

# GenUDC: High Quality 3D Mesh Generation With Unsigned Dual Contouring Representation – Supplementary Materials

Anonymous Authors

## 1 THE ARCHITECTURE OF NETWORKS

We show the architecture of our networks in Fig. A, Fig. B, Fig. C, Fig. D, and Fig. E. Specifically,  $\text{Conv}k\text{-}x$  is the convolution layer with  $k$  kernel size,  $x$  output channels, 1 stride, 1 padding.  $\text{ResBlock-}x$  is the ResNet block [3] with  $x$  output channels. We present the details of  $\text{ResBlock-}x$  in Fig. F.  $\text{SiLU}$  is the silu function.  $\text{GroupNorm-}x$  is the group normalization with  $x$  groups. The scale factor of DownSample and UpSample is 2. Attention is the attention block.  $\text{FC-}x$  is the fully connected layer with  $x$  output channels.

## 2 DIFFUSION MODEL

A diffusion model consists of two opposite processes: the forward process and the reverse process. Given the latent representation  $z_0 \sim p(z_0)$  as the data, the forward process adds the controlled Gaussian noise  $\epsilon$  to  $z_0$  for  $T$  times:

$$q(z_T|z_0) = \prod_{t=1}^T q(z_t|z_{t-1}), \quad (1)$$

$$q(z_t|z_{t-1}) = N(z_t; \sqrt{\alpha_t}z_{t-1}, \beta_t I), \quad (2)$$

where  $\alpha_t = 1 - \beta_t$ , and  $\beta_t$  is the predefined variance. In contrast, the reverse process denoises  $z_T$  to  $z_0$ :

$$p_\theta(z_0|z_T) = \prod_{t=1}^T p_\theta(z_{t-1}|z_t), \quad (3)$$

$$p_\theta(z_{t-1}|z_t) = N(z_{t-1}; \mu_\theta(z_t, t), \beta_t I). \quad (4)$$

According to the method in DDPM [4], we reparameterize  $\mu_\theta(z_t, t)$  as:

$$\mu_\theta(z_t, t) = \frac{1}{\sqrt{\alpha_t}} \left( z_t - \frac{\beta_t}{\sqrt{1 - \alpha_t}} \epsilon_\theta(z_t, t) \right), \quad (5)$$

$$z_t = \sqrt{\alpha_t}z_0 + \sqrt{1 - \alpha_t}\epsilon, \quad \alpha_t = \prod_{i=1}^t \alpha_i, \quad (6)$$

where  $\epsilon \sim N(0, 1)$  is the noise. In the cycle of the forward process and the reverse process, the noise  $\epsilon_\theta(z_t, t)$  is the only unknown value. Thus, we predict  $\epsilon_\theta(z_t, t)$  by a neural network parameterized as  $\theta$  to complete the cycle.

We train the network  $\epsilon_\theta$  with:

$$\mathcal{L}_{dm} = \mathbb{E}_{z,t,\epsilon \sim N(0,1)} \|\epsilon - \epsilon_\theta(z_t, t)\|_1. \quad (7)$$

In the forward process, we add the noise  $\epsilon$  to  $z_0$  for getting  $z_t$  as shown in Eq. (6), and train the network  $\epsilon_\theta$  to fit  $\epsilon$ . After training, our U-Net denoises  $z_T \sim N(0, 1)$  to  $z_0$  in the reverse process.

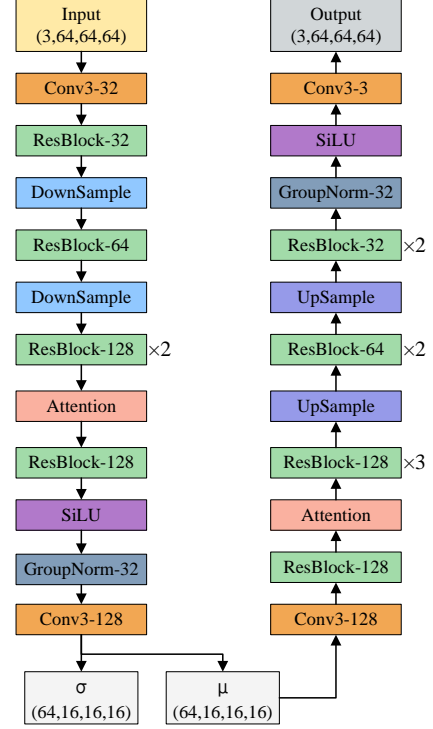


Figure A: Our VAE network for 64 resolution.

## 3 METRICS

**Chamfer distance (CD)** measures the similarity between two point clouds. The CD is formulated as:

$$CD(A, B) = \sum_{a \in A} \min_{b \in B} \|a - b\|_2^2 + \sum_{b \in B} \min_{a \in A} \|a - b\|_2^2, \quad (8)$$

where  $A$  and  $B$  are generated point clouds and reference point clouds.

**Earth mover's distance (EMD)** is a metric of dissimilarity between two distributions and can be also used to measure the similarity between two point clouds:

$$EMD(A, B) = \min_{\phi: A \rightarrow B} \sum_{a \in A} \|a - \phi(a)\|_2, \quad (9)$$

where  $\phi$  is the bijection between  $A$  and  $B$ .

**Light field descriptor (LFD)** [1, 2, 5] utilize silhouette images rendered from 20 camera poses to measure the structure similarity between two shapes.

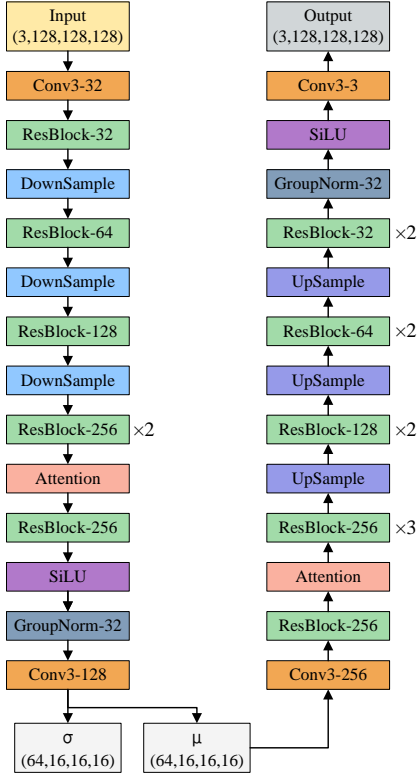


Figure B: Our VAE network for 128 resolution.

**Jensen-shannon divergence (JSD)** is calculated between two marginal point distribution of  $A$  and  $B$ :

$$JSD(P_X, P_Y) = \frac{1}{2} D_{KL}(P_X || M) + \frac{1}{2} D_{KL}(P_Y || M), \quad (10)$$

where  $P_X$  and  $P_Y$  are marginal distributions of points in the generated point clouds  $A$  and the reference point clouds  $B$  respectively. To approximate, we discretize the point cloud space into  $28^3$  voxels and assign each point to one of  $P_X$  and  $P_Y$ .

**Coverage (COV)** measures the diversity of generated dataset  $X$  in comparison to the reference dataset  $Y$ . For each  $x \in X$ , it finds a nearest neighbor  $y \in Y$  as a match. COV is the fraction of matched  $y$  in the reference dataset  $Y$ :

$$COV(X, Y) = \frac{|\{ \arg \min_{y \in Y} D(x, y) | x \in X \}|}{|Y|}, \quad (11)$$

where  $D(\cdot, \cdot)$  is a distance function, such as CD, EMD, or LFD.

**Minimum matching distance (MMD)** measures the quality of  $X$  referred to  $Y$ . For each  $y \in Y$ , it finds the nearest neighbor  $x$  in  $X$  and records  $D(x, y)$ . MMD is the mean of those distances:

$$MMD(X, Y) = \frac{1}{|Y|} \sum_{y \in Y} \arg \min_{x \in X} D(x, y) \quad (12)$$

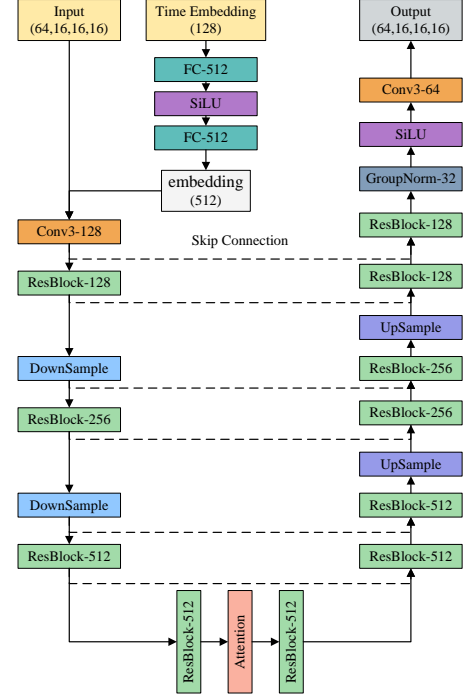


Figure C: Our diffusion model for 64 and 128 resolution.

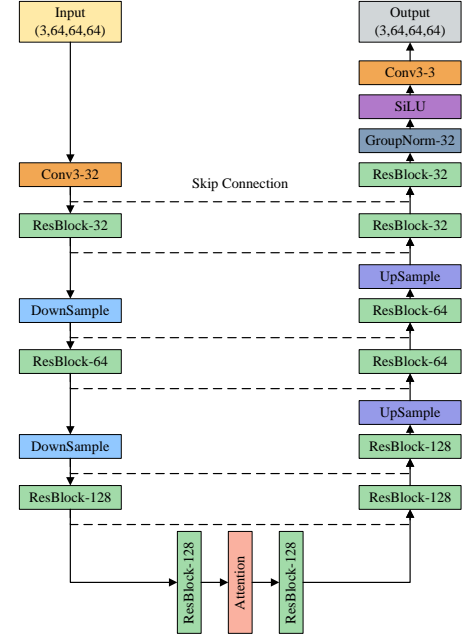


Figure D: Our vertex refiner (U-Net) for 64 resolution.

**1-nearest neighbor accuracy (1-NNa)** [6] measures the similarity between two distributions:

$$1-NNa(X, Y) = \frac{\sum_{x \in X} \mathbb{I}(n_x \in X)}{|X| + |Y|} + \frac{\sum_{y \in Y} \mathbb{I}(n_y \in Y)}{|X| + |Y|}, \quad (13)$$

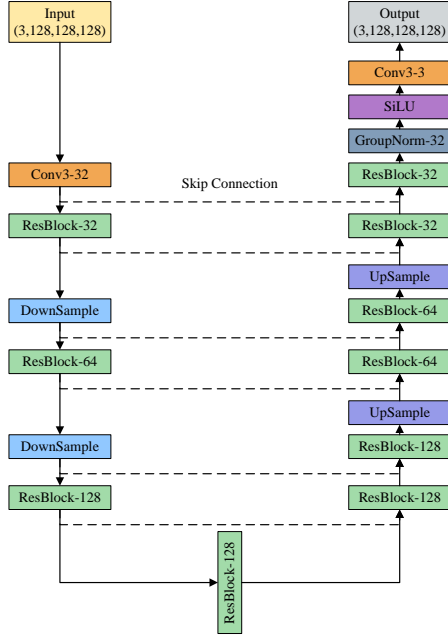


Figure E: Our vertex refiner (U-Net) for 128 resolution.

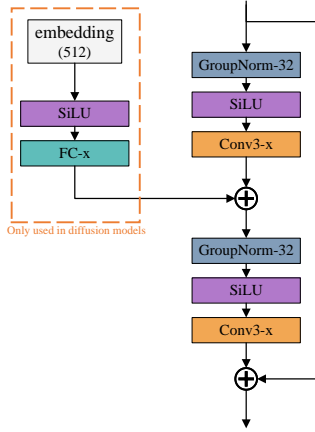


Figure F: The architecture of our ResNet Block.

where  $n_x$  is the nearest neighbor of  $x$  in  $X \cup Y - \{x\}$  and  $\mathbb{I}(\cdot)$  is a indicator function. For example, if  $n_x \in X$ ,  $\mathbb{I}(n_x \in X) = 1$ . If  $n_x \notin X$ ,  $\mathbb{I}(n_x \in X) = 0$ . Ideally, if  $X$  and  $Y$  are sampled from the same distribution, the 1-NNA value should be 50%. The closer the 1-NNA value is to 50%, the more similar  $X$  and  $Y$  are.

#### 4 GENUDC WITHOUT U-NET

We present the pipeline of GenUDC without U-Net in Fig. G. In GenUDC without U-Net, we remove the vertex refiner (U-Net) and concatenate the face part  $\mathcal{F}$  and the vertex part  $\mathcal{V}$  together to train the latent diffusion model.

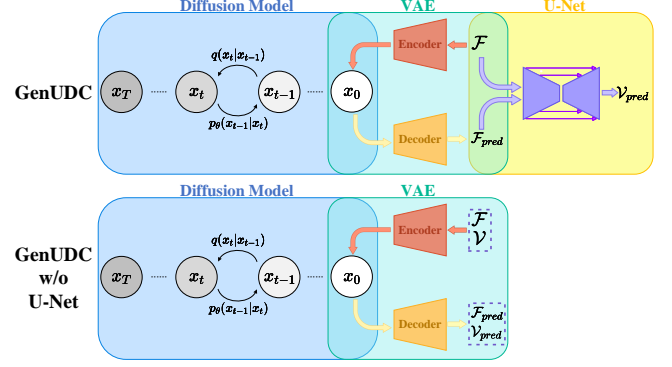
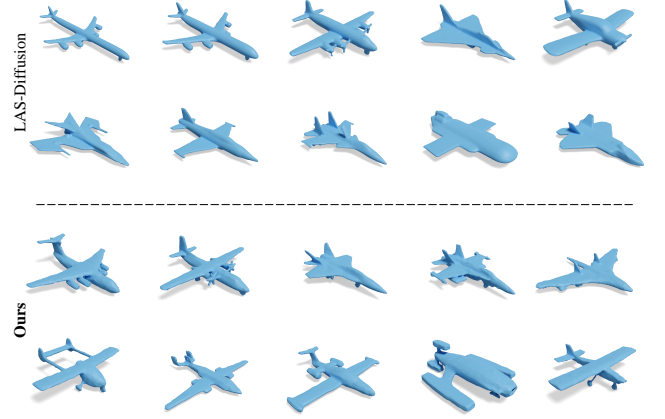


Figure G: The pipeline differences between GenUDC and GenUDC without U-Net.

Figure H: Qualitative evaluation of shape generation in  $128^3$  resolution.

#### 5 MORE VISUAL SAMPLES OF GENUDC

We present visual samples of 128 resolution in Fig. H. Both methods are visually good. It is difficult to distinguish which method is better according to those visual samples. However, Tab. 3 of the main paper proves that our variety and distribution are better than LAS-Diffusion [7].

#### REFERENCES

- [1] Ding-Yun Chen, Xiao-Pei Tian, Yu-Te Shen, and Ming Ouhyoung. 2003. On visual similarity based 3D model retrieval. In *CGF*.
- [2] Zhiqin Chen and Hao Zhang. 2019. Learning Implicit Fields for Generative Shape Modeling. In *CVPR*.
- [3] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep Residual Learning for Image Recognition. In *CVPR*.
- [4] Jonathan Ho, Ajay Jain, and Pieter Abbeel. 2020. Denoising diffusion probabilistic models. *NeurIPS*.
- [5] Moritz Ibing, Isaak Lim, and Leif Kobbelt. 2021. 3D Shape Generation With Grid-Based Implicit Functions. In *CVPR*.
- [6] David Lopez-Paz and Maxime Oquab. 2017. Revisiting classifier two-sample tests. In *ICLR*.
- [7] Xin-Yang Zheng, Hao Pan, Peng-Shuai Wang, Xin Tong, Yang Liu, and Heung-Yeung Shum. 2023. Locally Attentional SDF Diffusion for Controllable 3D Shape Generation. *SIGGRAPH*.