DiT-3D: Exploring Plain Diffusion Transformers for 3D Shape Generation

Shentong Mo[1] **Enze Xie**[2*] **Ruihang Chu**[3] **Lewei Yao**[2]
Lanqing Hong[2] **Matthias Nießner**[4] **Zhenguo Li**[2]

[1]MBZUAI, [2]Huawei Noah's Ark Lab, [3]CUHK, [4]TUM
https://DiT-3D.github.io

Abstract

Recent Diffusion Transformers (*e.g.* DiT [1]) have demonstrated their powerful effectiveness in generating high-quality 2D images. However, it is still being determined whether the Transformer architecture performs equally well in 3D shape generation, as previous 3D diffusion methods mostly adopted the U-Net architecture. To bridge this gap, we propose a novel Diffusion Transformer for 3D shape generation, namely DiT-3D, which can directly operate the denoising process on voxelized point clouds using plain Transformers. Compared to existing U-Net approaches, our DiT-3D is more scalable in model size and produces much higher quality generations. Specifically, the DiT-3D adopts the design philosophy of DiT [1] but modifies it by incorporating 3D positional and patch embeddings to adaptively aggregate input from voxelized point clouds. To reduce the computational cost of self-attention in 3D shape generation, we incorporate 3D window attention into Transformer blocks, as the increased 3D token length resulting from the additional dimension of voxels can lead to high computation. Finally, linear and devoxelization layers are used to predict the denoised point clouds. In addition, our transformer architecture supports efficient fine-tuning from 2D to 3D, where the pre-trained DiT-2D checkpoint on ImageNet can significantly improve DiT-3D on ShapeNet. Experimental results on the ShapeNet dataset demonstrate that the proposed DiT-3D achieves state-of-the-art performance in high-fidelity and diverse 3D point cloud generation. In particular, our DiT-3D decreases the 1-Nearest Neighbor Accuracy of the state-of-the-art method by 4.59 and increases the Coverage metric by 3.51 when evaluated on Chamfer Distance.

1 Introduction

In recent times, there has been a growing interest in exploring the potential of diffusion transformers for high-fidelity image generation, as evinced by a series of scholarly works [1, 2, 3, 4]. Notably, a seminal work by Peebles et al. [1] proposed the replacement of the widely-used U-Net backbone with a scalable transformer. Specifically, the proposed method operates on latent patches by training latent 2D diffusion models. However, the efficacy of plain diffusion transformers for 3D shape generation has yet to be explored, as most existing 3D diffusion approaches continue to adopt the U-Net backbone.

Generating high-fidelity point clouds for 3D shape generation is a challenging and significant problem. Early generative methods [5, 6, 7] addressed this problem by directly optimizing heuristic loss objectives, such as Chamfer Distance (CD) and Earth Mover's Distance (EMD). More recent works [8, 9, 10, 11] have explored the usage of the generative adversarial network (GAN)-based

---

[*]Corresponding author.

Figure 1: Examples of high-fidelity and diverse 3D point clouds produced from DiT-3D.

and flow-based models to generate 3D point clouds from a probabilistic perspective. Recently, researchers [12, 13, 14, 15] have turned to various denoising diffusion probabilistic models (DDPMs) to generate entire shapes from random noise. For instance, PVD [12] employed the point-voxel representation of 3D shapes as input to DDPMs. They reversed the diffusion process from observed point clouds to Gaussian noise by optimizing a variational lower bound to the likelihood function. Recently, the Diffusion Transformer (DiT) [1, 2] has been shown to surpass the U-Net architecture in 2D image generation, owing to its simple design and superior generative performance. Consequently, we investigate the potential of the Diffusion Transformer for 3D generation. However, extending the 2D DiT to 3D poses two significant challenges: (1) Point clouds are intrinsically unordered, unlike images where pixels are ordered; and (2) The tokens in 3D space have an additional dimension compared to 2D images, resulting in a substantial increase in computational cost.

This work introduces DiT-3D, a novel diffusion transformer architecture designed for 3D shape generation that leverages the denoising process of DDPM on 3D point clouds. The proposed model inherits the simple design of the modules in DiT-2D, with only minor adaptations to enable it to generalize to 3D generation tasks. To tackle the challenge posed by the unordered data structure of point clouds, we convert the point cloud into a voxel representation. DiT-3D employs 3D positional embedding and 3D patch embedding on the voxelized point clouds to extract point-voxel features and effectively process the unordered data. Furthermore, to address the computational cost associated with a large number of tokens in 3D space, we introduce a 3D window attention operator instead of the vanilla global attention in DiT-2D. This operator significantly reduces training time and memory usage, making DiT-3D feasible for large-scale 3D generation tasks. Finally, we utilize linear and devoxelization layers to predict the noised point clouds in the reversed process to generate final 3D shapes.

In order to address the computational cost associated with a large number of tokens in 3D space, we also introduce a parameter-efficient tuning method to utilize the pre-trained DiT-2D model on ImageNet as initialization for DiT-3D (window attention shares the same parameters with vanilla attention). Benefiting from the substantial similarity between the network structure and parameters of DiT-3D and DiT-2D, the representations learned on ImageNet significantly improve 3D generation, despite the significant domain disparity between 2D images and 3D point clouds. To our knowledge, we are the first to achieve parameter-efficient fine-tuning from 2D ImageNet pre-trained weights for high-fidelity and diverse 3D shape generation. In particular, we highly decrease the training parameters from 32.8MB to only 0.09MB.

We present a comprehensive evaluation of DiT-3D on a diverse set of object classes in the ShapeNet benchmark, where it achieves state-of-the-art performance compared to previous non-DDPM and DDPM-based 3D shape generation methods. Qualitative visualizations further emphasize the efficacy of DiT-3D in generating high-fidelity 3D shapes. Extensive ablation studies confirm the significance of 3D positional embeddings, window attention, and 2D pre-training in 3D shape generation. Moreover, we demonstrate that DiT-3D is easily scalable regarding patch sizes, voxel sizes, and model sizes. Our

findings align with those of DiT-2D, where increasing the model size leads to continuous performance improvements. In addition, our parameter-efficient fine-tuning from DiT-2D ImageNet pre-trained weights highly decreases the training parameters while achieving competitive generation performance. By only training 0.09MB parameters of models from the source class to the target class, we also achieve comparable results of quality and diversity in terms of all metrics.

Our main contributions can be summarized as follows:

- We present DiT-3D, the first plain diffusion transformer architecture for point cloud shape generation that can effectively perform denoising operations on voxelized point clouds.

- We make several simple yet effective modifications on DiT-3D, including 3D positional and patch embeddings, 3D window attention, and 2D pre-training on ImageNet. These modifications significantly improve the performance of DiT-3D while maintaining efficiency.

- Extensive experiments on the ShapeNet dataset demonstrate the state-of-the-art superiority of DiT-3D over previous non-DDPM and DDPM baselines in generating high-fidelity shapes.

2 Related Work

3D Shape Generation. 3D shape generation aims to synthesize high-fidelity point clouds or meshes using generative models, such as variational autoencoders [16, 17, 18], generative adversarial networks [19, 8, 20], and normalized flows [9, 10, 11]. Typically, PointFlow [9] utilized a probabilistic framework based on the continuous normalizing flow to generate 3D point clouds from two-level hierarchical distributions. ShapeGF [21] trained a score-matching energy-based network to learn the distribution of points across gradient fields using Langevin dynamics. More recently, GET3D [14] leveraged a signed distance field (SDF) and a texture field as two latent codes to learn a generative model that directly generates 3D meshes. In this work, we mainly focus on denoising diffusion probabilistic models for generating high-fidelity 3D point clouds starting from random noise, where point and shape distributions are not separated.

Diffusion Models. Diffusion models [22, 23, 24] have been demonstrated to be effective in many generative tasks, such as image generation [25], image restoration [26], speech generation [27], and video generation [28]. Denoising diffusion probabilistic models (DDPMs) [22, 23] utilized a forward noising process that gradually adds Gaussian noise to images and trained a reverse process that inverts the forward process. In recent years, researchers [29, 12, 13, 30, 15, 31, 32] have tried to explore diverse pipelines based on diffusion probabilistic models to achieve 3D shape generation. For example, PVD [12] applied DDPM based on PVCNNs [33] on the point-voxel representation of 3D shapes with structured locality into point clouds. To improve the generation quality, LION [13] used two DDPMs to learn a hierarchical latent space based on a global shape latent representation and a point-structured latent space separately. Different from them, we will solve the 3D shape generation problem in our approach by designing a plain transformer-based architecture backbone to replace the U-Net backbone for reversing the diffusion process from observed point clouds to Gaussian noise. Meanwhile, our 3D plain diffusion transformer supports multi-class training with learnable class embeddings as the condition and parameter-efficient fine-tuning with modality and domain transferability differ from DDPM-based 3D generation approaches discussed above.

Transformers in Diffusion Generation. Diffusion Transformers [1, 2, 3, 4] have recently shown their impressive capacity to generate high-fidelity images. For instance, Diffusion Transformer (DiT) [1] proposed a plain diffusion Transformer architecture to learn the denoising diffusion process on latent patches from a pre-trained pre-trained variational autoencoder model in Stable Diffusion [34]. U-ViT [2] incorporated all the time, condition, and noisy image patches as tokens and utilized a Vision transformer(ViT) [35]-based architecture with long skip connections between shallow and deep layers. More recently, UniDiffuser [3] designed a unified transformer for diffusion models to handle input types of different modalities by learning all distributions simultaneously. While those diffusion transformer approaches achieve promising performance in 2D image generation, how a plain diffusion transformer performs on 3D shape generation is still being determined. In contrast, we develop a novel plain diffusion transformer for 3D shape generation that can effectively perform denoising operations on voxelized point clouds. Furthermore, the proposed DiT-3D can support parameter-efficient fine-tuning with transferability across modality and domain.
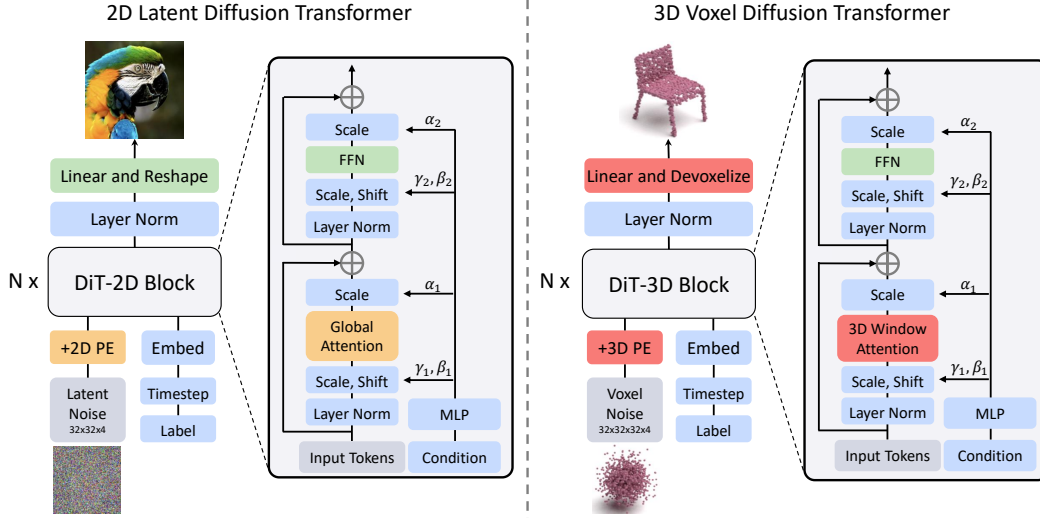
Figure 2: Illustration of the proposed Diffusion Transformers (DiT-3D) for 3D shape generation. The plain diffusion transformer takes voxelized point clouds as input, and a patchification operator is used to generate token-level patch embeddings, where 3D positional embeddings are added together. Then, multiple transformer blocks based on 3D window attention extract point-voxel representations from all input tokens. Finally, the unpatchified voxel tensor output from a linear layer is devoxelized to predict the noise in the point cloud space.

3 Method

Given a set of 3D point clouds, we aim to learn a plain diffusion transformer for synthesizing new high-fidelity point clouds. We propose a novel diffusion transformer that operates the denoising process of DDPM on voxelized point clouds, namely DiT-3D, which consists of two main modules: Design DiT for 3D Point Cloud Generation in Section 3.2 and Efficient Modality/Domain Transfer with Parameter-efficient Fine-tuning in Section 3.3.

3.1 Preliminaries

In this section, we first describe the problem setup and notations and then revisit denoising diffusion probabilistic models (DDPMs) for 3D shape generation and diffusion transformers on 2D images.

Problem Setup and Notations. Given a set $\mathcal{S} = \{\mathbf{p}_i\}_{i=1}^S$ of 3D shapes with M classes, our goal is to train a plain diffusion transformer from these point clouds for generating high-fidelity point clouds. For each point cloud $\mathbf{p}_i$, we have N points for x, y, z coordinates, that is $\mathbf{p}_i \in \mathbb{R}^{N \times 3}$. Note that we have a class label for the 3D shape $\mathbf{p}_i$, which is denoted as $\{y_i\}_{i=1}^M$ with y_i for the ground-truth category entry i as 1. During the training, we take the class label as input to achieve classifier-free guidance in conditional diffusion models, following the prior diffusion transformer (*i.e.*, DiT [1]) on images.

Revisit DDPMs on 3D Shape Generation. To solve the 3D shape generation problem, previous work [12] based on denoising diffusion probabilistic models (DDPMs) define a forward noising process that gradually applies noise to real data $\mathbf{x}_0$ as $q(\mathbf{x}_t|\mathbf{x}_{t-1}) = \mathcal{N}(\mathbf{x}_t; \sqrt{1-\beta_t}\mathbf{x}_{t-1}, \beta_t \mathbf{I})$, where β_t is a Gaussian noise value between 0 and 1. In particular, the denoising process produces a series of shape variables with decreasing levels of noise, denoted as $\mathbf{x}_T, \mathbf{x}_{T-1}, ..., \mathbf{x}_0$, where $\mathbf{x}_T$ is sampled from a Gaussian prior and $\mathbf{x}_0$ is the final output. With the reparameterization trick, we can have $\mathbf{x}_t = \sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}$, where $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$, $\alpha_t = 1 - \beta_t$, and $\bar{\alpha}_t = \prod_{i=1}^t \alpha_i$.

For the reverse process, diffusion models are trained to learn a denoising network $\boldsymbol{\theta}$ for inverting forward process corruption as $p_{\boldsymbol{\theta}}(\mathbf{x}_{t-1}|\mathbf{x}_t) = \mathcal{N}(\mathbf{x}_{t-1}; \boldsymbol{\mu}_{\boldsymbol{\theta}}(\mathbf{x}_t, t), \sigma_t^2 \mathbf{I})$. The training objective is to maximize a variational lower bound of the negative log data likelihood that involves all of $\mathbf{x}_0, ..., \mathbf{x}_T$

4

as

$$\mathcal{L} = \sum_t -p_{\boldsymbol{\theta}}(\mathbf{x}_0|\mathbf{x}_1) + \mathcal{D}_{\mathrm{KL}}(q(\mathbf{x}_{t-1}|\mathbf{x}_t,\mathbf{x}_0)||p_{\boldsymbol{\theta}}(\mathbf{x}_{t-1}|\mathbf{x}_t))) \tag{1}$$

where $\mathcal{D}_{\mathrm{KL}}(\cdot||\cdot)$ denotes the KL divergence measuring the distance between two distributions. Since both $p_{\boldsymbol{\theta}}(\mathbf{x}_{t-1}|\mathbf{x}_t))$ and $q(\mathbf{x}_{t-1}|\mathbf{x}_t,\mathbf{x}_0)$ are Gaussians, we can reparameterize $\boldsymbol{\mu}_{\boldsymbol{\theta}}(\mathbf{x}_t,t)$ to predict the noise $\boldsymbol{\epsilon}_{\boldsymbol{\theta}}(\mathbf{x}_t,t)$. In the end, the training objective can be reduced to a simple mean-squared loss between the model output $\boldsymbol{\epsilon}_{\boldsymbol{\theta}}(\mathbf{x}_t,t)$ and the ground truth Gaussian noise $\boldsymbol{\epsilon}$ as: $\mathcal{L}_{\mathrm{simple}} = \|\boldsymbol{\epsilon} - \boldsymbol{\epsilon}_{\boldsymbol{\theta}}(\mathbf{x}_t,t)\|^2$. After $p_{\boldsymbol{\theta}}(\mathbf{x}_{t-1}|\mathbf{x}_t))$ is trained, new point clouds can be generated by progressively sampling $\mathbf{x}_{t-1} \sim p_{\boldsymbol{\theta}}(\mathbf{x}_{t-1}|\mathbf{x}_t))$ by using the reparameterization trick with initialization of $\mathbf{x}_T \sim \mathcal{N}(\mathbf{0},\mathbf{I})$.

Revisit Diffusion Transformer (DiT) on 2D Image Generation. To generate high-fidelity 2D images, DiT proposed to train latent diffusion models (LDMs) with Transformers the backbone, consisting of two training models. They first extract the latent code $\mathbf{z}$ from an image sample $\mathbf{x}$ using an autoencoder with an encoder $f_{\mathrm{enc}}(\cdot)$ and a decoder $f_{\mathrm{dec}}(\cdot)$, that is, $\mathbf{z} = f_{\mathrm{enc}}(\mathbf{x})$. The decoder is used to reconstruct the image sample $\hat{\mathbf{x}}$ from the latent code $\mathbf{z}$, *i.e.*, $\hat{\mathbf{x}} = f_{\mathrm{dec}}(\mathbf{z})$. Based on latent codes $\mathbf{z}$, a latent diffusion transformer with multiple designed blocks is trained with time embedding $\mathbf{t}$ and class embedding $\mathbf{c}$, where a self-attention and a feed-forward module are involved in each block. Note that they apply patchification on latent code $\mathbf{z}$ to extract a sequence of patch embeddings and depatchification operators are used to predict the denoised latent code $\mathbf{z}$.

Although DDPMs achieved promising performance on 3D shape generation, they can only handle single-class training based on PVCNNs [33] as the encoder to extract 3D representations, and they cannot learn explicit class-conditional embeddings. Furthermore, we are not able to directly transfer their single-class pre-trained model to new classes with parameter-efficient fine-tuning. Meanwhile, we empirically observe that the direct extension of DiT [1] on point clouds does not work. To address this problem, we propose a novel plain diffusion transformer for 3D shape generation that can effectively achieve the denoising processes on voxelized point clouds, as illustrated in Figure 2.

3.2 Diffusion Transformer for 3D Point Cloud Generation

To enable denoising operations using a plain diffusion transformer, we propose several adaptations to 3D point cloud generation in Figure 2 within the framework of DiT [1]. Specifically, our DiT-3D model accepts voxelized point clouds as input and employs a patchification operator to generate token-level patch embeddings. We add 3D positional embeddings to these embeddings and extract point-voxel representations from all input tokens using multiple transformer blocks based on 3D window attention. Finally, we apply a devoxelized linear layer to the unpatchified voxel output, allowing us to predict the noise in the point cloud space.

Denoising on Voxelized Point Clouds. Point clouds are inherently unordered, unlike images where pixels follow a specific order. We encountered difficulty in our attempt to train a diffusion transformer on point coordinates due to the sparse distribution of points in the 3D embedding space. To address this issue, we decided to voxelize the point clouds into dense representations, allowing the diffusion transformers to extract point-voxel features. Our approach differs from DiT [1], which utilizes latent codes $\mathbf{z}$ to train the latent diffusion transformer. Instead, we directly train the denoising process on voxelized point clouds using the diffusion transformer. For each point cloud $\mathbf{p}_i \in \mathbb{R}^{N \times 3}$ with N points for x, y, z coordinates, we first voxelize it as input $\mathbf{v}_i \in \mathbb{R}^{V \times V \times V \times 3}$.

3D Positional and Patch Embeddings. With the voxel input $\mathbf{v}_i \in \mathbb{R}^{V \times V \times V \times 3}$, we introduce patchification operator with a patch size $p \times p \times p$ to generate a sequence of patch tokens $\mathbf{t} \in \mathbb{R}^{L \times 3}$. $L = (V/p)^3$ denotes the total number of patchified tokens. A 3D convolution layer is applied on patch tokens to extract patch embeddings $\mathbf{e} \in \mathbb{R}^{L \times D}$, where D is the dimension of embeddings. To adapt to our voxelized point clouds, we add frequency-based sine-cosine 3D positional embeddings instead of the 2D version in DiT [1] to all input tokens. Based on these patch-level tokens, we introduce time embeddings $\mathbf{t}$ and class embeddings $\mathbf{c}$ as input to achieve multi-class training with learnable class embeddings as the condition, which differs from existing 3D generation approaches with U-Net as the backbone.

3D Window Attention. Due to the increased token length resulting from the additional dimension in 3D space, the computational cost of 3D Transformers can be significantly high. To address this issue, we introduce efficient 3D window attention into Transformer blocks blocks to propagate point-voxel

features in efficient memory usage. For the original multi-head self-attention process with each of the heads Q, K, V have the same dimensions $L \times D$, where $L = (V/p)^3$ is the length of input tokens, we can have the attention operator as:

$$\text{Attention}(Q, K, V) = \text{Softmax}(\frac{QK^\top}{\sqrt{D_h}} V) \qquad (2)$$

where D_h is the dimension size of each head. The computational complexity of this process is $\mathcal{O}(L^2)$, which will be largely expensive for high voxel resolutions. Inspired by [36], we extend the 2D window attention operator to a 3D one for 3D input tokens instead of vanilla global attention. This process uses a window size of R to reduce the length of total input tokens as

$$\begin{aligned} \hat{K} &= \text{Reshape}(\frac{L}{R^3}, D \cdot R^3) \\ K &= \text{Linear}(D \cdot R^3, D)(\hat{K}) \end{aligned} \qquad (3)$$

where K is the input tokens to be reduced. Reshape $\left(\frac{L}{R^3}, D \cdot R^3\right)$ denotes to reshape K to the one with shape of $\frac{L}{R^3} \times (D \cdot R^3)$, and $\text{Linear}(C_{in}, C_{out})(\cdot)$ denotes to a linear layer with a C_{in}-dimensional tensor as input and a C_{out}-dimensional tensor as output. Therefore, the new K has the shape of $\frac{L}{R^3} \times D$. As a result, the complexity of the self-attention operator in Equation (2) is reduced from $\mathcal{O}(L^2)$ to $\mathcal{O}(\frac{L^2}{R^3})$. In our experiments, we set R to 4 in the default setting.

Devoxelized Prediction. Since the transformers blocks are implemented on voxelized point clouds, we can not directly use a standard linear decoder to predict the output noise $\epsilon_\theta(\mathbf{x}_t, t)$ from point clouds. In order to generate the output noise, we devoxelize output tokens from the linear decoder. We first apply the final layer norm and linearly decode each token into a $p \times p \times p \times L \times 3$ tensor, where L is the total number of input tokens. Then we unpatchify the decoded token into a voxel tensor with the shape of $V \times V \times V \times 3$. Finally, the unpatchified voxel tensor is devoxelized into a $N \times 3$ tensor as the output noise $\epsilon_\theta(\mathbf{x}_t, t)$, matching with the ground truth Gaussian noise ϵ in the point cloud space.

Model Scaling. Our DiT-3D is designed to be scalable, adapting to varying voxel sizes, patch sizes, and model sizes. Specifically, it can flexibly accommodate voxel dimensions of 16, 32, 64, patch dimensions of 2, 4, 8, and model complexity ranging from Small, Base, Large and Extra Large, as demonstrated in DiT [1]. For instance, a model designated as DiT-3D-S/4 refers that it utilizes the Small configuration of the DiT model [1], with a patch size p of 4.

3.3 Efficient Modality/Domain Transfer with Parameter-efficient Fine-tuning

Leveraging the scalability of the plain diffusion transformer, we investigate parameter-efficient fine-tuning for achieving modality and domain transferability. To facilitate modality transfer from 2D to 3D, we can leverage the knowledge pre-trained on large-scale 2D images using DiT [1]. For domain transfer from a source class to target classes, we train DiT-3D on a single class (*e.g.* chair) and transfer the model's parameters to other classes (*e.g.* airplane, car).

Modality Transfer: 2D (ImageNet) → 3D (ShapeNet). As large-scale pre-trained DiT checkpoints[2] are readily available, we can skip training our diffusion transformer from scratch. Instead, we can load most of the weights from the DiT [1] pre-trained on ImageNet [37] into our DiT-3D and continue with fine-tuning. To further optimize training efficiency, we adopt the parameter-efficient fine-tuning approach described in recent work, DiffFit [4], which involves freezing the majority of parameters and only training the newly-added scale factors, bias term, normalization, and class condition modules. It's worth noting that we initialize γ to 1, which is then multiplied with the frozen layers.

Domain Transfer: Source Class → Target Class. Given a pre-trained DiT-3D model on chair data, we can use the parameter-efficient fine-tuning approach to extend its applicability to new categories. Specifically, following the same methodology as described above, we leverage the fine-tuning strategy of DiffFit and obtain satisfactory generation results.

---

[2]https://github.com/facebookresearch/DiT/tree/main/diffusion

Table 1: Comparison results (%) on shape metrics of our DiT-3D and baseline models.

Method	Chair 1-NNA (↓) CD	Chair 1-NNA (↓) EMD	Chair COV (↑) CD	Chair COV (↑) EMD	Airplane 1-NNA (↓) CD	Airplane 1-NNA (↓) EMD	Airplane COV (↑) CD	Airplane COV (↑) EMD	Car 1-NNA (↓) CD	Car 1-NNA (↓) EMD	Car COV (↑) CD	Car COV (↑) EMD
r-GAN [8]	83.69	99.70	24.27	15.13	98.40	96.79	30.12	14.32	94.46	99.01	19.03	6.539
l-GAN (CD) [8]	68.58	83.84	41.99	29.31	87.30	93.95	38.52	21.23	66.49	88.78	38.92	23.58
l-GAN (EMD) [8]	71.90	64.65	38.07	44.86	89.49	76.91	38.27	38.52	71.16	66.19	37.78	45.17
PointFlow [9]	62.84	60.57	42.90	50.00	75.68	70.74	47.90	46.41	58.10	56.25	46.88	50.00
SoftFlow [10]	59.21	60.05	41.39	47.43	76.05	65.80	46.91	47.90	64.77	60.09	42.90	44.60
SetVAE [18]	58.84	60.57	46.83	44.26	76.54	67.65	43.70	48.40	59.94	59.94	49.15	46.59
DPF-Net [11]	62.00	58.53	44.71	48.79	75.18	65.55	46.17	48.89	62.35	54.48	45.74	49.43
DPM [29]	60.05	74.77	44.86	35.50	76.42	86.91	48.64	33.83	68.89	79.97	44.03	34.94
PVD [12]	57.09	60.87	36.68	49.24	73.82	64.81	48.88	52.09	54.55	53.83	41.19	50.56
LION [13]	53.70	52.34	48.94	52.11	67.41	61.23	47.16	49.63	53.41	51.14	**50.00**	**56.53**
GET3D [14]	75.26	72.49	43.36	42.77	–	–	–	–	75.26	72.49	15.04	18.38
MeshDiffusion [15]	53.69	57.63	46.00	46.71	66.44	76.26	47.34	42.15	81.43	87.84	34.07	25.85
DiT-3D (ours)	**49.11**	**50.73**	**52.45**	**54.32**	**62.35**	**58.67**	**53.16**	**54.39**	**48.24**	**49.35**	**50.00**	56.38

3.4 Relationship to DiT [1]

Our DiT-3D contains multiple different and efficient designs for 3D shape generation compared with DiT [1] on 2D image generation:

- We effectively achieve the diffusion space on voxelized point clouds, while DiT needs the latent codes from a pre-trained variational autoencoder as the denoising target.

- Our plain diffusion transformer first incorporates frequency-based sine-cosine 3D positional embeddings with patch embeddings for voxel structure locality.

- We are the first to propose efficient 3D window attention in the transformer blocks for reducing the complexity of the self-attention operator in DiT.

- We add a devoxelized operator to the final output of the last linear layer from DiT for denoising the noise prediction in the point cloud space.

4 Experiments

4.1 Experimental Setup

Datasets. Following most previous works [12, 13], we use ShapeNet [38] Chair, Airplane, and Car as our primary datasets for 3D shape generation. For each 3D shape, we sample 2,048 points from 5,000 provided points in [38] for training and testing. We also use the same dataset splits and pre-processing in PointFlow [9], which normalizes the data globally across the whole dataset.

Evaluation Metrics. For comprehensive comparisons, we follow prior work [12, 13] and use Chamfer Distance (CD) and Earth Mover's Distance (EMD) as our distance metrics in computing 1-Nearest Neighbor Accuracy (1-NNA) and Coverage (COV) as main metrics to measure generative quality. 1-NNA calculates the leave-one-out accuracy of the 1-NN classifier to quantify point cloud generation performance, which is robust and correlates with generation quality and diversity. A lower 1-NNA score is better. COV measures the number of reference point clouds matched to at least one generated shape, correlating with generation diversity. Note that a higher COV score is better but does not measure the quality of the generated point clouds since low-quality but diverse generated point clouds can achieve high COV scores.

Implementation. Our implementation is based on the PyTorch [39] framework. The input voxel size is $32 \times 32 \times 32 \times 3$, i.e., $V = 32$. The final linear layer is initialized with zeros, and other weights initialization follows standard techniques in ViT [35]. The models were trained for 10,000 epochs using the Adam optimizer [40] with a learning rate of $1e - 4$ and a batch size of 128. We set $T = 1000$ for experiments. In the default setting, we use S/4 with patch size $p = 4$ as the backbone. Note that we utilize 3D window attention in partial blocks (i.e., 0,3,6,9) and global attention in other blocks.

Figure 3: Qualitative visualizations of high-fidelity and diverse 3D point cloud generation.

4.2 Comparison to State-of-the-art Works

In this work, we propose a novel and effective diffusion transformer for 3D shape generation. In order to validate the effectiveness of the proposed DiT-3D, we comprehensively compare it to previous non-DDPM and DDPM baselines. 1) r-GAN, 1-GAN [8]: (2018'ICML): generative models based on GANs trained on point clouds (l-GAN) and latent variables (l-GAN); 2) PointFlow [9] (2019'ICCV): a probabilistic framework to generate 3D point clouds from a two-level hierarchy of distributions with the continuous normalizing flow; 3) SoftFlow [10] (2020'NeurIPS): a probabilistic framework for training normalizing flows on manifolds to estimate the distribution of various shapes; 4) SetVAE [18] (2021'CVPR): a hierarchical variational autoencoder for sets to learn latent variables for coarse-to-fine dependency and permutation invariance; 5) DPF-Net [11] (2020'ECCV): a discrete latent variable network that builds on normalizing flows with affine coupling layers; 6) DPM [29] (2021'ICCV): the first DDPM approach to learn the reverse diffusion process for point clouds as a Markov chain conditioned on shape latent; 7) PVD [12] (2021'ICCV): a strong DDPM baseline based on the point-voxel representation of 3D shapes; 8) LION [13] (2022'NeurIPS): a recent method based on two hierarchical DDPMs in global latent and latent points spaces; 9) GET3D [14] (2022'NeurIPS): a generative model that directly generates explicit textured 3D meshes based on two latent codes (a 3D SDF and a texture field); 10) MeshDiffusion [15] (2023'ICLR): a very recent DDPM method using graph structure of meshes and deformable tetrahedral grid parametrization of 3D mesh shapes.

For chair generation, we report the quantitative comparison results in Table 1. As can be seen, we achieved the best performance in terms of all metrics compared to previous non-DDPM and DDPM baselines. In particular, the proposed DiT-3D significantly outperforms DPF-Net [11], the current state-of-the-art normalizing flows baseline, decreasing by 12.89 1-NNA@CD & 7.80 1-NNA@EMD, and increasing by 7.74 COV@CD & 3.8 COV@EMD. Moreover, we achieve superior performance gains compared to MeshDiffusion [15], the current state-of-the-art DDPM baseline on meshes, which implies the importance of replacing the U-Net with a plain diffusion transformer from observed point clouds for generating high-fidelity 3D shapes. Meanwhile, our DiT-3D outperforms LION [15] by a large margin, where we achieve the performance gains of 4.59 1-NNA@CD & 1.61 1-NNA@EMD, and 3.51 COV@CD & 2.21 COV@EMD. These significant improvements demonstrate the superiority of our method in 3D shape generation. In addition, significant gains in airplane and car generations can be observed in Table 1. These qualitative results also showcase the effectiveness of applying a plain diffusion transformer to operate the denoising process from point clouds for generating high-fidelity and diverse shapes, as shown in Figure 3.

4.3 Experimental Analysis

In this section, we performed ablation studies to demonstrate the benefit of introducing three main 3D design components (voxel diffusion, 3D positional embeddings, and 3D window attention) in 3D shape generation. We also conducted extensive experiments to explore the efficiency of 3D window attention, modality and domain transferability, and scalability.

Table 2: **Ablation studies** on 3D adaptation components of our DiT-3D.

Voxel Diffusion	3D Pos Embed	3D Window Attention	Training Cost (hours)	1-NNA (↓) CD	1-NNA (↓) EMD	COV (↑) CD	COV (↑) EMD
✗	✗	✗	86.53	99.86	99.93	7.768	4.653
✓	✗	✗	91.85	67.46	69.47	38.97	41.74
✓	✓	✗	91.85	51.99	**49.94**	**54.76**	**57.37**
✓	✓	✓	**41.67**	**49.11**	50.73	52.45	54.32

Table 3: **Transferability studies** on modality and domain with parameter-efficient fine-tuning.

ImageNet Pre-train	Efficient Fine-tuning	Params (MB)	1-NNA (↓) CD	1-NNA (↓) EMD	COV (↑) CD	COV (↑) EMD
✗	✗	32.8	51.99	49.94	**54.76**	**57.37**
✓	✗	32.8	**49.07**	**49.76**	53.26	55.75
✓	✓	**0.09**	50.87	50.23	52.59	55.36

(a) Modality transfer.

Source Domain	Target Domain	Params (MB)	1-NNA (↓) CD	1-NNA (↓) EMD	COV (↑) CD	COV (↑) EMD
Chair	Chair	32.8	**51.99**	**49.94**	**54.76**	**57.37**
Airplane	Chair	**0.09**	52.56	50.75	53.71	56.32
Airplane	Airplane	32.8	**62.81**	**58.31**	**55.04**	**54.58**
Chair	Airplane	**0.09**	63.58	59.17	53.25	53.68

(b) Domain transfer.

Ablation on 3D Design Components. In order to validate the effectiveness of the introduced 3D adaptation components (voxel diffusion, 3D positional embeddings, and 3D window attention), we ablate the necessity of each module and report the quantitative results in Table 2. Note that no voxel diffusion means we directly perform the denoising process on point coordinates without voxelized point clouds and devoxelization prediction. We can observe that adding bearable voxel diffusion to the vanilla baseline highly decreases the results of 1-NNA (by 32.40 @CD and 30.46 @AUC) and increase the performance of COV (by 31.202 @CD and 37.087 @EMD), which demonstrates the benefit of voxelized point clouds and devoxelization prediction in denoising process for 3D shape generation. Meanwhile, introducing 3D positional embedding in the baseline with voxel diffusion also increases the shape generation performance in terms of all metrics. More importantly, incorporating 3D window attention and two previous modules together into the baseline significantly decreases the training cost by 44.86 hours and results of 1-NNA by 50.75 @CD and 49.2 @EMD, and raises the performance of COV by 44.682 @CD and 49.667 @EMD. These improving results validate the importance of the proposed 3D adaptation components in the plain diffusion transformer to operate the denoising process from observed point clouds for 3D shape generation.

Influence of 2D Pretrain (ImageNet). In order to show the modality transferability of the proposed approach from 2D ImageNet pre-trained weights to 3D generation with parameter-efficient fine-tuning, we report the ablation results of ImageNet pre-train and efficient fine-tuning on chair generation in Table 3a. From comparisons, two main observations can be derived: 1) With the initialization with 2D ImageNet pre-trained weights, the proposed DiT-3D improves the quality of shape generation by decreasing 1-NNA by 2.92@CD and 0.18@EMD. 2) Incorporating parameter-efficient fine-tuning into 2D ImageNet pre-trained weights highly decreases the training parameters while achieving competitive generation performance.

Transferability in Domain. In addition, we explore the parameter-efficient fine-tuning for domain transferability in Table 3b. By only training 0.09MB parameters of models from the source class to the target class, we can achieve a comparable performance of quality and diversity in terms of all metrics. These results indicate that our DiT-3D can support flexible transferability on modality and domain, which differs from previous 3D generation methods [12, 13] based on U-Net as the backbone of DDPMs.

Scaling Patch size, Voxel size and Model Size. To explore the scalability of our plain diffusion transformer to flexible designs, we ablate the patch size from $\{2, 4, 8\}$, voxel size from $\{16, 32, 64\}$, and the model size from $\{S/4, B/4, L/4, XL/4\}$. As seen in Table 4a, when the patch size is 2, the proposed DiT-3D achieves the best performance. This trend is also observed in the original DiT [1] work for 2D image generation. In addition, increasing the voxel size from 16 to 64 for the input of the diffusion denoising process raises the performance in terms of all metrics, as shown in Table 4b. More importantly, we can still observe performance gains by scaling up the proposed plain diffusion transformer to XL/4 when the model is trained for 2,000 epochs. These promising results further demonstrate the strong scalability of our DiT-3D to flexible patch size, voxel size, and model sizes for generating high-fidelity 3D shapes.

Table 4: **Scalability studies** on flexible patch, voxel, and model sizes.

Patch Size	1-NNA (↓) CD	EMD	COV (↑) CD	EMD
8	53.84	51.20	50.01	52.49
4	51.99	49.94	**54.76**	**57.37**
2	**51.78**	**49.69**	54.54	55.94

(a) Patch size.

Voxel Size	1-NNA (↓) CD	EMD	COV (↑) CD	EMD
16	54.00	50.60	50.73	52.26
32	51.99	49.94	54.76	57.37
64	**50.32**	**49.73**	55.45	57.32

(b) Voxel size.

Model Size	Params (MB)	1-NNA (↓) CD	EMD	COV (↑) CD	EMD
S/4	32.8	56.31	55.82	47.21	50.75
B/4	130.2	55.59	54.91	50.09	52.80
L/4	579.0	52.96	53.57	51.88	54.41
XL/4	674.7	**51.95**	**52.50**	**52.71**	**54.31**

(c) Model size.

5   Conclusion

In this work, we present DiT-3D, a novel plain diffusion transformer for 3D shape generation, which can directly operate the denoising process on voxelized point clouds. Compared to existing U-Net approaches, our DiT-3D is more scalable in model size and produces much higher quality generations. Specifically, we incorporate 3D positional and patch embeddings to aggregate input from voxelized point clouds. We then incorporate 3D window attention into Transformer blocks to reduce the computational cost of 3D Transformers, which can be significantly high due to the increased token length resulting from the additional dimension in 3D. Finally, we leverage linear and devoxelization layers to predict the denoised point clouds. Due to the scalability of the Transformer, DiT-3D can easily support parameter-efficient fine-tuning with modality and domain transferability. Empirical results demonstrate the state-of-the-art performance of the proposed DiT-3D in high-fidelity and diverse 3D point cloud generation.

References

[1] William Peebles and Saining Xie. Scalable diffusion models with transformers. *arXiv preprint arXiv:2212.09748*, 2022. 1, 2, 3, 4, 5, 6, 7, 9

[2] Fan Bao, Shen Nie, Kaiwen Xue, Yue Cao, Chongxuan Li, Hang Su, and Jun Zhu. All are worth words: A vit backbone for diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 1, 2, 3

[3] Fan Bao, Shen Nie, Kaiwen Xue, Chongxuan Li, Shi Pu, Yaole Wang, Gang Yue, Yue Cao, Hang Su, and Jun Zhu. One transformer fits all distributions in multi-modal diffusion at scale. *arXiv preprint arXiv:2303.06555*, 2023. 1, 3

[4] Enze Xie, Lewei Yao, Han Shi, Zhili Liu, Daquan Zhou, Zhaoqiang Liu, Jiawei Li, and Zhenguo Li. Difffit: Unlocking transferability of large diffusion models via simple parameter-efficient fine-tuning. *arXiv preprint arXiv:2304.06648*, 2023. 1, 3, 6

[5] Haoqiang Fan, Hao Su, and Leonidas J. Guibas. A point set generation network for 3d object reconstruction from a single image. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 605–613, 2017. 1

[6] Thibault Groueix, Matthew Fisher, Vladimir G. Kim, Bryan C. Russell, and Mathieu Aubry. A papier-mâché approach to learning 3d surface generation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 216–224, 2018. 1

[7] Andrey Kurenkov, Jingwei Ji, Animesh Garg, Viraj Mehta, JunYoung Gwak, Christopher Bongsoo Choy, and Silvio Savarese. Deformnet: Free-form deformation network for 3d shape reconstruction from a single image. In *Proceedings of IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 858–866, 2017. 1

[8] Panos Achlioptas, Olga Diamanti, Ioannis Mitliagkas, and Leonidas Guibas. Learning representations and generative models for 3d point clouds. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2018. 1, 3, 7, 8

[9] Guandao Yang, Xun Huang, Zekun Hao, Ming-Yu Liu, Serge Belongie, and Bharath Hariharan. Pointflow: 3d point cloud generation with continuous normalizing flows. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 4541–4550, 2019. 1, 3, 7, 8

[10] Hyeongju Kim, Hyeonseung Lee, Woohyun Kang, Joun Yeop Lee, and Nam Soo Kim. Softflow: Probabilistic framework for normalizing flow on manifolds. In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*, 2020. 1, 3, 7, 8

[11] Roman Klokov, Edmond Boyer, and Jakob Verbeek. Discrete point flow networks for efficient point cloud generation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, page 694–710, 2020. 1, 3, 7, 8

[12] Linqi Zhou, Yilun Du, and Jiajun Wu. 3d shape generation and completion through point-voxel diffusion. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 5826–5835, 2021. 2, 3, 4, 7, 8, 9, 13, 14

[13] Xiaohui Zeng, Arash Vahdat, Francis Williams, Zan Gojcic, Or Litany, Sanja Fidler, and Karsten Kreis. Lion: Latent point diffusion models for 3d shape generation. In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*, 2022. 2, 3, 7, 8, 9, 13

[14] Jun Gao, Tianchang Shen, Zian Wang, Wenzheng Chen, Kangxue Yin, Daiqing Li, Or Litany, Zan Gojcic, and Sanja Fidler. Get3d: A generative model of high quality 3d textured shapes learned from images. In *Proceedings of Advances In Neural Information Processing Systems (NeurIPS)*, 2022. 2, 3, 7, 8

[15] Zhen Liu, Yao Feng, Michael J. Black, Derek Nowrouzezahrai, Liam Paull, and Weiyang Liu. Meshdiffusion: Score-based generative 3d mesh modeling. In *Proceedings of International Conference on Learning Representations (ICLR)*, 2023. 2, 3, 7, 8

[16] Yaoqing Yang, Chen Feng, Yiru Shen, and Dong Tian. Foldingnet: Point cloud auto-encoder via deep grid deformation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 206–215, 2018. 3

[17] Matheus Gadelha, Rui Wang, and Subhransu Maji. Multiresolution tree networks for 3d point cloud processing. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018. 3

[18] Jinwoo Kim, Jaehoon Yoo, Juho Lee, and Seunghoon Hong. Setvae: Learning hierarchical composition for generative modeling of set-structured data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 15059–15068, 2021. 3, 7, 8, 13

[19] Diego Valsesia, Giulia Fracastoro, and Enrico Magli. Learning localized generative models for 3d point clouds via graph convolution. In *Proceedings of International Conference on Learning Representations (ICLR)*, 2019. 3

[20] Dong Wook Shu, Sung Woo Park, and Junseok Kwon. 3d point cloud generative adversarial network based on tree structured graph convolutions. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 3859–3868, 2019. 3

[21] Ruojin Cai, Guandao Yang, Hadar Averbuch-Elor, Zekun Hao, Serge Belongie, Noah Snavely, and Bharath Hariharan. Learning gradient fields for shape generation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020. 3

[22] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *Proceedings of Advances In Neural Information Processing Systems (NeurIPS)*, pages 6840–6851, 2020. 3

[23] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. In *Proceedings of International Conference on Learning Representations (ICLR)*, 2021. 3

[24] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *ArXiv*, 2021. 3

[25] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily Denton, Seyed Kamyar Seyed Ghasemipour, Burcu Karagol Ayan, S. Sara Mahdavi, Rapha Gontijo Lopes, Tim Salimans, Jonathan Ho, David J Fleet, and Mohammad Norouzi. Photorealistic text-to-image diffusion models with deep language understanding. *arXiv preprint arXiv:2205.11487*, 2022. 3

[26] Chitwan Saharia, Jonathan Ho, William Chan, Tim Salimans, David J. Fleet, and Mohammad Norouzi. Image super-resolution via iterative refinement. *arXiv preprint arXiv:2104.07636*, 2021. 3

[27] Zhifeng Kong, Wei Ping, Jiaji Huang, Kexin Zhao, and Bryan Catanzaro. Diffwave: A versatile diffusion model for audio synthesis. In *Proceedings of International Conference on Learning Representations (ICLR)*, 2021. 3

[28] Jonathan Ho, William Chan, Chitwan Saharia, Jay Whang, Ruiqi Gao, Alexey Gritsenko, Diederik P. Kingma, Ben Poole, Mohammad Norouzi, David J. Fleet, and Tim Salimans. Imagen video: High definition video generation with diffusion models. *arXiv preprint arXiv:2210.02303*, 2022. 3

[29] Shitong Luo and Wei Hu. Diffusion probabilistic models for 3d point cloud generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2837–2845, 2021. 3, 7, 8, 13, 14

[30] Gimin Nam, Mariem Khlifi, Andrew Rodriguez, Alberto Tono, Linqi Zhou, and Paul Guerrero. 3d-ldm: Neural implicit 3d shape generation with latent diffusion models. *arXiv preprint arXiv:2212.00842*, 2022. 3

[31] Muheng Li, Yueqi Duan, Jie Zhou, and Jiwen Lu. Diffusion-sdf: Text-to-shape via voxelized diffusion. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 3

[32] Ruihang Chu, Enze Xie, Shentong Mo, Zhenguo Li, Matthias Nießner, Chi-Wing Fu, and Jiaya Jia. Diffcomplete: Diffusion-based generative 3d shape completion. *arXiv preprint arXiv:2306.16329*, 2023. 3

[33] Zhijian Liu, Haotian Tang, Yujun Lin, and Song Han. Point-voxel cnn for efficient 3d deep learning. In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*, 2019. 3, 5

[34] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10684–10695, 2022. 3

[35] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *Proceedings of International Conference on Learning Representations (ICLR)*, 2021. 3, 7

[36] Yanghao Li, Hanzi Mao, Ross B. Girshick, and Kaiming He. Exploring plain vision transformer backbones for object detection. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2022. 6

[37] Jia Deng, Wei Dong, Richard Socher, Li-Jia. Li, Kai Li, and Li Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 248–255, 2009. 6

[38] Angel X. Chang, Thomas Funkhouser, Leonidas Guibas, Pat Hanrahan, Qixing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, Jianxiong Xiao, Li Yi, and Fisher Yu. Shapenet: An information-rich 3d model repository. *arXiv preprint arXiv:1512.03012*, 2015. 7

[39] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. PyTorch: An imperative style, high-performance deep learning library. In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*, pages 8026–8037, 2019. 7

[40] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 7

Appendix

In this appendix, we first provide additional experimental analyses on multi-class training and DDPM sampling steps in Section A. Furthermore, we showcase qualitative visualizations for comparisons with state-of-the-art works, visualizations of the diffusion process, and more high-fidelity visualizations in Section B. Finally, we thoroughly discuss this work's limitation and broader impact in Section C.

A Additional Experimental Analyses

A.1 Results on Multi-class Training

In order to show the effectiveness of our DiT-3D on multi-class training, we change the training classes from {Chair}, {Chair,Car}, {Chair,Car,Airplane} and test on chair class in Table 5. We can observe that the proposed diffusion transformer achieves competitive generation results against category-specific models for all metrics by using learnable class embeddings as the condition after multi-class training. This benefits us in training only one global model for all classes simultaneously instead of training class-specific models multiple times, which differs from previous DDPM-based approaches without class embeddings involved.

Table 5: **Exploration studies** on multi-class training. One global model for all three classes achieves competitive results against category-specific models trained on only one class.

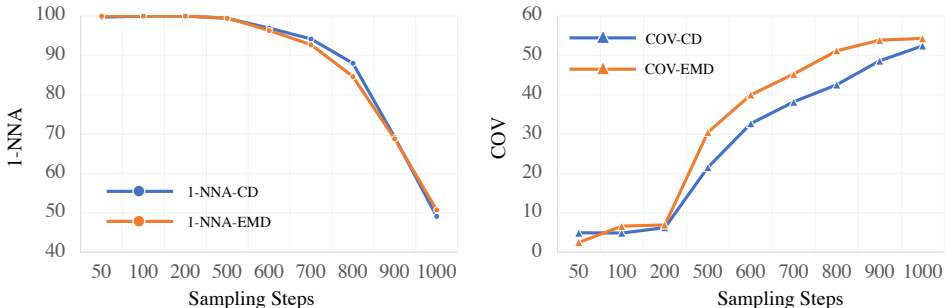| Train	Test	1-NNA ($\downarrow$)		COV ($\uparrow$)	
Class	Class	CD	EMD	CD	EMD
Chair	Chair	**51.99**	**49.94**	**54.76**	**57.37**
Chair, Car	Chair	52.68	50.62	54.15	56.83
Chair, Car, Airplane	Chair	53.35	51.84	52.81	55.30

Figure 4: Effect of sampling steps on 3D shape generation (*Chair*) during the inference stage.

A.2 Effect of Sampling Steps

Furthermore, we explore the effect of DDPM sampling steps T on the final performance during the inference stage in Figure 4. As can be seen, the proposed DiT-3D achieves the best results (lowest 1-NNA and highest COV) for all metrics (CD and EMD) when the number of sampling steps is set to 1000. This trend is consistent with similar conclusions in the prior DDPM work [13].

B Qualitative Visualizations

B.1 Comparisons with State-of-the-art Works

In order to qualitatively evaluate the generated 3D shapes, we compare the proposed DiT-3D with SetVAE [18], DPM [29], and PVD [12] on generated 3D point clouds of all three class in Figure 5. From comparisons, we can observe that the qualities of 3D point clouds generated by our framework are superior to SetVAE [18], a hierarchical variational autoencoder for sets to learn latent variables
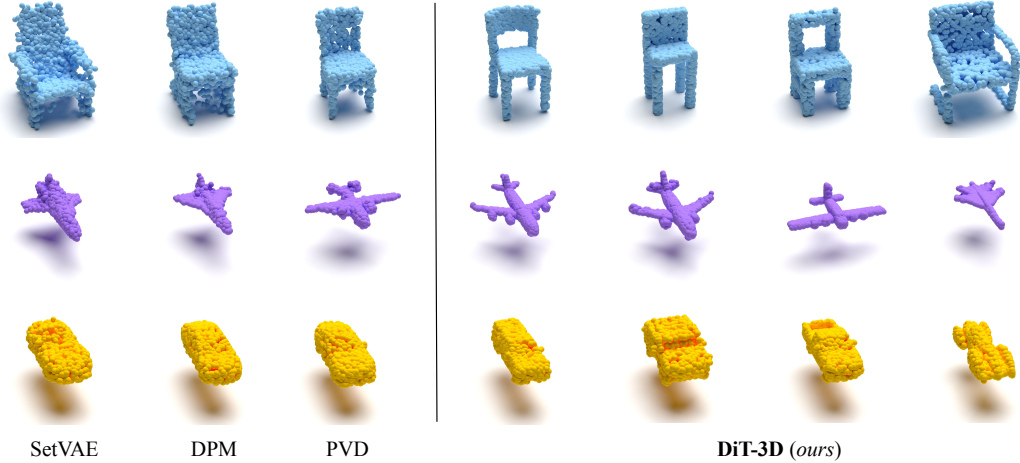
Figure 5: Qualitative comparisons with state-of-the-art works. The proposed DiT-3D generates high-fidelity and diverse point clouds of 3D shapes for each category.

for coarse-to-fine dependency and permutation invariance. Meanwhile, we achieve much better results than DPM [29], the first diffusion denoising probabilistic model on point cloud generation. More importantly, the proposed DiT-3D achieves high-fidelity and diverse results compared to the strong diffusion model based on point voxels, PVD [12]. These visualizations further showcase the superiority of our DiT-3D in generating high-fidelity and diverse shapes by using a plain diffusion transformer to operate the denoising process from point clouds.

B.2 Visualizations of Diffusion Process

Furthermore, we visualize the diffusion process of generated *Chair* shapes from 1000 sampling steps in Figure 6, generating 6 shapes from random noise to the final 3D shapes for each sample. From left to right, we can observe that our DiT-3D achieves a meaningful diffusion process to produce high-fidelity and diverse shapes. When the number of sampling steps is closer to 1000, the generated shapes are more realistic, while they are more like random noises in the initial few sampling steps. These qualitative diffusion process results also showcase the effectiveness of applying a plain diffusion transformer to generate high-fidelity and diverse shapes. The results of the diffusion process for *Airplane* and *Car* shapes generated from 1000 sampling steps are reported in Figure 7 and 8.

B.3 More Visualizations of Generated Shapes

To qualitatively showcase the high-fidelity and diverse properties of generated shapes, we visualize more generated samples from all three classes in Figure 9, 10, and 11. These qualitative visualizations demonstrate the effectiveness of the proposed 3D design components in a plain diffusion transformer to produce high-fidelity and diverse shapes by achieving the denoising process from point clouds of three categories directly.

C Discussion

Limitation & Future Work. This work thoroughly explores the plain diffusion transformer on point clouds for generating high-fidelity and diverse 3D shapes. However, we have yet to explore the potential of other 3D modalities, such as signed distance fields (SDFs) and meshes, or scaling our DiT-3D to large-scale training on more 3D shapes. These directions are promising, and we will leave them as the future work.

Broader Impact. The proposed DiT-3D generates high-fidelity and diverse 3D shapes from training samples in the existing ShapeNet benchmark, which might cause the model to learn internal biases in the data. These biased problems should be carefully solved for the deployment of real applications.
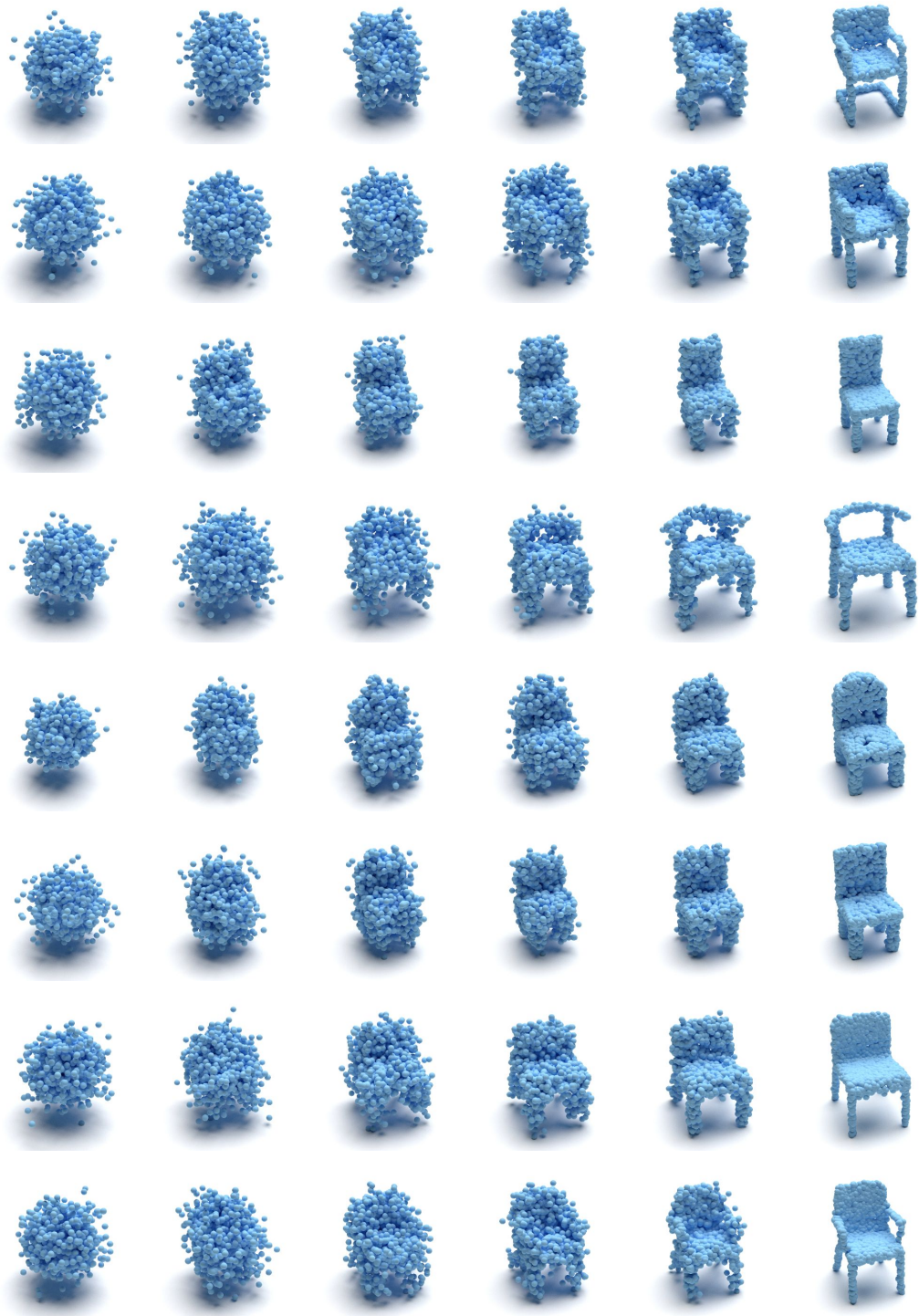
Figure 6: Qualitative visualizations of the diffusion process on *Chair* shape generation. The results of generating from random noise to final 3D shapes are shown in left-to-right order.
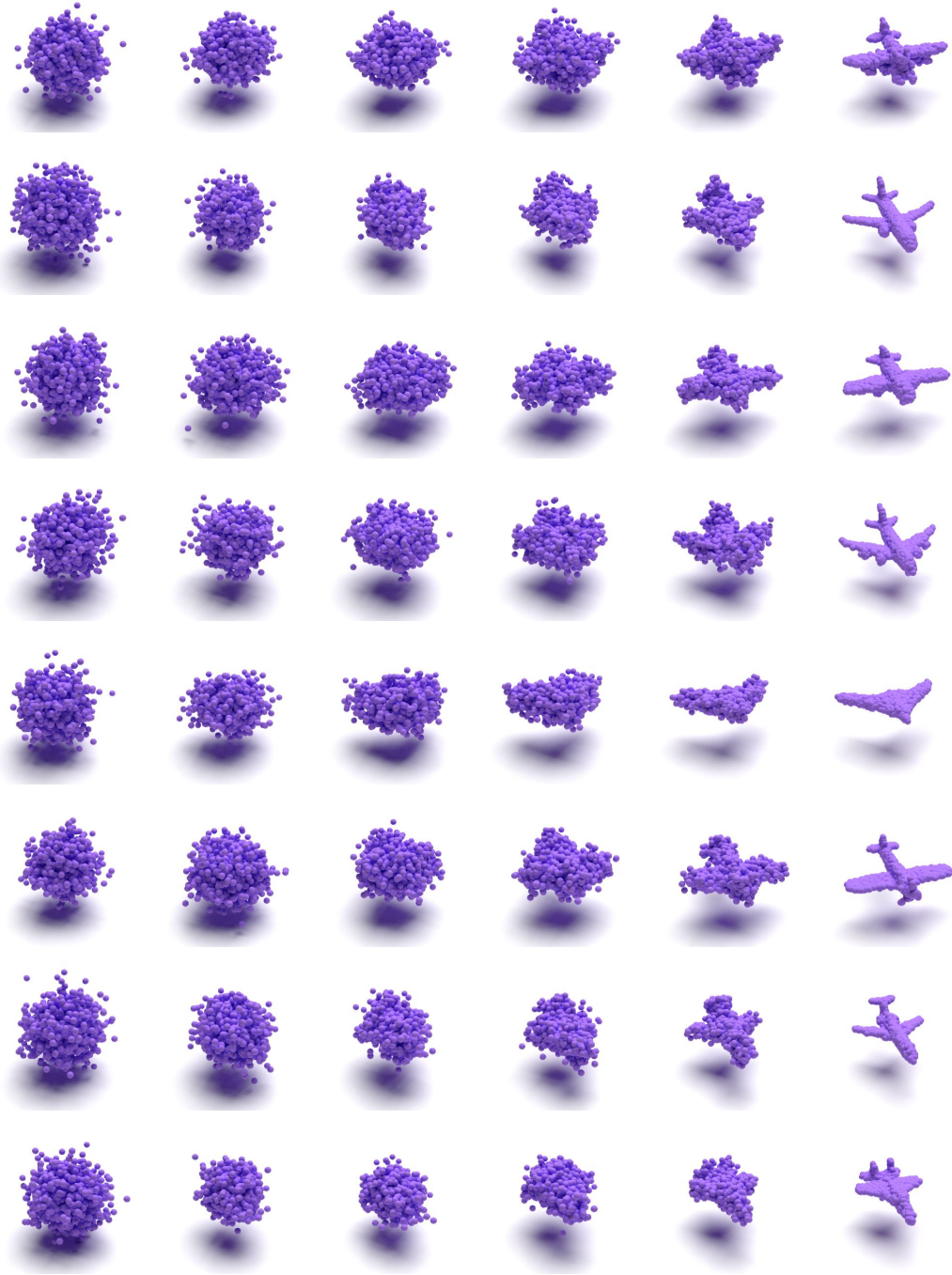
Figure 7: Qualitative visualizations of the diffusion process on *Airplane* shape generation. The results of generating from random noise to final 3D shapes are shown in left-to-right order.

Figure 8: Qualitative visualizations of the diffusion process on *Car* shape generation. The results of generating from random noise to final 3D shapes are shown in left-to-right order.
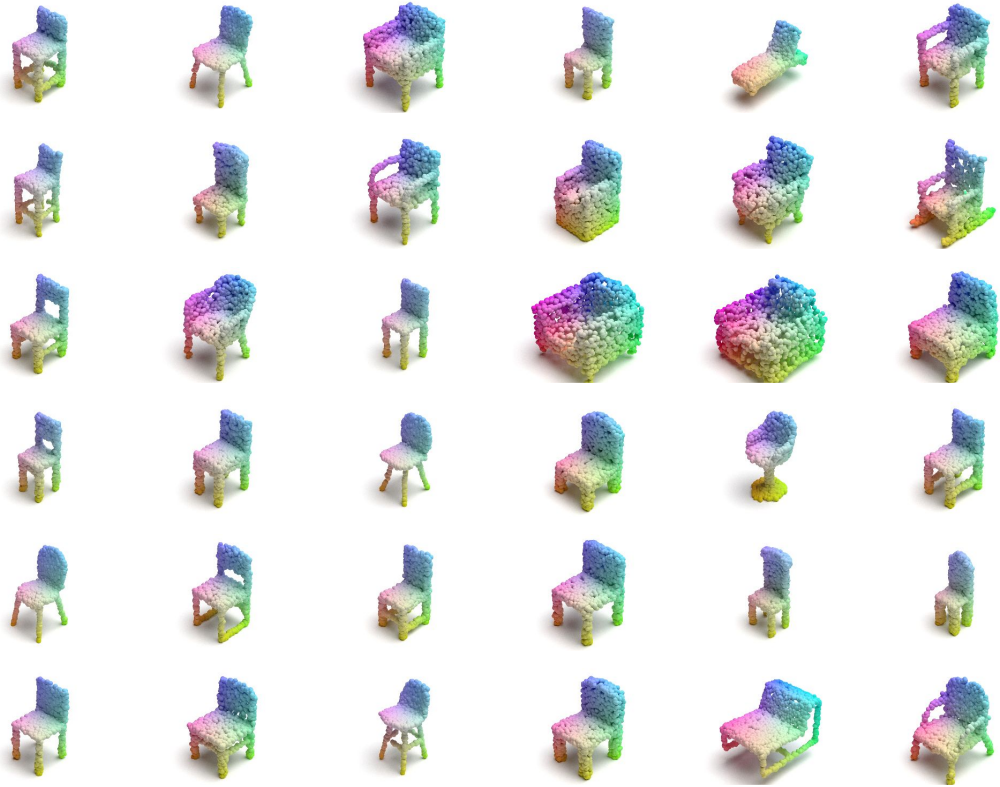
Figure 9: Qualitative visualizations of high-fidelity and diverse results on *Chair* shape generation.
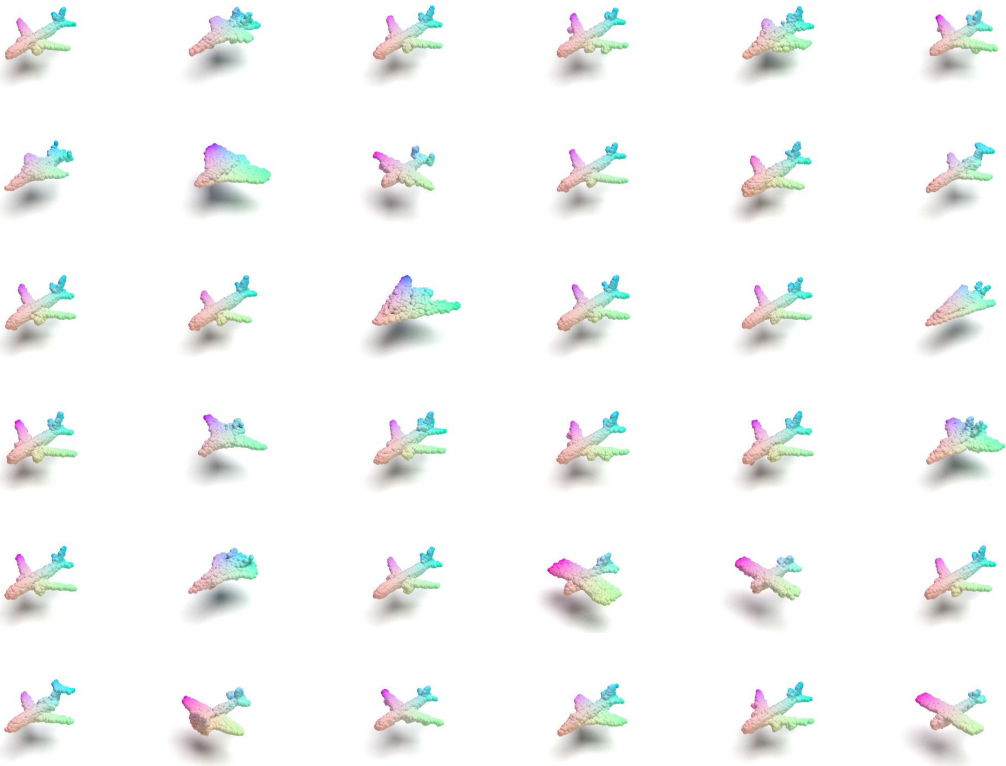

Figure 10: Qualitative visualizations of high-fidelity and diverse results on *Airplane* shape generation.

Figure 11: Qualitative visualizations of high-fidelity and diverse results on *Car* shape generation.