Learning Dual Hierarchical Representation for 3D Surface Reconstruction - Supplementary Material -

Jiyoon Shin[1], Youngwook Kim[2], Sangwoo Hong[1], and Jungwoo Lee[1]

[1] Seoul National University
[2] Kookmin University
jiyoonshin@cml.snu.ac.kr, youngwook@kookmin.ac.kr,
{tkddn0606, junglee}@snu.ac.kr

Abstract. In this supplementary material, we give further details of the main paper. Architecture details including layers of the transformer block and hyperparameter choices are given in Section 1. Implementation details are formulated in Section 2 and the procedures for encoding sparse point clouds are illustrated in Section 3. Section 4 shows additional experimental results.

1 Architecture Details

1.1 Transformer Block

Layer details of the transformer block of the feature-to-occupancy hierarchical decoder are given in this subsection. The self-attn layer of Equation 4 is demonstrated as:

$$\begin{aligned} &\text{self-attn}(Y_{l-1}) = \text{cat}(A_1, \cdots, A_H)W^{self}, \\ &\text{where } A_h = \text{Attn}(Y_{l-1}W_h^Q, Y_{l-1}W_h^K, Y_{l-1}W_h^V) \in \mathbb{R}^{M \times d_H}. \end{aligned} \tag{1}$$

The cross-attn layer of Equation 5 is formulated as:

$$\begin{aligned} &\text{cross-attn}(Y_l', z_{R-(l-1)}) = \text{cat}(A_1, \cdots, A_H)W^{cross}, \\ &\text{where } A_h = \text{Attn}(Y_l'W_h^Q, z_{R-(l-1)}W_h^K, z_{R-(l-1)}W_h^V) \in \mathbb{R}^{M \times d_H}. \end{aligned} \tag{2}$$

The projections of both equations are parameter matrices

$$\begin{aligned} W_h^Q \in \mathbb{R}^{D \times d_H}, W_h^K \in \mathbb{R}^{D \times d_H}, W_h^V \in \mathbb{R}^{D \times d_H}, \\ W^{self} \in \mathbb{R}^{Hd_H \times D}, W^{cross} \in \mathbb{R}^{Hd_H \times D}, \end{aligned} \tag{3}$$

where d_H denotes the feature dimension in each head and H is the number of attention heads.

Note that regardless of the different-resolution latent code inputs, decoder weights are shared across all stages as the decreased resolutions disappear by the key and value multiplication ($K^T V$) of attention layers.

1.2 Hyperparameter Choices

Architecture hyperparameter choices for the main paper experiments are given in this subsection. Refer to Section 3 of the main paper for a detailed description of the symbols.

(Subsection 3.1) Hierarchical Latent Feature Code Set Encoder. From a sparse voxelized input shape of resolution $N = 128$, $R = 7$ feature grids are encoded, each with channels $C_k = [1, 16, 32, 64, 128, 128, 128]$. During training, subsamples of size $M = 10,000$ are used.

(Subsection 3.2) Feature-to-Occupancy Decoder. $D = 12$ is used for the hidden dimension of the transformer decoder, and $H = 8$ attention heads are used.

(Subsection 3.3) Occupancy Field Prediction. The MLP of the occupancy field prediction consists of 6 fully connected layers.

2 Implementation Details

2.1 Implementations

The Adam optimizer is employed with an initial learning rate of 1×10^{-4} and StepLR scheduler with parameters $step\_size = 50, gamma = 0.1$. Training lasts for 200 epochs with a mini-batch size of 4, and the computation is limited to a single Nvidia V100 GPU for all models. The spatial positional embedding Y_0 is initialized for each query point set using the Kaiming uniform distribution. Implemented hyperparameters for hierarchical losses are tabulated in Table 1.

2.2 Datasets

We utilize the complete ShapeNet version 2 dataset, which consists of 13 categories. The original dataset contains triangle meshes, which were made watertight following the preprocessing method by [15], and were divided into training and testing sets according to the split provided by [4]. Ground truth occupancies

Table 1: Hyperparameters. Implemented hyperparameters for hierarchical losses are tabulated below.

l	1	2	3	4	5	6	7
λ_l	0.05	0.1	0.2	0.35	0.5	0.65	0.7
ε_l	0.1	0.08	0.06	0.04	0.01	0.005	0

are calculated as boolean values using libraries [3]. This process involves projecting points and mesh triangles onto a 2D plane, determining intersection depths, counting ray-triangle intersections, and utilizing this data to determine the status of each point.

2.3 Metrics

Formal definitions of all three metrics (IoU, F-Score, and Chamfer distance) used for quantitative evaluations are provided below.

IoU. Intersection over Union (IoU) [8] measures how well volumes match, and a higher value indicates better results. For all points that are inside or on the predicted mesh $\mathcal{M}_{\text{pred}}$ and ground truth mesh $\mathcal{M}_{\text{GT}}$, volumetric IoU is defined as the quotient of the two volumes' intersection and their union:

$$\mathrm{IoU}(\mathcal{M}_{\text{pred}}, \mathcal{M}_{\text{GT}}) \equiv \frac{|\mathcal{M}_{\text{pred}} \cap \mathcal{M}_{\text{GT}}|}{|\mathcal{M}_{\text{pred}} \cup \mathcal{M}_{\text{GT}}|}. \tag{4}$$

F-Score. F-Score [6,12–14] measures the ratio of good predictions, and a higher value indicates better results. With a distance threshold d, the F-Score is defined as:

$$\mathrm{F-Score}(d) = \frac{2P(d)R(d)}{P(d)+R(d)}, \tag{5}$$

where $P(d)$ and $R(d)$ denote the precision and recall, respectively. Precision $P(d)$ quantifies the accuracy of reconstruction by the portion of reconstructed points lying within distance d to the ground truth:

$$P(d) = \frac{1}{|\mathcal{R}|} \sum_{r \in \mathcal{R}} \Big[\min_{g \in \mathcal{G}} \|g - r\| < d \Big]. \tag{6}$$

Also, recall $R(d)$ quantifies the completeness of reconstruction by the portion of ground-truth points lying within distance d to the reconstruction:

$$R(d) = \frac{1}{|\mathcal{G}|} \sum_{g \in \mathcal{G}} \Big[\min_{r \in \mathcal{R}} \|g - r\| < d \Big]. \tag{7}$$

$[\cdot]$ is the Iverson bracket, and $\mathcal{R}$ and $\mathcal{G}$ indicate the reconstructed and ground-truth point set, respectively. $\mathrm{F-Score}(d)$ has the property that if either $P(d) \to 0$ or $R(d) \to 0$, then $\mathrm{F-Score}(d) \to 0$. F-Score results reported in the paper use a value of $d = 1\%$, and implementation settings provided by [12].

[3] https://github.com/autonomousvision/convolutional_occupancy_networks

Chamfer distance. Chamfer distance [8] measures the average error of all points, and a lower value indicates better results. For the predicted mesh $\mathcal{M}_{\text{pred}}$ and ground truth mesh $\mathcal{M}_{\text{GT}}$, the Chamfer$-L_1$ distance is defined as:

$$\begin{aligned}\text{Chamfer}-L_1(\mathcal{M}_{\text{pred}}, \mathcal{M}_{\text{GT}}) \equiv \\ &\frac{1}{2|\partial\mathcal{M}_{\text{pred}}|}\int_{\partial\mathcal{M}_{\text{pred}}} \min_{q\in\partial\mathcal{M}_{\text{GT}}} \|p-q\|dp + \\ &\frac{1}{2|\partial\mathcal{M}_{\text{GT}}|}\int_{\partial\mathcal{M}_{\text{GT}}} \min_{p\in\partial\mathcal{M}_{\text{pred}}} \|p-q\|dq,\end{aligned} \tag{8}$$

where the surfaces of the two meshes are denoted by $\partial\mathcal{M}_{\text{pred}}$ and $\partial\mathcal{M}_{\text{GT}}$, respectively. Additionally, the accuracy score and completeness score of $\mathcal{M}_{\text{pred}}$ wrt. $\mathcal{M}_{\text{GT}}$ is defined below:

$$\text{Accuracy}(\mathcal{M}_{\text{pred}}|\mathcal{M}_{\text{GT}}) \equiv \frac{1}{2|\partial\mathcal{M}_{\text{pred}}|}\int_{\partial\mathcal{M}_{\text{pred}}} \min_{q\in\partial\mathcal{M}_{\text{GT}}} \|p-q\|dp, \tag{9}$$

$$\text{Completeness}(\mathcal{M}_{\text{pred}}|\mathcal{M}_{\text{GT}}) \equiv \frac{1}{2|\partial\mathcal{M}_{\text{GT}}|}\int_{\partial\mathcal{M}_{\text{GT}}} \min_{p\in\partial\mathcal{M}_{\text{pred}}} \|p-q\|dq. \tag{10}$$

Note that the Chamfer$-L_1$ distance is the mean of Accuracy and Completeness score.

3 Sparse Point Cloud Encoding

In this section, the encoding process of sparse point clouds is formulated (Subsection 4.4). It follows a similar procedure as encoding voxel inputs. After encoding, the processes are identical to those illustrated in the main paper.

The sparse point cloud input shape $X \in \mathcal{X}$, where $\mathcal{X} = \mathbb{R}^{N\times 3}$, is first subsampled into a set of "more sparse" point clouds via Farthest Point Sampling as

$$\{X_k\}_{k\in[1,\ldots,R]} = \text{farthest-point-sampling}(X), \quad X_k \in \mathbb{R}^{K\times 3}. \tag{11}$$

The sparse point clouds are then encoded into a set of multi-scale features with a mini-PointNet-like module [11, 16] as

$$\forall k \in [1,\ldots,R], \quad F_k = \text{PointNet}(X_k), \quad F_k \in \mathcal{F}_k^K. \tag{12}$$

$\mathcal{F}_k \in \mathbb{R}^{C_k}$ is a deep feature with channels C_k, $K = \frac{N}{2^{k-1}}$ is the sparse point cloud size varying with scale, and R is the number of features. Features of early stages include local details of the shape while features of late stages capture global structures.

Given a continuous query point $q \in \mathbb{R}^3$ from a query set $Q \in \mathbb{R}^{M\times 3}$, a hierarchical latent feature code set is acquired by grid-sampling [5] the particular location on each feature as

$$\forall k \in [1,\ldots,R], \quad \mathrm{z}_k^q = \text{grid-sample}(F_k, q) \tag{13}$$

where $\mathrm{z}_k^q \in \mathbb{R}^{C_k}$ and thus $\mathrm{z}_k \in \mathbb{R}^{M\times C_k}$. Trilinear interpolation is used to align continuous 3D points on the discrete features.

4 Additional Results

4.1 Shape Reconstruction

Additional reconstruction results of diverse shapes from 13 categories of ShapeNet [2] are visualized in Figure 1 and 2. Our method visualizes solid structures with the inclusion of details, patterns, and subtle parts.

4.2 Point Cloud Completion

Per-Class Evaluations. In Table 2a, 2b, 2c, per-class evaluations of point cloud completion are provided in terms of IoU, F-Score, and Chamfer distance, respectively. The mean and std computed over all 13 ShapeNet categories are indicated below the category measures. Our method is compared against OccNet [8], ConvONet [10], IF-Net [3], SAP [9], POCO [1], and DCC-DIF [7]. Similar to the results shown by voxel reconstructions, our method outperforms baselines by nearly all measures. Additionally, it shows robustness between various categories by revealing the smallest variance in all three metrics.

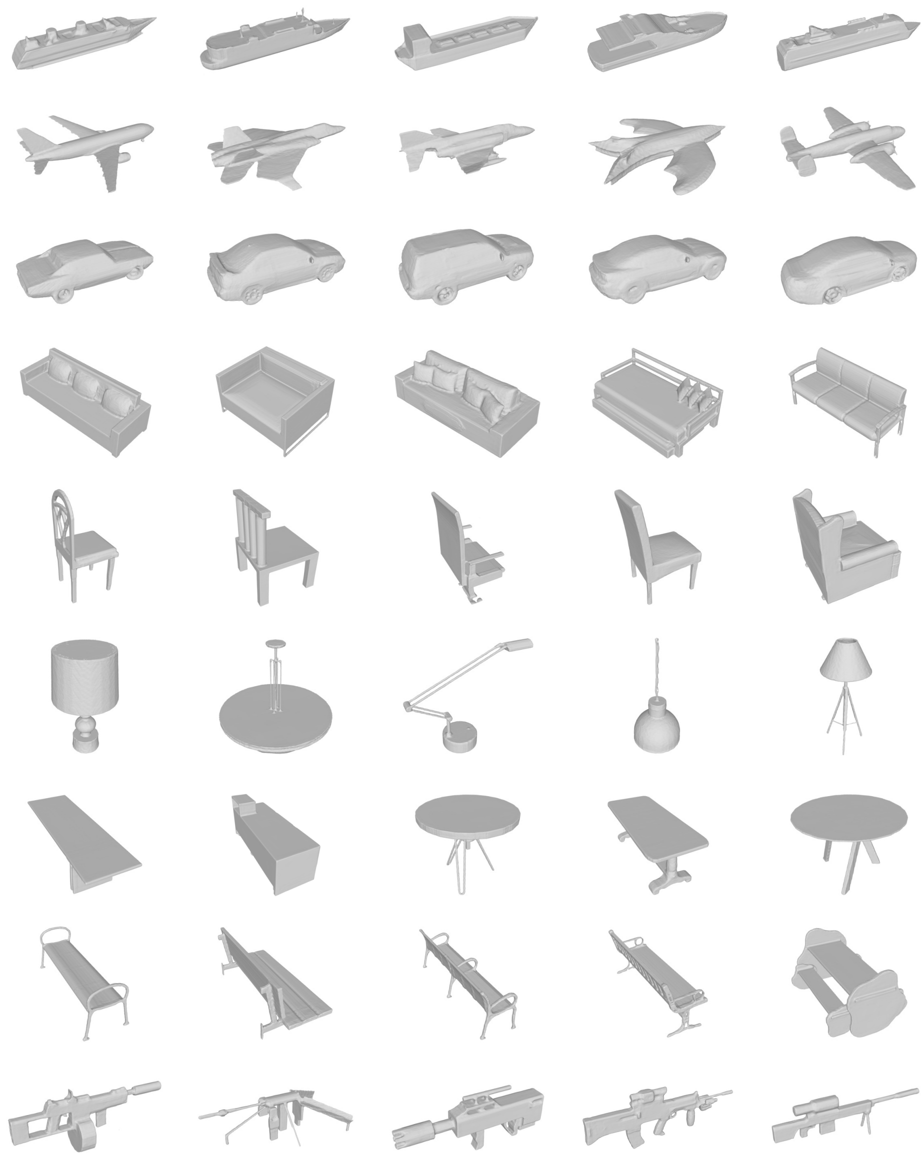

Fig. 1: Reconstruction of shapes from various categories. From the top row, shapes from categories: vessel, airplane, car, sofa, chair, lamp, table, bench, and riffle. Please zoom in to see the details of the shapes.

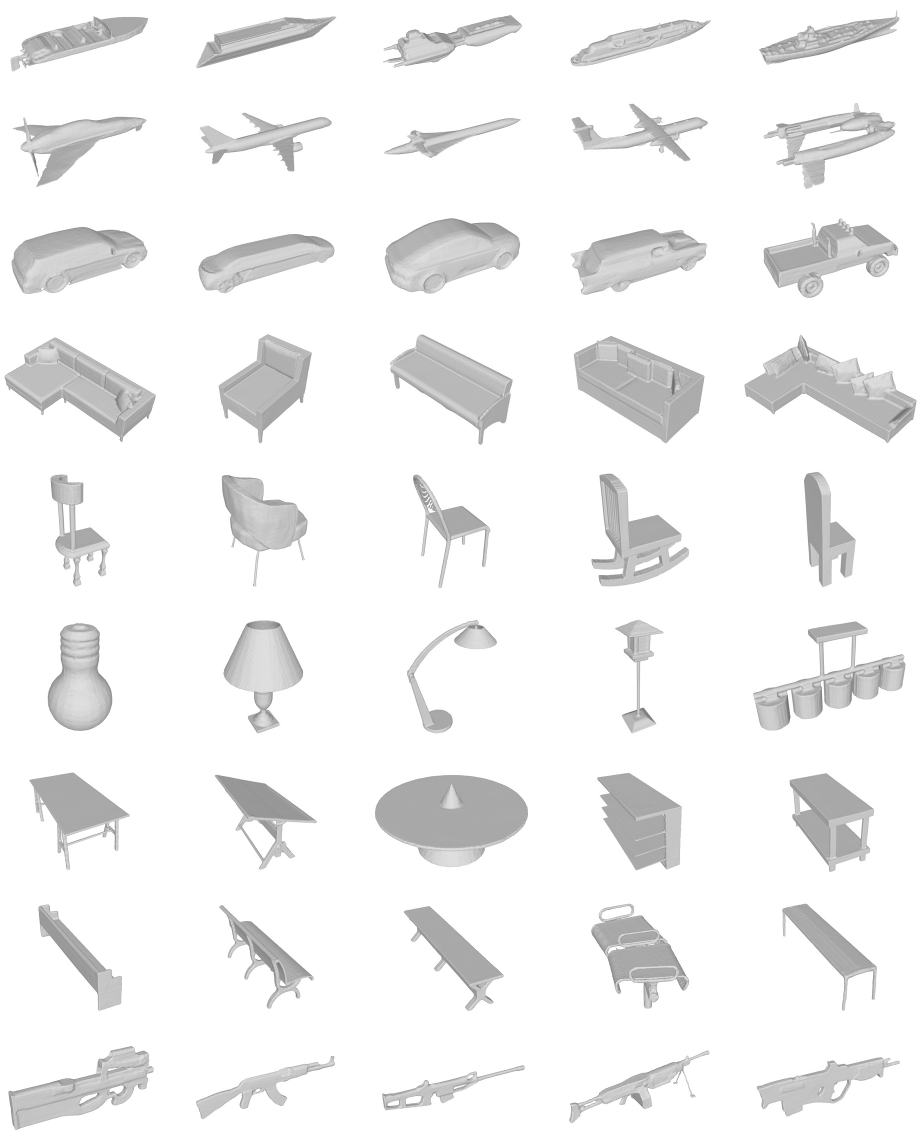

Fig. 2: Reconstruction of shapes from various categories. From the top row, shapes from categories: vessel, airplane, car, sofa, chair, lamp, table, bench, and riffle. Please zoom in to see the details of the shapes.

Table 2: Point cloud completion accuracy by three measures.

(a) Point cloud completion accuracy under ShapeNet in terms of IoU (↑).

Category	IoU ↑						
	OccNet	ConvONet	IF-Net	SAP	POCO	DCC-DIF	DHR (Ours)
Airplane	0.760	0.848	0.891	0.910	**0.941**	0.938	0.940
Bench	0.716	0.790	0.880	0.830	0.855	0.932	**0.938**
Cabinet	0.867	0.922	0.854	0.872	0.885	0.908	**0.928**
Car	0.835	0.876	0.911	0.928	0.912	0.918	**0.935**
Chair	0.736	0.852	0.873	0.851	0.840	0.889	**0.911**
Display	0.817	0.903	0.862	0.894	0.893	0.909	**0.913**
Lamp	0.566	0.792	0.878	0.888	0.917	0.917	**0.924**
Loudspeaker	0.828	0.913	0.849	0.807	0.849	0.920	**0.944**
Rifle	0.694	0.826	0.923	0.899	0.907	**0.926**	0.919
Sofa	0.872	0.923	0.881	0.909	0.895	0.931	**0.963**
Table	0.759	0.859	0.859	0.912	0.906	0.929	**0.933**
Telephone	0.915	0.942	0.843	0.911	**0.949**	0.905	0.930
Vessel	0.748	0.858	0.862	0.920	0.903	0.897	**0.924**
Mean	0.777	0.870	0.874	0.887	0.896	0.917	**0.931**
Std	0.089	0.047	0.023	0.035	0.033	0.014	**0.013**

(b) Point cloud completion accuracy under ShapeNet in terms of F-Score (↑).

Category	F-Score ↑						
	OccNet	ConvONet	IF-Net	SAP	POCO	DCC-DIF	DHR (Ours)
Airplane	0.878	0.967	0.944	0.988	0.978	0.984	**0.988**
Bench	0.875	0.944	0.926	0.964	0.966	**0.989**	0.980
Cabinet	0.860	0.929	0.930	0.958	0.950	0.946	**0.962**
Car	0.775	0.833	0.874	**0.976**	0.912	0.950	0.971
Chair	0.772	0.929	0.945	0.961	0.958	**0.967**	0.959
Display	0.821	0.955	0.961	0.965	0.979	0.972	**0.987**
Lamp	0.627	0.910	0.881	0.871	0.940	**0.977**	0.964
Loudspeaker	0.862	0.880	0.935	0.951	0.956	0.971	**0.984**
Rifle	0.859	0.969	0.925	0.946	0.954	0.986	**0.991**
Sofa	0.747	0.942	0.902	0.981	0.973	0.970	**0.994**
Table	0.849	0.953	0.934	0.955	0.951	0.975	**0.986**
Telephone	0.948	0.987	0.968	0.966	0.963	0.959	**0.979**
Vessel	0.773	0.927	0.927	**0.992**	0.989	0.958	0.976
Mean	0.819	0.933	0.927	0.961	0.960	0.970	**0.979**
Std	0.077	0.041	0.027	0.029	0.019	0.013	**0.011**

(c) Point cloud completion accuracy under ShapeNet in terms of Chamfer distance (↓).

Category	Chamfer distance ↓						
	OccNet	ConvONet	IF-Net	SAP	POCO	DCC-DIF	DHR (Ours)
Airplane	0.565	0.333	0.245	0.350	0.113	0.103	**0.063**
Bench	0.592	0.410	0.187	0.397	0.124	0.089	**0.078**
Cabinet	0.738	0.543	0.099	0.409	0.117	0.090	**0.051**
Car	0.981	0.802	0.231	0.519	0.135	0.054	**0.042**
Chair	0.890	0.494	0.158	0.475	0.131	0.112	**0.089**
Display	0.762	0.420	0.301	0.368	0.148	0.098	**0.079**
Lamp	1.350	0.645	0.106	0.520	0.138	0.125	**0.090**
Loudspeaker	1.169	0.647	0.243	0.520	0.128	0.118	**0.057**
Rifle	0.603	0.308	0.277	0.324	0.097	0.107	**0.067**
Sofa	0.695	0.456	0.313	0.413	0.138	0.135	**0.074**
Table	0.717	0.427	0.086	0.387	0.150	0.094	**0.077**
Telephone	0.411	0.295	0.256	0.225	0.100	0.109	**0.081**
Vessel	0.850	0.449	0.235	0.480	0.106	0.127	**0.067**
Mean	0.794	0.479	0.211	0.415	0.126	0.105	**0.070**
Std	0.247	0.142	0.074	0.084	0.017	0.020	**0.014**

References

1. Boulch, A., Marlet, R.: POCO: point convolution for surface reconstruction. In: Proc. Conf. on Computer Vision and Pattern Recognition (CVPR). pp. 6292–6304 (2022)
2. Chang, A.X., Funkhouser, T.A., Guibas, L.J., Hanrahan, P., Huang, Q., et al.: Shapenet: An information-rich 3d model repository. CoRR **abs/1512.03012** (2015)
3. Chibane, J., Alldieck, T., Pons-Moll, G.: Implicit functions in feature space for 3d shape reconstruction and completion. In: Proc. Conf. on Computer Vision and Pattern Recognition (CVPR). pp. 6968–6979 (2020)
4. Choy, C.B., Xu, D., Gwak, J., Chen, K., Savarese, S.: 3d-r2n2: A unified approach for single and multi-view 3d object reconstruction. In: Proc. European Conf. on Computer Vision (ECCV). pp. 628–644 (2016)
5. Jaderberg, M., Simonyan, K., Zisserman, A., Kavukcuoglu, K.: Spatial transformer networks. In: Proc. Advances in Neural Information Processing Systems (NeurIPS). pp. 2017–2025 (2015)
6. Knapitsch, A., Park, J., Zhou, Q., Koltun, V.: Tanks and temples: Benchmarking large-scale scene reconstruction. ACM Trans. on Graphics (TOG) pp. 1–13 (2017)
7. Li, T., Wen, X., Liu, Y., Su, H., Han, Z.: Learning deep implicit functions for 3d shapes with dynamic code clouds. In: Proc. Conf. on Computer Vision and Pattern Recognition (CVPR). pp. 12840–12850 (2022)
8. Mescheder, L.M., Oechsle, M., Niemeyer, M., Nowozin, S., Geiger, A.: Occupancy networks: Learning 3d reconstruction in function space. In: Proc. Conf. on Computer Vision and Pattern Recognition (CVPR). pp. 4460–4470 (2019)
9. Peng, S., Jiang, C., Liao, Y., Niemeyer, M., Pollefeys, M., Geiger, A.: Shape as points: A differentiable poisson solver. In: Proc. Advances in Neural Information Processing Systems (NeurIPS). pp. 13032–13044 (2021)
10. Peng, S., Niemeyer, M., Mescheder, L.M., Pollefeys, M., Geiger, A.: Convolutional occupancy networks. In: Proc. European Conf. on Computer Vision (ECCV). pp. 523–540 (2020)
11. Qi, C.R., Su, H., Mo, K., Guibas, L.J.: Pointnet: Deep learning on point sets for 3d classification and segmentation. In: Proc. Conf. on Computer Vision and Pattern Recognition (CVPR). pp. 652–660 (2017)
12. Tatarchenko, M., Richter, S.R., Ranftl, R., Li, Z., Koltun, V., Brox, T.: What do single-view 3d reconstruction networks learn? In: Proc. Conf. on Computer Vision and Pattern Recognition (CVPR). pp. 3405–3414 (2019)
13. Wang, D., Cui, X., Chen, X., Zou, Z., Shi, T., Salcudean, S., Wang, Z.J., Ward, R.: Multi-view 3d reconstruction with transformers. In: Proc. Int. Conf. on Computer Vision (ICCV). pp. 5702–5711 (2021)
14. Xie, H., Yao, H., Zhang, S., Zhou, S., Sun, W.: Pix2vox++: Multi-scale context-aware 3d object reconstruction from single and multiple images. International Journal of Computer Vision pp. 2919–2935 (2020)
15. Xu, Q., Wang, W., Ceylan, D., Mech, R., Neumann, U.: Disn: Deep implicit surface network for high-quality single-view 3d reconstruction. Proc. Advances in Neural Information Processing Systems (NeurIPS) **32** (2019)
16. Zhang, B., Nießner, M., Wonka, P.: 3dilg: Irregular latent grids for 3d generative modeling. Proc. Advances in Neural Information Processing Systems (NeurIPS) pp. 21871–21885 (2022)