ALTO: Alternating Latent Topologies for Implicit 3D Reconstruction
Supplemental Material

Zhen Wang[1*] Shijie Zhou[1*] Jeong Joon Park[2] Despoina Paschalidou[2]
Suya You[3] Gordon Wetzstein[2] Leonidas Guibas[2] Achuta Kadambi[1]
[1]University of California, Los Angeles [2]Stanford University [3]DEVCOM Army Research Laboratory

Supplementary Content

This supplement is organized as follows:

- Section A contains network architecture details;
- Section B contains more details on the training and inference settings;
- Section C contains more ablation studies of our method;
- Section D contains both quantitative and qualitative results on ShapeNet dataset;
- Section E contains more qualitative results on Synthetic Room dataset;
- Section F contains additional qualitative results on ScanNet dataset;
- Section G contains evaluation of different noise levels;
- Section H contains the code link of the comparison baselines; and
- Section I contains discussion on the limitation of our method and future work.

A. Network Architecture

PointNet: Given the input un-oriented point cloud $\mathcal{P} = \{\boldsymbol{p}_i \in \mathbb{R}^3\}_{i=1}^{S}$, where S is the number of input points, we map the input coordinates to point features using a fully-connected layer and a ResNet-FC [3] block. Instead of using global features as in [11], we use locally-pooled features to fuse local features. Specifically, we aggregate features within the same plane or voxel cell from a 2D triplanar or 3D volumetric grids using max-pooling. We concatenate the locally pooled features with the feature before pooling and then input to the next ResNet block. To obtain the final point features, there are totally 5 ResNet blocks used.

Our ALTO U-Net: Our alternation U-Net architecture is similar to traditional U-Net [2, 12], except that we replace the convolution-only block with our ALTO block where point and grid (either 2D or 3D) features are converted back and forth as depicted in Fig. 3 of main paper. The input and output feature dimensions is set to be 32. There is no ALTO block in the final block of the U-Net.

Our Attention-based Decoder: For the triplane representation, we implement 3 single-head attention for 3 feature planes respectively, where the hidden dimension is equal to the feature dimension 32. For the volume representation, we implement a multi-head attention with h heads. To maximize the flexibility of our method for different datasets and experiments, we set the number of heads h as a hyperparameter and the hidden dimension as $h \times$ feature dimension(32). The following occupancy network consisting of 5 stacked ResNet-FC blocks with skip connections is used to predict the occupancy probability of query point features. For all experiments, we use a hidden dimension equal to the attention output feature dimension and 5 ResNet blocks for the occupancy network.

*Equal contribution.

Total # of alternation blocks	IoU ↑	Chamfer-L_1 ↓	NC↑	F-score↑
0	0.831	0.55	0.912	0.892
3	0.847	0.50	0.914	0.910
6	**0.863**	**0.47**	**0.922**	**0.924**

Table A. **Ablation study of total number of ALTO alternation blocks on ShapeNet dataset with 300 input points.**

Method	IoU ↑	Chamfer-L_1 ↓	NC↑	F-score↑
ConvONet (3×128^2) [10]	0.805	0.44	0.903	0.948
ConvONet (64^3) [10]	0.849	0.42	0.915	0.964
ALTO (3×128^2, Encoder Only)	0.834	0.43	0.906	0.960
ALTO (3×128^2)	0.895	0.37	0.910	0.974
ALTO (64^3, Encoder Only)	0.903	0.36	0.920	0.978
ALTO (64^3)	**0.914**	**0.35**	**0.921**	**0.981**

Table B. **Ablation study of our attention-based decoder for different latent topologies used (i.e. point-triplane and point-voxel alternations) on Synthetic Room dataset.** Input points 10K with noise added. Boldface font represents the preferred results.

B. Training and Inference Details

Object-Level Reconstruction: For object-level reconstruction in ShapeNet, we use alternation between latent topologies: point and triplane, because triplane representation is found to tend to give better results for object-level reconstruction in ConvONet [10]. The dimension of each 2D feature plane is set as 64^2. The depth of our ALTO U-Net is 4, and we do not downsample or upsample in the top two levels of the U-Net, so the lowest resolution of the U-Net is 16^2.

Scene-Level Reconstruction: For scene-level reconstruction, we use alternation between two topologies: point and feature volume. The dimension of the feature volume is set as 64^3. The depth of our ALTO U-Net is 4, and similarly we do not downsample or upsample in the top two levels, so the lowest resolution of the U-Net is 16^3. At decoder stage, we set the hyperparameter $h = 4$ for experiments on Synthetic Room dataset and $h = 1$ for experiments on ScanNet dataset which we find the best performance in practice.

Mesh Generation: We use a form of Marching Cubes (MC) [7] to evaluate occupancy values from implicit representations on a 3D grid. As a result of Marching Cube, the vertices are usually placed in the middle of segments, which causes discretization effects [1]. To deal with this issue, we apply the refinement method from POCO [1], which takes both the generated vertices and their floor to predict their occupancy values again. After that, we compare two values, mask out non-perfect vertices, take the average between the generated vertices and their floor, and repeat 10 times to improve the granularity. For object-level reconstruction, we use resolution 128 and for scene-level reconstruction, we use resolution 256 for marching cubes.

Hardware: We describe the detailed setups that have been used for inference evaluation:

- CUDA version: 11.1
- PyTorch version: 1.9.0
- GPU: single NVIDIA GeForce RTX 3090
- CPU: AMD RYZEN PRO 3955WX 16-Cores CPU

C. Ablation Studies

C.a. Alternation blocks

In Tab. A, we report the performance of method with different number of alternations between point and grid forms within each block in the ALTO U-Net. 0 represents no point-grid alternations (i.e. staying with only grid form), 3 represents

that there is only point-grid alternation in the top two levels of our ALTO U-Net, and 6 represents that there is point-grid alternations in each level of our ALTO U-Net. As we can see the results, we can observe the trend that increasing the number of ALTO blocks improves the results for all the metrics.

C.b. Attention-based decoder

We also report the results of the ablation study of our attention-based decoder on synthetic room dataset in Tab. B. As demonstrated in the table, with our attention-based decoder, it improves results for both triplanar (3×128^3) and volumetric representations (64^3).

C.c. Alternating vs parallel latent topologies

We experimentally show that our alternating strategy significantly outperforms the simultaneous strategy (Tab. C). We believe that alternating strategy's *information exchange* between the *points* and *grids* in each layer is the critical factor.

	ShapeNet dataset				Synthetic Room dataset			
Method	IoU↑	Chamfer-L_1↓	NC↑	F-score↑	IoU↑	Chamfer-L_1↓	NC↑	F-score↑
ALTO w/ parallel topologies	0.873	0.47	0.935	0.931	0.832	0.42	0.919	0.960
ALTO	**0.905**	**0.35**	**0.940**	**0.964**	**0.914**	**0.35**	**0.921**	**0.981**

Table C. **Alternating topologies show better performance than the parallel latent topologies.** Boldface font represents the preferred results.

C.d. Skip connections

In Tab. D, we show that consecutive layers and UNet-style skip connections are important.

Method	IoU ↑	Chamfer-L_1↓	NC↑	F-score↑
ALTO w/o layer skip	0.890	0.40	0.932	0.952
ALTO w/o UNet skip	0.898	0.37	0.936	0.959
ALTO	**0.905**	**0.35**	**0.940**	**0.964**

Table D. **Ablations for skip connections.**

D. Additional Results on ShapeNet

D.a. Quantitative results

We show per-category quantitative results in ShapeNet with various point density levels: 3K input points (Tab. E), 1K input points (Tab. F) and 300 input points (Tab. G). It is notable that when point clouds get sparser, ALTO performs better than POCO on all four metrics for all categories. We also show comparison with SAP [9] in Tab. H.

D.b. Qualitative results

Besides 1K input points for ShapeNet as we show in Fig. 6 of the main paper, we show additional qualitative results in ShapeNet with 3K input points in Fig. A and 300 input points in Fig. B.

E. Additional Results on Synthetic Room Dataset

We show additional qualitative results in Synthetic Room dataset with 10K inputs points in Fig. C and 3K inputs points in Fig. D.

Additionally, we compare our method to SA-ConvONet [13] (c.f. Tab. I) under the same settings. Note that our method performs better than SA-ConvONet with 10K input points, while SA-ConvONet takes 30K. We also outperform 3D-ILG [15] (IoU **0.919** vs 0.866; Ch-L_1 0.34 vs **0.31**; F-score **0.983** vs 0.979) on IoU and F-score, though having less input points (ALTO 10,000 vs 3D-ILG 16,384).

Method	IoU ↑				Chamfer-L_1 ↓			
	ONet [8]	ConvONet [10]	POCO [1]	ALTO	ONet [8]	ConvONet [10]	POCO [1]	ALTO
Airplane	0.734	0.849	0.902	**0.908**	0.64	0.34	0.23	**0.22**
Bench	0.682	0.830	0.865	**0.890**	0.67	0.35	0.28	**0.26**
Cabinet	0.855	0.940	0.960	**0.965**	0.82	0.46	0.37	**0.34**
Car	0.830	0.886	0.921	**0.924**	1.04	0.75	**0.41**	0.43
Chair	0.720	0.871	0.919	**0.925**	0.95	0.46	0.33	**0.32**
Display	0.799	0.927	0.956	**0.962**	0.82	0.36	0.28	**0.27**
Lamp	0.546	0.785	**0.877**	0.868	1.59	0.59	**0.33**	0.34
Loudspeaker	0.826	0.918	**0.957**	0.953	1.18	0.64	**0.41**	**0.41**
Rifle	0.668	0.846	0.897	**0.898**	0.66	0.28	**0.19**	**0.19**
Sofa	0.865	0.936	0.963	**0.966**	0.73	0.42	0.30	**0.29**
Table	0.739	0.888	0.924	**0.937**	0.76	0.38	0.31	**0.29**
Telephone	0.896	0.955	0.968	**0.977**	0.46	0.27	0.22	**0.21**
Vessel	0.729	0.865	**0.927**	0.924	0.94	0.43	**0.25**	0.26
mean	0.761	0.884	0.926	**0.931**	0.87	0.44	**0.30**	**0.30**
Method	NC ↑				F-score ↑			
	ONet [8]	ConvONet [10]	POCO [1]	ALTO	ONet [8]	ConvONet [10]	POCO [1]	ALTO
Airplane	0.886	0.931	0.944	**0.949**	0.829	0.965	**0.994**	0.992
Bench	0.871	0.921	0.928	**0.941**	0.827	0.964	0.988	**0.991**
Cabinet	0.913	0.956	0.961	**0.967**	0.833	0.956	0.979	**0.982**
Car	0.874	0.893	0.894	**0.917**	0.747	0.849	**0.946**	0.940
Chair	0.886	0.943	0.956	**0.959**	0.730	0.939	**0.985**	**0.985**
Display	0.926	0.968	0.975	**0.976**	0.795	0.971	**0.994**	0.993
Lamp	0.809	0.900	**0.929**	0.924	0.581	0.892	**0.975**	0.962
Loudspeaker	0.903	0.939	**0.952**	0.951	0.727	0.892	**0.964**	0.955
Rifle	0.849	0.929	**0.949**	**0.949**	0.818	0.980	**0.998**	0.996
Sofa	0.928	0.958	0.967	**0.971**	0.832	0.953	**0.989**	0.987
Table	0.917	0.959	0.966	**0.968**	0.824	0.967	**0.991**	0.990
Telephone	0.970	0.983	0.985	**0.987**	0.930	0.989	**0.998**	**0.998**
Vessel	0.857	0.919	**0.940**	**0.940**	0.734	0.931	**0.989**	0.982
mean	0.891	0.938	0.950	**0.954**	0.785	0.942	**0.984**	0.981

Table E. **Performance on ShapeNet with input noisy point cloud 3K.** Boldface font represents the preferred results.

F. Additional Results on ScanNet

We demonstrate the Sim2Real qualitative results with the model trained on Synthetic Room dataset and tested on ScanNet in Fig. 8 of the main paper. We show in Fig. E of the supplement material the Sim2Real results with different point density levels (i.e. N_{Train}=10k, N_{Test}=3k) to further demonstrate the generalization capability of our method ALTO.

G. Evaluate robustness to different noise levels

We test our model with two additional noise levels, 0.0 and 0.25, which shows that ALTO outperforms ConvONet [10] and POCO [1] on all metrics (Tab. J). ALTO also outperforms SAP [9] (Chamfer-L_1: 0.54, F-score: 0.896, NC: 0.917) at the high noise level of 0.25, even though SAP is specifically designed to cope with high noise.

H. Comparison Code Links

We list all the links of the code of the comparisons baselines in Tab. K. Our code is available at https://visual.ee.ucla.edu/alto.htm/.

I. Limitation and Future Work

For our current method, we are not learning a probabilistic generative model that can learn the distribution of the input data, which limits the diversity of the shapes our model can generate. Moreover, we are uniformly sampling points as in previous work such as [10]. More efficient sampling strategy that samples more points on densely populated regions and less on sparsely populated regions can be adopted to capture more details on the fine-grained areas.

	IoU ↑				Chamfer-L_1 ↓			
Method	ONet [8]	ConvONet [10]	POCO [1]	ALTO	ONet [8]	ConvONet [10]	POCO [1]	ALTO
Airplane	0.748	0.825	0.850	**0.872**	0.59	0.39	0.32	**0.29**
Bench	0.702	0.798	0.804	**0.856**	0.62	0.40	0.38	**0.30**
Cabinet	0.862	0.926	0.936	**0.953**	0.76	0.50	0.46	**0.37**
Car	0.837	0.867	0.878	**0.901**	0.99	0.83	0.60	**0.50**
Chair	0.736	0.837	0.867	**0.894**	0.89	0.55	0.44	**0.39**
Display	0.812	0.911	0.930	**0.946**	0.78	0.41	0.34	**0.31**
Lamp	0.567	0.741	0.807	**0.820**	1.44	0.68	**0.50**	**0.50**
Loudspeaker	0.831	0.899	0.923	**0.933**	1.14	0.72	0.54	**0.48**
Rifle	0.680	0.801	0.850	**0.862**	0.63	0.36	0.27	**0.25**
Sofa	0.873	0.921	0.937	**0.952**	0.69	0.47	0.38	**0.33**
Table	0.757	0.858	0.880	**0.913**	0.70	0.44	0.38	**0.33**
Telephone	0.897	0.946	0.953	**0.968**	0.46	0.29	0.26	**0.23**
Vessel	0.736	0.840	0.880	**0.893**	0.91	0.51	0.37	**0.33**
mean	0.772	0.859	0.884	**0.905**	0.82	0.50	0.40	**0.35**
	NC ↑				F-score ↑			
Method	ONet [8]	ConvONet [10]	POCO [1]	ALTO	ONet [8]	ConvONet [10]	POCO [1]	ALTO
Airplane	0.894	0.922	0.920	**0.933**	0.850	0.946	0.970	**0.976**
Bench	0.882	0.911	0.902	**0.925**	0.849	0.943	0.956	**0.979**
Cabinet	0.925	0.949	0.945	**0.957**	0.852	0.939	0.951	**0.972**
Car	0.904	0.885	0.867	**0.889**	0.763	0.819	0.868	**0.912**
Chair	0.893	0.931	0.930	**0.946**	0.753	0.902	0.943	**0.965**
Display	0.930	0.961	0.962	**0.970**	0.805	0.956	0.976	**0.984**
Lamp	0.820	0.885	0.895	**0.905**	0.606	0.845	0.924	**0.926**
Loudspeaker	0.914	0.929	0.928	**0.936**	0.740	0.863	0.908	**0.926**
Rifle	0.859	0.916	0.928	**0.936**	0.828	0.957	0.984	**0.987**
Sofa	0.937	0.950	0.950	**0.960**	0.846	0.932	0.961	**0.974**
Table	0.918	0.950	0.949	**0.961**	0.842	0.947	0.964	**0.979**
Telephone	0.972	0.980	0.979	**0.984**	0.940	0.983	0.990	**0.994**
Vessel	0.866	0.906	0.913	**0.923**	0.740	0.899	0.952	**0.961**
mean	0.901	0.929	0.928	**0.940**	0.801	0.918	0.950	**0.964**

Table F. **Performance on ShapeNet with input noisy point cloud 1K.** Boldface font represents the preferred results.

As our method is general in encoding 3D point features, it can be generalized to not just occupancy fields, but also radiance fields trained from images. Similarly, it can be applied to a broader range of neural fields such as semantic field [14] and affordance field [4].

	IoU ↑				Chamfer-L_1 ↓			
Method	ONet [8]	ConvONet [10]	POCO [1]	ALTO	ONet [8]	ConvONet [10]	POCO [1]	ALTO
Airplane	0.760	0.782	0.744	**0.825**	0.57	0.48	0.57	**0.39**
Bench	0.716	0.743	0.707	**0.801**	0.60	0.50	0.56	**0.39**
Cabinet	0.867	0.900	0.889	**0.927**	0.73	0.52	0.58	**0.46**
Car	0.834	0.843	0.817	**0.867**	0.99	0.76	0.83	**0.67**
Chair	0.736	0.787	0.776	**0.840**	0.89	0.67	0.71	**0.52**
Display	0.817	0.885	0.878	**0.917**	0.76	0.47	0.49	**0.38**
Lamp	0.567	0.663	0.681	**0.747**	1.38	1.02	0.93	**0.76**
Loudspeaker	0.827	0.870	0.867	**0.901**	1.16	0.78	0.79	**0.64**
Rifle	0.691	0.757	0.742	**0.801**	0.61	0.43	0.45	**0.35**
Sofa	0.872	0.898	0.893	**0.926**	0.69	0.52	0.53	**0.42**
Table	0.758	0.813	0.794	**0.868**	0.72	0.52	0.57	**0.42**
Telephone	0.916	0.939	0.927	**0.952**	0.41	0.31	0.33	**0.27**
Vessel	0.748	0.797	0.795	**0.846**	0.85	0.63	0.60	**0.47**
mean	0.778	0.821	0.808	**0.863**	0.80	0.59	0.61	**0.47**
	NC ↑				F-score ↑			
Method	ONet [8]	ConvONet [10]	POCO [1]	ALTO	ONet [8]	ConvONet [10]	POCO [1]	ALTO
Airplane	0.897	0.901	0.867	**0.914**	0.864	0.902	0.867	**0.938**
Bench	0.878	0.886	0.864	**0.906**	0.860	0.912	0.882	**0.947**
Cabinet	0.916	0.931	0.917	**0.943**	0.856	0.916	0.896	**0.943**
Car	0.875	0.864	0.835	**0.873**	0.757	0.810	0.766	**0.850**
Chair	0.889	0.905	0.885	**0.923**	0.754	0.850	0.833	**0.910**
Display	0.926	0.947	0.938	**0.956**	0.813	0.926	0.916	**0.957**
Lamp	0.813	0.853	0.834	**0.875**	0.618	0.771	0.781	**0.857**
Loudspeaker	0.897	0.911	0.897	**0.916**	0.737	0.832	0.819	**0.871**
Rifle	0.863	0.890	0.883	**0.909**	0.838	0.919	0.918	**0.952**
Sofa	0.928	0.935	0.924	**0.946**	0.846	0.906	0.899	**0.941**
Table	0.917	0.933	0.917	**0.945**	0.839	0.913	0.894	**0.947**
Telephone	0.970	0.975	0.970	**0.978**	0.942	0.975	0.971	**0.984**
Vessel	0.860	0.879	0.867	**0.898**	0.758	0.850	0.851	**0.909**
mean	0.895	0.908	0.892	**0.922**	0.806	0.883	0.869	**0.924**

Table G. **Performance on ShapeNet with input noisy point cloud 300.** Boldface font represents the preferred results.

	Ch-L_1↓	F-score↑	NC↑
SAP [9]	0.34	0.975	0.944
ALTO	**0.30**	**0.980**	**0.952**

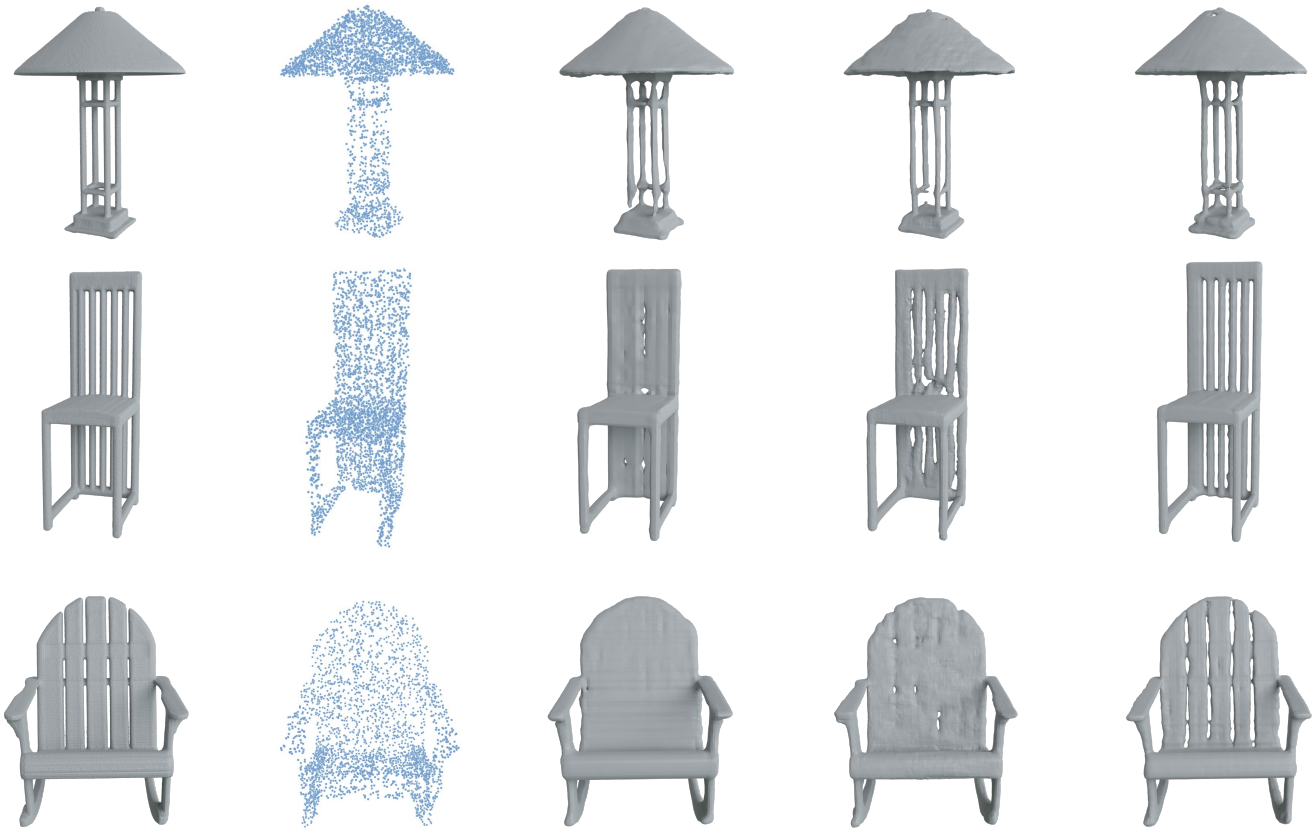
Table H. **Comparison with additional baseline: SAP [9]**

	Ch-L_1↓	NC↑	FS (τ)↑	FS (2τ)↑
SA-ConvONet [13]	0.495	90.04	93.85	98.82
ALTO	**0.348**	**92.03**	**98.23**	**99.63**

Table I. **Comparison with additional baseline: SA-ConvONet [13]**

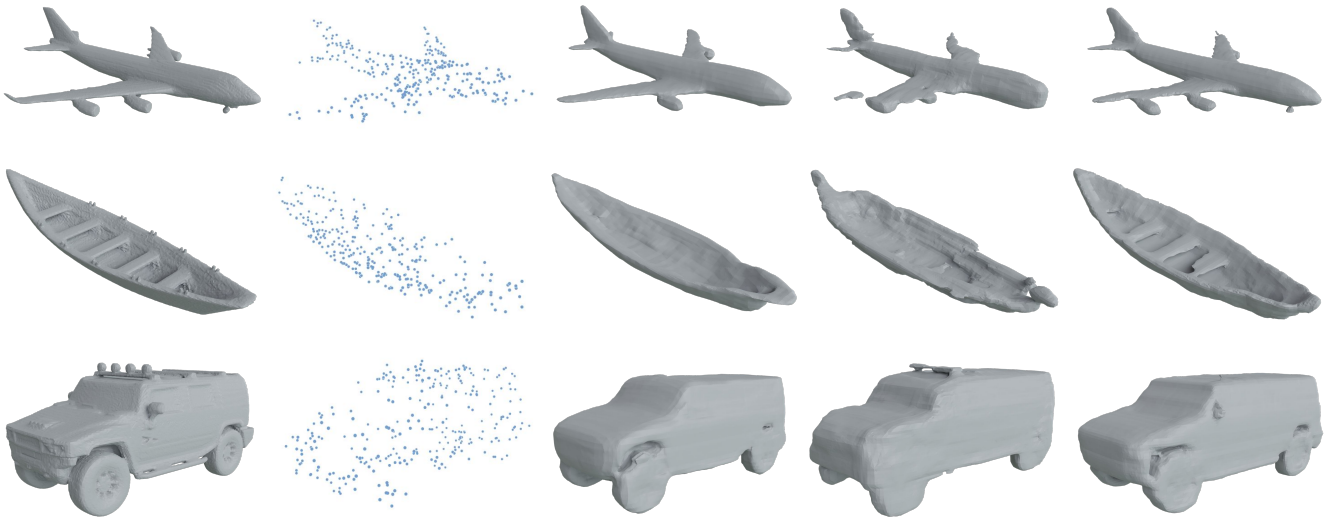
	noise level 0			noise level 0.25		
Method	Chamfer-L_1↓	F-score↑	NC↑	Chamfer-L_1↓	F-score↑	NC↑
ConvONet [10]	0.40	0.958	0.929	0.66	0.849	0.913
POCO [1]	0.28	0.983	**0.958**	0.66	0.846	0.893
ALTO	**0.27**	**0.984**	**0.958**	**0.53**	**0.901**	**0.918**

Table J. **Performance comparison on different noise levels on ShapeNet dataset with 3K input points.**



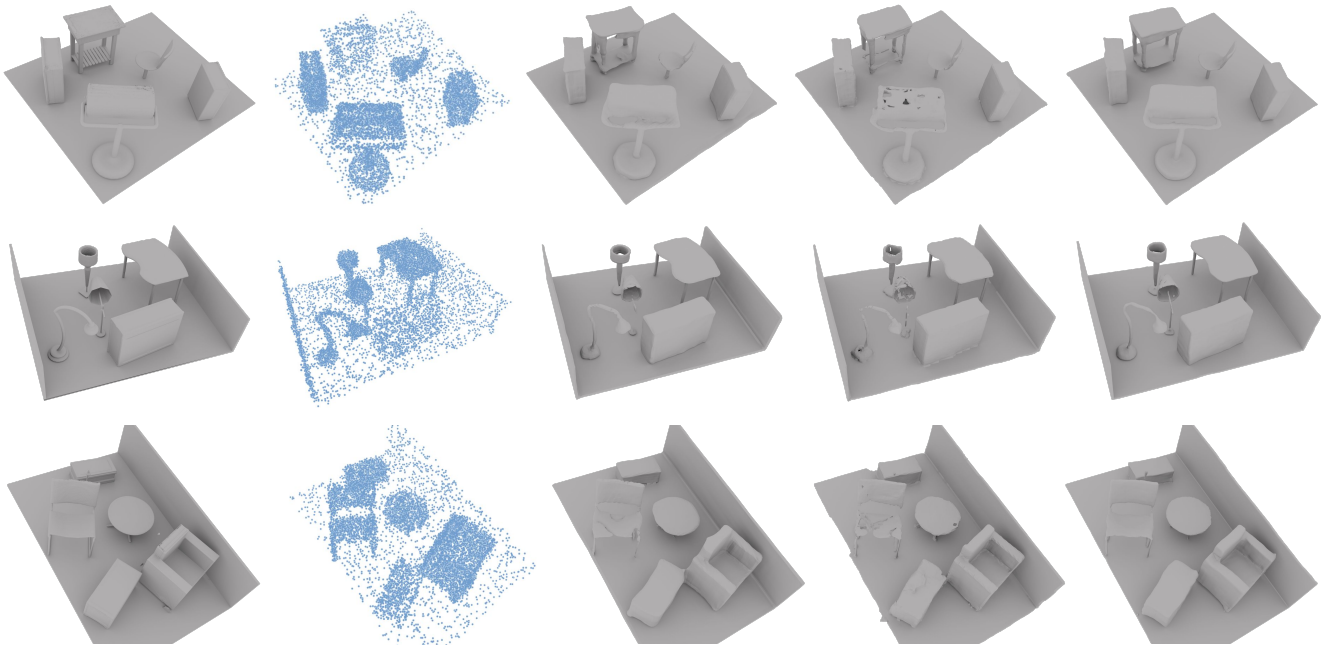
(a) Ground Truth (b) Input Points (c) ConvONet [10] (d) POCO [1] (e) **ALTO**

Figure A. **Qualitative comparison on object-level reconstruction ShapeNet dataset.** Trained and tested on 3k noisy points.



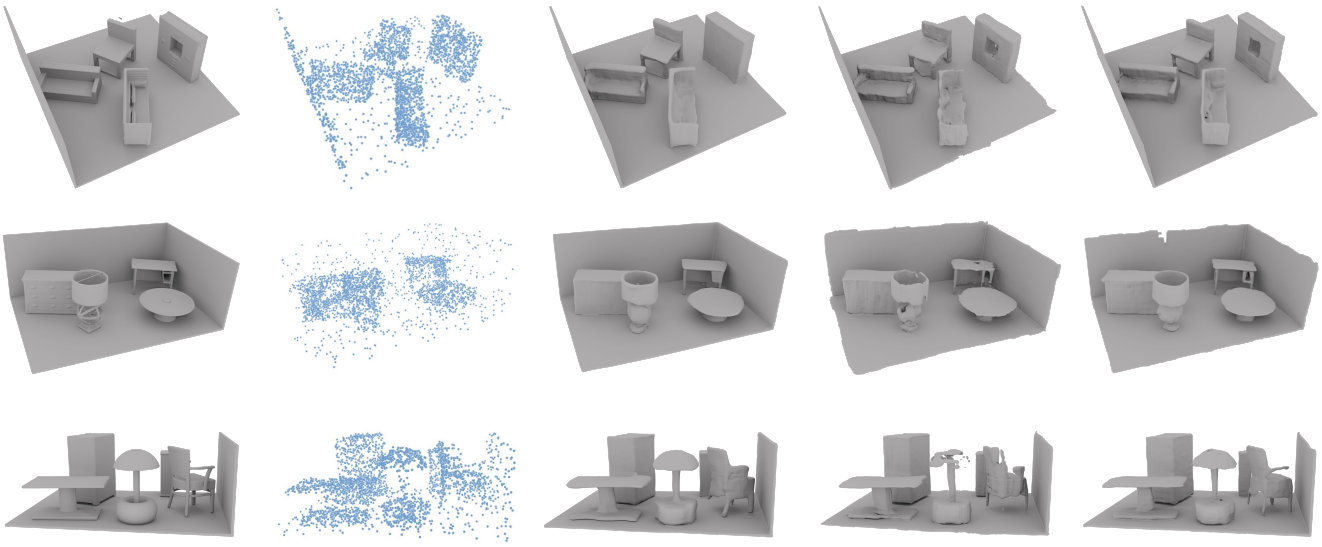
(a) Ground Truth (b) Input Points (c) ConvONet [10] (d) POCO [1] (e) **ALTO**

Figure B. **Qualitative comparison on object-level reconstruction ShapeNet dataset.** Trained and tested on 300 noisy points.



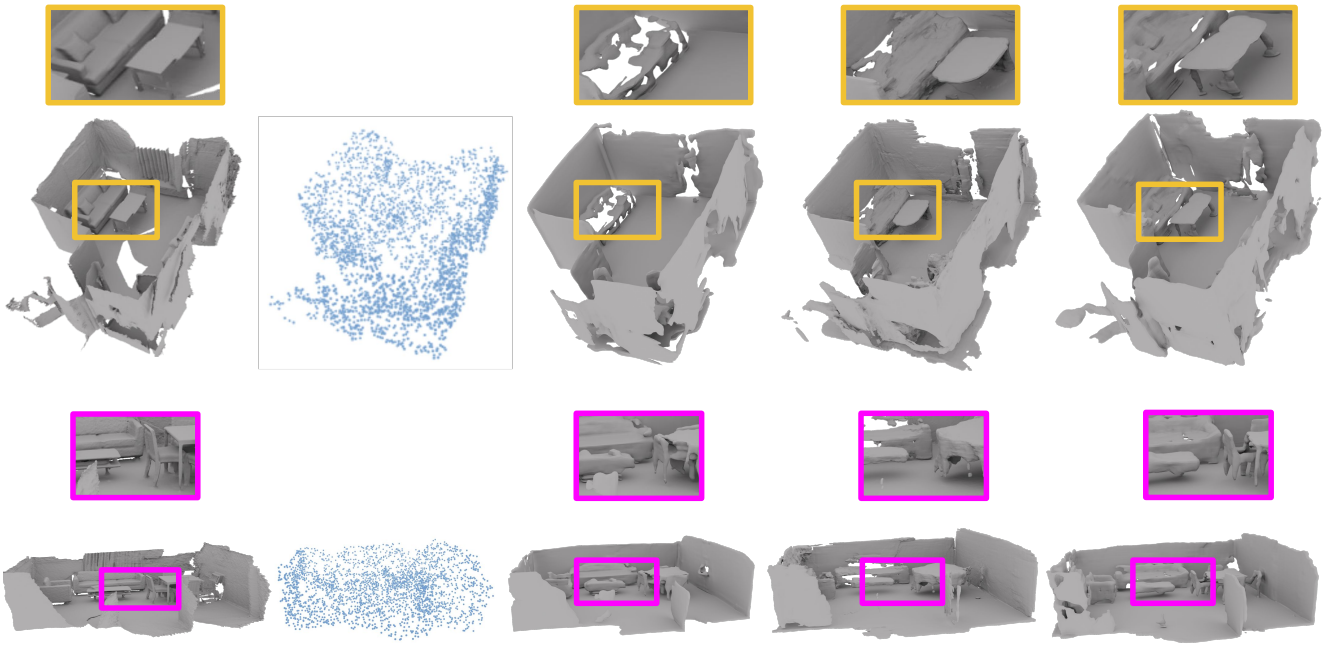
(a) Ground Truth (b) Input Points (c) ConvONet [10] (d) POCO [1] (e) **ALTO**

Figure C. **Qualitative comparison on scene-level reconstruction Synthetic Room dataset.** Trained and tested on 10k noisy points.



(a) Ground Truth (b) Input Points (c) ConvONet [10] (d) POCO [1] (e) **ALTO**

Figure D. **Qualitative comparison on scene-level reconstruction Synthetic Room dataset.** Trained and tested on 3K noisy points.



(a) Ground Truth (b) Input Points (c) ConvONet [10] (d) POCO [1] (e) **ALTO**

Figure E. **Qualitative comparison on scene-level reconstruction ScanNet.**

Methods	Links
SPSR [5]	https://github.com/mmolero/pypoisson
ONet [8]	https://github.com/autonomousvision/occupancy_networks
ConvONet [10]	https://github.com/autonomousvision/convolutional_occupancy_networks
DP-ConvONet [6]	https://github.com/dsvilarkovic/dynamic_plane_convolutional_onet
POCO [1]	https://github.com/valeoai/POCO

Table K. **The link for the baseline methods we compare.**

References

[1] 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, 5, 6, 7, 8, 9

[2] Özgün Çiçek, Ahmed Abdulkadir, Soeren S Lienkamp, Thomas Brox, and Olaf Ronneberger. 3d u-net: learning dense volumetric segmentation from sparse annotation. In *International conference on medical image computing and computer-assisted intervention*, pages 424–432. Springer, 2016. 1

[3] 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. 1

[4] Zhenyu Jiang, Yifeng Zhu, Maxwell Svetlik, Kuan Fang, and Yuke Zhu. Synergies between affordance and geometry: 6-dof grasp detection via implicit representations. *arXiv preprint arXiv:2104.01542*, 2021. 5

[5] Michael Kazhdan and Hugues Hoppe. Screened poisson surface reconstruction. *ACM Transactions on Graphics (ToG)*, 32(3):1–13, 2013. 9

[6] Stefan Lionar, Daniil Emtsev, Dusan Svilarkovic, and Songyou Peng. Dynamic plane convolutional occupancy networks. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1829–1838, 2021. 9

[7] William E Lorensen and Harvey E Cline. Marching cubes: A high resolution 3d surface construction algorithm. *ACM siggraph computer graphics*, 21(4):163–169, 1987. 2

[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. 4, 5, 6, 9

[9] Songyou Peng, Chiyu Jiang, Yiyi Liao, Michael Niemeyer, Marc Pollefeys, and Andreas Geiger. Shape as points: A differentiable poisson solver. *Advances in Neural Information Processing Systems*, 34:13032–13044, 2021. 3, 4, 6

[10] Songyou Peng, Michael Niemeyer, Lars Mescheder, Marc Pollefeys, and Andreas Geiger. Convolutional occupancy networks. In *European Conference on Computer Vision*, pages 523–540. Springer, 2020. 2, 4, 5, 6, 7, 8, 9

[11] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 652–660, 2017. 1

[12] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015. 1

[13] Jiapeng Tang, Jiabao Lei, Dan Xu, Feiying Ma, Kui Jia, and Lei Zhang. Sa-convonet: Sign-agnostic optimization of convolutional occupancy networks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6504–6513, 2021. 3, 6

[14] Suhani Vora, Noha Radwan, Klaus Greff, Henning Meyer, Kyle Genova, Mehdi SM Sajjadi, Etienne Pot, Andrea Tagliasacchi, and Daniel Duckworth. Nesf: Neural semantic fields for generalizable semantic segmentation of 3d scenes. *arXiv preprint arXiv:2111.13260*, 2021. 5

[15] Biao Zhang, Matthias Nießner, and Peter Wonka. 3dilg: Irregular latent grids for 3d generative modeling. *arXiv preprint arXiv:2205.13914*, 2022. 3