Supplemental Materials: PVT: An Implicit Surface Reconstruction Framework via Point Voxel Geometric-aware Transformer

Chuanmao Fan * [1], Chenxi Zhao * [2], and Ye Duan †[2]

[1]University of Missouri-Columbia, Columbia, MO, USA
[2]Clemson University, Clemson, SC, USA

Abstract

This supplementary material is organized as follows: section 1 details the network training process. section 2 explain the data preparation methods. Section 3 demonstrate more results on shapenet and scenes reconstruction results.

1. Network and training

Platform and hardware: The proposed PVT network is implemented in pytorch [9]. The training and testing are conducted using a middle-range desktop computer with an Nvidia RTX A5000 GPU of 24 GB memory.

Loss functions: We implement two loss functions for occupancy and unsigned distance field learning, respectively. Binary cross entropy loss of Equation 1 is used for occupancy learning, and truncated regression loss of Equation 2 is used for unsigned distance field learning.

$$L_o(W) = \frac{1}{|B|}\sum_{i=1}^{|B|}\sum_{j=1}^{K}\left|BCE\left(s_{q,i,j}, s_i^j\right)\right| \quad (1)$$

$$L_u(W) = \frac{1}{|B|}\sum_{i=1}^{|B|}\sum_{j=1}^{K}\left|trun\left(s_{q,i,j},\delta\right) - trun\left(|s_i^j|,\delta\right)\right| \quad (2)$$

Here B is the mini-batch data size, K is the number of query points for each object, $s_{q,i,j}$ is the prediction value for a given query point q_i^j, BCE is the binary cross entropy loss for occupancy field, $trun$ is the truncation function with threshold δ, $trunc(x,\delta) := min(\delta, x)$, with the threshold set as 0.15.

Training: The network is trained using the Adam optimizer [6] with parameters $\beta_1 = 0.9, \beta_2 = 0.999$, and an initial learning rate of $1.0e^{-4}$. The learning rate decreases by 0.1× with the step scheduler at 50 and 100 epochs, respectively. We use the same ratio of 7:2:1 for training, validation, and testing for datasets except shapenet. Shapenet datasets inlcude 26K objects. We follow IFnet [5] 's train/val/test split and train only 20 epochs.

Metrics: **Chamfer distance (CD)** as the metric for performance evaluation More specifically, we sample points on both the reconstruction and the ground truth surface to serve as the proxy for computing the chamfer distance between the two surfaces. The chamfer distance between the two shapes represented by point cloud P_a and P_b respectively can thus be measured as the sum of the average of the minimum distances from P_a to P_b and from P_b to P_a. In the paper, we follow ONet [8], IFnet [5], we compute both $CDl1$ and $CDl2$.

$$Chamferl1(P_a, P_b) = \frac{Completeness}{2|P_a|} + \frac{Accuracy}{2|P_b|}$$
$$Chamferl2(P_a, P_b) = \frac{Completeness^2}{2|P_a|} + \frac{Accuracy^2}{2|P_b|} \quad (3)$$

where

*co-first author cf7b6@umsystem.edu, chenxiz@clemson.edu
†Email: duan@clemson.edu

$$
\begin{aligned}
Completness &= \sum_{p_a \in P_a} \min_{p_b \in P_b} d(p_a, p_b) \\
Accuracy &= \sum_{p_b \in P_b} \min_{p_a \in P_a} d(p_b, p_a) \\
Completness^2 &= \sum_{p_a \in P_a} \min_{p_b \in P_b} d(p_a, p_b)^2 \\
Accuracy^2 &= \sum_{p_b \in P_b} \min_{p_a \in P_a} d(p_b, p_a)^2
\end{aligned} \tag{4}
$$

Normal Consistency (NC). The normal consistency between two points cloud P_a and P_b is defined by the following equation:

$$
\begin{aligned}
NC(P_a, P_b) = &\frac{1}{2|P_a|} \sum_{p_a \in P_a} N_{p_a} N_{nearp_a, P_b} \\
&+ \frac{1}{2|P_b|} \sum_{p_b \in P_b} N_{p_b} N_{nearp_b, P_a}
\end{aligned} \tag{5}
$$

where N_{nearp_a,P_b} is the nearest point of p_a of P_a in point cloud P_b. and N_p is the normal of point p on the mesh. **F-Score (FS)**. F-Score between the two point clouds P_a and P_b given a threshold t is defined as follows:

$$
F - Score(P_a, P_b, t) = \frac{2Recall \cdot Precision}{Recall + Precession} \tag{6}
$$

where

$$
Recall(P_a, P_b, t) = |p_a \in P_a, s.t. \min_{p_b \in P_b} d(p_a, p_b)| \tag{7}
$$

$$
Precision(P_a, P_b, t) = |p_b \in P_b, s.t. \min_{p_a \in P_a} d(p_b, p_a)| \tag{8}
$$

We follows ONet [8], ConvONet [10] and POCO [3], we set t = 1%, 0.5%.

Intersection over Union (IoU) measure the volumetric alignment between the predicted mesh and ground truth mesh. We basically sample a large number of points in unite cube of the reconstruction volume. and then count the number of points that lie in or outside of the predicted mesh and ground truth mesh. then the IOU is computed as follows:

$$
IoU(M_a, M_b) = \frac{TP}{TP + FP + FN} \tag{9}
$$

where TP (resp. FP, FN) are the number of the true positive points i.e. those correctly predicted as inside occupancy (reps. the number of points wrongly predicted as inside actually being outside points, and the number of points wrongly predicted as outside but actually being inside of the ground truth mesh). We sample one Million points within the reconstruction unit volume for this IOU measurement.

2. Data and processing

For training, we prepare three types of data for a given mesh object:
1. N input points will be sampled from the given mesh. The given mesh is normalized to [-0.5, 0.5] before sampling.
2. K query points will be generated by adding isotropic Gaussian noise displacement $n \sim N(0, \Sigma)$ to each sampled surface point, *i.e.* $q = p + n$, where $\Sigma \in R^{3\times3}$ is the diagonal covariance matrix with variance setting $\Sigma_{0,0} = \Sigma_{1,1} = \Sigma_{2,2} = \sigma$ defining the displacement scales. We prepare three sets of query points $K1, K2$, and $K3$, with 500,000 points in each set, and σ equals to $0.25, 0.02, 0.003$, respectively for each mesh object. We then randomly pick 15%, 35%, and 50% from $K1, K2$, and $K3$, respectively, and combine them together as the final K = 0.15×$K1$ + 0.35×$K2$ + 0.50×$K3$ query points for each object for training.
3. Ground truth occupancy and distance value of every query point for occupancy field and unsigned distance field training, respectively.

Shapenet: The shapenet dataset [13] contains 13 classes of objects with watertight surface. Each object will be normalized to a unit sphere with 3000 points sampled for each object mesh.

Shapenet car with complex inner structures: The shapenet car dataset [4] contains shapes with complex inner structures. There is a total of 5756 objects in the dataset with 10,000 points sampled for each object.

Garments/FAUST: In order to evaluate the performance of open surface reconstruction, we collect 307 garments data from MGN [1], with 3000 sampled points for each object. To enlarge the size of the training data, we further combine it with the FAUST [2] dataset, which contains 300 real, high-resolution human scans of 10 different persons in 30 different poses.

Gibson: Gibson [12] is a large-scale indoor scene 3D dataset collected with 3D scanning devices. We pick 35 scenes from the dataset and divide each scene into 2.5 cubemeter blocks. In training, the blocks are normalized to the range of [-0.5, 0.5] meters. During testing, blocks are processed individually and merged back into the final scene reconstruction.

	Model size (MB)	field construction (Second)	surface reconstruction (Second)	total (Second)
GeoUDF [11]	3.02	1.42	23.20	24.62
GridFormer [7]	52.00	2.78	1.81	4.59
Ours (UDF)	63.69	0.03*	3.22	3.25
Ours(Occupancy)	63.69	1.70	0.23	1.93

Table 1. Computation complexity comparison between baselines and our methods. Model size is in MB. The computation time consists of 3D distnace field construction in seconds, surface construction time in seconds and total processing time in seconds. *Our UDF model only has input point encoding time in field construction, while surface reconstruction time is dense surface point generation time

3. More results

Table 1 shows comparison of computation complexity between the proposed methods and baselines.

Shapenet: Figure 1- 4 show more qualitative results on Shapenet.

Scene:Figure 5 and 6 shows more qualitative results on Scene.

References

[1] Bharat Lal Bhatnagar, Garvita Tiwari, Christian Theobalt, and Gerard Pons-Moll. Multi-garment net: Learning to dress 3d people from images. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 5420–5430, 2019. 2

[2] Federica Bogo, Javier Romero, Matthew Loper, and Michael J Black. Faust: Dataset and evaluation for 3d mesh registration. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3794–3801, 2014. 2

[3] Alexandre Boulch and Renaud Marlet. Poco: Point convolution for surface reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6302–6314, 2022. 2

[4] Angel X Chang, Thomas Funkhouser, Leonidas Guibas, Pat Hanrahan, Qixing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, et al. Shapenet: An information-rich 3d model repository. *arXiv preprint arXiv:1512.03012*, 2015. 2

[5] Julian Chibane, Thiemo Alldieck, and Gerard Pons-Moll. Implicit functions in feature space for 3d shape reconstruction and completion. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6970–6981, 2020. 1

[6] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 1

[7] Shengtao Li, Ge Gao, Yudong Liu, Yu-Shen Liu, and Ming Gu. Gridformer: Point-grid transformer for surface reconstruction. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2024. 3

[8] Lars Mescheder, Michael Oechsle, Michael Niemeyer, Sebastian Nowozin, and Andreas Geiger. Occupancy networks: Learning 3d reconstruction in function space. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4460–4470, 2019. 1, 2

[9] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32, 2019. 1

[10] Songyou Peng, Michael Niemeyer, Lars Mescheder, Marc Pollefeys, and Andreas Geiger. Convolutional occupancy networks. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part III 16*, pages 523–540. Springer, 2020. 2

[11] Siyu Ren, Junhui Hou, Xiaodong Chen, Ying He, and Wenping Wang. Geoudf: Surface reconstruction from 3d point clouds via geometry-guided distance representation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14214–14224, 2023. 3

[12] Fei Xia, Amir R. Zamir, Zhi-Yang He, Alexander Sax, Jitendra Malik, and Silvio Savarese. Gibson env: real-world perception for embodied agents. In *Computer Vision and Pattern Recognition (CVPR), 2018 IEEE Conference on*. IEEE, 2018. 2, 8, 9

[13] Qiangeng Xu, Weiyue Wang, Duygu Ceylan, Radomir Mech, and Ulrich Neumann. Disn: Deep implicit surface network for high-quality single-view 3d reconstruction. *Advances in neural information processing systems*, 32, 2019. 2

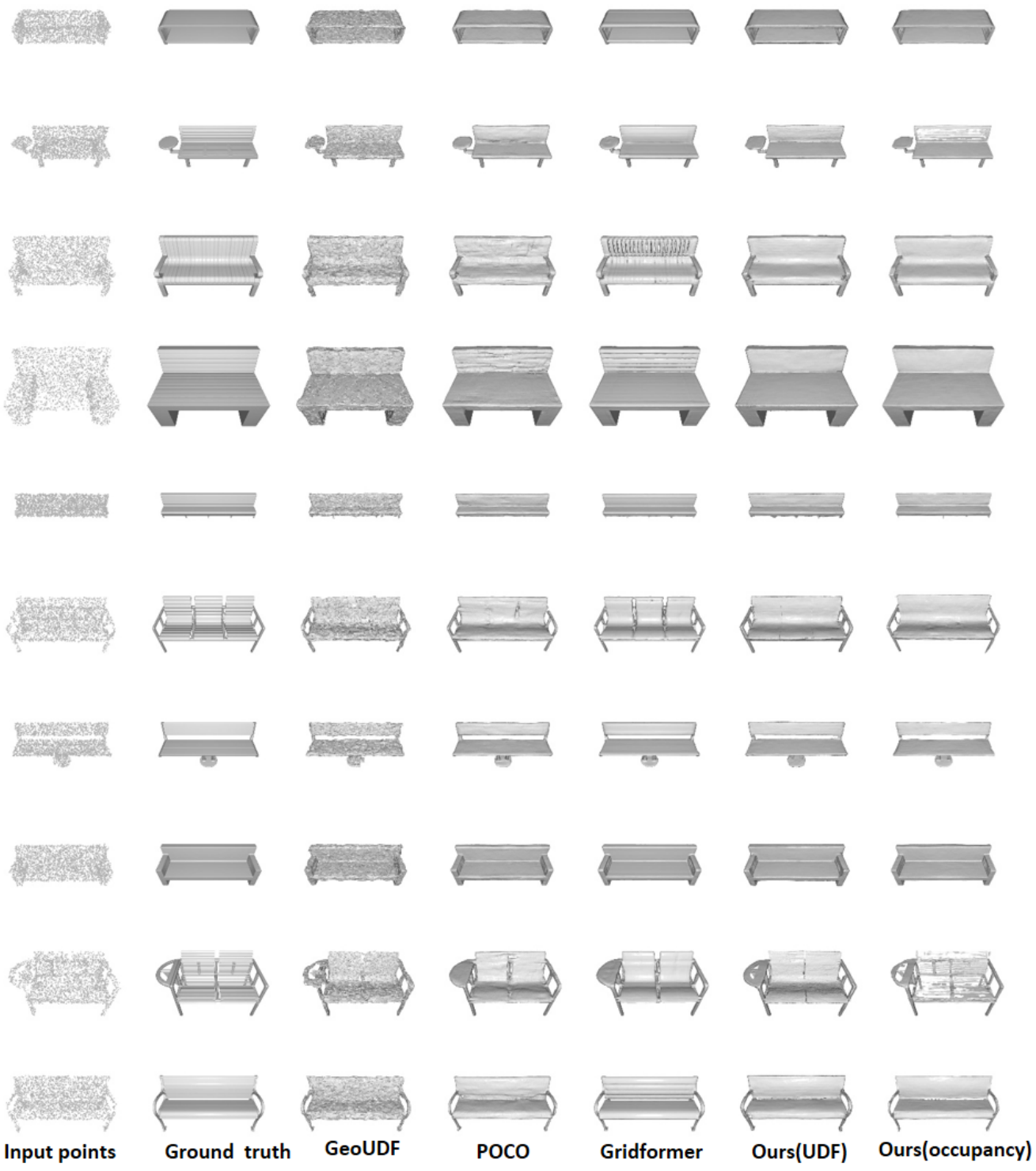

Figure 1. Shapenet reconstruction with 3000 input points and Gaussian noise of standard deviation 0.005. From left to right: input points, groundtruth, IFnet, POCO, GridFormer, Ours

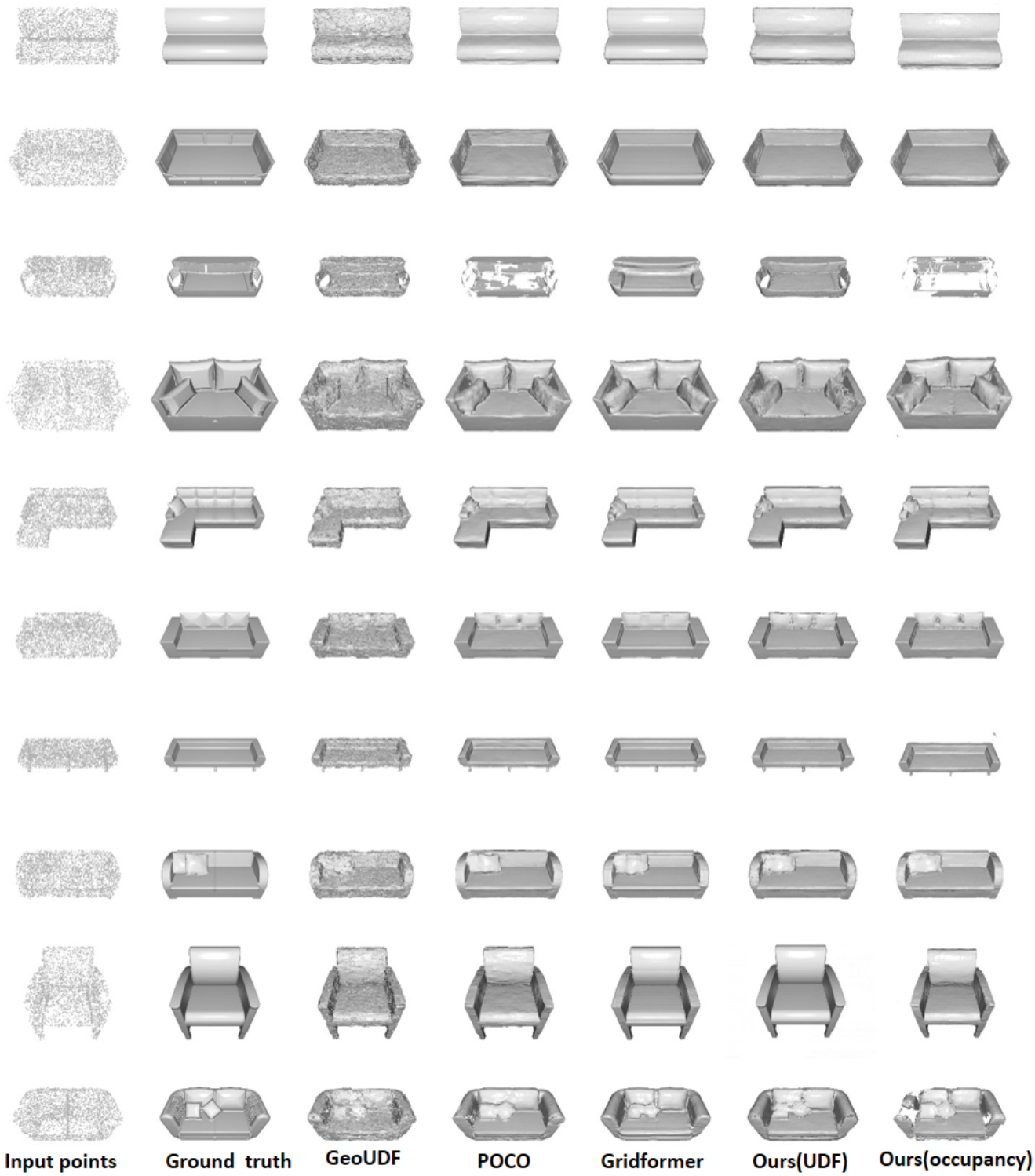

Figure 2. Shapenet reconstruction with 3000 input points and Gaussian noise of standard deviation 0.005. From left to right: input points, groundtruth, IFnet, POCO, GridFormer, Ours

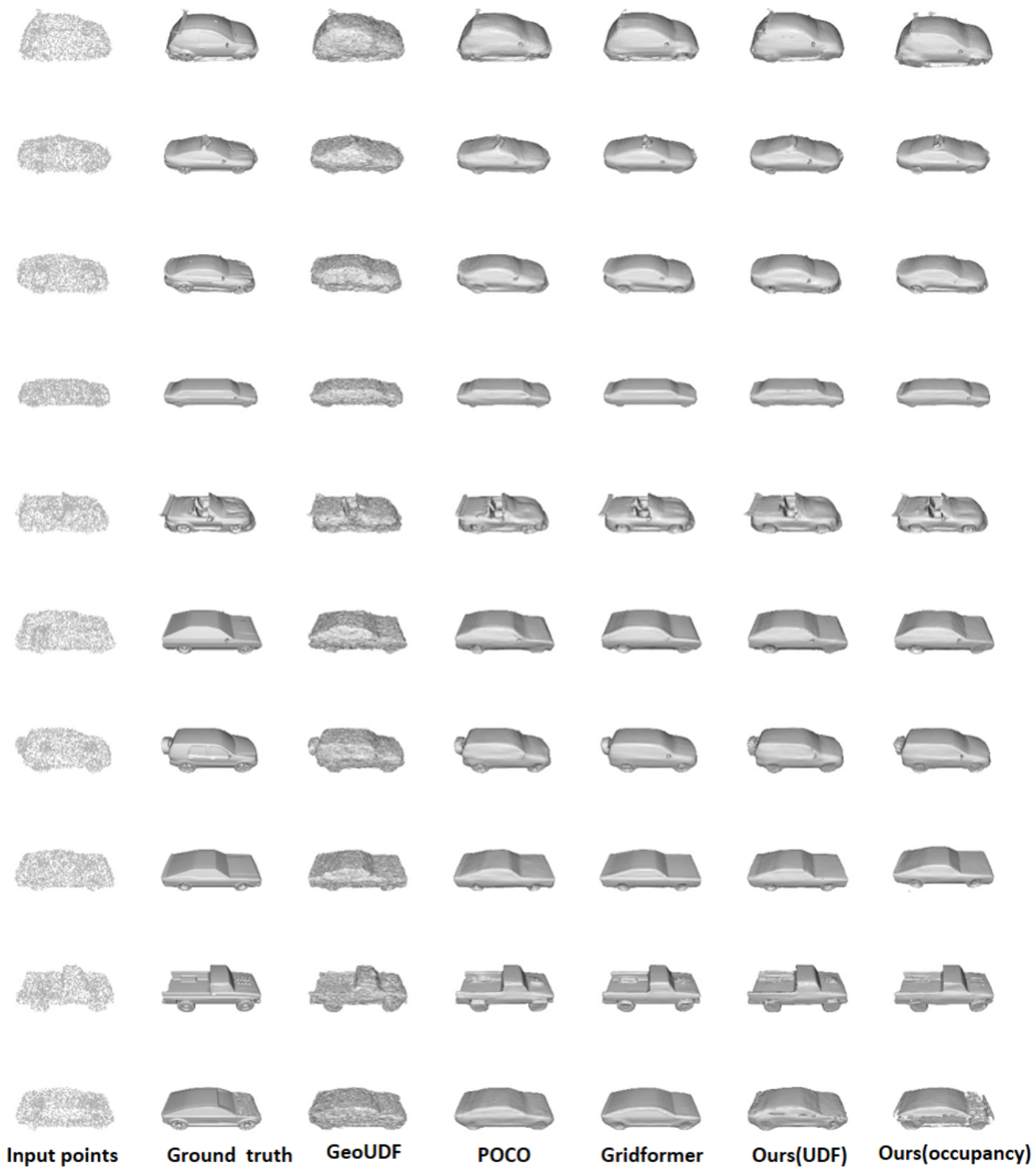

Figure 3. Shapenet reconstruction with 3000 input points and Gaussian noise of standard deviation 0.005. From left to right: input points, groundtruth, IFnet, POCO, GridFormer, Ours

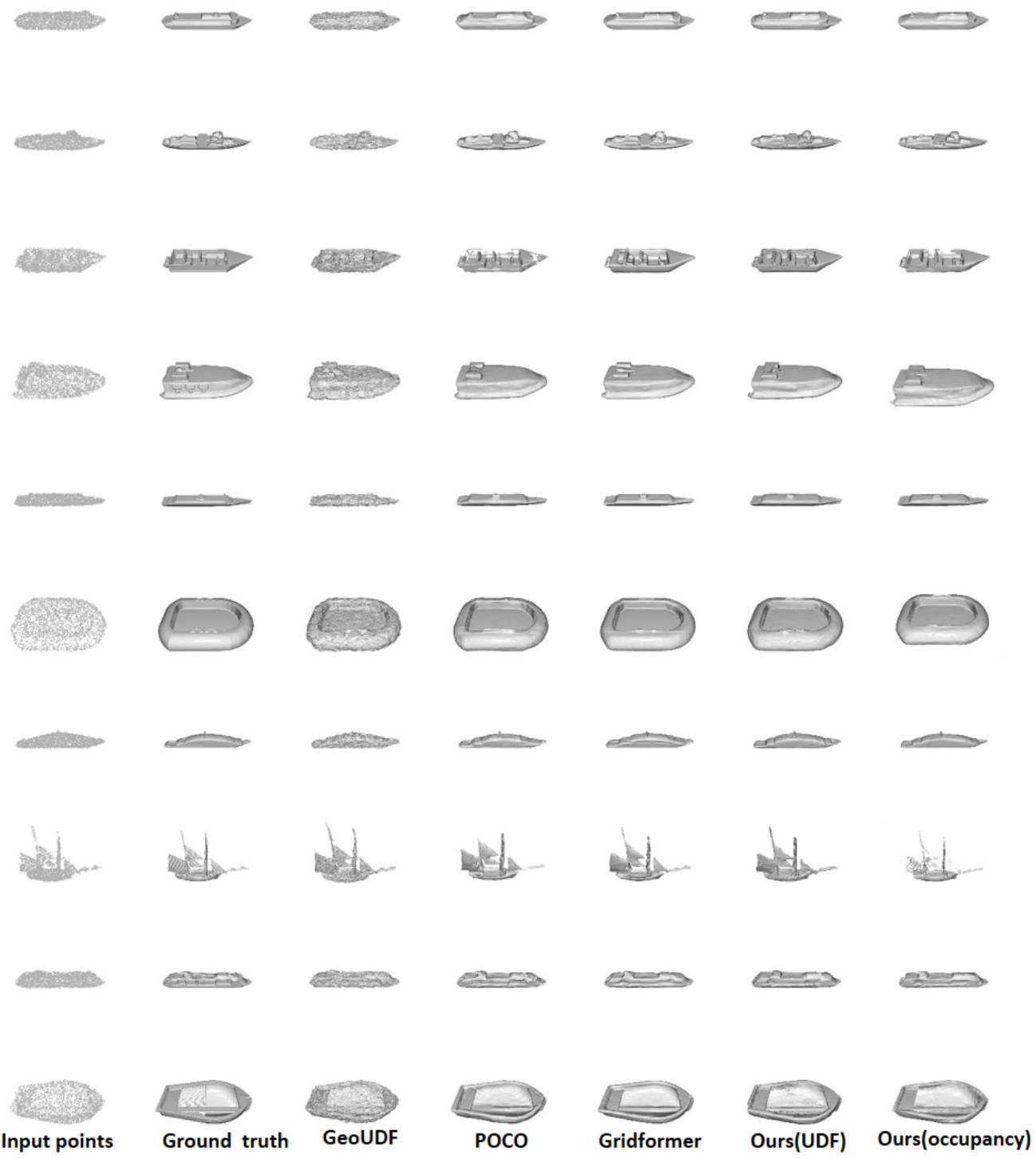

Figure 4. Shapenet reconstruction with 3000 input points and Gaussian noise of standard deviation 0.005. From left to right: input points, groundtruth, IFnet, POCO, GridFormer, Ours

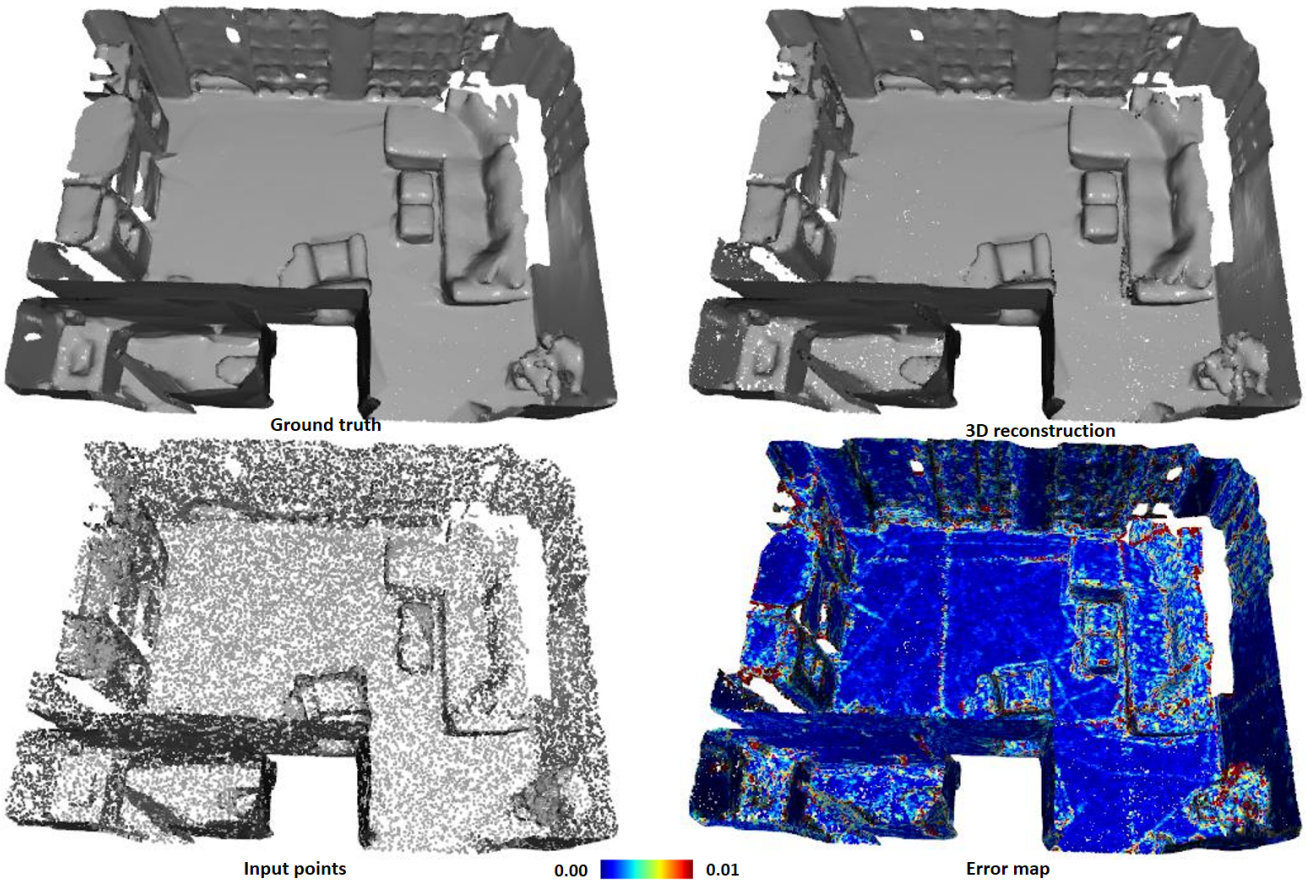

Figure 5. Large scale indoor scene reconstruction result of the Gibson dataset [12]. From top to bottom: input points, ground truth mesh, reconstruction, error map. Here the error map is measured in the meter scale, e.g. ”0.01” in the color bar means an error of one centimeter.

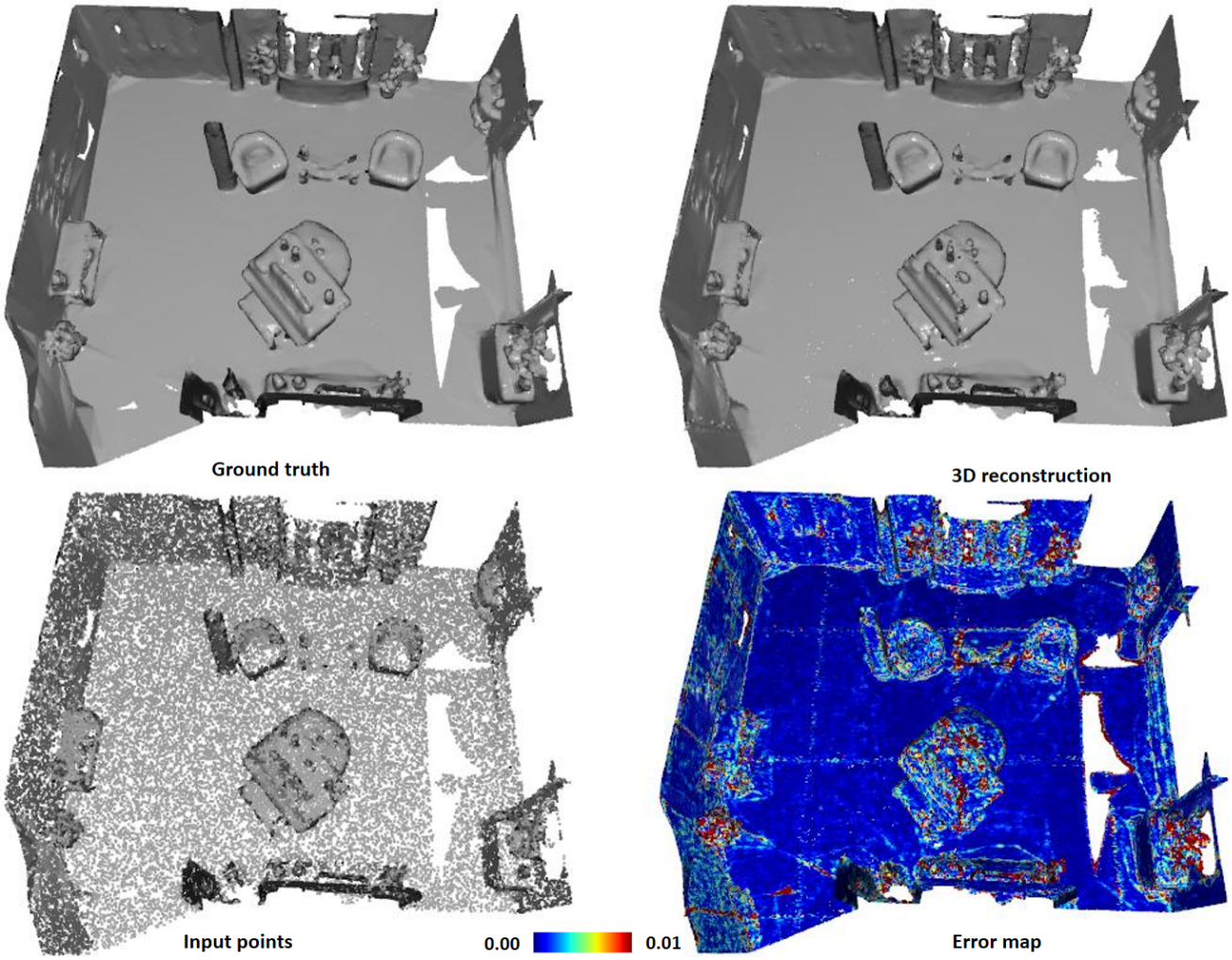

Figure 6. Large scale indoor scene reconstruction result of the Gibson dataset [12]. From top to bottom: input points, ground truth mesh, reconstruction, error map. Here the error map is measured in the meter scale, e.g. ”0.01” in the color bar means an error of one centimeter.