Supplemental Material: Deep Implicit Moving Least-Squares Functions for 3D Reconstruction

Shi-Lin Liu[1,3] Hao-Xiang Guo[2,3] Hao Pan[3] Pengshuai Wang[3] Xin Tong[3] Yang Liu[3]

[1]University of Science and Technology of China [2]Tsinghua University [3]Microsoft Research Asia

freelin@mail.ustc.edu.cn ghx17@mails.tsinghua.edu.cn

{haopan,penwan,xtong,yangliu}@microsoft.com

A. Training data preparation

For training data involved with the ShapeNet dataset, we use data preprocessing tools from [3] to generate watertight meshes via TSDF fusion. We then normalize each mesh into a $[-1, 1]^3$ bounding box with 5% padding and compute signed distance function (SDF) values and gradients using the OpenVDB library (`https://www.openvdb.org`). We generate $256 \times 256 \times 256$ SDF grids, denoted by $\mathcal{F} = \{(i, j, k, s_{i,j,k}, \nabla s_{i,j,k})\}$, and collect SDF samples subset in a progressive manner: we first gather depth-6 SDF samples ((*i.e.*, samples whose indices satisfy: $i, j, k \mod 4 = 0$)) with absolute SDF values less than $\frac{1}{8}$, this threshold guarantees coverage of generated octree nodes. To better capture shape details, similar to the sampling strategy in [2], we add more SDF samples near the surface, to be concrete, the depth-7 SDF samples with absolute SDF values less than $\frac{1}{16}$.

B. Evaluation metrics

We reuse the evaluation tools of [3] to compute the following metrics. We denote M_g and M_p as the ground-truth mesh and the mesh of the predicted result. $\mathcal{X} := \{\mathbf{x}_1, \ldots, \mathbf{x}_{N_g}\}$ and $\mathcal{Y} := \{\mathbf{y}_1, \ldots, \mathbf{y}_{N_p}\}$ are randomly sample points on these two meshes, respectively. We define $\mathcal{P}_{g2p}(\mathbf{x}) = \arg\min_{\mathbf{y}\in\mathcal{Y}} \|\mathbf{x}-\mathbf{y}\|$ and $\mathcal{P}_{p2g}(\mathbf{y}) = \arg\min_{\mathbf{x}\in\mathcal{X}} \|\mathbf{x}-\mathbf{y}\|$. $\mathbf{n}(\cdot)$ denote an operator that returns the normal vector of a given point.

- L_1 Chamfer distance.

$$\begin{aligned}\mathtt{CD}_1 = &\frac{1}{2N_g}\sum_{i=1}^{N_g}\|\mathbf{x}_i - \mathcal{P}_{g2p}(\mathbf{x}_i)\| + \\ &\frac{1}{2N_p}\sum_{i=1}^{N_p}\|\mathbf{y}_i - \mathcal{P}_{p2g}(\mathbf{y}_i)\|.\end{aligned}$$

- Normal consistency.

$$\begin{aligned}\mathtt{NC} = &\frac{1}{2N_g}\sum_{i=1}^{N_g}|\mathbf{n}(\mathbf{x}_i)\cdot\mathbf{n}(\mathcal{P}_{g2p}(\mathbf{x}_i))| + \\ &\frac{1}{2N_p}\sum_{i=1}^{N_p}|\mathbf{n}(\mathbf{y}_i)\cdot\mathbf{n}(\mathcal{P}_{p2g}(\mathbf{y}_i))|.\end{aligned}$$

- IOU is the volumetric intersection of two meshes divided by the volume of their union. To compute this metric, 100k points are sampled in the bounding box and are determined whether they are in or outside two meshes.

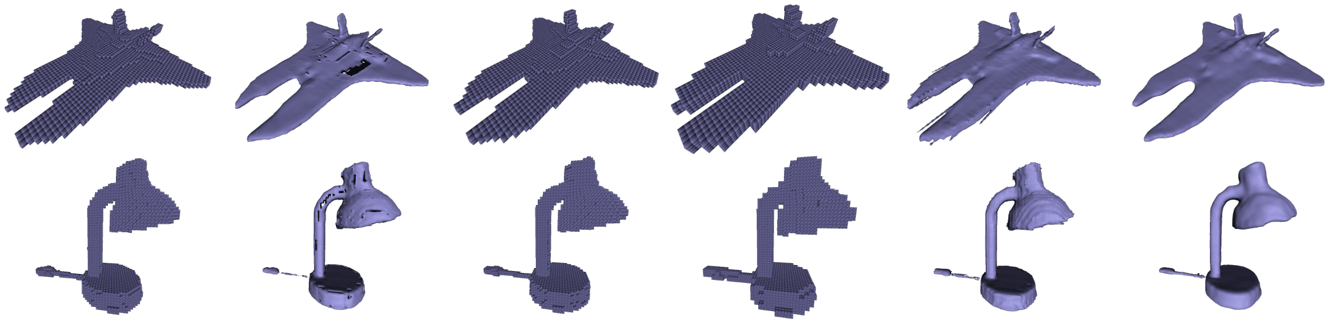
- F-Score is the harmonic mean between Precision and Recall. Precision is the percentage of points on M_p that lie within distance τ to M_g, Recall is the percentage of points on M_g that lie within distance τ to M_p.

$$\mathtt{F\text{-}Score} = \frac{2 * \mathtt{precision} * \mathtt{Recall}}{\mathtt{precision} + \mathtt{Recall}}.$$

We also compute the light field descriptor(LFD) to evaluate the perceptional similarity of the results to the ground-truth by following the setup of [1]. LFD is computed in the following way: each generated shape is rendered from various views and results in a set of projected images, then each projected image is encoded using Zernike moments and Fourier descriptors.

C. Network architecture

The number of network parameters for our network IMLSNet(7,7,1) reported in the paper is 4.6 M. The detailed setup of IMLSNet(7,6,1) and IMLSNet(7,7,1) is illustrated in the first and second rows of Table 1. We also did an ablation study by setting the channel number of depth-2 and depth-3 octree nodes to 128 and reduced the network parameter size while achieving comparable performances as shown in Table 2. The networks are denoted by IMLSNet(6,6,1)* and IMLSNet(7,7,1)*.



(a) (b) (c) (d) (e) (f)

Figure 1: (a): Non-empty finest octants predicated by the network based on the octree-aided deep local implicit function. (b): Reconstruction results from (a). (c): Ground-truth non-empty finest octants. (d): The expanded octree. (e): Reconstruction results based on (d). (f) Our IMLSNet results.

Network	2	3	4	5	6	7	size
IMLSNet(7,6,1)	256	256	128	64	32	16	4.6M
IMLSNet(7,7,1)	256	256	128	64	32	16	4.6M
IMLSNet(7,6,1)*	128	128	128	64	32	16	1.5M
IMLSNet(7,7,1)*	128	128	128	64	32	16	1.6M

Table 1: Network parameters of IMLSNets. Feature channel dimensions on each octree depth (from 2 to 7) are listed.

Network	$\text{CD}_1 \downarrow$	**NC** $\uparrow$	**IoU** $\uparrow$	**F-Score** $\uparrow$
IMLSNet(7,6,1)	0.0310	0.9430	0.9134	0.9813
IMLSNet(7,7,1)	**0.0306**	**0.9440**	**0.9135**	**0.9833**
IMLSNet(7,6,1)*	0.0311	0.9425	0.9129	0.9814
IMLSNet(7,7,1)*	0.0307	0.9434	0.9132	0.9827

Table 2: Quantitative evaluation of IMLSNet with different network settings on the task of 3D object reconstruction.

D. Octree-aided deep local implicit function

As discussed in Section 4, our ablation study shows that the octree-aided deep local implicit function has issues in obtaining complete zero-iso surfaces. We confirm this fact by examining the generated non-empty octants which cover the missing regions but the local implicit function does not generate the iso-surface in them. Fig. 1-a illustrates these octants which cover the ground-truth non-empty octants (see Fig. 1-c) but fail to generate a complete shape (Fig. 1-b). We speculate that the implicit function passes through the neighboring region which is not covered by the current finest octants. Based on this speculation, we split all the $d-1$-depth octants to expand the octree (see Fig. 1-d) and extract the surface via marching cubes. It turns out that the implicit surface appears in those regions (see Fig. 1-e), however, the reconstruction error is higher than our IMLSNet (Fig. 1-f).

Network	δ	$\text{CD}_1 \downarrow$	**NC** $\uparrow$	**IoU** $\uparrow$	**F-Score** $\uparrow$
IMLSNet	1.0×10^{-3}	**0.0288**	**0.9476**	**0.9226**	**0.9859**
ConvOccNet	1.0×10^{-3}	0.0495	0.9349	0.8573	0.9442
IMLSNet	3.0×10^{-3}	**0.0290**	**0.9473**	**0.9219**	**0.9857**
ConvOccNet	3.0×10^{-3}	0.0439	0.9377	0.8831	0.9461
IMLSNet	5.0×10^{-3}	**0.0306**	**0.9440**	**0.9135**	**0.9833**
ConvOccNet	5.0×10^{-3}	0.0441	0.9383	0.8842	0.9421
IMLSNet	7.5×10^{-3}	**0.0372**	**0.9291**	**0.8754**	**0.9705**
ConvOccNet	7.5×10^{-3}	0.0536	0.9345	0.8435	0.9221

Table 3: Robustness test to noise.

E. Evaluation of object reconstruction

In Table 4, we report the numerical metrics of the tasks of object reconstruction from point clouds for each shape category. Fig. 2 presents more visual results reconstructed from our network. All the evaluations demonstrate the superiority of our method over other approaches in terms of reconstruction accuracy and the capacity of recovering details and thin regions. In Fig. 3 we present more results of our ablation study of different network settings.

F. Robustness test on noise levels

We did a robustness test on the input noise. The network IMLSNet(7,7,1) and ConvOccNet were trained with noisy data whose Gaussian noise is with standard deviation $\delta = 5 \times 10^{-3}$. We add different noise levels ($\delta = 1 \times 10^{-3}, 3 \times 10^{-3}, 7.5 \times 10^{-3}$) to the test data of 13 shape classes and feed to our network and ConvOccNet for evaluating their performance. From Table 3, we can see with lower noise levels, our network always performs better than ConvOccNet. With a higher level noise ($\delta = 7.5 \times 10^{-3}$), The network performance of both methods degrades gracefully, and our method still outperforms ConvOccNet.

Category	CD$_1$ ↓ O-CNN-C	CD$_1$ ↓ IMLSNet points	CD$_1$ ↓ ConvOccNet	CD$_1$ ↓ IMLSNet	NC ↑ O-CNN-C	NC ↑ IMLSNet points	NC ↑ ConvOccNet	NC ↑ IMLSNet
airplane	0.0634	0.0316	0.0336	**0.0245**	0.9181	0.9292	0.9311	**0.9371**
bench	0.0646	0.0356	0.0352	**0.0301**	0.9136	0.9194	0.9205	**0.9220**
cabinet	0.0709	0.0375	0.0461	**0.0348**	0.9411	0.9505	**0.9561**	0.9546
car	0.0765	0.0419	0.0750	**0.0395**	0.8668	0.8709	**0.8931**	0.8820
chair	0.0664	0.0383	0.0459	**0.0348**	0.9407	0.9487	0.9427	**0.9503**
display	0.0655	0.0339	0.0368	**0.0292**	0.9598	0.9710	0.9677	**0.9732**
lamp	0.0667	0.0367	0.0595	**0.0312**	0.9111	0.9206	0.9003	**0.9218**
speaker	0.0729	0.0413	0.0632	**0.0396**	0.9363	0.9440	0.9387	**0.9473**
rifle	0.0617	0.0300	0.0280	**0.0207**	0.9320	0.9428	0.9293	**0.9433**
sofa	0.0657	0.0350	0.0414	**0.0309**	0.9492	0.9602	0.9579	**0.9631**
table	0.0663	0.0360	0.0385	**0.0319**	0.9461	0.9599	0.9588	**0.9621**
telephone	0.0610	0.0295	0.0270	**0.0229**	0.9737	0.9827	0.9823	**0.9839**
vessel	0.0641	0.0336	0.0430	**0.0271**	0.9221	0.9280	0.9187	**0.9319**
mean	0.0666	0.0355	0.0441	**0.0306**	0.9316	0.9406	0.9382	**0.9440**
bag	0.0704	0.0386	0.0538	**0.0351**	0.9342	0.9420	0.9417	**0.9455**
bathtub	0.0663	0.0378	0.0526	**0.0350**	0.9478	0.9599	0.9537	**0.9622**
bed	0.0720	0.0428	0.0608	**0.0412**	0.9192	0.9246	0.9119	**0.9278**
bottle	0.0619	0.0332	0.0421	**0.0279**	0.9610	0.9696	0.9657	**0.9708**
pillow	0.0631	0.0340	0.0548	**0.0303**	0.9652	0.9743	0.9660	**0.9757**
mean	0.0667	0.0373	0.0528	**0.0339**	0.9455	0.9541	0.9478	**0.9564**

Category	IoU ↑ O-CNN-C	IoU ↑ IMLSNet points	IoU ↑ ConvOccNet	IoU ↑ IMLSNet	F-Score ↑ O-CNN-C	F-Score ↑ IMLSNet points	F-Score ↑ ConvOccNet	F-Score ↑ IMLSNet
airplane	n/a	n/a	0.8485	**0.8910**	0.8101	0.9923	0.9653	**0.9918**
bench	n/a	n/a	0.8298	**0.8480**	0.7995	0.9867	0.9643	**0.9860**
cabinet	n/a	n/a	0.9398	**0.9495**	0.7887	0.9833	0.9558	**0.9811**
car	n/a	n/a	0.8858	**0.9052**	0.7474	0.9604	0.8490	**0.9521**
chair	n/a	n/a	0.8709	**0.9033**	0.7993	0.9824	0.9387	**0.9815**
display	n/a	n/a	0.9275	**0.9491**	0.8109	0.9935	0.9708	**0.9935**
lamp	n/a	n/a	0.7840	**0.8583**	0.7999	0.9806	0.8910	**0.9785**
speaker	n/a	n/a	0.9188	**0.9450**	0.7789	0.9676	0.8924	**0.9633**
rifle	n/a	n/a	0.8459	**0.8856**	0.8263	0.9961	0.9799	**0.9962**
sofa	n/a	n/a	0.9362	**0.9541**	0.8052	0.9886	0.9531	**0.9873**
table	n/a	n/a	0.8877	**0.9076**	0.8013	0.9875	0.9674	**0.9870**
telephone	n/a	n/a	0.9537	**0.9647**	0.8252	0.9978	0.9882	**0.9978**
vessel	n/a	n/a	0.8663	**0.9140**	0.8085	0.9880	0.9313	**0.9868**
mean	n/a	n/a	0.8842	**0.9135**	0.8001	0.9850	0.9421	**0.9833**
bag	n/a	n/a	0.9229	**0.9461**	0.7859	0.9766	0.9187	**0.9745**
bathtub	n/a	n/a	0.8431	**0.9079**	0.8014	0.9861	0.9084	**0.9851**
bed	n/a	n/a	0.8612	**0.9052**	0.7727	0.9661	0.8985	**0.9622**
bottle	n/a	n/a	0.9468	**0.9663**	0.8282	0.9916	0.9515	**0.9907**
pillow	n/a	n/a	0.9354	**0.9700**	0.8219	0.9948	0.9005	**0.9941**
mean	n/a	n/a	0.9019	**0.9391**	0.8020	0.9831	0.9155	**0.9813**

Table 4: Quantitative evaluation of different networks on the test data of 13 shape classes and the full data of 5 unseen shape classes.

References

[1] Zhiqin Chen and Hao Zhang. Learning implicit fields for generative shape modeling. In *CVPR*, 2019. 1

[2] Jeong Joon Park, Peter Florence, Julian Straub, Richard Newcombe, and Steven Lovegrove. DeepSDF: learning continuous signed distance functions for shape representation. In *CVPR*, 2019. 1

[3] Songyou Peng, Michael Niemeyer, Lars Mescheder, Marc Pollefeys, and Andreas Geiger. Convolutional occupancy networks. In *ECCV*, 2020. 1

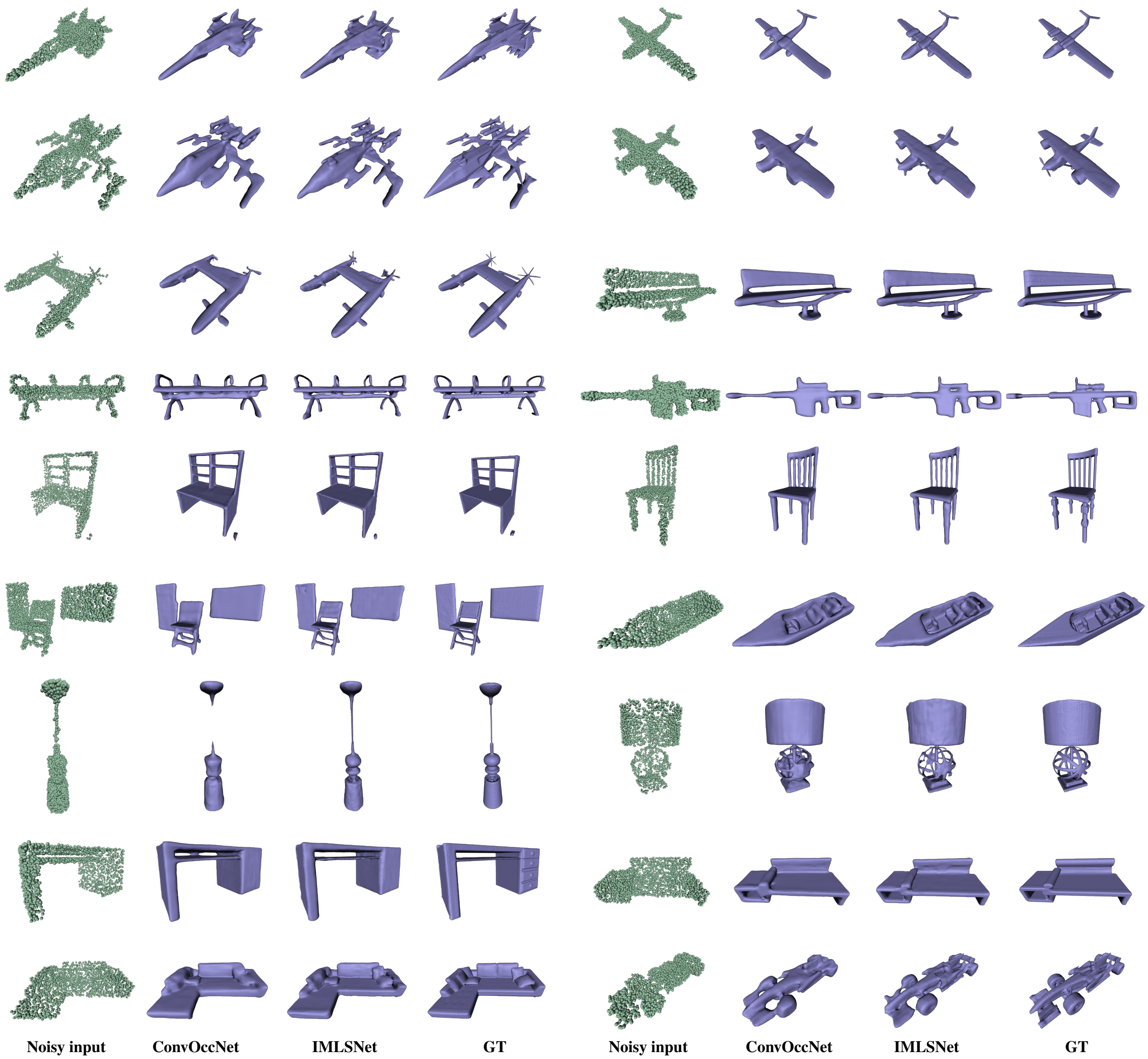


Figure 2: More results of object reconstruction from point clouds.

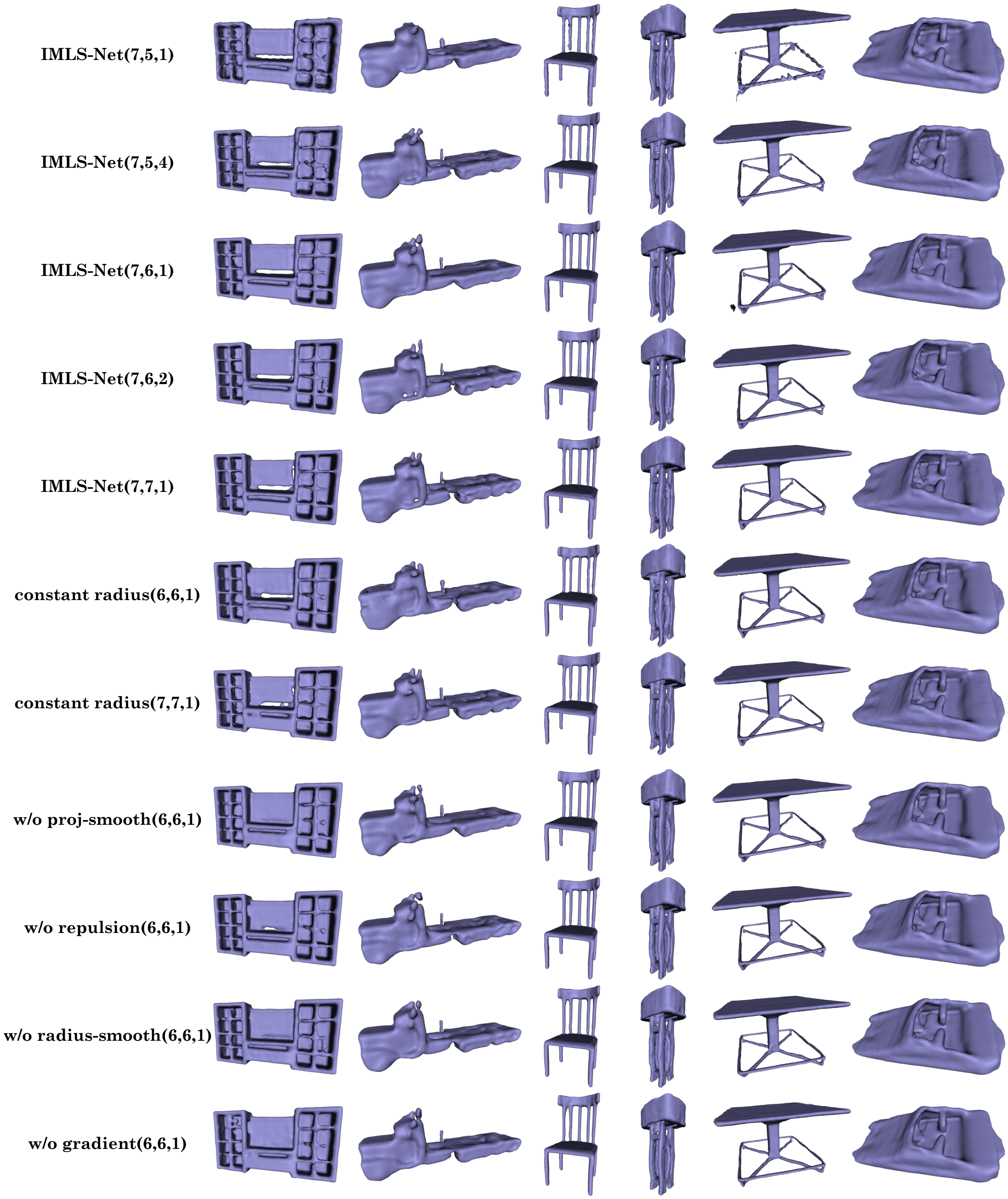


Figure 3: More results of our ablation study on network settings. The inputs are the noisy point clouds (see Fig. 3 of the main body of our paper).