TFDM: Time-Variant Frequency-Based Point Cloud Diffusion with State Space Model

Jiaxu Liu[1] Li Li[2] Hubert P. H. Shum[1] Toby P. Breckon[1]
[1]Durham University [2]Huawei Noah's Ark Lab
jiaxu@liu-research.org i@luisli.org hubert.shum@durham.ac.uk toby.breckon@durham.ac.uk

Abstract

Diffusion models have achieved remarkable success across generative tasks. Recent advances in 2D diffusion have shown the strong potential of the Mamba state space model, owing to its efficiency in modeling long-range dependencies and sequential data. However, integrating state space models into 3D point cloud generation remains largely unexplored. To address this gap, we propose a diffusion framework for point cloud generation that incorporates a dual latent Mamba block (DM-Block) and a time-variant frequency encoder (TF-Encoder). In the DM-Block, we introduce a 3D Spiral Serialization (3DSS) strategy that organizes points according to point-cloud frequency characteristics for Mamba-based modeling, while operating in latent space to reduce the computational cost of direct 3D processing. Meanwhile, the TF-Encoder exploits the diffusion model's coarse-to-fine denoising behavior by prioritizing key points within the U-Net architecture, improving the recovery of fine geometric details in the final stages. Experiments on ShapeNet-v2 and ModelNet40 show that our method achieves state-of-the-art performance (0.14% on 1-NNA-Abs50 EMD and 58.10% on COV EMD) across several categories and metrics, while reducing the number of trainable parameters and inference time by up to 10× and 9×, respectively. The source code is available at https://github.com/JudgeLJX/TFDM.

1. Introduction

Point clouds are widely used in 3D tasks for their fidelity, ease of acquisition, and simple structure. Their generation is increasingly vital in applications such as VR, robotics [22, 23], mesh modeling, and scene reconstruction [9, 15, 33, 51]. However, unlike 2D images [16, 40], point clouds are discrete and unordered, making generative modeling especially challenging.

Early generative methods, including variational autoencoders (VAE) [25, 27], generative adversarial networks (GAN) [1, 5], and normalizing flows [20, 49], often suffer from limited stability or fidelity. Denoising diffusion models [31] (DDMs) have recently achieved stronger results by progressively corrupting point clouds with Gaussian noise and reversing the process to generate data. However, their iterative nature incurs substantial computational overhead and limits scalability.

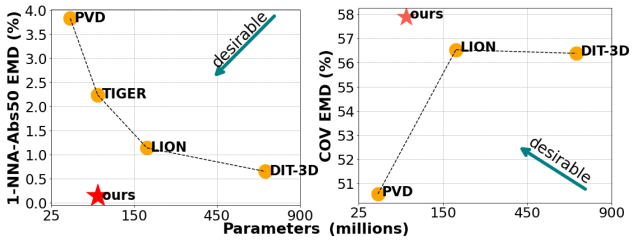

Figure 1. 1-NNA-Abs50 EMD (left) & COV EMD (right) performance (%) *vs.* parameter size (mil) on ShapeNet-v2 Car category (Sec. 4). Lower 1-NNA-Abs50 EMD indicates better generation quality and fidelity. Higher COV EMD means better diversity.

Recent advances in sequence modeling, particularly the Mamba architecture [13, 29], have demonstrated strong potential for efficient long-range dependency modeling. While Transformers have achieved impressive results in 3D perception and diffusion tasks [32, 45, 55], their quadratic complexity limits scalability for large point sets. In contrast, Mamba-based architectures [28, 54] model global dependencies with linear complexity, making them more suitable for large-scale 3D data. By serializing point clouds into sequences via space-filling curves (e.g., Z-order or our proposed 3DSS), Mamba can efficiently capture spatially coherent structures.

We further explore the compatibility between diffusion modeling and state-space sequence models. The iterative denoising process naturally forms a temporal dependency structure, which can be effectively modeled by Mamba. Building on this, we integrate Mamba into the diffusion framework and design a spiral space-filling serialization scheme that preserves local geometric coherence. This enables the model to capture global structure and fine-grained details throughout the denoising process, improving both efficiency and generation quality.

In addition to temporal consistency, diffusion models also need to balance local (high-frequency) and global (low-frequency) structures during denoising. Recent work has explored frequency analysis in both 2D and 3D domains [19, 34, 41, 48, 50], including integration into diffusion models [34, 50] and Mamba-based 2D architectures [17]. Several studies [12] have further combined frequency analysis and Mamba in 2D diffusion. However, extending such frequency modeling to 3D point clouds remains challenging:

unlike 2D data, where Fourier or spectral methods apply naturally, point clouds are irregularly sampled and inherently discrete, making frequency decomposition within state space models considerably more difficult.

To address these gaps, we propose TFDM, a point cloud diffusion architecture based on the Mamba framework. To improve the efficiency of long-sequence modeling, we introduce dual latent Mamba blocks (DM-Block) in latent space, resulting in a compact yet effective alternative to conventional Mamba-based architectures. Moreover, motivated by observations in 2D diffusion that coarse structures (low-frequency components) are recovered early while fine details (high-frequency components) emerge later, we extend this principle to point clouds. The generation process first establishes a coarse global shape and then progressively refines local contours, where high-frequency components correspond to edges and corners, and low-frequency components correspond to flatter regions. Based on this observation, we emphasize high-frequency regions during later diffusion stages. Specifically, within a U-Net architecture with multiple downsampling layers, we design a time-variant frequency-based encoder (TF-Encoder) that replaces conventional farthest point sampling with our frequency-based strategy to better select keypoints at later timesteps, thereby improving fine-detail recovery. In parallel, we propose a spiral curve serialization scheme tailored to this frequency-aware framework. This ordering arranges points progressively from the center toward the boundary, reflecting a common spatial-frequency tendency in structured point clouds, where coarse, low-frequency geometry typically lies in the interior, while higher-frequency details tend to appear near object boundaries. As a result, the sequence is organized according to frequency-related characteristics rather than arbitrary spatial order. This design establishes an intrinsic connection between the DM-Block and TF-Encoder, enabling coherent frequency-aware modeling throughout the diffusion process.

Our contributions are summarized as follows. The source code is in the supplementary material and will be released.

- We present the first integration of frequency-domain analysis into diffusion models within the Mamba framework, enabling efficient point cloud diffusion modeling. In addition, we design a novel 3D Spiral Serialization (3DSS) strategy that bridges the Mamba structure and the frequency-aware latent representation.
- We propose an end-to-end architecture (**TFDM**) that integrates a **T**ime-variant **F**requency-based Encoder (TF-Encoder) with a **D**ual latent **M**amba Block (DM-Block) to enhance high-frequency point-cloud details. It adapts to the diffusion timestep within the Mamba latent space of a point cloud DDPM, enabling precise detail refinement.
- We conduct extensive experiments on the ShapeNet-v2 [6] and ModelNet40 [47] benchmark datasets. Our method achieves state-of-the-art performance (ShapeNet-v2: 0.14% on 1-NNA-Abs50 EMD and 58.10% on COV EMD) across multiple categories while reducing the number of parameters by up to 10× and inference time by up to 9×.

2. Related Work

Diffusion Models: Denoising diffusion probabilistic models [16] generate data by reversing a progressive noising process in a Markov chain. They overcome common drawbacks of earlier generative paradigms in other generative frameworks [5, 25, 58]. Extensions to 3D point clouds [20, 38] have demonstrated strong generative potential, yet remain computationally demanding due to iterative denoising, especially with large datasets. To mitigate this issue, our approach performs diffusion in a latent space, substantially reducing cost while preserving quality.

Point Cloud Generation: Point cloud generation has been studied using autoregressive models [7, 43, 44], flow-based approaches [20, 26, 52], and GANs [5]. Recently, diffusion models [32, 38, 53] have emerged, incorporating strategies such as point–voxel fusion [57], hierarchical latent spaces [53], frequency-aware loss [56], and adapted Transformers [18, 32, 38]. While these models improve fidelity, they typically incur computational overhead. Our work offers a more efficient alternative that emphasizes fine-grained geometry while preserving global structural consistency.

State Space Models: State space models [13, 14, 29] employ linear recurrences to capture long-range dependencies with fewer parameters. Mamba [13] extends this formulation by enabling linear-time inference and has demonstrated strong performance in point cloud classification, segmentation [28, 54], and 2D diffusion tasks [17, 42]. In addition, several studies have explored alternative space-filling curves [3, 21] to better preserve spatial coherence. Building on these insights, we identify Mamba's untapped potential for 3D point cloud diffusion and incorporate it as the core sequential modeling component in our framework. Furthermore, we introduce 3DSS for our frequency-based architecture, enabling frequency-consistent ordering of points.

Frequency Analysis: Frequency decomposition is effective in distinguishing high-frequency (edge-like) and low-frequency (smooth) structures in both point clouds [19, 48] and images [41, 50]. Recent diffusion models [34, 50] leverage wavelet or spectral transforms to enhance detail and reduce redundancy. While Zhou et al. [56] proposed a frequency-based module for 3D point cloud diffusion, it was not combined with state space modeling. In contrast, we integrate frequency-aware sampling into a Mamba-based diffusion framework, improving detail fidelity with efficiency.

3. Methodology

Our framework is motivated by the observation that point cloud diffusion extends beyond spatial structure to an inherently sequential process: as denoising progresses, the model gradually reconstructs fine-grained geometry from

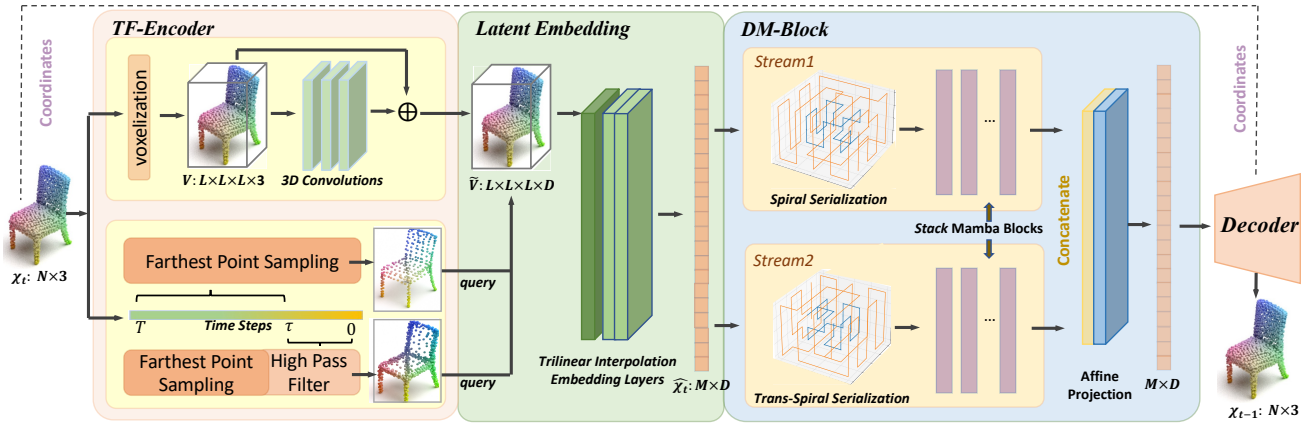

Figure 2. **Overview of TFDM**. Given a point cloud at timestep t, the network predicts the noise in $\mathcal{X}_t$ to recover $\mathcal{X}_{t-1}$. The input is first processed by a time-variant frequency-based encoder, followed by a latent embedding module that produces the latent point cloud $\hat{\mathcal{X}}_t$. This is fed into dual-stream Mamba blocks with proposed 3DSS methods to extract complementary features. An affine transformation block aligns the outputs from both streams, and the final latent representation is decoded back to 3D space.

Figure 3. Qualitative results of joint training on 10 categories, presented in the following order: *bag, keyboard, mug, pillow, rocket, earphone, basket, bed, bowl, and cap*.

noise. This aligns with the recurrent nature of state space models, which efficiently capture long-range dependencies. By serializing point clouds in the latent space, we reinterpret them as continuous sequences on manifolds, enabling effective Mamba-based modeling. Based on this, we design a diffusion architecture that integrates 3D spiral serialization, dual-stream Mamba blocks, and a time-varying frequency encoder. We first formulate the generative diffusion objective. Building on Mamba's capability for 3D representation learning, we propose a diffusion framework for point clouds (see Fig. 2). Specifically, we introduce a frequency-based filter (Sec. 3.1) to extract key components in a time-variant frequency encoder (Sec. 3.2) for keypoint sampling across diffusion timesteps. Our dual-stream latent Mamba architecture (Sec. 3.3), together with the proposed 3DSS, combines state space modeling, frequency analysis, and diffusion for high-fidelity point cloud generation.

Problem Definition: Given a point cloud $\mathcal{X} \in \mathbb{R}^{N\times 3}$ consisting of N points, the goal is to generate a high-fidelity point cloud from Gaussian noise $p(\mathbf{x}_T)$ by learning the transition probability $p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t)$. Specifically, we model the mean of the transition distribution while keeping a predetermined variance throughout the reverse diffusion process. Similar to TIGER [38] and PVD [57], we adopt a U-Net backbone [39] for $\mu_\theta(\mathbf{x}_t, t)$ and incorporate a newly designed Mamba layer and frequency-based keypoint selection to enhance its capability. To sample the point cloud, we denoise from $p(\mathbf{x}_T)$ over T timesteps by minimizing the MSE loss $\mathbb{E}_{t\sim[1,T]} \parallel \epsilon_\theta(\mathbf{x}_t, t) - \epsilon_0 \parallel_2^2$ between the predicted noise $\epsilon_\theta(\mathbf{x}_t, t)$ and the ground-truth noise ϵ_0, ensuring accurate denoising across timesteps.

3.1. Point Cloud Graph Filter

Unlike the 2D domain, where spectral analysis methods such as Fourier and wavelet transforms are directly applicable [19, 34, 48], the irregular, non-Euclidean nature of point clouds [35, 36] demands new approaches for defining frequency components. The absence of a structured grid in point cloud data [2, 55] complicates the direct adoption of traditional spectral techniques, thus motivating a tailored method to effectively capture and process the inherent frequency characteristics of point cloud geometry.

Graph Construction: To capture geometric structure and local topology in point clouds, we build a k-nearest neighbors (k-NN) graph and extract high-frequency points from its signals without trainable parameters.

Given a point cloud $\mathcal{X} = \{x_i, \ldots, x_N\}$ with corresponding d-dimensional features $\mathbf{f}_i \in \mathbb{R}^d$, $i \in \{1, \ldots, N\}$, we construct a k-NN graph $\mathcal{G} = (\mathcal{V}, \tilde{\mathcal{A}}_u, \tilde{\mathcal{A}}_w)$. Each point x_i corresponds to a node $v_i \in \mathcal{V}$, and $\tilde{\mathcal{A}}_u, \tilde{\mathcal{A}}_w \in \mathbb{R}^{N\times N}$ are the normalized unweighted and weighted adjacency matrices encoding point dependencies in feature space. The unweighted $\tilde{\mathcal{A}}^u_{ij}$ and weighted edges $\tilde{\mathcal{A}}^w_{ij}$ connecting nodes v_i and v_j are:

$$\begin{aligned} \tilde{\mathcal{A}}^u_{ij} &= \mathbb{1}(x_j \in \mathcal{N}(x_i)), \\ \tilde{\mathcal{A}}^w_{ij} &= \kappa(\|x_i - x_j\|^2) \cdot \tilde{\mathcal{A}}^u_{ij}, \end{aligned} \tag{1}$$

where $\kappa(\cdot)$ is a non-negative function, e.g., a Gaussian function, to ensure that $\tilde{\mathcal{A}}_w$ is a diagonally dominant matrix;

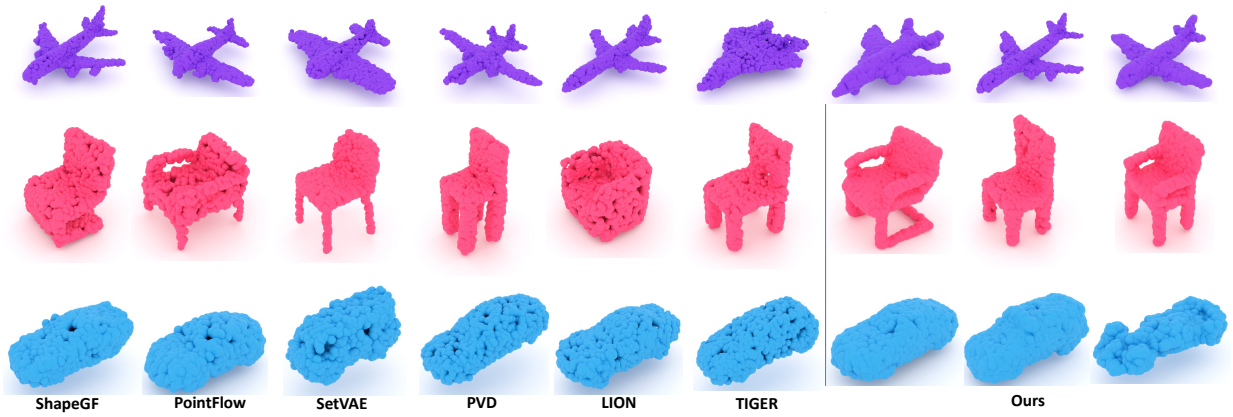

Figure 4. Qualitative comparisons for three illustrative object categories $\{airplanes, chairs, cars\}$: our approach (right) and other leading contemporary approaches (left/middle). TFDM generates high-quality and diverse point clouds.

$\mathcal{N}$ represents the neighborhood; $\mathbb{1}(\cdot)$ represents the indicator function which returns 1 if the specified condition (the function input) is true and 0 if it is false.

Point Cloud High-pass Filter: Our high-pass graph filter design is inspired by the 2D case, where high-frequency components (*e.g.*, edges) produce strong spatial responses. Following GDA [48], we adopt the Laplacian operator to construct the graph filter: $h(\tilde{\mathcal{A}}_w) = I - \tilde{\mathcal{A}}_w$. Given a graph signal $\mathbf{s}_d \in \mathbb{R}^N$ for each $d \in \{1, \ldots, D\}$, the filtered signal is $\mathbf{y}_d = h(\tilde{\mathcal{A}}_w) \cdot \mathbf{s}_d \in \mathbb{R}^N$. The corresponding frequency response of $h(\mathcal{A}_w)$ is:

$$\widehat{h}(\tilde{\mathcal{A}}_w) = \text{diag}(1 - \tilde{\lambda}_1, 1 - \tilde{\lambda}_2, \ldots, 1 - \tilde{\lambda}_N), \qquad (2)$$

where $\text{diag}(\cdot)$ denotes the diagonal matrix operator. The eigenvalues $\tilde{\lambda}_i$ are ordered in reverse to represent descending frequencies. Since $1 - \tilde{\lambda}_i < 1 - \tilde{\lambda}_{i+1}$, low-frequency components are suppressed, making this a high-pass filter.

We apply the filter $h(\tilde{\mathcal{A}}_w)$ to the point cloud $\mathcal{X}$ to obtain the filtered point $h(\tilde{\mathcal{A}}_w)\mathcal{X}$, with each point computed as:

$$(h(\tilde{\mathcal{A}}_w)\mathcal{X})_i = x_i - \sum_j^N (\tilde{\mathcal{A}}_w)_{i,j} x_j. \qquad (3)$$

This operation preserves local variation, because the filtered point in Eq. (3) measures the difference between a point feature and a linear combination of its neighboring features.

Based on the filtered signal, we compute its l_2 norm in Eq. (3) and select the top-M points with the largest responses to capture dominant high-frequency components. This selection introduces frequency decomposition into the point cloud domain despite its inherent irregularity, providing a compact yet informative set of frequency-aware keypoints.

3.2. Time-Variant Frequency Point Cloud Encoder

As shown in Fig. 5, we propose TF-Encoder, a timestep-dependent encoder designed to match the coarse-to-fine reconstruction dynamics of diffusion. Unlike static sampling strategies, TF-Encoder adaptively adjusts the emphasis on geometric frequency components across diffusion timesteps. Early diffusion steps mainly reconstruct global geometric structures, which correspond to low-frequency components, whereas later steps progressively refine high-frequency local details. To align with this progression, TF-Encoder gradually increases the proportion of points sampled from high-frequency regions during later diffusion stages.

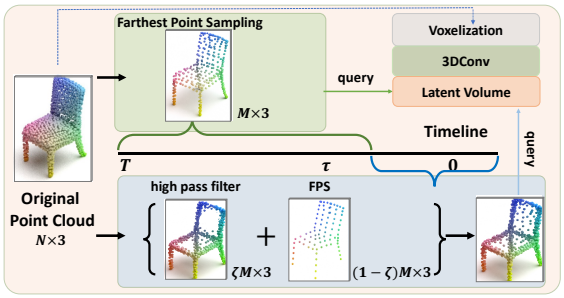

Figure 5. **Frequency-based keypoint selection** (within the encoder) applies different strategies across timesteps to downsample the point cloud. The resulting points are used to query the latent volume, producing the latent point cloud.

Voxel-Based Feature Extraction: Specifically, we use the PVCNN [30] backbone, which enables efficient computation by downsampling the point cloud into a voxel grid. For a point cloud at time step t, denoted by $\mathcal{X}_t \in \mathbb{R}^{N\times 3}$, our TF-Encoder $\mathcal{E}$ transforms it into a latent space $\hat{\mathcal{X}}_t \in \mathbb{R}^{M\times D}$, where $M < N$ denote the numbers of subsampled and original points, respectively. To aggregate voxelized features, the point cloud $\mathcal{X}_t$ with normalized coordinates $c = (x, y, z)$ is voxelized into voxel grids $\{\boldsymbol{V}_{m,p,q}\}$, where $\boldsymbol{V} \in \mathbb{R}^{L\times L\times L}$ has resolution L. The latent feature of each grid is computed as the mean of the point features within it:

$$\boldsymbol{V}_{m,p,q} = \frac{1}{K_{m,p,q}} \sum_{i=1}^{n} \boldsymbol{I}[\text{floor}(x_i \times r) = m,$$
$$\text{floor}(y_i \times r) = p,\ \text{floor}(z_i \times r) = q] \times f_i, \qquad (4)$$

where r denotes the voxel resolution and $\boldsymbol{I}$ is an indicator function that determines whether coordinate c_i belongs to the voxel grid $\{m, p, q\}$. $K_{m,p,q}$ represents the number of points falling within the grid $\{m, p, q\}$, and floor$(\cdot)$ is the floor function. After voxelization, multiple 3D convolutional layers with Swish activation [37] and GroupNorm [46] are applied to obtain the latent volume $\tilde{\boldsymbol{V}} \in \mathbb{R}^{L\times L\times L\times D}$ with D channels.

Time-Variant Frequency-Aware Sampling : Unlike standard farthest point sampling (FPS) pipelines used in PVCNN-style backbones, we combine a high-pass graph filter (Sec. 3.1) with FPS in a time-variant manner. This design ensures that during early timesteps $\tau \leq t \leq T$, we maintain a balanced selection of low-frequency structures and a subset of high-frequency regions to preserve global shape coherence. As the process advances to later timesteps $0 < t < \tau$, our method adaptively biases sampling toward high-frequency points, enabling a more precise representation of subtle edges, corners, and complex geometric details.

Formally, at each time step, for M target points, we select ζM points using the graph-based high-pass filter, while the remaining $(1-\zeta)M$ points are sampled via FPS. As diffusion progresses, ζ is switched to a larger value after timestep τ, allowing the model to emphasize high-frequency geometric details during later diffusion stages. This adaptive strategy enables the TF-Encoder to align with the diffusion trajectory and provide timestep-specific frequency emphasis. The extracted point cloud is given by:

$$\mathcal{X}_t^* = \begin{cases} \zeta h(\tilde{\mathcal{A}}_w)\mathcal{X}_t + (1-\zeta)F(\mathcal{X}_t^{N-M}), & t < \tau, \\ F(\mathcal{X}_t), & \tau \leq t \leq T, \end{cases} \tag{5}$$

where $F(\cdot)$ denotes farthest point sampling, $\mathcal{X}_t^{N-M}$ is the original point cloud excluding the points selected by the high-pass filter.

Subsequently, we employ trilinear interpolation by querying the latent volume $\tilde{\boldsymbol{V}}$ with the sampled point cloud $\mathcal{X}_t^*$ to obtain the latent features $\hat{\mathcal{X}}_t$. The coordinates of both $\mathcal{X}_t^*$ and $\mathcal{X}_t$ are preserved to facilitate upsampling and positional embedding. Having obtained frequency-aware keypoints across timesteps, the next challenge is to model their temporal and spatial dependencies efficiently. To this end, we introduce the Dual Latent Mamba Block (DM-Block), which integrates state-space modeling with our frequency-aware latent representation.

3.3. Dual Latent Mamba Blocks

Although time-variant frequency emphasis helps refine point selection, directly applying a state space model to raw points at each timestep is computationally expensive because of the high dimensionality and unordered nature of point clouds [2, 35, 36]. To address this, we propose Dual Latent Mamba Blocks (DM-Block), which operate in a latent space and serialize the downsampled point set into a 1D sequence suitable for Mamba modeling. This design preserves local neighborhood relationships through diverse space-filling curves while exploiting Mamba's ability to model long-range dependencies efficiently. Building on this motivation, we further introduce a spiral-based serialization strategy to enhance spatial-frequency coherence in the Mamba modeling process.

3D Spiral Space-Filling Curve Serialization For Frequency: To enhance the sequential modeling capability of the DM-Block, we reorder latent points using a space-filling curve. Unlike conventional Hilbert and Z-order curves, we propose 3DSS, a spiral curve that traverses points from the center outward (Fig. 2). Formally, we define $\pi(k) : \{0, \ldots, L^3-1\} \rightarrow \{0, \ldots, L-1\}^3$ as

$$\pi(k) = \Big(\mathrm{Spiral}_2\big(u_k;\ r_k - |z_k - c_z|\big),\ z_k \Big), \tag{6}$$

where r_k denotes the current radius layer, c_z is the z-coordinate of the grid center, z_k indexes the current slice, u_k is the traversal index of the 2D spiral on the (x, y) plane at slice z_k, and $\mathrm{Spiral}_2(u_k; r)$ maps the traversal index u_k to a 2D location in the spiral of radius r on the corresponding (x, y) slice. Intuitively, the proposed 3D spiral serialization converts 3D coordinates into a 1D sequence by expanding from the center in cubic shells and enumerating points within each shell using a slice-wise 2D spiral ordering. This construction yields a smooth center-to-boundary traversal that preserves spatial locality between consecutive elements, which is beneficial for sequence-based models.

This ordering also aligns with a spatial-frequency tendency in point clouds: smooth planar regions usually correspond to low-frequency components, whereas edges, corners, and fine geometric structures correspond to higher frequencies [8]. Consistent with this observation, we find that in many ShapeNet-v2 and ModelNet40 objects [6, 47], interior regions tend to capture coarse, low-frequency geometry, while peripheral or transitional regions (e.g., boundaries and edges) contain richer high-frequency details (Fig. 6). The proposed 3DSS therefore places points with similar frequency characteristics close to each other in the sequence, promoting both spatial and frequency coherence.

We further evaluate multiple ordering schemes, including Hilbert, Z-order, and our 3DSS, along with their transposed variants (Trans-Hilbert, Trans-Z, and Trans-Spiral). This reordering strategy preserves spatial locality within the serialized representation, enabling the DM-Block to better capture local correlations as neighboring points remain close in sequence. Although the proposed 3D spiral serialization may not be optimal for highly irregular or hollow structures, it is well suited for the object categories considered in this work, where the geometric distribution roughly follows a center-to-boundary organization.

Formally, a space-filling curve is defined as a bijective mapping $\phi : \mathbb{Z} \rightarrow \mathbb{Z}^3$ that traverses every point in a discrete 3D space while maintaining spatial continuity. Given a space-filling curve $\mathcal{C}$, the latent point cloud $\hat{\mathcal{X}}_t$ is reordered according

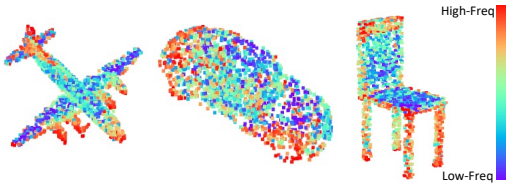

Figure 6. Illustration of the frequency distribution in different categories. Red and blue denote high- and low-frequency regions.

Table 1. Comparison results (%) jointly trained on 10 categories, demonstrating strong cross-category generalization.

Method	CD ↓ (1-NNA-Abs50)	EMD ↓ (1-NNA-Abs50)	CD ↑ (COV)	EMD ↑ (COV)
DPM	9.71	21.54	43.65	38.94
PVD	7.52	17.43	44.12	44.32
Tiger	0.88	0.98	56.25	57.64
Ours	**0.85**	**0.43**	**56.41**	**60.68**

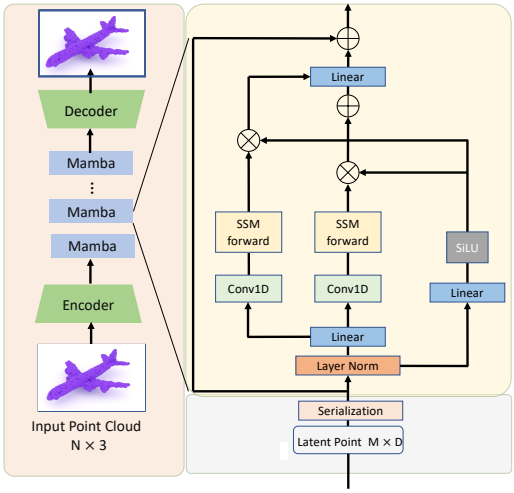

Figure 7. Illustration of our proposed Latent mamba block, which includes Layer Norm, Linear Layer, forward and backward state space model with its corresponding Conv1D block (N.B. we only perform serialization before the first block).

to $\mathcal{C}$, producing the serialized latent representation:

$$\hat{\mathcal{X}}_t^c = \mathcal{C}(\mathcal{X}_t^*)\hat{\mathcal{X}}_t, \quad \text{where } \hat{\mathcal{X}}_t^c \in \mathbb{R}^{M\times D}. \tag{7}$$

Bidirectional Latent Mamba: To capture forward and backward dependencies efficiently along the serialized latent sequence $\hat{\mathcal{X}}_t^c$, we employ a bidirectional variant of Mamba. Each Mamba block (Fig. 7) consists of layer normalization [4], causal one-dimensional convolution, SiLU activation [11], and residual connections. The serialized latent point cloud sequence $\hat{\mathcal{X}}_t^c$ is processed through stack Mamba blocks. Given input $\mathcal{Z}_{l-1}$, each block performs the transformation:

$$\begin{aligned}
\mathcal{Z}_{l-1}^l &= \text{LN}(\mathcal{Z}_{l-1}), \quad \mathcal{Z}^{'} = s(\text{Linear}(\mathcal{Z}_{l-1}^l)),\\
\mathcal{Z}_l^d &= \text{SSM}_d(\text{Conv1D}(\text{Linear}(\mathcal{Z}_{l-1}^l))), \quad d \in \{f, b\},\\
\mathcal{Z}_l &= \text{Linear}(\mathcal{Z}^{'} \odot (\mathcal{Z}_l^f + \mathcal{Z}_l^b)) + \mathcal{Z}_{l-1},
\end{aligned} \tag{8}$$

where s denotes the SiLU activation function, $\mathcal{Z}_l^f$ and $\mathcal{Z}_l^b$ denote the forward and backward SSM branches, respectively, and $\mathcal{Z}_l$ is the output of the l-th block. The Mamba output $\hat{\mathcal{X}}_{\text{out}} \in \mathbb{R}^{M\times D}$ is obtained after passing through a stack of such blocks.

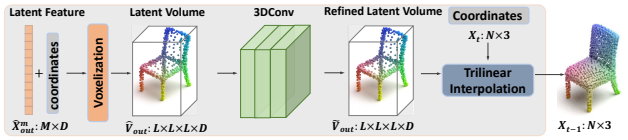

Figure 8. Decoder overview. The final prediction X_{t-1} is obtained by querying the latent volume V_{out} with the coordinates.

Two Streams Affine Fusion: To further enhance representational power, we run two parallel streams with distinct space-filling orders (e.g., Z vs. Transposed-Z), each capturing complementary structural cues. We fuse their outputs using a learnable affine transformation followed by a projection network, producing an aggregated representation that retains global shape coherence and local detail sensitivity. Specifically, for the features from the two streams, $\hat{\mathcal{X}}_{\text{out}}^{c1}$ and $\hat{\mathcal{X}}_{\text{out}}^{c2}$, we perform the affine fusion:

$$\hat{\mathcal{X}}_{\text{out}}^m = \text{Proj}((\hat{\mathcal{X}}_{\text{out}}^{c1} \odot \gamma^{c1} + \delta^{c1}) \oplus (\hat{\mathcal{X}}_{\text{out}}^{c2} \odot \gamma^{c2} + \delta^{c2})), \tag{9}$$

where $\gamma^{c1}, \gamma^{c2} \in \mathbb{R}^D$ and $\delta^{c1}, \delta^{c2} \in \mathbb{R}^D$ are scale and shift factors, respectively. The operator $\odot$ denotes element-wise multiplication, and $\oplus$ denotes concatenation. $\text{Proj}(\cdot)$ denotes a projection network that maps the concatenated features from $\mathbb{R}^{M\times 2D}$ to $\mathbb{R}^{M\times D}$. The final output feature $\hat{\mathcal{X}}_{\text{out}}^m \in \mathbb{R}^{M\times D}$ therefore aggregates both global and local information.

Point Cloud Decoder: Finally, a point cloud decoder upsamples the latent point cloud to predict the noise ϵ_θ. As shown in Fig. 8, we employ trilinear interpolation to map the latent point cloud $\hat{\mathcal{X}}_{out}^m \in \mathbb{R}^{M\times D}$, together with its coordinates, back to 3D space $\mathcal{X}_t \in \mathbb{R}^{N\times 3}$. Similar to Sec. 3.2, we voxelize $\hat{\mathcal{X}}_{out}^m$ into a volume $\tilde{\boldsymbol{V}}_{out} \in \mathbb{R}^{L\times L\times L\times D}$ and process it with a 3D convolutional network that preserves overall geometric fidelity. We then query using $\mathcal{X}_t$ to obtain the final noise prediction ϵ_θ. By adopting the TF-Encoder and DM-Block, we overcome the computational bottleneck of raw 3D data processing while retaining high-frequency details at the appropriate diffusion stage. This design integrates time-variant frequency emphasis with state space modeling in a simple yet effective manner.

4. Experiments

ShapeNet-v2 Benchmark Dataset: For a fair comparison on ShapeNet-v2 [6], we follow common practice and focus on three key categories: *chair*, *car*, and *airplane*. From each shape, we sample 2048 points from the 5000 points available in the training and test sets, with normalization applied across the entire dataset. We follow the preprocessing steps and data split strategy of PointFlow [49].

ModelNet40 Dataset. ModelNet40 [47] is a widely used 3D benchmark originally designed for object recognition, consisting of 12,308 CAD models from 40 categories. The

Table 2. Comparison results (%) on ShapeNet-v2 with shape metrics: Absolute 50-Shifted 1-Nearest Neighbor Accuracy (1-NNA-Abs50) and Coverage (COV), Chamfer Distance (CD) and Earth Mover's Distance (EMD), where CD is multiplied by 10^3 and EMD is multiplied by 10^2; – denotes unavailable result from original authors; **Best**/2nd best highlighted.

Method	Chair				Airplane				Car			
	1-NNA-Abs50 (↓)		COV (↑)		1-NNA-Abs50 (↓)		COV (↑)		1-NNA-Abs50 (↓)		COV (↑)	
	CD	EMD	CD	EMD	CD	EMD	CD	EMD	CD	EMD	CD	EMD
PointFlow [49]	12.84	10.40	46.84	47.35	25.68	20.74	47.04	40.52	8.10	6.52	35.40	44.60
SoftFlow [24]	9.21	10.55	41.39	47.43	26.05	15.80	46.24	40.25	18.58	15.98	36.34	45.25
SetVAE [25]	8.84	10.57	46.83	44.36	24.80	15.65	48.10	40.35	13.04	15.53	40.99	46.59
DPM [31]	10.05	24.77	44.86	35.50	26.42	36.91	48.64	33.83	18.89	29.97	44.03	34.94
PVD [57]	7.89	23.68	40.66	42.71	16.44	26.26	47.34	42.15	4.55	3.83	41.19	50.56
LION [53]	3.70	2.34	48.94	52.11	17.41	11.23	47.16	49.63	3.41	1.14	50.00	56.53
DiT-3D [32]	**0.89**	**0.73**	**52.45**	54.32	**12.35**	8.67	**53.16**	**54.39**	**1.76**	0.65	50.00	56.38
TIGER [38]	4.61	2.71	-	-	21.85	**5.82**	-	-	4.31	2.24	-	-
FrePolad [56]	3.53	3.23	50.28	50.93	15.25	12.10	45.16	47.80	1.89	0.26	50.14	55.23
NSO [20]	5.51	7.63	-	-	18.63	11.85	-	-	9.66	3.55	-	-
ConTiCom-3D [10]	6.11	3.09	-	-	24.22	18.54	-	-	5.43	4.22	-	-
TFDM (ours)	3.25	1.68	49.84	**54.98**	18.31	8.88	51.38	52.25	3.89	**0.14**	**50.56**	**58.10**

Table 3. Comparison results (%) on ModelNet40.

Method	Class	CD ↓ (1-NNA-Abs50)	EMD ↓ (1-NNA-Abs50)	CD ↑ (COV)	EMD ↑ (COV)
PVD	stairs	22.43	26.54	46.65	47.80
TIGER		18.88	10.25	49.50	50.55
LION		20.15	**8.10**	47.42	48.92
Ours		**17.50**	8.95	**51.25**	**52.34**
PVD	person	8.64	15.49	43.54	45.85
TIGER		6.64	3.45	51.05	53.52
LION		5.58	2.89	49.85	52.21
Ours		**5.50**	**2.56**	**54.23**	**55.16**

dataset contains manually cleaned CAD objects without texture or color information. Following common practice, we uniformly sample 2,048 points from each mesh surface and normalize each shape into a unit sphere.

Evaluation Metrics: Following common practice in prior work [31, 57], we use 1-NNA (and its variant 1-NNA-Abs50) and COV to evaluate generation quality and diversity, together with CD and EMD, which quantify discrepancies at the point and distribution levels. Because 1-NNA can be ambiguous to interpret, we propose 1-NNA-Abs50 (Absolute 50-Shifted 1-NNA) as a clearer alternative. It transforms 1-NNA x into $|x - 50|$, making it more sensitive to deviations from the ideal 50%; a lower score indicates a distribution more consistent with real data. See the supplementary material for details.

Implementation: We set $k = 32$ and $\zeta = 0.875$ for the k-NN and high-pass filtering in the frequency-based encoder. Diffusion is performed over 1,000 timesteps with $\tau = 50$. Each Mamba stream uses 8 blocks with a latent dimension of 512 and 256 points. Training is conducted for 10,000 epochs using Adam (lr=2e-4, weight decay=1e-4) on an NVIDIA A100 GPU.

4.1. Comparison with State-of-the-Art

Generalization Across Categories: Previous methods [20, 32, 57] often train category-specific models, which limits generalization across diverse object classes. To evaluate this, we conduct joint training without category conditioning on ten categories from ShapeNet-v2 (*cap, keyboard, earphone, pillow, bag, rocket, basket, bed, mug, bowl*) (Fig. 3). Training on such diverse shapes highlights the model's generalization capability and creates a challenging multimodal setting. As shown in Tab. 1, our method consistently outperforms all baseline models under identical settings. We further evaluate the efficiency and scalability of our proposed architecture.

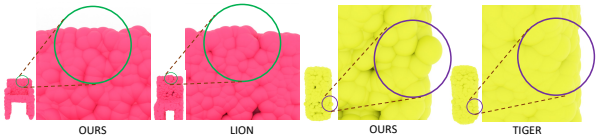

Figure 9. Chair (pink) - smooth (ours), deformed (other), Car side-view mirror (yellow) - retained (ours), missing (other).

Performance: We next perform per-class training on three common categories (*chair, airplane, car*). In Tab. 2, we compare TFDM with multiple point cloud generation approaches. Notably, TIGER (CVPR24), FrePolad (ECCV24), NSO (ICLR25), and ConTiCom-3D (2025) are all recent methods. Among these, TIGER, FrePolad, ConTiCom-3D, and NSO are relatively lightweight, yet we outperform them on the *chair* and *car* categories. For example, TFDM achieves a 0.25% improvement in 1-NNA-Abs50 CD and a 0.98% improvement in 1-NNA-Abs50 EMD on *chairs* over the strongest competing method, as well as a 0.12% gain on *cars*. While DiT-3D attains strong results, it requires substantial computation (1700 GPU hours and 711M parameters). In contrast, TFDM exceeds DiT-3D on three of four metrics for the *car* class, including a 0.51% gain in 1-NNA-Abs50 EMD and a 1.72% gain in COV EMD, confirming both the efficiency and effectiveness of our framework. To evaluate cross-dataset generalization, we further test our method on two ModelNet40 [47] categories, namely *person* and *stairs*. As shown in Tab. 3, our method achieves competitive performance on both categories. Notably, the *stairs* category exhibits a distinct geometric structure compared with the object categories in ShapeNet, which further verifies the generalization capability. The qualitative comparisons in Fig. 4 and Fig. 9 show our method produces smoother surfaces and preserves fine-grained structures.

Efficiency: As summarized in Tab. 4, our full model

Table 4. Comparison on training and inference time, model size, and the corresponding evaluation results. Time is measured on the same device, and averaged over three categories: chair, airplane and car. 'SS' indicates our single-stream model.

Method	Para (M)	Training Time (h)	Inference Time (s)	EMD (1-NNA-Abs50) ↓
TIGER	70.11	164	9.73	2.24
DIT-3D	711.88	1688	100.13	0.65
LION	144.25	550	27.12	1.14
Ours (SS)	**48.84**	**138**	**8.12**	0.85
Ours (Full)	70.25	192	11.41	**0.14**

achieves the best overall performance while requiring only marginally more training time than TIGER. Compared with other top-performing methods, TFDM significantly reduces both parameter count and training hours. For further efficiency, the single-stream variant achieves the fastest inference speed and lowest computational cost, with only minor accuracy trade-offs relative to DiT-3D and our full model.

4.2. Ablation Studies

Different combinations of serialization methods: We evaluate various serialization combinations within the two-stream architecture to determine the most effective information-flow configuration. For the *car* category, pairing our proposed *3DSS* and *transposed-3DSS* serializations achieves the best performance among all variants. Comparing rows (i) and (h) in Tab. 5, this configuration yields improvements of 0.64% in 1-NNA-Abs50 CD, 0.21% in 1-NNA-Abs50 EMD, 0.31% in COV CD, and 1.45% in COV EMD over other combinations, indicating more effective cross-stream feature interaction.

Effectiveness of the frequency-based model: We next assess the contribution of the proposed time-variant frequency mechanism by comparing models with and without this component. As shown by rows (d) and (e) in Tab. 5, integrating frequency decomposition consistently improves performance across all metrics, achieving gains of 0.09% (1-NNA-Abs50 CD), 0.71% (1-NNA-Abs50 EMD), 0.13% (COV CD), and 1.08% (COV EMD). This demonstrates that frequency-aware modeling enhances detail recovery in later diffusion stages, making it well suited for fine-grained geometric refinement (see supplementary for visual evidence).

Effectiveness of the Mamba block: To examine the benefit of Mamba-based state-space modeling, we replace the latent Mamba block (row c) with standard 3D convolutional blocks (row a) and Transformer blocks (row b). As reported in Tab. 5, Mamba blocks significantly boost both generation quality and diversity, outperforming the convolutional design by 3.26% and 4.85% in 1-NNA-Abs50 CD and EMD, respectively. Compared with the Transformer-based variant, Mamba improves 0.33% and 0.55% in 1-NNA-Abs50 EMD and COV CD, while using roughly half the parameters, highlighting its efficiency advantage.

Effectiveness of the two-stream Mamba layer design: We further assess the impact of using two serialization streams versus a single stream. As shown in Tab. 5, the two-stream configuration (row g) consistently outperforms the single-stream baseline (row c) across all metrics, with gains of 1.34% and 1.09% in 1-NNA-Abs50 CD and EMD, and 1.01% and 1.52% in COV CD and EMD, respectively. This indicates that complementary traversal orders enable the model to capture richer geometric cues and improve spatial–frequency consistency.

Table 5. Component-wise ablation of TFDM on ShapeNet-v2 (car category): latent block, serialization strategy, frequency-based component, and sampling strategy. H-Trans represents Hilbert-Trans, and NNA represents the 1-NNA-Abs50 metric.

	Serialization Strategy	Freq Decom	Latent Block	CD ↓ (NNA)	EMD ↓ (NNA)	CD ↑ (COV)	EMD ↑ (COV)
(a)	None		Conv	9.24	5.95	45.26	50.43
(b)	None		Transformer	6.21	1.43	49.65	54.15
(c)	Hilbert		Mamba	5.98	1.10	49.10	54.21
(d)	Hilbert	✓	Mamba	4.76	0.85	49.99	56.10
(e)	3DSS	✓	Mamba	4.43	0.66	50.23	56.34
(f)	Z + Z-Trans		Mamba	4.53	0.54	50.01	55.52
(g)	Hilbert + H-Trans		Mamba	4.64	0.64	50.11	55.73
(h)	Hilbert + H-Trans	✓	Mamba	4.53	0.35	50.25	56.65
(i)	3DSS + 3DSS-Trans	✓	Mamba	**3.89**	**0.14**	**50.56**	**58.10**
(k)	Hilbert	Rand	Mamba	6.45	1.68	47.54	53.21

Table 6. Ablations on hyperparameters τ and ζ.

	τ	ζ	CD ↓ (1-NNA-Abs50)	EMD ↓ (1-NNA-Abs50)	CD ↑ (COV)	EMD ↑ (COV)
(a)	25	0.875	3.54	1.99	49.01	53.99
(b)	50	0.875	3.25	1.68	49.84	54.98
(c)	50	0.75	4.15	2.37	48.93	53.46

Effect of Sampling Strategy. To evaluate the effectiveness of the proposed time-variant frequency sampling, we compare it with a simple random sampling strategy in which points are uniformly selected regardless of the diffusion timestep. As shown in Tab. 5 (k), the proposed TF-sampling consistently improves both generation fidelity and diversity compared with random sampling. This result indicates that aligning the sampling strategy with the coarse-to-fine diffusion process benefits geometric reconstruction.

Effectiveness of Hyperparameters: We evaluate the hyperparameters τ and ζ on the *chair* category. As shown in Tab. 6, $\tau = 50$ and $\zeta = 0.875$ yield the best overall performance, balancing detail preservation and computational efficiency.

5. Conclusion

We propose a novel point cloud diffusion architecture that leverages state-space modeling and frequency analysis for generation. Our DM-Block employs latent representations and a newly designed 3DSS to enhance frequency-aware feature propagation within Mamba blocks, enabling more effective diffusion modeling. To recover fine-grained details in later timesteps, we introduce TF-Encoder, a time-variant frequency-based point extractor with minimal overhead. Experiments show that our method achieves state-of-the-art performance across categories while remaining highly efficient, with up to 10× fewer parameters and 9× faster inference than competitive approaches.

References

[1] Zeeshan Ahmad, Zain ul Abidin Jaffri, Meng Chen, and Shudi Bao. Understanding gans: Fundamentals, variants, training challenges, applications, and open problems. *Multimedia Tools and Applications*, 84(12):10347–10423, 2025. 1

[2] Yasuhiro Aoki, Hunter Goforth, Rangaprasad Arun Srivatsan, and Simon Lucey. Pointnetlk: Robust & efficient point cloud registration using pointnet. In *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, pages 7163–7172, 2019. 3, 5

[3] Omer F Atli, Bilal Kabas, Fuat Arslan, Arda C Demirtas, Mahmut Yurt, Onat Dalmaz, and Tolga Cukur. I2i-mamba: Multi-modal medical image synthesis via selective state space modeling. *arXiv preprint arXiv:2405.14022*, 2024. 2

[4] Jimmy Lei Ba. Layer normalization. *arXiv preprint arXiv:1607.06450*, 2016. 6

[5] Florian Bordes, Sina Honari, and Pascal Vincent. Learning to generate samples from noise through infusion training. In *Int. Conf. Learn. Represent. (ICLR)*, 2022. 1, 2

[6] Angel X Chang, Thomas Funkhouser, Leonidas Guibas, Pat Hanrahan, Qixing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, et al. Shapenet: An information-rich 3d model repository. *arXiv preprint arXiv:1512.03012*, 2015. 2, 5, 6

[7] An-Chieh Cheng, Xueting Li, Sifei Liu, Min Sun, and Ming-Hsuan Yang. Autoregressive 3d shape generation via canonical mapping. In *Eur. Conf. Comput. Vis. (ECCV)*, pages 89–104. Springer, 2022. 2

[8] Rosa Pia Devanna, Miguel Torres-Torriti, Kamil Sacilik, Necati Cetin, and Fernando Auat Cheein. Exploring the frequency domain point cloud processing for localisation purposes in arboreal environments. *Algorithms*, 18(8):522, 2025. 5

[9] Yi Du, Zhipeng Zhao, Shaoshu Su, Sharath Golluri, Haoze Zheng, Runmao Yao, and Chen Wang. Superpc: a single diffusion model for point cloud completion, upsampling, denoising, and colorization. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 16953–16964, 2025. 1

[10] Sebastian Eilermann, René Heesch, and Oliver Niggemann. A continuous-time consistency model for 3d point cloud generation, 2025. 7

[11] Stefan Elfwing, Eiji Uchibe, and Kenji Doya. Sigmoid-weighted linear units for neural network function approximation in reinforcement learning. *Neural networks*, 107: 3–11, 2018. 6

[12] Linjie Fu, Xia Li, Xiuding Cai, Yingkai Wang, Xueyao Wang, Yali Shen, and Yu Yao. Md-dose: A diffusion model based on the mamba for radiotherapy dose prediction. *arXiv preprint arXiv:2403.08479*, 2024. 1

[13] Albert Gu and Tri Dao. Mamba: Linear-time sequence modeling with selective state spaces. *arXiv preprint arXiv:2312.00752*, 2023. 1, 2

[14] Albert Gu, Karan Goel, and Christopher Ré. Efficiently modeling long sequences with structured state spaces. *arXiv preprint arXiv:2111.00396*, 2021. 2

[15] Anchit Gupta, Wenhan Xiong, Yixin Nie, Ian Jones, and Barlas Oğuz. 3dgen: Triplane latent diffusion for textured mesh generation. *arXiv preprint arXiv:2303.05371*, 2023. 1

[16] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Adv. Neural Inform. Process. Syst. (NeurIPS)*, 33:6840–6851, 2020. 1, 2

[17] Vincent Tao Hu, Stefan Andreas Baumann, Ming Gui, Olga Grebenkova, Pingchuan Ma, Johannes S Fischer, and Björn Ommer. Zigma: A dit-style zigzag mamba diffusion model. *arXiv preprint arXiv:2403.13802*, 2024. 1, 2

[18] Dongshuo Huang, Xiaoshui Huang, Chengdong Zhang, and Yilei Shi. Lpcg: A self-conditional architecture for labeled point cloud generation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 3635–3643, 2025. 2

[19] Hao Huang and Yi Fang. Adaptive wavelet transformer network for 3d shape representation learning. In *Int. Conf. Learn. Represent.*, 2022. 1, 2, 3

[20] Ka-Hei Hui, Chao Liu, Xiaohui Zeng, Chi-Wing Fu, and Arash Vahdat. Not-so-optimal transport flows for 3d point cloud generation. In *The Thirteenth International Conference on Learning Representations*, 2025. 1, 2, 7

[21] Zhihan Ju and Wanting Zhou. Vm-ddpm: Vision mamba diffusion for medical image synthesis. *arXiv preprint arXiv:2405.05667*, 2024. 2

[22] Ivan Kapelyukh, Vitalis Vosylius, and Edward Johns. Dall-e-bot: Introducing web-scale diffusion models to robotics. *IEEE Robotics and Automation Letters*, 8(7):3956–3963, 2023. 1

[23] Kento Kawaharazuka, Jihoon Oh, Jun Yamada, Ingmar Posner, and Yuke Zhu. Vision-language-action models for robotics: A review towards real-world applications. *IEEE Access*, 2025. 1

[24] Hyeongju Kim, Hyeonseung Lee, Woo Hyun Kang, Joun Yeop Lee, and Nam Soo Kim. Softflow: Probabilistic framework for normalizing flow on manifolds. *Adv. Neural Inform. Process. Syst. (NeurIPS)*, 33:16388–16397, 2020. 7

[25] Jinwoo Kim, Jaehoon Yoo, Juho Lee, and Seunghoon Hong. Setvae: Learning hierarchical composition for generative modeling of set-structured data. In *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, pages 15059–15068, 2021. 1, 2, 7

[26] Roman Klokov, Edmond Boyer, and Jakob Verbeek. Discrete point flow networks for efficient point cloud generation. In *Eur. Conf. Comput. Vis. (ECCV)*, pages 694–710. Springer, 2020. 2

[27] Zongjian Li, Bin Lin, Yang Ye, Liuhan Chen, Xinhua Cheng, Shenghai Yuan, and Li Yuan. Wf-vae: Enhancing video vae by wavelet-driven energy flow for latent video diffusion model. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 17778–17788, 2025. 1

[28] Dingkang Liang, Xin Zhou, Wei Xu, Xingkui Zhu, Zhikang Zou, Xiaoqing Ye, Xiao Tan, and Xiang Bai. Pointmamba: A simple state space model for point cloud analysis. In *Adv. Neural Inform. Process. Syst. (NeurIPS)*, 2024. 1, 2

[29] Xiao Liu, Chenxu Zhang, Fuxiang Huang, Shuyin Xia, Guoyin Wang, and Lei Zhang. Vision mamba: A comprehensive survey and taxonomy. *IEEE Transactions on Neural Networks and Learning Systems*, 2025. 1, 2

[30] Zhijian Liu, Haotian Tang, Yujun Lin, and Song Han. Point-voxel cnn for efficient 3d deep learning. *Adv. Neural Inform. Process. Syst. (NeurIPS)*, 32, 2019. 4

[31] Shitong Luo and Wei Hu. Diffusion probabilistic models for 3d point cloud generation. In *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, 2021. 1, 7

[32] Shentong Mo, Enze Xie, Ruihang Chu, Lanqing Hong, Matthias Niessner, and Zhenguo Li. Dit-3d: Exploring plain diffusion transformers for 3d shape generation. *Adv. Neural Inform. Process. Syst. (NeurIPS)*, 36, 2024. 1, 2, 7

[33] Lucas Nunes, Rodrigo Marcuzzi, Benedikt Mersch, Jens Behley, and Cyrill Stachniss. Scaling diffusion models to real-world 3d lidar scene completion. In *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, pages 14770–14780, 2024. 1

[34] Hao Phung, Quan Dao, and Anh Tran. Wavelet diffusion models are fast and scalable image generators. In *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, pages 10199–10208, 2023. 1, 2, 3

[35] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, pages 652–660, 2017. 3, 5

[36] Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *Adv. Neural Inform. Process. Syst. (NeurIPS)*, 30, 2017. 3, 5

[37] Prajit Ramachandran, Barret Zoph, and Quoc V. Le. Searching for activation functions, 2018. 5

[38] Zhiyuan Ren, Minchul Kim, Feng Liu, and Xiaoming Liu. Tiger: Time-varying denoising model for 3d point cloud generation with diffusion process. In *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, pages 9462–9471, 2024. 2, 3, 7

[39] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. *CoRR*, abs/1505.04597, 2015. 3

[40] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *Int. Conf. Mach. Learn. (ICML)*, pages 2256–2265. PMLR, 2015. 1

[41] Jingyu Song, Haiyong Xu, Gangyi Jiang, Mei Yu, Yeyao Chen, Ting Luo, and Yang Song. Frequency domain-based latent diffusion model for underwater image enhancement. *Pattern Recognition*, 160:111198, 2025. 1, 2

[42] Yao Teng, Yue Wu, Han Shi, Xuefei Ning, Guohao Dai, Yu Wang, Zhenguo Li, and Xihui Liu. Dim: Diffusion mamba for efficient high-resolution image synthesis. *arXiv preprint arXiv:2405.14224*, 2024. 2

[43] Si-Tong Wei, Rui-Huan Wang, Chuan-Zhi Zhou, Baoquan Chen, and Peng-Shuai Wang. Octgpt: Octree-based multiscale autoregressive models for 3d shape generation. In *Proceedings of the Special Interest Group on Computer Graphics and Interactive Techniques Conference Conference Papers*, pages 1–11, 2025. 2

[44] Cheng Wen, Baosheng Yu, and Dacheng Tao. Learning progressive point embeddings for 3d point cloud generation. In *Int. Conf. Comput. Vis. (ICCV)*, pages 10266–10275, 2021. 2

[45] Xiaoyang Wu, Yixing Lao, Li Jiang, Xihui Liu, and Hengshuang Zhao. Point transformer v2: Grouped vector attention and partition-based pooling. *Adv. Neural Inform. Process. Syst. (NeurIPS)*, 35:33330–33342, 2022. 1

[46] Yuxin Wu and Kaiming He. Group normalization. In *Eur. Conf. Comput. Vis. (ECCV)*, pages 3–19, 2018. 5

[47] Zhirong Wu, Shuran Song, Aditya Khosla, Fisher Yu, Linguang Zhang, Xiaoou Tang, and Jianxiong Xiao. 3d shapenets: A deep representation for volumetric shapes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1912–1920, 2015. 2, 5, 6, 7

[48] Mutian Xu, Junhao Zhang, Zhou Peng, Mingye Xu, Xiaojuan Qi, and Yu Qiao. Learning geometry-disentangled representation for complementary understanding of 3d object point cloud. *AAAI*, 35:3056–3064, 2021. 1, 2, 3, 4

[49] Guandao Yang, Xun Huang, Zekun Hao, Ming-Yu Liu, Serge Belongie, and Bharath Hariharan. Pointflow: 3d point cloud generation with continuous normalizing flows. In *Int. Conf. Comput. Vis. (ICCV)*, 2019. 1, 6, 7

[50] Xingyi Yang, Daquan Zhou, Jiashi Feng, and Xinchao Wang. Diffusion probabilistic model made slim. In *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, pages 22552–22562, 2023. 1, 2

[51] Kaixin Yao, Longwen Zhang, Xinhao Yan, Yan Zeng, Qixuan Zhang, Lan Xu, Wei Yang, Jiayuan Gu, and Jingyi Yu. Cast: Component-aligned 3d scene reconstruction from an rgb image. *ACM Transactions on Graphics (TOG)*, 44(4): 1–19, 2025. 1

[52] LAN Yushi, Shangchen Zhou, Zhaoyang Lyu, Fangzhou Hong, Shuai Yang, Bo Dai, Xingang Pan, and Chen Change Loy. Gaussiananything: Interactive point cloud flow matching for 3d generation. In *The Thirteenth International Conference on Learning Representations*, 2025. 2

[53] 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 *Adv. Neural Inform. Process. Syst. (NeurIPS)*, pages 10021–10039. Curran Associates, Inc., 2022. 2, 7

[54] Tao Zhang, Xiangtai Li, Haobo Yuan, Shunping Ji, and Shuicheng Yan. Point could mamba: Point cloud learning via state space model. *arXiv preprint arXiv:2403.00762*, 2024. 1, 2

[55] Hengshuang Zhao, Li Jiang, Jiaya Jia, Philip HS Torr, and Vladlen Koltun. Point transformer. In *Int. Conf. Comput. Vis. (ICCV)*, pages 16259–16268, 2021. 1, 3

[56] Chenliang Zhou, Fangcheng Zhong, Param Hanji, Zhilin Guo, Kyle Fogarty, Alejandro Sztrajman, Hongyun Gao, and Cengiz Oztireli. Frepolad: Frequency-rectified point latent diffusion for point cloud generation. In *Eur. Conf. Comput. Vis. (ECCV)*, 2024. 2, 7

[57] Linqi Zhou, Yilun Du, and Jiajun Wu. 3d shape generation and completion through point-voxel diffusion. In *Int. Conf. Comput. Vis. (ICCV)*, pages 5826–5835, 2021. 2, 3, 7

[58] Jiapeng Zhu, Ceyuan Yang, Kecheng Zheng, Yinghao Xu, Zifan Shi, Yifei Zhang, Qifeng Chen, and Yujun Shen. Exploring sparse moe in gans for text-conditioned image synthesis. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 18411–18423, 2025. 2