MASH: Masked Anchored SpHerical Distances for 3D Shape Representation and Generation - Supplemental Material

CHANGHAO LI, University of Science and Technology of China, China
YU XIN, University of Science and Technology of China, China
XIAOWEI ZHOU, State Key Laboratory of CAD & CG, Zhejiang University, China
ARIEL SHAMIR, Reichman University, Israel
HAO ZHANG, Simon Fraser University, Canada
LIGANG LIU, University of Science and Technology of China, China
RUIZHEN HU*, Shenzhen University, China

CCS Concepts: • **Computing methodologies → Shape modeling**; *Neural networks.*

ACM Reference Format:
Changhao Li, Yu Xin, Xiaowei Zhou, Ariel Shamir, Hao Zhang, Ligang Liu, and Ruizhen Hu. 2025. MASH: Masked Anchored SpHerical Distances for 3D Shape Representation and Generation - Supplemental Material . In *Special Interest Group on Computer Graphics and Interactive Techniques Conference Conference Papers (SIGGRAPH Conference Papers '25), August 10–14, 2025, Vancouver, BC, Canada.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3721238.3730610

1 SHAPE APPROXIMATION

Optimization details. To get more uniform sampling on the surfaces determined by the anchors, we set the number of uniformly pre-sampled directions $N_{\text{dir}} = 1000$, and then keep 80% of sampled points of all anchors by the farthest point sampling algorithm. Besides, in order to calculate the boundary-continuous loss term, we sample another $N_{\text{mask}} = 90$ points on the vision mask boundary of each anchor. When optimizing MASH parameters, we use AdamW [Loshchilov 2017] as our optimizer with an initial learning rate of 2e-3, and we use ReduceLROnPlateau as our scheduler with a final learning rate of 1e-3, a factor of 0.8, and patience of 2.

Hyperparameter choices. The key hyperparameters in our MASH representation are anchor numbers M, mask degrees K, and spherical harmonic degrees L. When L is 0 or 1, the surface patch of each anchor approximates a plane or a near-spherical surface, and when L exceeds 2, the excessive number of SH basis functions leads to increased computational costs. Therefore, to balance accuracy and computational efficiency, we choose L to be 2. After extensive experiments, we find that regarding the object complexity of the ShapeNet-V2 dataset, setting the mask degree $K = 3$ and the anchor number $M = 400$ can accurately represent most of the objects in this dataset while maintaining high computational efficiency. Table 1 shows how the approximation error changes with different K and L when $M = 400$. Table 2 shows how the approximation error changes with different M with $K = 3$ and $L = 2$. The results clearly show that as M, K, and L increase, the expressive power of MASH and the coverage ratio of the object surface gradually improve. This demonstrates the potential of MASH to effectively represent arbitrarily complex shapes by reasonably adjusting these hyperparameters.

Table 1. Quantitative comparison of shape approximation results with different mask and SH degrees. To ensure fair comparison, we set $M = 400$, $K = 3$ and $L = 2$ if not mentioned otherwise.

	K=0	K=1	K=2	K=3	K=4
L1-CD ↓	11.281	6.392	5.698	5.450	**5.434**
F-Score ↑	0.872	0.993	0.996	**0.997**	**0.997**
	L=0	L=1	L=2	L=3	L=4
L1-CD ↓	12.593	6.828	5.450	5.411	**5.323**
F-Score ↑	0.814	0.989	0.997	0.997	**0.998**
	L=2	L=3	L=4	L=5	L=6
L_c↓	6.569	5.926	5.714	5.444	**5.268**

Table 2. Quantitative comparison of shape approximation results with different numbers of MASH anchors, where $K = 3$ and $L = 2$.

	M=400	M=200	M=100	M=50	M=20	M=10
L1-CD ↓	**5.450**	6.013	7.174	11.267	16.536	18.990
F-Score ↑	**0.997**	0.994	0.988	0.924	0.851	0.812
Time (s)	39.027	28.560	23.979	20.817	18.083	14.352

Table 3. Ablation studies of some key components of our optimization scheme.

Method	L1-CD↓	L2-CD↓	FScore↑	D_H↓	S_{cos}↑	NIC↓
w/o inversion	5.909	33.367	0.981	0.027	0.961	22.344
w/o uniform	5.762	27.872	0.996	0.026	0.978	20.080
Ours	**5.450**	**22.523**	**0.997**	**0.019**	**0.980**	**18.040**

Importance of Inverse Transformation and Uniform Sampling. We also conduct an experiment to show the importance of the inverse transformation and uniform sampling. If inverse transformation is not used, the representation ability of each anchor will decrease

*Corresponding author: Ruizhen Hu (ruizhen.hu@gmail.com)

SIGGRAPH Conference Papers '25, August 10–14, 2025, Vancouver, BC, Canada

ACM ISBN 979-8-4007-1540-2/2025/08…$15.00
https://doi.org/10.1145/3721238.3730610

Table 4. Comparison of computation time and memory usage when different numbers of points Q are sampled on the given shape, under the default setting of M=400, K=3, and L=2.

$\lvert Q\rvert$	PS	L_f	L_c	L_ω	TC	Memory	L1-CD
1K	2.806	3.372	3.391	3.416	13.236	1690	19.901
2K	4.307	4.421	4.386	4.129	17.628	1691	13.684
5K	4.336	4.878	4.797	4.651	18.852	1703	11.257
10K	5.692	7.136	7.210	7.205	27.104	1719	6.833
20K	5.727	7.902	7.897	7.884	29.371	1745	5.263
40K	7.675	10.366	10.263	9.901	38.762	1839	4.971

and the planar area of the given shape will become one of the most difficult parts to be fit with spherical harmonics. In addition, if we directly sample the parameterized coordinates $\{\omega, \phi\}$ uniformly without considering the distribution of the sampled points on the object surface, the distribution of sampling points in some regions will be too sparse. As a result, the corresponding surface patches could deviate from the real distribution of the object surface due to the lack of sufficient supervision signals. The results in Table 3 confirmed the effectiveness of applying the inverse transformation and uniform sampling.

Timing and memory usage. Table 4 shows the detailed computation time and memory usage during the MASH optimization process when different numbers of points are sampled from the given shape, averaging over 100 shapes sampled from the ShapeNet-V2 dataset. The units for time are the second, and the units for memory usage are MB. Specifically, Point Sampling (PS) represents the total time taken to sample points from MASH, L_f, L_c, and L_ω correspond to the total time taken for the corresponding loss calculations, and Time Consumption (TC) is the total duration of the optimization algorithm.

Note that when fitting point clouds with different sizes, we need to correspondingly adjust the number of pre-sampled directions N_{dir} to prevent overfitting. Therefore, we set N_{dir}=$10 \cdot |Q|/M$. As $|Q|$ increases, anchors can more accurately expand and cover along the object's surface, thereby gradually improving the fitting efficiency.

Scene MASH optimization. Other than 3D shapes, we also make an additional attempt to fit the point cloud of the indoor scenes with MASH by increasing the anchor number M to 800. Some example results are shown in Figure 1, which demonstrates the potential of MASH in the representation and processing of large-scale data.

2 SURFACE RECONSTRUCTION

Reconstruction details. Most of the technical details about the surface reconstruction have been provided in the main paper. Here we show how the normal n_{scr} of the sampled points is computed:

$$n_x = \frac{\partial y}{\partial \phi}\frac{\partial z}{\partial \theta} - \frac{\partial z}{\partial \phi}\frac{\partial y}{\partial \theta} \tag{1}$$

$$n_y = \frac{\partial z}{\partial \phi}\frac{\partial x}{\partial \theta} - \frac{\partial x}{\partial \phi}\frac{\partial z}{\partial \theta} \tag{2}$$

$$n_z = \frac{\partial x}{\partial \phi}\frac{\partial y}{\partial \theta} - \frac{\partial y}{\partial \phi}\frac{\partial x}{\partial \theta} \tag{3}$$

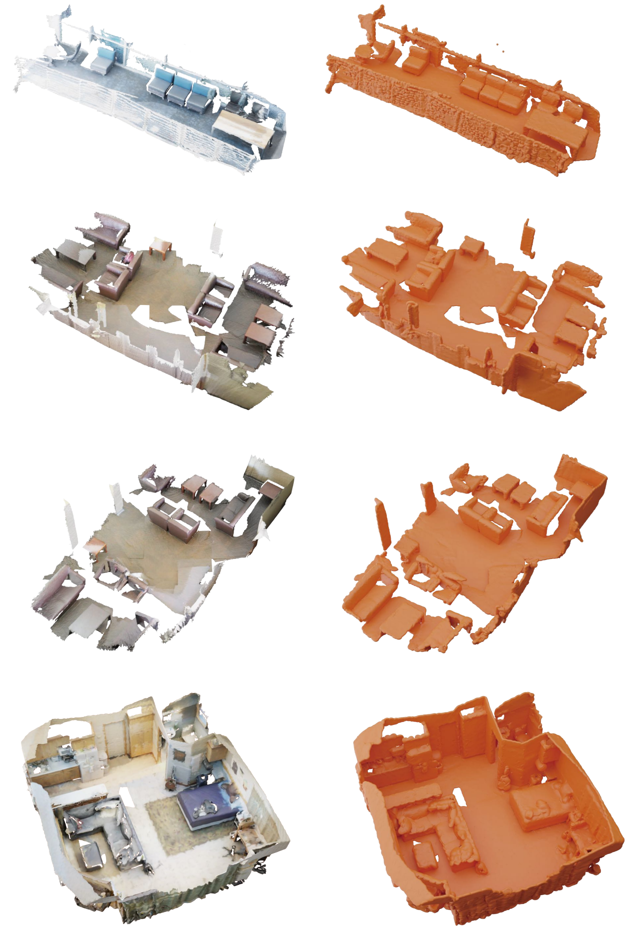

Fig. 1. Qualitative results on fitting and reconstruction for large-scale indoor scene point clouds with MASH.

where $\{\theta, \phi\}$ are the spherical coordinates and $\{x, y, z\}$ is the position of that sampled point.

Evaluation metrics. We compare our reconstruction results with ground truth (GT) meshes using common evaluation metrics, including *Hausdorff distance (D_H), L1 Chamfer Distance (L1-CD), L2 Chamfer Distance (L2-CD), F-Score, mesh cosine similarity (S_{cos}),* and *Normal InConsistency (NIC)* commonly used in previous works [Erler et al. 2020; Fan et al. 2017; Huttenlocher et al. 1993; Hwang and Sung 2024; Lin et al. 2022; Park et al. 2019; Xu et al. 2023], which are defined as:

$$D_H(P,Q) = \max(\max_{p\in P} d(p,Q), \max_{q\in Q} d(q,P)) \tag{4}$$

$$L1-CD(P,Q) = \frac{1}{|P|}\sum_{p\in P} d(p,Q) + \frac{1}{|Q|}\sum_{q\in Q} d(q,P) \tag{5}$$

$$L2-CD(P,Q) = \frac{1}{|P|}(\sum_{p\in P} d^2(p,Q) + \frac{1}{|Q|}\sum_{q\in Q} d^2(q,P))^{\frac{1}{2}} \tag{6}$$

$$F-Score(P,Q) = \frac{2 \cdot TP(P,Q) \cdot TP(Q,P)}{TP(P,Q) + TP(Q,P)} \tag{7}$$

$$S_{cos}(P,Q) = \frac{1}{|P|}\sum_{p \in P} |d_n(p,Q)| + \frac{1}{|Q|}\sum_{q \in Q} |d_n(q,P)| \tag{8}$$

$$NIC(P,Q) = \frac{1}{|P|}\sum_{p \in P} d_r(p,Q) \tag{9}$$

where the true-positive function TP is defined as:

$$TP(P,Q) = \frac{1}{|P|}\sum_{p \in P} \mathbb{I}(d(p,Q) \le 0.01) \tag{10}$$

and the normal similarity function d_n and the radian angle difference function d_r are defined as:

$$d_n(p,Q) = v(p) \cdot v(\arg\min_{q \in Q}(||p-q||_2)) \tag{11}$$

$$d_r(p,Q) = \min(\arccos(d_n(p,Q)), \pi - \arccos(d_n(p,Q))) \tag{12}$$

with $v(p)$ to be the surface normal at point p.

When calculating all these metrics, we sample 50k points by farthest point sampling from the ground truth and reconstructed surfaces, respectively. In order to make metrics more intuitive, we sample the object surface twice with different initial points by and report all metrics in Table 5 as *FPS*. This setting reflects the approximate upper or lower bound of each metric.

Surface reconstruction comparison. For object surface reconstruction, we conduct experiments on the ShapeNet-V2 dataset [Chang et al. 2015] and list detailed comparison results on the seven main categories with the metrics of L1-CD and F-Score in Table 5. Note that since the results of D_H and S_{cos} on different categories are relatively similar, we mainly report their results on the complete dataset in the main paper. More visual examples are also presented in Figure 2.

Surface reconstruction on noisy data. We also conduct experiments on noisy inputs across all categories, in which the noise conforms to a Gaussian distribution with the mean of 0 and the variance of $0.2\%L$ or $0.5\%L$ where L is the length of the bounding box diagonal corresponding to each object. The quantitative comparisons are shown in Table 6, and our method gets consistently better results.

Some representative surface reconstruction results of each method on the Chair category are presented in Figure 3. The global consistency of the normal directions estimated by PGR will decrease as the noise increases, resulting in obvious holes in the results due to the flipping of normal directions. Since ARONet and ConvONet are learning-based methods, they have stronger anti-noise capabilities. However, their methods can still be misled by noise that does not belong to the object surface, leading to incorrect judgments of the object topology. Different from their methods, since the surface of the object is approximately distributed at the mean of the noise data, our method moves the surface patch of each anchor to near the mean position of local noise data by trying to fit all given points and ultimately achieves higher reconstruction quality.

Surface reconstruction on sparse and non-uniform data. To further verify the robustness of MASH in surface reconstruction, we conducted evaluations on sparse and non-uniform data. For data acquisition, we first uniformly sampled an initial point cloud 10 times the size of the input point cloud on the surface of the object from the ShapeNet-V2 dataset. We then obtained the initial point cloud through random down sampling, and further added 0.2% Gaussian noise to it. This approach ensured the non-uniformity of the input point cloud.

As shown in Table 7 and Table 8, when 8,192 non-uniform points are used as input, our proposed method still maintains a consistent lead. However, when only 1,024 sparse non-uniform points are served as input, the uncertainty of the gradient descent direction of our loss terms used for optimization significantly increases. This prevents MASH from accurately extending along the object surface, resulting in a decline in reconstruction quality. Therefore, we propose a new method *GEN+ICP+R10*, which treats reconstruction as a conditional generation process to obtain robust MASH from sparse point cloud inputs. Due to the stochasticity of the generation, we generate 10 results and select the one closest to the input point cloud in L1-CD through ICP alignment. Although our generator was trained with MASH on an 8,096-point setting, it generalizes very well for generation-based reconstruction on very sparse (1,024) inputs, outperforming other learning-based reconstruction baselines.

In addition, we found that the LiDAR data in the KITTI dataset [Geiger et al. 2013] also has sparsity and non-uniformity, so we also evaluated our method on these data. As shown in Figure 4 and Table 9, our default MASH outperforms all metrics and baselines, thanks to its capability of handling thin-sheet structures, further validating its effectiveness in this real-world input setting.

Surface reconstruction on high-genus data. To further verify the representablity of our proposed surface reconstruction method, we also evaluated the expressive power of MASH on high-genus objects in the Thingi10K [Zhou and Jacobson 2016] dataset. Since there is no corresponding ground truth mesh for the LiDAR point cloud as a reference, we stitch all the frames of the LiDAR point cloud and clip out the dense LiDAR point cloud of the corresponding area to serve as an approximation of the ground truth mesh. As shown in Figure 5, our method is capable of recovering more complete and accurate geometric information of objects. In addition, compared to implicit representations, MASH can represent the geometric shapes of objects more compactly and accurately, without encountering generalization issues caused by the bias of the training dataset.

3 SHAPE GENERATION

Technical details. We use the optimal transport conditional flow matching [Tong et al. 2023] to train our diffusion models on 8 RTX4090 with a batch size of 1024 for $T = 1,000$ epochs. The learning rate is linearly increased to $lr_{max} = 2e-4$ in the first $t_0 = 80$ epochs, and then gradually decreases using the cosine decay schedule $lr_{max} * 0.5^{1+cos(\frac{t-t_0}{T-t_0})}$ until reaching $1e-6$.

To further extract the mesh from the generated MASH representation, we train an extra network to decode the occupancy grid and then extract the mesh by using ODC [Hwang and Sung 2024].

Table 5. Detailed quantitative comparison with surface reconstruction baselines, including SPR [Kazhdan and Hoppe 2013], PGR [Lin et al. 2022], ARO-Net [Wang et al. 2023] and CONet [Peng et al. 2020]on the ShapeNet dataset. For ease of comparison of results, we multiply the L1-CD by 1000.

		SPR	PGR	CONet	ARONet	Ours	FPS
L1-CD ↓	airplane	63.562	4.839	13.254	11.569	**4.194**	3.426
	chair	102.852	7.824	19.374	15.713	**6.646**	5.471
	car	135.148	8.052	25.073	23.173	**6.795**	5.383
	lamp	88.064	6.598	16.903	17.958	**4.534**	3.850
	rifle	79.631	4.479	11.393	13.171	**3.829**	2.916
	sofa	112.850	7.857	26.782	18.069	**6.444**	5.249
	table	77.971	7.595	22.339	15.965	**6.578**	5.473
	mean	89.565	6.381	17.732	15.697	**5.450**	4.782
F-Score ↑	airplane	0.544	0.997	0.871	0.873	**0.998**	1.000
	chair	0.436	0.984	0.801	0.886	**0.997**	1.000
	car	0.391	0.964	0.627	0.720	**0.987**	0.998
	lamp	0.517	0.979	0.811	0.801	**0.999**	0.999
	rifle	0.525	0.987	0.931	0.924	**0.998**	1.000
	sofa	0.422	0.970	0.698	0.727	**0.996**	0.999
	table	0.530	0.988	0.802	0.892	**0.997**	0.999
	mean	0.497	0.988	0.812	0.880	**0.997**	0.999
D_H ↓	mean	0.272	0.023	0.131	0.117	**0.019**	0.012
S_{cos} ↑	mean	0.684	0.974	0.821	0.898	**0.980**	0.991

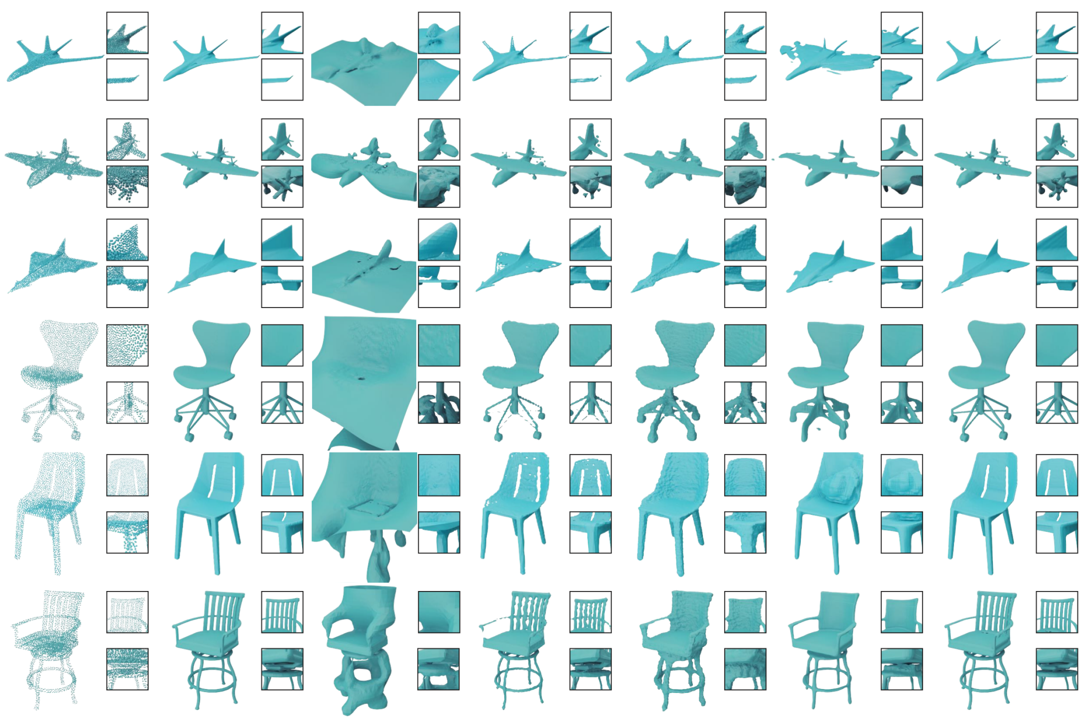

Fig. 2. Qualitative results on surface reconstruction with different methods. We additionally select two details for each result to show the performance of all methods better.

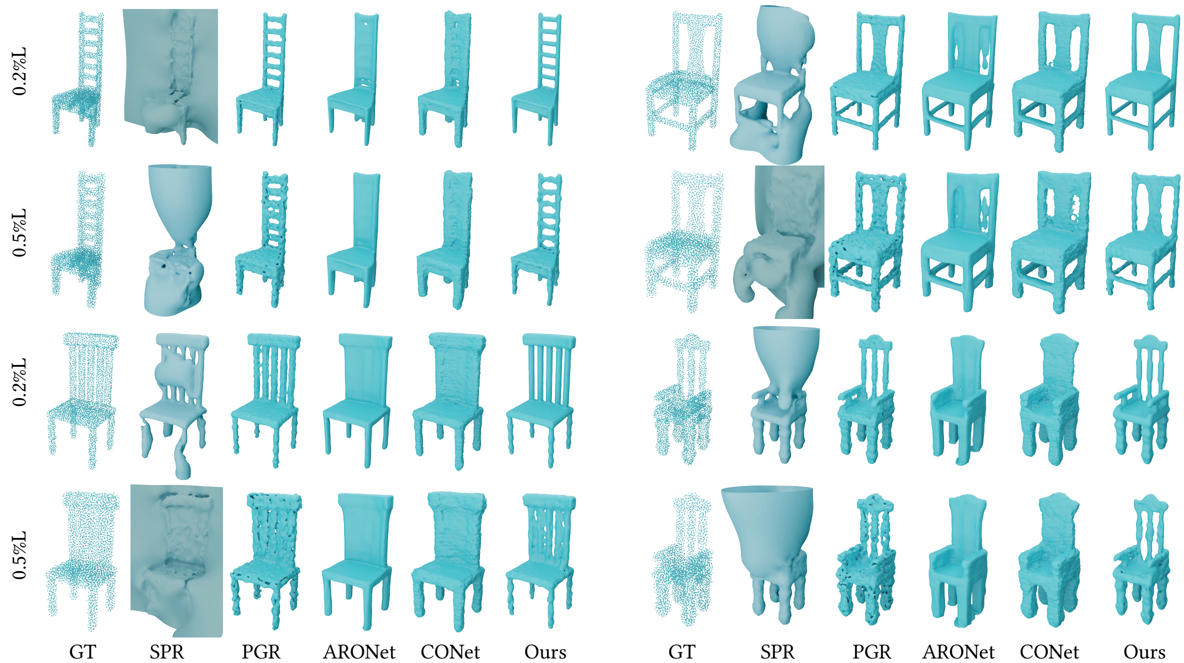

Fig. 3. Reconstructions on the chair category with two different noise levels.

Table 6. Quantitative comparison with surface reconstruction baselines on noisy input. For ease of comparison of results, we multiply the L1-CD by 1000 here.

		SPR	PGR	CONet	ARONet	Ours
0.2%L	L1-CD ↓	89.724	7.752	18.203	16.825	**6.177**
	F-Score ↑	0.494	0.949	0.803	0.878	**0.962**
	D_H ↓	0.271	0.024	0.142	0.122	**0.020**
	S_{cos} ↑	0.542	0.966	0.803	0.884	**0.971**
0.5%L	L1-CD ↓	94.172	19.875	21.434	18.872	**12.718**
	F-Score ↑	0.493	0.802	0.781	0.804	**0.894**
	D_H ↓	0.268	0.069	0.148	0.125	**0.059**
	S_{cos} ↑	0.326	0.828	0.801	0.881	**0.912**

Table 7. Quantitative comparison with surface reconstruction baselines on the ShapeNet-V2 dataset. For each shape, we randomly sampled 8192 points with 0.2% Gaussian noise as input to account for non-uniformity.

Method	SPR	PGR	CONet	ARONet	MASH	GEN+ICP+R10
L1-CD↓	93.452	8.714	19.031	17.301	**5.932**	6.723
L2-CD↓	342.803	17.809	53.071	42.806	**6.021**	6.300
F-Score↑	0.466	0.892	0.763	0.782	**0.958**	0.949
D_H↓	0.291	0.048	0.133	0.121	**0.023**	0.069
S_{cos}↑	0.623	0.897	0.820	0.837	**0.972**	0.963
NIC↓	67.712	23.622	31.796	29.801	**19.823**	21.107

Meanwhile, to enable our shape decoder to adapt to the potentially noisy generated MASH data, we borrow the key idea of VAE [Pinheiro Cinelli et al. 2021]. Specifically, We map each channel of all anchors in the dataset to an approximate normal distribution by

Table 8. Quantitative comparison with surface reconstruction baselines on the ShapeNet-V2 dataset. For each shape, we randomly sampled 1024 points with 0.2% Gaussian noise as input to account for non-uniformity.

Method	SPR	PGR	CONet	ARONet	MASH	GEN+ICP+R10
L1-CD↓	131.059	14.927	16.682	16.593	18.113	**7.172**
L2-CD↓	462.081	36.092	47.041	33.502	37.730	**8.628**
F-Score↑	0.326	0.722	0.698	0.763	0.538	**0.892**
D_H↓	0.442	0.194	0.196	0.192	0.231	**0.116**
S_{cos}↑	0.320	0.782	0.798	0.760	0.632	**0.899**
NIC↓	56.711	31.542	29.514	30.298	46.033	**24.046**

Table 9. Quantitative comparison with surface reconstruction baselines on the LiDAR data of KITTI dataset.

Method	SPR	PGR	CONet	ARONet	MASH	MASH-1600
L1-CD↓	8.062	2.211	3.826	2.254	0.180	**0.101**
L2-CD↓	64.087	7.433	15.627	4.825	0.058	**0.015**
F-Score@0.1↑	0.017	0.061	0.024	0.037	0.722	**0.877**
D_H↓	33.282	11.086	26.028	13.252	2.372	**1.125**

carefully designing a strictly monotonically increasing piecewise linear function. This allows us to perform variational MASH sampling more effectively and obtain a more stable shape decoder.

Compared to Shape2VecSet [Zhang et al. 2023], both the training and sampling time of our model is less than one-third, with the same network backbone. More specifically, Shape2VecSet takes about 28 GPU days to train the AutoEncoder and another 92 GPU days to train the diffusion model, while our model takes about 32 GPU days to train with the same setting. When generating shapes on a single

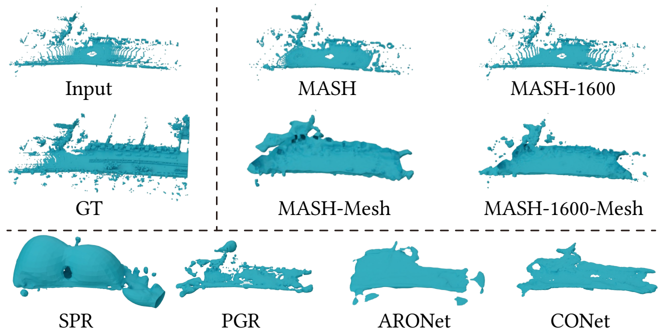

Fig. 4. Qualitative results on surface reconstruction with different methods on LiDAR data of KITTI dataset.

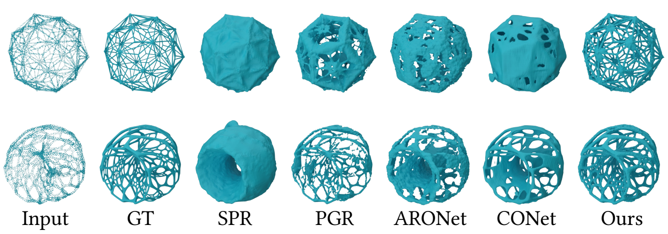

Fig. 5. Qualitative results on surface reconstruction with different methods on high-genus shapes of Thingi10K dataset.

RTX 4090 GPU, Shape2VecSet takes about 0.2 seconds to generate a vector set and about 14 seconds to decode it into a 3D shape. In contrast, our model takes 0.05 seconds to generate a MASH, and another 4 seconds to convert the MASH into a triangular mesh if required.

Evaluation metrics. The metrics used to measure the visual similarity between generated shapes and the dataset are defined as:

$$\text{Render-FID} = \|\mu_r - \mu_g\|^2 + \text{Tr}(\Sigma_r + \Sigma_g - 2(\Sigma_r\Sigma_g)^{\frac{1}{2}}) \tag{13}$$

$$\text{Render-KID} = \text{MMD}\left(\frac{1}{|\mathcal{R}|}\sum_{x\in\mathcal{R}}\max_{y\in\mathcal{G}} D(x,y)\right)^2 \tag{14}$$

where g and r denote the generated and training datasets respectively. μ and Σ are the statistical mean and covariance matrix of the feature distribution extracted by the Inception-V3 network [Szegedy et al. 2016]. $D(x, y)$ is a polynomial kernel function to evaluate the similarity of two samples, $\mathcal{G}$ and $\mathcal{R}$ are feature distributions of the generated set and reference set, respectively. The function $\text{MMD}(\cdot)$ is Maximum Mean Discrepancy.

To measure the geometric similarity between the generated 3D shapes and the GT shapes, we sample 50K points from each and use L1-CD as the primary evaluation metrics. To measure the alignment between images and shapes, we sample 8,192 points from the generated shapes and use ULIP-2 [Xue et al. 2024] as the primary evaluation metric. Specifically, ULIP-I is defined as $\text{ULIP-I}(I, S) = \langle E_I, E_S\rangle$, corresponding to the inner product of normalized ULIP features of image I and generated shape S.

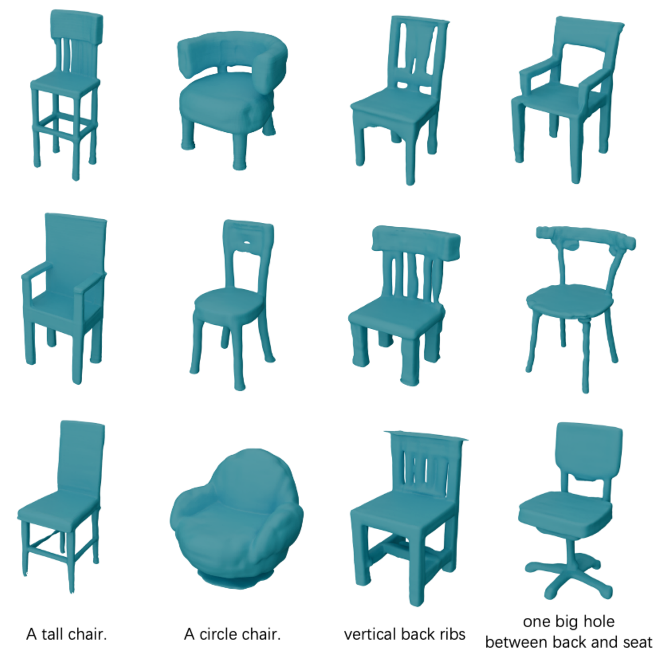

Fig. 6. Qualitative results on text-conditioned generation.

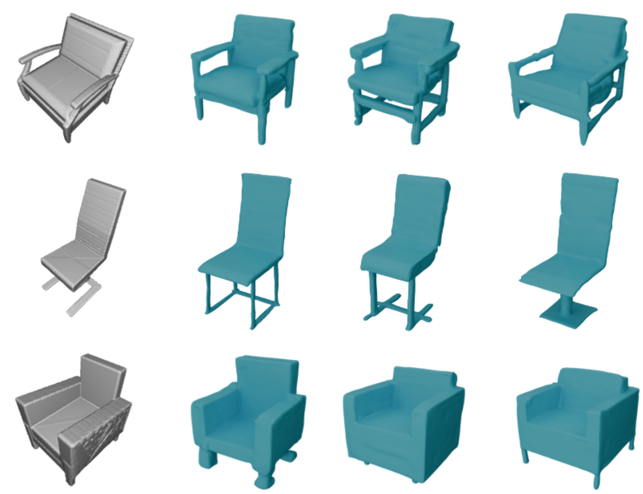

Fig. 7. Qualitative results on image-conditioned generation.

Multi-modal shape generation. In addition to category-conditioned and image-conditioned generative models, we also trained a multimodal generative model using the ShapeNet-V2 and ShapeGlot [Achlioptas et al. 2019] datasets.

Specifically, we use the ULIP-2 pre-trained model to encode three different modalities including text, images, and point clouds, to obtain the corresponding feature embeddings and serve them as conditions of the generative model. The text label comes from the ShapeGlot dataset, but only for the chair category.For the image

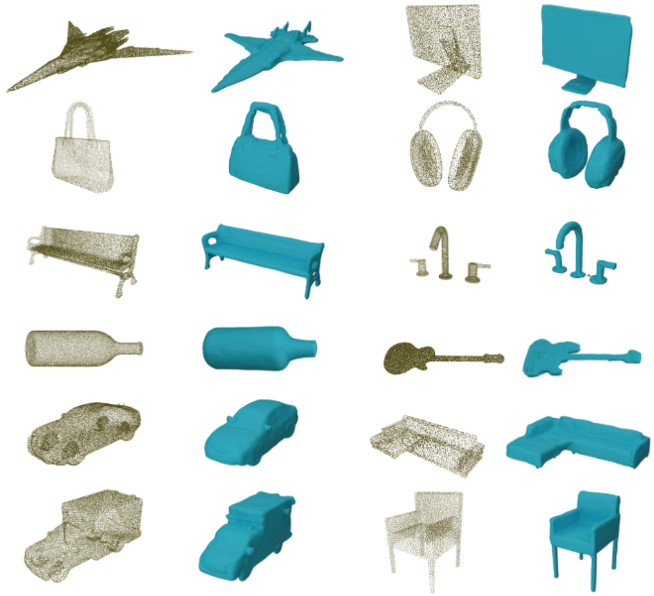

Fig. 8. Qualitative results on point cloud-conditioned generation.

condition, we create an image dataset by rendering all shapes in the ShapeNet-V2 dataset from 12 different viewpoints. For the point cloud condition, we sample a point cloud with 8,192 points through the farthest point sampling for each shape in ShapeNet-V2.

The generated results are shown in Figure 6, 7, and 8. Our multi-modal generative model can accurately generate shapes that meet our expectations based on different modalities of input.

Image-conditioned shape generation. We show more visual comparisons of the image-conditioned shape generation results in Figure 9 and more our results in Figure 10.

REFERENCES

Panos Achlioptas, Judy Fan, Robert Hawkins, Noah Goodman, and Leonidas J Guibas. 2019. ShapeGlot: Learning language for shape differentiation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 8938–8947.

Angel X. Chang, Thomas A. Funkhouser, Leonidas J. Guibas, Pat Hanrahan, Qi-Xing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, Jianxiong Xiao, Li Yi, and Fisher Yu. 2015. ShapeNet: An Information-Rich 3D Model Repository. *CoRR* abs/1512.03012 (2015). arXiv:1512.03012 http://arxiv.org/abs/1512.03012

Philipp Erler, Paul Guerrero, Stefan Ohrhallinger, Niloy J. Mitra, and Michael Wimmer. 2020. Points2Surf: Learning Implicit Surfaces from Point Clouds. In *Computer Vision – ECCV 2020*, Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm (Eds.). Springer International Publishing, Cham, 108–124.

H. Fan, H. Su, and L. Guibas. 2017. A Point Set Generation Network for 3D Object Reconstruction from a Single Image. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE Computer Society, Los Alamitos, CA, USA, 2463–2471. https://doi.org/10.1109/CVPR.2017.264

Andreas Geiger, Philip Lenz, Christoph Stiller, and Raquel Urtasun. 2013. Vision meets Robotics: The KITTI Dataset. *International Journal of Robotics Research (IJRR)* (2013).

D. P. Huttenlocher, G. A. Klanderman, and W. A. Rucklidge. 1993. Comparing Images Using the Hausdorff Distance. *IEEE Trans. Pattern Anal. Mach. Intell.* 15, 9 (sep 1993), 850–863. https://doi.org/10.1109/34.232073

Jisung Hwang and Minhyuk Sung. 2024. Occupancy-Based Dual Contouring. In *SIGGRAPH Asia 2024 Conference Papers*. 1–11.

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

Siyou Lin, Dong Xiao, Zuoqiang Shi, and Bin Wang. 2022. Surface Reconstruction from Point Clouds without Normals by Parametrizing the Gauss Formula. *ACM Trans. Graph.* 42, 2, Article 14 (oct 2022), 19 pages. https://doi.org/10.1145/3554730

I Loshchilov. 2017. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101* (2017).

Jeong Joon Park, Peter Florence, Julian Straub, Richard Newcombe, and Steven Lovegrove. 2019. DeepSDF: Learning Continuous Signed Distance Functions for Shape Representation. arXiv:1901.05103 [cs.CV]

Songyou Peng, Michael Niemeyer, Lars Mescheder, Marc Pollefeys, and Andreas Geiger. 2020. Convolutional Occupancy Networks. In *Computer Vision – ECCV 2020*, Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm (Eds.). Springer International Publishing, Cham, 523–540.

Lucas Pinheiro Cinelli, Matheus Araújo Marins, Eduardo Antúnio Barros da Silva, and Sérgio Lima Netto. 2021. Variational autoencoder. In *Variational Methods for Machine Learning with Applications to Deep Networks*. Springer, 111–149.

Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. 2016. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2818–2826.

Alexander Tong, Nikolay Malkin, Guillaume Huguet, Yanlei Zhang, Jarrid Rector-Brooks, Kilian Fatras, Guy Wolf, and Yoshua Bengio. 2023. Conditional flow matching: Simulation-free dynamic optimal transport. *arXiv preprint arXiv:2302.00482* 2, 3 (2023).

Yizhi Wang, Zeyu Huang, Ariel Shamir, Hui Huang, Hao Zhang, and Ruizhen Hu. 2023. ARO-Net: Learning Implicit Fields from Anchored Radial Observations. *CVPR* (2023).

Rui Xu, Zhiyang Dou, Ningna Wang, Shiqing Xin, Shuangmin Chen, Mingyan Jiang, Xiaohu Guo, Wenping Wang, and Changhe Tu. 2023. Globally consistent normal orientation for point clouds by regularizing the winding-number field. *ACM Transactions on Graphics (TOG)* 42, 4 (2023), 1–15.

Le Xue, Ning Yu, Shu Zhang, Artemis Panagopoulou, Junnan Li, Roberto Martín-Martín, Jiajun Wu, Caiming Xiong, Ran Xu, Juan Carlos Niebles, et al. 2024. Ulip-2: Towards scalable multimodal pre-training for 3d understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 27091–27101.

Biao Zhang, Jiapeng Tang, Matthias Niessner, and Peter Wonka. 2023. 3dshape2vecset: A 3d shape representation for neural fields and generative diffusion models. *ACM Transactions on Graphics (TOG)* 42, 4 (2023), 1–16.

Qingnan Zhou and Alec Jacobson. 2016. Thingi10k: A dataset of 10,000 3d-printing models. *arXiv preprint arXiv:1605.04797* (2016).

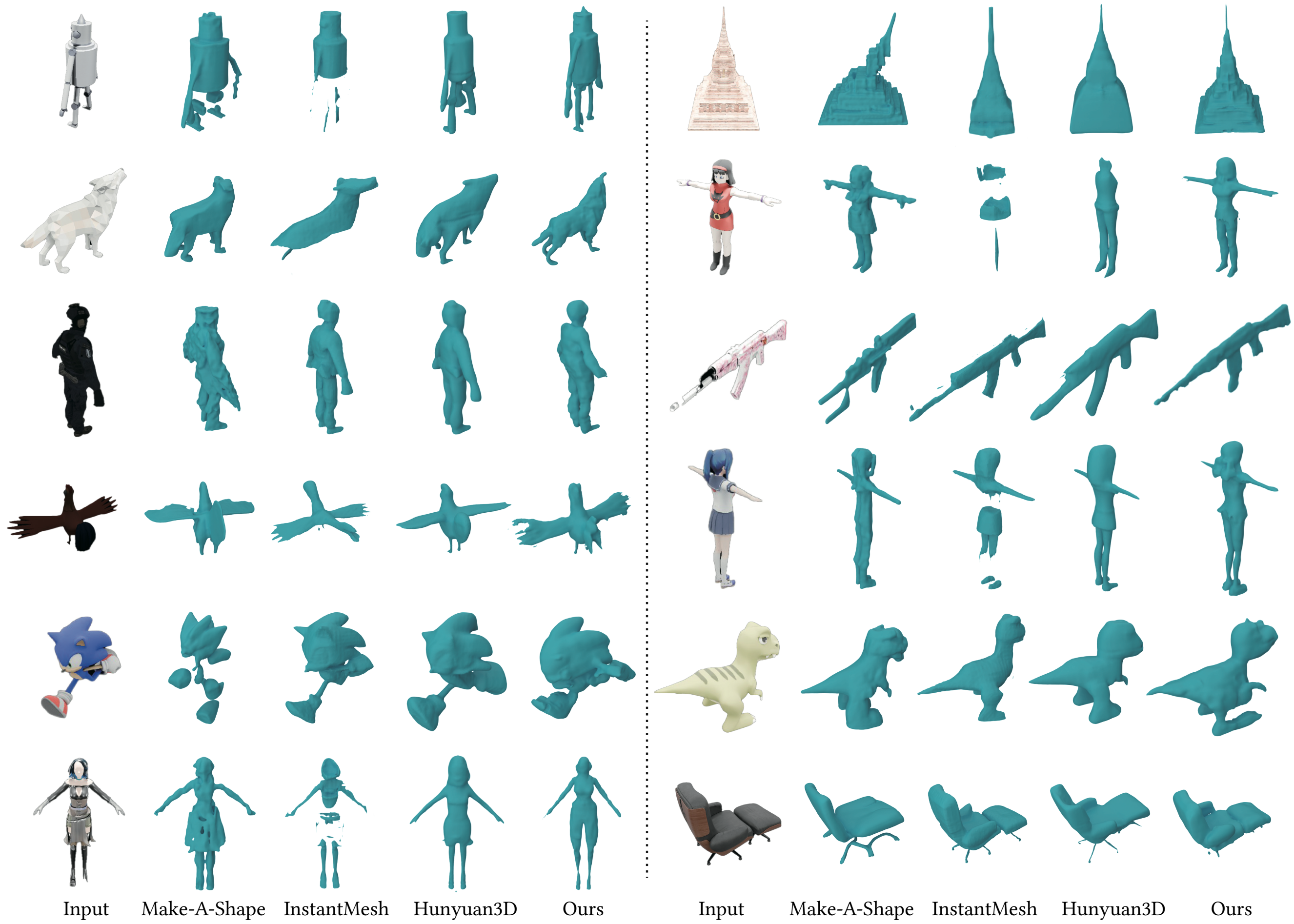

Fig. 9. More qualitative results on single image to 3D generation compared with different methods.

Fig. 10. More qualitative results on single image to 3D generation based on MASH.