Pyramid Diffusion for Fine 3D Large Scene Generation

Yuheng Liu[1,2*], Xinke Li[3*], Xueting Li[4], Lu Qi[5†], Chongshou Li[1], Ming-Hsuan Yang[5,6]

[1] Southwest Jiaotong University [2] University of Leeds [3] City University of Hong Kong
[4] NVIDIA [5] The University of California, Merced [6] Yonsei University

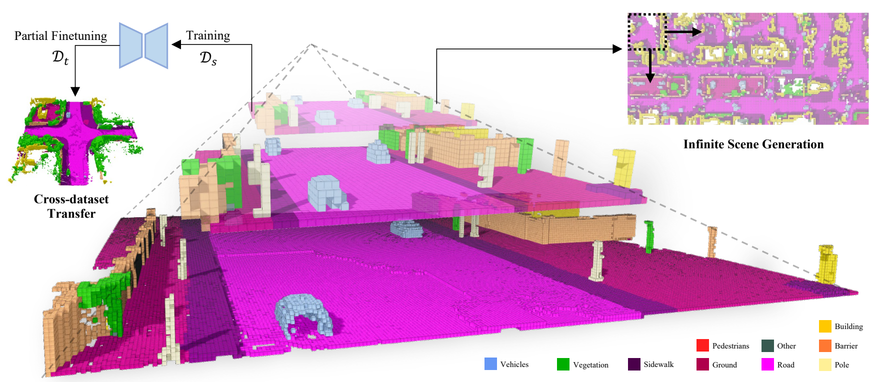

Fig. 1: We present Pyramid Discrete Diffusion Model, a method that progresses from generating coarse- to fine-grained scenes, mirroring the top-down sequence of the pyramid structure shown. The model is extended for cross-dataset and infinite scene generation, with detailed scene intricacies illustrated on the flanking sides of the image. $\mathcal{D}_s$ and $\mathcal{D}_t$ refer to a source dataset and a target dataset, respectively.

Abstract. Diffusion models have shown remarkable results in generating 2D images and small-scale 3D objects. However, their application to the synthesis of large-scale 3D scenes has been rarely explored. This is mainly due to the inherent complexity and bulky size of 3D scenery data, particularly outdoor scenes, and the limited availability of comprehensive real-world datasets, which makes training a stable scene diffusion model challenging. In this work, we explore how to effectively generate large-scale 3D scenes using the coarse-to-fine paradigm. We introduce a framework, the Pyramid Discrete Diffusion model (PDD), which employs scale-varied diffusion models to progressively generate high-quality outdoor scenes. Experimental results of PDD demonstrate

* Equal contribution.
† Corresponding author.

our successful exploration in generating 3D scenes both unconditionally and conditionally. We further showcase the data compatibility of the PDD model, due to its multi-scale architecture: a PDD model trained on one dataset can be easily fine-tuned with another dataset. Code is available at https://github.com/yuhengliu02/pyramid-discrete-diffusion.

Keywords: 3D Scene Generation · Diffusion Models · Transfer Learning

1 Introduction

3D scene generation is the task of creating digital representations that mimic the three-dimensional complexities of our real-world environment, allowing for a more nuanced understanding of the tangible surroundings. This technique plays an essential role in fundamental computer vision tasks such as autonomous driving [21,42,44], virtual reality [30,32,39], and robotic manipulation [8,16,46]. However, high-quality large 3D scenes are extremely challenging to synthesize due to their inherently bulky size, and the lack of large-scale 3D scene datasets [43].

Meanwhile, recent advances in diffusion models have shown impressive results in generating 2D images [35–37] or small-scale 3D objects [24, 33]. Yet, it is not a trivial task to employ diffusion models in 3D scene generation. On one hand, state-of-the-art diffusion models leave substantial memory footprints and demand considerable training time, posing a particular challenge when generating 3D scenes with large scales and intricate details. On the other hand, diffusion models require a large amount of training data [29,48], while capturing large-scale 3D scenes is itself a challenging and ongoing research topic [4, 23]. As a result, only a few attempts have been made to apply diffusion models directly to 3D outdoor scenes [18], which resulted in unstable generation and thus suboptimal performance.

To resolve these challenges, existing works focus on conditional generation and resort to additional signals such as Scene Graphs [41] or 2D maps [28] for guidance. Nonetheless, such conditional guidance is not always accessible, thereby restricting the generalizability of these approaches. Inspired by the coarse-to-fine philosophy widely used in image super-resolution [14, 31, 38], we introduce the Pyramid Discrete Diffusion model (PDD), a framework that progressively generates large 3D scenes without relying on additional guidance.

We begin by generating small-scale 3D scenes and progressively increase the scale. At each scale level, we learn a separate diffusion model. This model takes the generated scene from the previous scale as a condition (except for the first diffusion model which takes noise as input) and synthesizes a 3D scene of a larger scale. Intuitively, this multi-scale generation process breaks down a challenging unconditional generation task (*i.e.*, high-quality 3D scene generation) into several more manageable conditional generation tasks. This separation allows each diffusion model to specialize in generating either coarse structure (smaller scale) or intricate details (larger scale). Moreover, at the highest scale, we employ a technique known as scene subdivision, which involves dividing a large scene into multiple smaller segments that are synthesized using a shared diffusion model.

This approach mitigates the issues of oversized models caused by the bulky size of the 3D scenes. Additionally, a noteworthy outcome of our multi-scale design is its capacity to facilitate cross-data transfer applications, which substantially reduces training resources. Lastly, we further propose a natural extension of our PDD framework with scene subdivision for infinite 3D scene generation, thereby demonstrating the scalability of the proposed method.

The main contributions of this work are as follows:

- We practically implement a coarse-to-fine strategy for 3D outdoor scene generation via designing a novel pyramid diffusion model.
- We conduct extensive experiments on our pyramid diffusion, demonstrating its generation of higher quality 3D scenes with comparable computational resources of existing approaches. In addition, we introduce new metrics to evaluate the quality of 3D scene generation from various perspectives.
- Our proposed method showcases broader applications, enabling the generation of scenes from synthetic datasets to real-world data. Furthermore, our approach can be extended to facilitate the creation of infinite scenes.

2 Related Work

Diffusion Models for 2D Images. Recent advancements in the generative model have seen the diffusion models [14, 31, 38] rise to prominence, especially in applications in 2D image creation [9,35,36]. In order to generate high-fidelity images via diffusion models, a multi-stage diffusion process is proposed and employed as per [13, 15, 37]. This process starts with the generation of a coarse-resolution image using an initial diffusion model. Subsequently, a second diffusion model takes this initial output as input, refining it into a finer-resolution image. These cascaded diffusions can be iteratively applied to achieve the desired image resolution. We note that the generation of fine-grained 3D data presents more challenges than 2D due to the addition of an extra dimension. Consequently, our work is motivated by the aforementioned multistage 2D approaches to explore their applicability in 3D contexts. Furthermore, we aim to leverage the advantages of this structure to address the scarcity of datasets in 3D scenes.

Diffusion Models for 3D Generation. As a sparse and memory-efficient representation, 3D point clouds has been widely used in various computer vision applications such as digital human [27, 40, 49], autonomous driving [20], and 3D scene reconstruction [17]. Point clouds generation aims to synthesize a 3D point clouds from a random noise [5,6], or scanned lidar points [18]. Though the memory efficiency of point clouds is a valuable property, it poses high challenges in the task of point cloud generation. Existing works largely focus on using Generative Adversarial Networks (GANs), Variational Autoencoders (VAEs), or Vector Quantized Variational Autoencoders (VQ-VAEs) as the backbone for this task [1, 5, 6]. However, these models have limited capacity for high-fidelity generation and are notoriously known for unstable training. As an alternative to the generative models discussed above, diffusion models have revolutionized the computer vision community with their impressive performance in 2D image

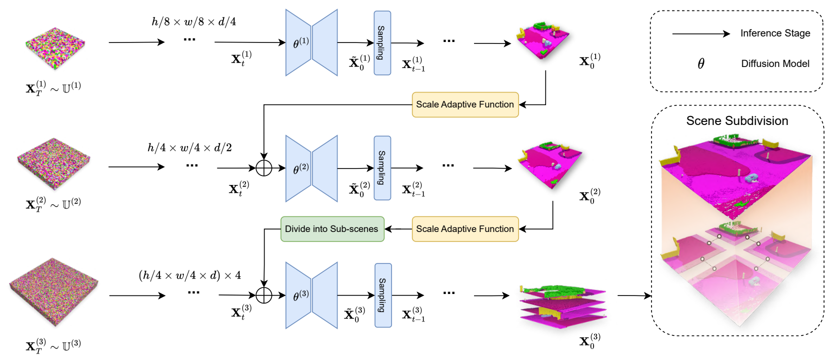

Fig. 2: Framework of the proposed Pyramid Discrete Diffusion model. In our structure, there are three different scales. Scenes generated by a previous scale can serve as a condition for the current scale after processing through our scale adaptive function. Furthermore, for the final scale processing, the scene from the previous scale is subdivided into four sub-scenes. The final scene is reconstructed into a large scene using our Scene Subdivision module.

generation [35–37]. Yet, applying diffusion models for 3D point cloud generation has not been thoroughly explored hitherto. Point-Voxel Diffusion [50] proposes to generate a raw point cloud through the diffusion process while LION [47] and DPM [26] use the latent representation of a point cloud during the denoising process. However, all these methods focus on object-level point clouds and cannot be naively extended to scene-level point clouds. Most relevant to our work is [18], where a diffusion model is trained on a scene-level point cloud dataset for the synthesis task. However, due to the capacity limitation of diffusion models, generating a scene-level point cloud with a single diffusion model leads to unsatisfying results, such as undesired wholes or the lack of fine-grained objects. In this work, we propose a pyramid discrete diffusion model that reduces the difficulty at each pyramid level, thus producing scene point clouds with more realistic and fine-grained details.

3D Large-scale Scene Generation. Generating large-scale 3D scenes is an important but highly challenging task. A generative model on 3D scenes potentially provides infinite training data for tasks such as scene segmentation, autonomous driving, etc. Existing works [4, 22, 23, 45] simplify this task by first generating 2D scenes and then "lifting" them to 3D. Though such design is efficient for city scenes populated with regular geometries (e.g., buildings), it does not generalize easily to scenes with more fine-grained objects (e.g., pedestrians, cars, trees, etc.) In this paper, we directly generate 3D outdoor scenes using diffusion models, which include abundant small objects with semantics.

3 Approach

The proposed Pyramid Discrete Diffusion (PDD) model comprises multi-scale models capable of step-by-step generation of high-quality 3D scenes from smaller scales. The PDD first extends the standard discrete diffusion for 3D data (Section 3.2) and then proposes a scene subdivision method to further reduce memory requirements (Section 3.3). Finally, we demonstrate two practical applications of PDD in specific scenarios (Section 3.4).

3.1 Discrete Diffusion

We focus on learning a data distribution based on 3D semantic scenes. Specifically, the semantic scene is represented in a one-hot format, *i.e.,* $\mathbf{X} \in \{0,1\}^{h \times w \times d \times c}$, where h, w, and d indicate the dimensions of the scene, respectively, and c denotes the size of the one-hot label.

Discrete diffusion [2] has been proposed to generate discrete data including semantic scenes. It involves applying the Markov transition matrix on discrete states for noise diffusion. In the forward process, an original scene $\mathbf{X}_0$ is gradually corrupted into a t-step noised map $\mathbf{X}_t$ with $t = 1, \cdots, T$. Each forward step can be defined by a Markov uniform transition matrix $\mathbf{Q}_t$ as $\mathbf{X}_t = \mathbf{X}_{t-1}\mathbf{Q}_t$. Based on the Markov property, we can derive the t-step scene $\mathbf{X}_t$ straight from $\mathbf{X}_0$ with a cumulative transition matrix $\bar{\mathbf{Q}}_t = \mathbf{Q}_1\mathbf{Q}_2 \cdots \mathbf{Q}_t$:

$$q\left(\mathbf{X}_t \mid \mathbf{X}_0\right) = \operatorname{Cat}\left(\mathbf{X}_t; \mathbf{P} = \mathbf{X}_0 \bar{\mathbf{Q}}_t\right), \tag{1}$$

where $\operatorname{Cat}(\mathbf{X}; \mathbf{P})$ is a multivariate categorical distribution over the one-hot semantic labels $\mathbf{X}$ with probabilities given by $\mathbf{P}$. Finally, the semantic scene $\mathbf{X}_T$ at the last step T is supposed to be in the form of a uniform discrete noise. In the reverse process, a learnable model parametrized by θ is used to predict denoised semantic labels by $\tilde{p}_\theta\left(\tilde{\mathbf{X}}_0 \mid \mathbf{X}_t\right)$. The reparametrization trick is applied subsequently to get the reverse process $p_\theta\left(\mathbf{X}_{t-1} \mid \mathbf{X}_t\right)$:

$$p_\theta\left(\mathbf{X}_{t-1} \mid \mathbf{X}_t\right) = \mathbb{E}_{\tilde{p}_\theta\left(\tilde{\mathbf{X}}_0 \mid \mathbf{X}_t\right)} q\left(\mathbf{X}_{t-1} \mid \mathbf{X}_t, \tilde{\mathbf{X}}_0\right). \tag{2}$$

A loss consisting of the two KL divergences is proposed to learn better reconstruction ability for the model, given by

$$\begin{aligned} \mathcal{L}_\theta = & \, d_{\mathrm{KL}}\left(q\left(\mathbf{X}_{t-1} \mid \mathbf{X}_t, \mathbf{X}_0\right) \| p_\theta\left(\mathbf{X}_{t-1} \mid \mathbf{X}_t\right)\right) \\ & + \lambda d_{\mathrm{KL}}\left(q\left(\mathbf{X}_0\right) \| \tilde{p}_\theta\left(\tilde{\mathbf{X}}_0 \mid \mathbf{X}_t\right)\right), \end{aligned} \tag{3}$$

where λ is an auxiliary loss weight and d_{KL} stands for KL divergence. In the following, we focus on extending the discrete diffusion into the proposed PDD.

3.2 Pyramid Discrete Diffusion

We propose PDD that operates various diffusion processes across multiple scales (or resolutions), as depicted in Figure 2. Given a 3D scene data $\mathbf{Z} \in \{0,1\}^{h\times w\times d\times c}$, we define a 3D pyramid including different scales of $\mathbf{Z}$, *i.e.*, $\{\mathbf{Z}^{(1)}, \cdots, \mathbf{Z}^{(l)}, \cdots, \mathbf{Z}^{(L)}\}$, where a larger l indicates a larger scene scale. Formally, let $h_l \times w_l \times d_l \times c$ denote the dimension of $\mathbf{Z}^{(l)}$, $h_{l+1} \geq h_l$, $w_{l+1} \geq w_l$ and $d_{l+1} \geq d_l$ are kept for $l = 1, \cdots, L-1$. We note that such a pyramid can be obtained by applying different down-sample operators, such as pooling functions, on $\mathbf{Z}$. For each scale in the pyramid, we construct a conditional discrete diffusion model parameterized by θ_l. The l-th model for $l \neq 1$ is given by:

$$\tilde{p}_{\theta_l}\left(\tilde{\mathbf{X}}_0^{(l)} \mid \mathbf{X}_t^{(l)}, \mathbf{Z}^{(l-1)}\right) = \tilde{p}_{\theta_l}\left(\tilde{\mathbf{X}}_0^{(l)} \mid \mathrm{Concat}\left(\mathbf{X}_t^{(l)}, \phi^{(l)}(\mathbf{Z}^{(l-1)})\right)\right), \quad (4)$$

where $\mathbf{X}_t^{(l)}$ and $\mathbf{X}_0^{(l)}$ are with the same size of $\mathbf{Z}^{(l)}$, and $\phi^{(l)}$ is a Scale Adaptive Function (SAF) for upsamling $\mathbf{Z}^{(l-1)}$ into the size of $\mathbf{Z}^{(l)}$. As a case in point, SAF can be a tri-linear interpolation function depending on the data. Additionally, we maintain the first model $\tilde{p}_{\theta_1}$ as the original non-conditional model.

During the training process, PDD learns L denoising models separately at varied scales of scene pyramids in the given dataset. Given that $\mathbf{Z}^{(l-1)}$ is essentially a lossy-compressed version of $\mathbf{Z}^{(l)}$, the model training can be viewed as learning to restore the details of a coarse scene. In the inference process, denoising model p_{θ_1} is performed initially according to Equation (2) and the rest of PDD models are executed in sequence from $l = 2$ to L via the sampling,

$$\mathbf{X}_{t-1}^{(l)} \sim p_{\theta_l}(\mathbf{X}_{t-1}^{(l)} \mid \mathbf{X}_t^{(l)}, \mathbf{X}_0^{(l-1)}), \quad (5)$$

where $\mathbf{X}_0^{(l-1)}$ is the denoised result of $\tilde{p}_{\theta_{l-1}}$.

Except for the high-quality generation, the proposed PDD bears two merits: 1) Diffusion models in PDD can be trained in parallel due to their independence, which allows for a flexible computation reallocation during training. 2) Due to its multi-stage generation process, PDD is fitting for restoring scenes of arbitrary coarse-grained scale by starting from the intermediate processes, thereby extending the method's versatility.

3.3 Scene Subdivision

To overcome the memory constraint for generating large 3D scenes, we propose the scene subdivision method. We divide a 3D scene $\mathbf{Z}^{(l)}$ along z-axis into I overlapped sub-components as $\{\mathbf{Z}_i^{(l)}\}_{i=1}^{I}$. For the instance of four subscenes case, let $\mathbf{Z}_i^{(l)} \in \{0,1\}^{(1+\delta_l)h_l \backslash 2\times(1+\delta_l)w_l \backslash 2\times d_l \times c}$ denote one subscene and δ_l denote the overlap ratio, the shared l-th diffusion model in PDD is trained to reconstruct $\mathbf{Z}_i^{(l)}$ for $i = 1, \cdots, 4$. Subsequently, sub-scenes are merged into a holistic one by a fusion algorithm, *i.e.*, voting on the overlapped parts to ensure the continuity of the 3D scene.

In the training process, to ensure context-awareness of the entire scene during the generation of a sub-scene, we train the model by adding the overlapped regions with other sub-scenes as the condition. In the inference process, the entire scene is generated in an autoregressive manner. Apart from the first sub-scene generated without context, all other sub-scenes utilize the already generated overlapped region as a condition, *i.e.*,

$$\mathbf{X}_{t-1,i}^{(l)} \sim p_\theta \left(\mathbf{X}_{t-1,i}^{(l)} \mid \mathbf{X}_{t,i}^{(l)}, \mathbf{X}_{0,i}^{(l+1)}, \sum_{j \neq i} \Delta_{ij} \odot \mathbf{X}_{0,j}^{(l+1)} \right), \tag{6}$$

where j is the index of generated sub-scenes before i-th scene, and Δ_{ij} is a binary mask between $\mathbf{X}_{0,i}^{(l+1)}$ and $\mathbf{X}_{0,j}^{(l+1)}$ representing the overlapped region on $\mathbf{X}_{0,j}^{(l+1)}$ with 1 and the separate region with 0. Scene Subdivision module can reduce the model parameters, as diffusion model could be shared by four sub-scenes. In practice, we only implement the scene subdivision method on the largest scale which demands the largest memory.

3.4 Applications

Beyond its primary function as a generative model, we introduce two novel applications for PDD. First, **cross-dataset transfer** aims at adapting a model trained on a source dataset to a target dataset [51]. Due to the flexibility of input scale, PDD can achieve this by retraining or fine-tuning the smaller-scale models in the new dataset while keeping the larger-scale models. The strategy leveraging PDD improves the efficiency of transferring 3D scene generation models between distinct datasets. Second, **infinite scene generation** is of great interest in fields such as autonomous driving [10] and urban modeling [19] which require a huge scale of 3D scenes. PDD can extend its scene subdivision technique. By using the edge of a previously generated scene as a condition as in Equation (6), it can iteratively create larger scenes, potentially without size limitations.

4 Experimental Results

4.1 Evaluation Protocols

Since the metrics used in 2D generation such as FID [12] are not directly applicable in the 3D, we introduce and implement three metrics to assess the quality of the generated 3D scenes. We note that more implementation details can be found in the supplementary material.

Semantic Segmentation results on the generated scenes are used to evaluate the effectiveness of models in creating semantically coherent scenes. Specifically, two architectures, the voxel-based SparseUNet [11] and point-based PointNet++ [34], are implemented to perform the segmentation tasks. We report the mean Intersection over Union (mIoU) and Mean Accuracy (MAs) for evaluation.

Table 1: Comparison of various diffusion models on 3D semantic scene generation of CarlaSC. DiscreteDiff [2], LatentDiff [18], and P-DiscreteDiff refer to the original discrete diffusion, latent discrete diffusion, and our approach, respectively. Conditioned models work based on the context of unlabeled point clouds or the coarse version of the ground truth scene. A higher *Segmentation Metric* value is better, indicating semantic consistency. A lower *Feature-based Metric* value is preferable, representing closer proximity to the original dataset. The brackets with V represent voxel-based network and P represent point-based network.

Method	Model	Condition	Segmentation Metric				Feature-based Metric	
			mIoU (V)	MA (V)	mIoU (P)	MA (P)	F3D (↓)	MMD (↓)
Ground Truth	-	-	52.19	72.40	32.90	47.68	0.246	0.108
Unconditioned	DiscreteDiff [2]	-	40.05	63.65	25.54	38.71	1.361	0.599
	LatentDiff [18]	-	38.01	62.39	26.69	45.87	0.331	0.211
	P-DiscreteDiff (Ours)	-	**68.02**	**85.66**	**33.89**	**52.12**	**0.315**	**0.200**
Conditioned	DiscreteDiff [2]	Point cloud	38.55	59.97	28.41	44.06	0.357	**0.261**
	DiscreteDiff [2]	Coarse scene (s_1)	52.52	77.23	27.93	43.13	0.359	0.284
	P-DiscreteDiff (Ours)	Coarse scene (s_1)	**55.75**	**78.70**	**29.78**	**46.61**	**0.342**	0.274

F3D is a 3D adaption of the 2D Fréchet Inception Distance (FID) [12], which is based on a pre-trained autoencoder with an 3D CNN architecture. We calculate and report the Fréchet distance (by 10^{-3} ratio) between the generated scenes and real scenes in the feature domain.

Maximum Mean Discrepancy (MMD) is a statistical measure to quantify the disparity between the distributions of generated and real scenes. Similar to our F3D approach, we extract features via the same pre-trained autoencoder and present the MMD between 3D scenes.

4.2 Experiment Settings

Datasets. We use CarlaSC [43] and SemanticKITTI [3] for experiments. Specifically, we conduct our main experiments as well as ablation studies on the synthesis dataset CarlaSC due to its large data volume and diverse semantic objects. Our primary model is trained on the training set of CarlaSC with 10 categories and 32,400 scans. SemanticKITTI, which is a real-world collected dataset, is used for our cross-dataset transfer experiment. Both datasets are adjusted to ensure consistency in semantic categories, with further details in the supplementary material.

Model Architecture. The primary proposed PDD is performed on three scales of a 3D scene pyramid, *i.e.*, s_1, s_2 and s_4 in Table 2. We implement 3D-UNets [7] for three diffusion models in PDD based on the scales. Notably, the model applied on s_4 scale is with the input/output size of s'_3 due to the use of scene subdivision, while such a size of other models follows the working scale size. In the ablation study, we also introduce the scale s_3 in the experiment. Additionally, we implement two baseline methods merely on scale s_4 which are the original discrete diffusion [2] and the latent diffusion model with VQ-VAE decoder [18].

Training Setting. We train each PDD model using the same training setting except for the batch size. Specifically, we set the learning rate of 10^{-3} for the

Table 2: Different scales in the 3D scene pyramid.

Scale Rep.	3D Scene Size
s_1	$32 \times 32 \times 4$
s_2	$64 \times 64 \times 8$
s_3	$128 \times 128 \times 8$
s_3'	$136 \times 136 \times 16$
s_4	$256 \times 256 \times 16$

Table 3: Average SSIM between each generated scene and the closest scene in the training set. We generate $1k$ scenes and calculat the average.

Dataset	Method	SSIM
CarlaSC	Uncon-Gen	0.72
	Val Set	0.65
Sem-KITTI (*fine-tuned*)	Uncon-Gen	0.74
	Val Set	0.67

Table 4: Comparison of different diffusion pyramids on 3D semantic scene generation.

Pyramid	Conditioned	mIoU(V)	mIoU(P)	F3D(↓)	MMD(↓)
s_4	×	40.0	25.5	1.36	0.60
$s_1 \to s_4$	×	67.0	32.1	0.32	0.24
$s_1 \to s_2 \to s_4$	×	**68.0**	**33.9**	**0.32**	**0.20**
$s_1 \to s_2 \to s_3 \to s_4$	×	**68.0**	33.4	**0.32**	0.23
$s_1 \to s_4$	✓	52.5	27.9	0.36	0.28
$s_1 \to s_2 \to s_4$	✓	55.8	**29.8**	**0.34**	**0.27**
$s_1 \to s_2 \to s_3 \to s_4$	✓	**55.9**	29.6	**0.34**	0.28

AdamW optimizer [25], and the time step $T = 100$ for the diffusion process, and 800 for the max epoch. The batch sizes are set to 128, 32, and 16 for the models working on s_1, s_2 and s_4 scales. However, for the baseline method based on the s_4 scale, we use the batch size of 8 due to memory constraints. We note that all diffusion models are trained on four NVIDIA A100 GPUs. In addition, we apply the trilinear interpolation for the scene fusion algorithm and set the overlap ratio in scene subdivision, δ_l to 0.0625.

4.3 Main Results

Generation Quality. We compare our approach with two baselines, the original Discrete Diffusion [2] and the Latent Diffusion [18]. The result reported in Table 1 demonstrates the notable performance of our method across all metrics in both unconditional and conditional settings in comparable computational resources with existing method. Our proposed method demonstrates a notable advantage in segmentation tasks, especially when it reaches around 70% mIoU for SparseUNet, which reflects its ability to generate scenes with accurate semantic coherence. We also provide visualizations of different model results in Figure 3, where the proposed method demonstrates better performance in detail generation and scene diversity for random 3D scene generations.

Additionally, we conduct the comparison on conditioned 3D scene generation. We leverage the flexibility of input scale for our method and perform the generation by models in s_2 and s_4 scales conditioned on a coarse ground truth scene in s_1 scale. We benchmark our method against the discrete diffusion conditioned on

Fig. 3: Visualization of unconditional generation results on CarlaSC. We compare with two baseline models – DiscreteDiff [2] and LatentDiff [18] and show synthesis from our models with different scales. Our method produces more diverse scenes compared to the baseline models. Furthermore, with more levels, our model can synthesize scenes with more intricate details.

unlabeled point clouds and the same coarse scenes. Results in Table 1 and Figure 5 present the impressive results of our conditional generation comparison. It is also observed that the point cloud-based model can achieve decent performance on F3D and MMD, which could be caused by 3D point conditions providing more structural information about the scene than the coarse scene. Despite the informative condition of the point cloud, our method can still outperform it across most metrics.

None-overfitting Verification. The MMD and F3D metrics (defined in Supp A) numerically illustrate the statistical feature distance between generated scenes and the training set. Our method achieves the lowest MMD and F3D among all baseline methods as shown in Table 1. However, we argue that this *does not indicate overfitting* to the dataset for the following reasons. First, our MMD and F3D are larger than those of the ground truth. Furthermore, we leverage structural similarity (SSIM) to search and show that a generated scene is different from its nearest neighbour in the dataset. Specifically, we generate $1k$ scenes and identify their closest matches in the training set using the SSIM metric. The average SSIM of these $1k$ scenes are calculated and presented in Table 3. Additionally, we apply the same methodology to the Validation Set to establish an

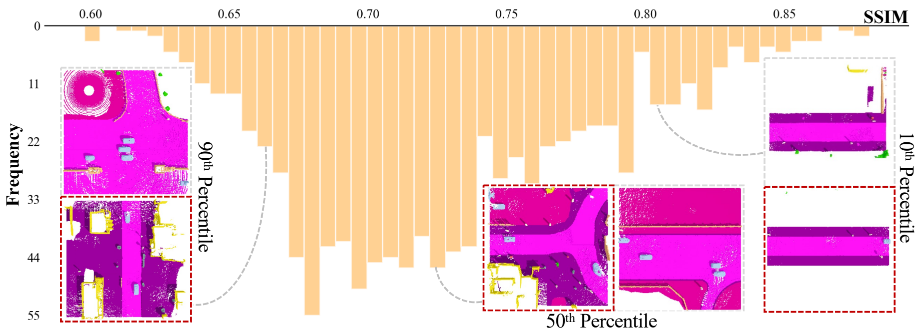

Fig. 4: Data retrieval visualization. We generate $1k$ scenes using PDD on CarlaSC dataset, and retrieve the most similar scene in training set for each scene using SSIM (SSIM=1 means identical), and plot SSIM distribution. Scenes at various percentiles are displayed (red box: generated scenes; grey box: scenes in training set), those with the highest 10% similarity (i.e., 10^{th} percentile of SSIM) are very similar to the training set, but still not completely identical.

oracle baseline. Table 3 shows that our generated scenes are comparable to the baseline, verifying that our method does not overfit the training set. To further support this, we use distribution plots, as shown in Figure 4, to validate the similarity between our generated scenes and the training set. We also display three pairs at different percentiles in this figure, showing that scenes with lower SSIM scores differ more from their nearest matches in the training set. This visual evidence reinforces that PDD effectively captures the distribution of the training set instead of merely memorizing it.

4.4 Ablation Studies

Pyramid Diffusion. Our experiments explore the impact of varying refinement scales on the quality of generated scenes. According to Table 4, both conditional and unconditional scene generation quality show incremental improvements with additional scales. Balancing training overhead and generation quality, a three-scale model with the scale of s_4 progression offers an optimal compromise between performance and computational cost. We find that as the number of scales increases, there is indeed a rise in performance, particularly notable upon the addition of the second scale. However, the progression from a three-scale pyramid to a four-scale pyramid turns out to be insignificant. Given the greater training overhead for a four-scale pyramid compared to a three-scale one, we choose the latter as our main structure.

Scene Subdivision. We explore the optimal mask ratio for scene subdivision and report on Figure 6, which shows an inverse correlation between the mask ratio and the effectiveness of F3D and MMD metrics; higher mask ratios may result in diminished outcomes due to increased overfitting, leading to reduced randomness in the generated results. The lowest mask ratio test, 0.0625, achieves

Fig. 5: Visualization of conditional generation results on CarlaSC. *PC* stands for point cloud condition.

Table 5: Generation results on CarlaSC in different scales on the diffusion pyramid without any conditions. All output scales are lifted to s_4 using the upsampling method.

Model No.	Scale	mIoU(↑)	MA(↑)	F3D(↓)	MMD(↓)
1	s_1	18.0	42.7	0.29	0.16
2	s_2	43.7	66.8	0.29	0.18
3	s_4	68.0	85.7	0.32	0.23

the best results across all metrics with detailed retention. Thus, we set a mask ratio of 0.0625 as the standard for our scene subdivision module. Further analysis shows that higher overlap ratios in scene subdivision result in quality deterioration, mainly due to increased discontinuities when merging sub-scenes using scene fusion algorithm.

Performance on Different Scales. To facilitate a comprehensive understanding of the progressive improvement achieved by our coarse-to-fine method, we evaluate the quality of scenes generated at different scales and present the findings in Table 5. Even at the smaller scale s_1, we observe high F3D and MMD scores, indicating its capacity in synthesizing scenes with both reasonable and diverse layouts. As we advance to larger scales (i.e., s_2 and s_3), the mIoU and MA scores consistently increase, demonstrating that our model focuses on learning intricate details in later stages. Meanwhile, F3D and MMD metrics show stability without significant decline, indicating a balanced enhancement in scene complexity and fidelity.

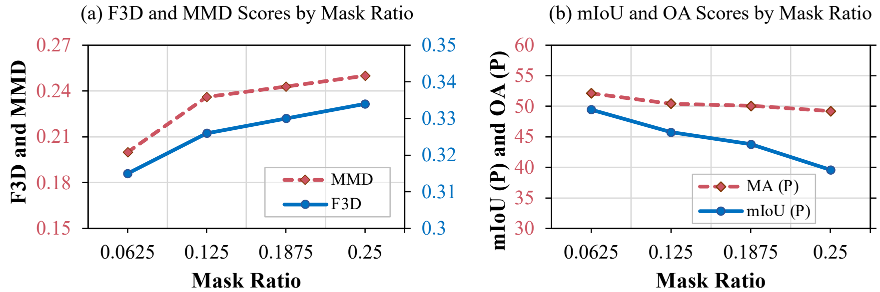

Fig. 6: Effects of mask ratio on unconditional generation results.

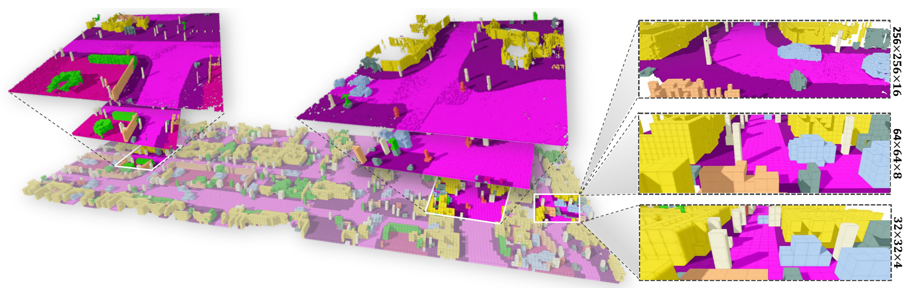

Fig. 7: Infinite Scene Generation. Thanks to the pyramid representation, PDD can be readily applied for unbounded scene generation. This involves the initial efficient synthesis of a large-scale coarse 3D scene, followed by subsequent refinement at higher levels.

4.5 Applications

Cross-dataset. Figure 8 and Figure 9 showcase our model's performance on the transferred dataset from CarlaSC to SemanticKITTI for both unconditional and conditional scene generation. The PDD shows enhanced scene quality after finetuning with SemanticKITTI data, as indicated by improved results in Table 6. This fine-tuning effectively adapts the model to the dataset's complex object distributions and scene dynamics. Notably, despite the higher training effort of the Discrete Diffusion (DD) approach, our method outperforms DD even without fine-tuning, using only coarse scenes from SemanticKITTI. This demonstrates the strong cross-data transfer capability of our approach.

Infinite Scene Generation. Figure 7 visualizes the process of generating large-scale infinite scenes using our PDD model. As discussed in Section 3.4, we first use the small-scale model to swiftly generate a coarse infinite 3D scene (bottom level in Figure 7). We then leverage larger-scale models to progressively add intricate details (middle and top level in Figure 7), enhancing realism. This approach enables our model to produce high-quality, continuous cityscapes without additional inputs, overcoming the limitations of conventional datasets with finite scenes and supporting downstream tasks like 3D scene segmentation.

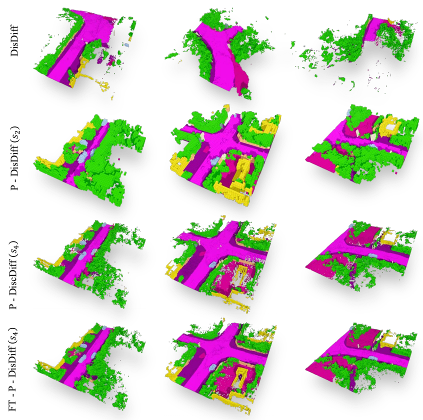

Fig. 8: SemanticKITTI unconditional generation. *FT* stands for finetuning pre-trained model from CarlaSC.

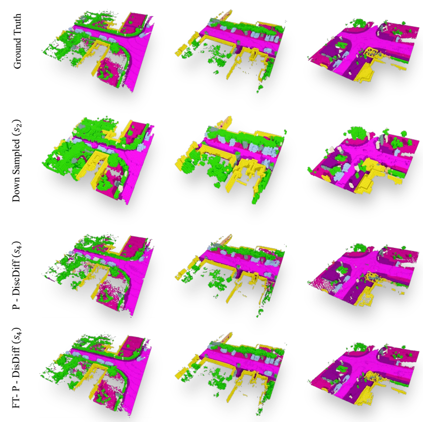

Fig. 9: SemanticKITTI conditional generation. *FT* stands for finetuning from CarlaSC models.

Table 6: Generation results on SemanticKITTI. *Finetuned Scales* set to None indicates training from scratch and others stand for finetuning corresponding pre-trained CarlaSC model.

Method	Finetuned Scales	Conditioned	mIoU(V)	mIoU(P)	F3D(↓)	MMD(↓)
DD [2]	s_4	×	29.1	16.0	0.46	0.31
PDD	None	✓	33.4	**22.8**	**0.27**	0.32
PDD	s_2, s_4	✓	**43.9**	**22.8**	0.28	**0.16**
PDD	s_1	×	31.3	23.2	0.22	0.13
PDD	s_1, s_2, s_4	×	**44.7**	**24.2**	**0.21**	**0.11**

5 Conclusion

In this work, we introduce the Pyramid Discrete Diffusion model (PDD) to address the significant challenges associated with 3D large scene generation, particularly in the context of limited scale and available datasets. The PDD demonstrates a progressive approach to generating high-quality 3D outdoor scenes, transitioning seamlessly from coarse to fine details. Compared to the other methods, the PDD can generate high-quality scenes within limited resource constraints and does not require additional data sources. Our extensive experimental results show that PDD consistently performs favourably in both unconditional and conditional generation tasks, establishing itself as a reliable and robust solution for the creation of realistic and intricate scenes. Furthermore, the proposed PDD method has great potential in efficiently adapting models trained on synthetic data to real-world datasets and suggests a promising solution to the current challenge of limited real-world data.

Acknowledgement

Supported by the Intelligence Advanced Research Projects Activity (IARPA) via Department of Interior/ Interior Business Center (DOI/IBC) contract number 140D0423C0074. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright annotation thereon. Disclaimer: The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of IARPA, DOI/IBC, or the U.S. Government. Y. Liu and C. Li are supported in part by the National Science Foundation of China (Grant No: 62202395).

References

1. Anvekar, T., Tabib, R.A., Hegde, D., Mudengudi, U.: Vg-vae: A venatus geometry point-cloud variational auto-encoder. In: CVPR (2022)
2. Austin, J., Johnson, D.D., Ho, J., Tarlow, D., Van Den Berg, R.: Structured denoising diffusion models in discrete state-spaces. In: NeurIPS (2021)
3. Behley, J., Garbade, M., Milioto, A., Quenzel, J., Behnke, S., Stachniss, C., Gall, J.: Semantickitti: A dataset for semantic scene understanding of lidar sequences. In: ICCV (2019)
4. Chen, Z., Wang, G., Liu, Z.: Scenedreamer: Unbounded 3d scene generation from 2d image collections. In: arXiv preprint arXiv: 2302.01330 (2023)
5. Cheng, A.C., Li, X., Liu, S., Sun, M., Yang, M.H.: Learning 3d dense correspondence via canonical point autoencoder. In: ECCV (2022)
6. Cheng, A.C., Li, X., Sun, M., Yang, M.H., Liu, S.: Learning 3d dense correspondence via canonical point autoencoder. In: NeurIPS (2021)
7. Çiçek, Ö., Abdulkadir, A., Lienkamp, S.S., Brox, T., Ronneberger, O.: 3d u-net: learning dense volumetric segmentation from sparse annotation. In: MICCAI (2016)
8. Cong, Y., Chen, R., Ma, B., Liu, H., Hou, D., Yang, C.: A comprehensive study of 3-d vision-based robot manipulation. IEEE Transactions on Cybernetics (2021)
9. Dhariwal, P., Nichol, A.: Diffusion models beat gans on image synthesis. In: NeurIPS (2021)
10. Geiger, A., Lenz, P., Urtasun, R.: Are we ready for autonomous driving? the kitti vision benchmark suite. In: CVPR (2012)
11. Graham, B., Engelcke, M., Van Der Maaten, L.: 3d semantic segmentation with submanifold sparse convolutional networks. In: CVPR (2018)
12. Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., Hochreiter, S.: Gans trained by a two time-scale update rule converge to a local nash equilibrium. In: NeurIPS (2017)
13. Ho, J., Chan, W., Saharia, C., Whang, J., Gao, R., Gritsenko, A., Kingma, D.P., Poole, B., Norouzi, M., Fleet, D.J., et al.: Imagen video: High definition video generation with diffusion models. arXiv preprint arXiv:2210.02303 (2022)
14. Ho, J., Jain, A., Abbeel, P.: Denoising diffusion probabilistic models. In: NeurIPS (2020)
15. Ho, J., Saharia, C., Chan, W., Fleet, D.J., Norouzi, M., Salimans, T.: Cascaded diffusion models for high fidelity image generation. The Journal of Machine Learning Research (2022)

16. Huang, W., Wang, C., Zhang, R., Li, Y., Wu, J., Fei-Fei, L.: Voxposer: Composable 3d value maps for robotic manipulation with language models. arXiv preprint arXiv:2307.05973 (2023)
17. Lan, Z., Yew, Z.J., Lee, G.H.: Robust point cloud based reconstruction of large-scale outdoor scenes. In: CVPR (2019)
18. Lee, J., Im, W., Lee, S., Yoon, S.E.: Diffusion probabilistic models for scene-scale 3d categorical data. arXiv preprint arXiv:2301.00527 (2023)
19. Li, X., Li, C., Tong, Z., Lim, A., Yuan, J., Wu, Y., Tang, J., Huang, R.: Campus3d: A photogrammetry point cloud benchmark for hierarchical understanding of outdoor scene. In: ACM MM (2020)
20. Li, Y., Ma, L., Zhong, Z., Liu, F., Chapman, M.A., Cao, D., Li, J.: Deep learning for lidar point clouds in autonomous driving: A review. NeurIPS (2020)
21. Li, Y., Yu, A.W., Meng, T., Caine, B., Ngiam, J., Peng, D., Shen, J., Lu, Y., Zhou, D., Le, Q.V., et al.: Deepfusion: Lidar-camera deep fusion for multi-modal 3d object detection. In: CVPR (2022)
22. Li, Z., Wang, Q., Snavely, N., Kanazawa, A.: Infinitenature-zero: Learning perpetual view generation of natural scenes from single images. In: ECCV (2022)
23. Lin, C.H., Lee, H.Y., Menapace, W., Chai, M., Siarohin, A., Yang, M.H., Tulyakov, S.: Infinicity: Infinite-scale city synthesis. arXiv preprint arXiv:2301.09637 (2023)
24. Liu, M., Shi, R., Chen, L., Zhang, Z., Xu, C., Wei, X., Chen, H., Zeng, C., Gu, J., Su, H.: One-2-3-45++: Fast single image to 3d objects with consistent multi-view generation and 3d diffusion. arXiv preprint arXiv:2311.07885 (2023)
25. Loshchilov, I., Hutter, F.: Decoupled weight decay regularization. arXiv preprint arXiv:1711.05101 (2017)
26. Luo, S., Hu, W.: Diffusion probabilistic models for 3d point cloud generation. In: CVPR (2021)
27. Ma, Q., Yang, J., Tang, S., Black, M.J.: The power of points for modeling humans in clothing. In: ICCV (2021)
28. Mascaro, R., Teixeira, L., Chli, M.: Diffuser: Multi-view 2d-to-3d label diffusion for semantic scene segmentation. In: ICRA (2021)
29. Moon, T., Choi, M., Lee, G., Ha, J.W., Lee, J.: Fine-tuning diffusion models with limited data. In: NeurIPS 2022 Workshop on Score-Based Methods (2022)
30. Moro, S., Komuro, T.: Generation of virtual reality environment based on 3d scanned indoor physical space. In: ISVC (2021)
31. Nichol, A.Q., Dhariwal, P.: Improved denoising diffusion probabilistic models. In: ICML (2021)
32. Ögün, M.N., Kurul, R., Yaşar, M.F., Turkoglu, S.A., Avci, Ş., Yildiz, N.: Effect of leap motion-based 3d immersive virtual reality usage on upper extremity function in ischemic stroke patients. Arquivos de neuro-psiquiatria (2019)
33. Poole, B., Jain, A., Barron, J.T., Mildenhall, B.: Dreamfusion: Text-to-3d using 2d diffusion. arXiv (2022)
34. Qi, C.R., Yi, L., Su, H., Guibas, L.J.: Pointnet++: Deep hierarchical feature learning on point sets in a metric space. In: NeurIPS (2017)
35. Ramesh, A., Dhariwal, P., Nichol, A., Chu, C., Chen, M.: Hierarchical text-conditional image generation with clip latents. arXiv preprint arXiv:2204.06125 (2022)
36. Rombach, R., Blattmann, A., Lorenz, D., Esser, P., Ommer, B.: High-resolution image synthesis with latent diffusion models. In: CVPR (2022)
37. Saharia, C., Chan, W., Saxena, S., Li, L., Whang, J., Denton, E.L., Ghasemipour, K., Gontijo Lopes, R., Karagol Ayan, B., Salimans, T., et al.: Photorealistic text-to-image diffusion models with deep language understanding. In: NeurIPS (2022)

38. Sohl-Dickstein, J., Weiss, E.A., Maheswaranathan, N., Ganguli, S.: Deep unsupervised learning using nonequilibrium thermodynamics. arXiv preprint arXiv:1503.03585 (2015)
39. Sra, M., Garrido-Jurado, S., Schmandt, C., Maes, P.: Procedurally generated virtual reality from 3d reconstructed physical space. In: Proceedings of the 22nd ACM Conference on Virtual Reality Software and Technology (2016)
40. Su, S.Y., Bagautdinov, T., Rhodin, H.: Npc: Neural point characters from video. In: ICCV (2023)
41. Tang, J., Nie, Y., Markhasin, L., Dai, A., Thies, J., Nießner, M.: Diffuscene: Scene graph denoising diffusion probabilistic model for generative indoor scene synthesis. arXiv preprint arXiv:2303.14207 (2023)
42. Tang, Y., He, H., Wang, Y., Mao, Z., Wang, H.: Multi-modality 3d object detection in autonomous driving: A review. Neurocomputing (2023)
43. Wilson, J., Song, J., Fu, Y., Zhang, A., Capodieci, A., Jayakumar, P., Barton, K., Ghaffari, M.: Motionsc: Data set and network for real-time semantic mapping in dynamic environments. IEEE Robotics and Automation Letters **7**(3) (2022)
44. Wu, P., Jia, X., Chen, L., Yan, J., Li, H., Qiao, Y.: Trajectory-guided control prediction for end-to-end autonomous driving: A simple yet strong baseline. In: NeurIPS (2022)
45. Xie, H., Chen, Z., Hong, F., Liu, Z.: Citydreamer: Compositional generative model of unbounded 3d cities. arXiv preprint arXiv:2309.00610 (2023)
46. Xu, Z., He, Z., Wu, J., Song, S.: Learning 3d dynamic scene representations for robot manipulation. arXiv preprint arXiv:2011.01968 (2020)
47. Zeng, X., Vahdat, A., Williams, F., Gojcic, Z., Litany, O., Fidler, S., Kreis, K.: Lion: Latent point diffusion models for 3d shape generation. In: NeurIPS (2022)
48. Zheng, H., Nie, W., Vahdat, A., Anandkumar, A.: Fast training of diffusion models with masked transformers. arXiv preprint arXiv:2306.09305 (2023)
49. Zheng, Y., Yifan, W., Wetzstein, G., Black, M.J., Hilliges, O.: Pointavatar: Deformable point-based head avatars from videos. In: CVPR (2023)
50. Zhou, L., Du, Y., Wu, J.: 3d shape generation and completion through point-voxel diffusion. In: ICCV (2021)
51. Zhuang, F., Qi, Z., Duan, K., Xi, D., Zhu, Y., Zhu, H., Xiong, H., He, Q.: A comprehensive survey on transfer learning. arXiv preprint arXiv:1911.02685 (2020)