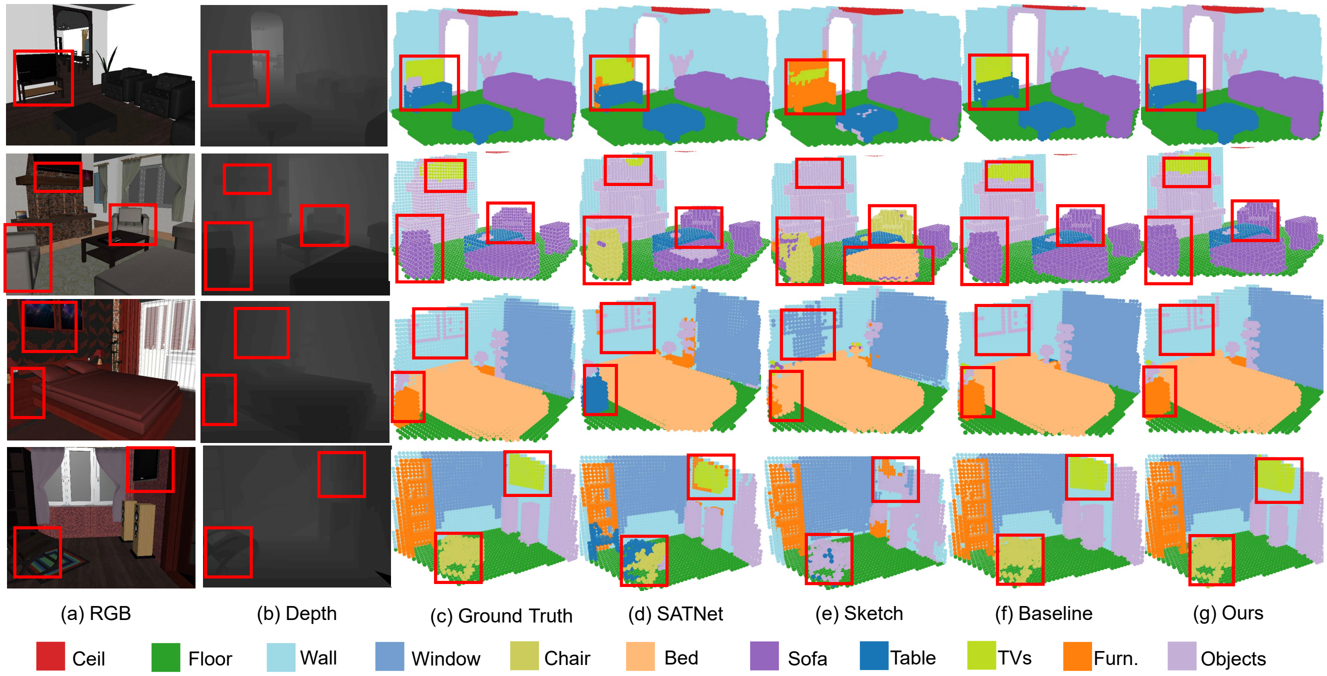

Figure 2. **Semantic Scene Completion results on SUNCG-RGBD dataset.** From left to right: (a) RGB input, (b) Depth, (c) ground truth, (d) results of SATNet [6], (e) results of Sketch [2], (f) baseline (without using instance completion), (g) our results. Our results achieve higher voxel-level accuracy compared with SATNet [6] and Sketch [2]. Better viewed in color and zoom in.

instances and scene information.

Methods	Dataset	Visible		Invisible	
		SC	SSC	SC	SSC
Baseline	NYU	98.1	47.3	64.5	38.7
+S-I-S Once	NYU	97.6	53.6	70.9	45.3
+Iter S-I-S	NYU	**98.1**	**55.5**	**72.3**	**46.8**
Baseline	NYUCAD	99.6	62.2	80.7	49.1
+S-I-S Once	NYUCAD	99.6	68.2	84.1	54.9
+Iter S-I-S	NYUCAD	**99.6**	**69.5**	**84.7**	**56.3**
Baseline	SUNCG-RGBD	93.5	85.6	86.2	59.0
+S-I-S Once	SUNCG-RGBD	95.9	88.1	88.9	64.4
+Iter S-I-S	SUNCG-RGBD	**96.3**	**88.4**	**88.9**	**65.0**

Table 5. **Visible and Invisible Region Results of the three datastes**. Bold numbers represent the best scores.

5. Visualization Results

Figures 1 and 2 illustrate the visualization results on NYUCAD and SUNCG-RGBD, respectively. Although the previous methods work well for some scenes, they usually fail to deal with shape details and nearby objects nearby objects whose semantic categories are easily mixed-up in the scene completion. However, our proposed method leverages and propagates instance-level and scene-level information can obtain a more comprehensive and accurate understanding of 3D scene. For NYUCAD, we compare our method with state-of-the-art method Sketch [1] and the classic ssc method SSCNet [10]. For SUNCG-RGBD, we compare with Sketch [1] and SATNet [7], who proposes SUNCG-RGBD dataset.

References

[1] Xiaokang Chen, Kwan-Yee Lin, Chen Qian, Gang Zeng, and Hongsheng Li. 3d sketch-aware semantic scene completion via semi-supervised structure prior. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4193–4202, 2020. 1, 4

[2] Xiaokang Chen, Kwan-Yee Lin, Chen Qian, Gang Zeng, and Hongsheng Li. 3d sketch-aware semantic scene completion via semi-supervised structure prior. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 3, 4

[3] Yu-Xiao Guo and Xin Tong. View-volume network for semantic scene completion from a single depth image. In *IJCAI*, pages –, 2018. 1

[4] Peng Wang Yu Liu Jie Li, Kai Han and Xia Yuan. Anisotropic convolutional networks for 3d semantic scene completion. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2020. 1

[5] Jie Li, Yu Liu, Dong Gong, Qinfeng Shi, Xia Yuan, Chunxia Zhao, and Ian Reid. Rgbd based dimensional decomposition residual network for 3d semantic scene completion. In *CVPR*, pages –, 2019. 1

[6] Shice Liu, Yu Hu, Yiming Zeng, Qiankun Tang, Beibei Jin, Yinhe Han, and Xiaowei Li. See and think: Disentangling

semantic scene completion. In *Advances in Neural Information Processing Systems*, pages 263–274, 2018. 1, 4

[7] Shice Liu, Yu Hu, Yiming Zeng, Qiankun Tang, Beibei Jin, Yinhe Han, and Xiaowei Li. See and think: Disentangling semantic scene completion. In *Advances in Neural Information Processing Systems*, pages 263–274, 2018. 4

[8] Xiankai Lu, Chao Ma, Bingbing Ni, Xiaokang Yang, Ian Reid, and Ming-Hsuan Yang. Deep regression tracking with shrinkage loss. In *Proceedings of the European conference on computer vision (ECCV)*, pages 353–369, 2018. 2

[9] Shuran Song, Fisher Yu, Andy Zeng, Angel X Chang, Manolis Savva, and Thomas Funkhouser. Semantic scene completion from a single depth image. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1746–1754, 2017. 1

[10] Shuran Song, Fisher Yu, Andy Zeng, Angel X Chang, Manolis Savva, and Thomas Funkhouser. Semantic scene completion from a single depth image. In *CVPR*, pages 1746–1754, 2017. 3, 4

[11] Haozhe Xie, Hongxun Yao, Shangchen Zhou, Jiageng Mao, Shengping Zhang, and Wenxiu Sun. Grnet: Gridding residual network for dense point cloud completion. *arXiv preprint arXiv:2006.03761*, 2020. 1, 2

[12] Jiahui Zhang, Hao Zhao, Anbang Yao, Yurong Chen, Li Zhang, and Hongen Liao. Efficient semantic scene completion network with spatial group convolution. In *ECCV*, pages 733–749, 2018. 1