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]National University of Singapore
[4]NVIDIA, [5]The Univeristy of California, Merced, [6]Google Research

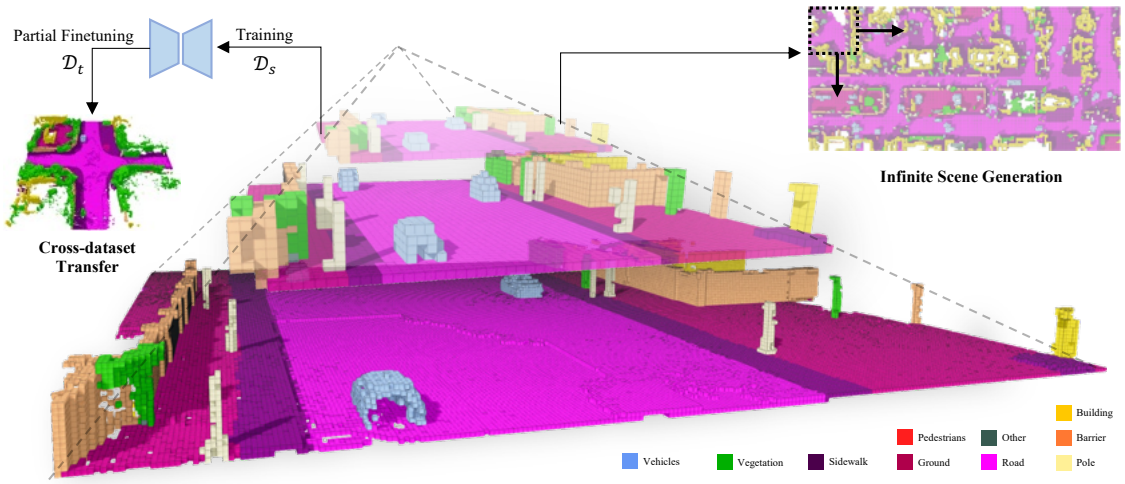

Figure 1. We present Pyramid Discrete Diffusion Model, a method that progresses from generating coarse- to fine-grained scenes, mirroring the top-down sequence of the pyramid structure shown. The model is extended for cross-dataset and infinite scene generation, with detailed scene intricacies illustrated 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

In this paper, we aim to apply the popular diffusion model for large 3D scene generation. Directly transferring the 2D techniques to 3D scene generation is challenging due to significant resolution reduction and the scarcity of comprehensive real-world 3D scene datasets. To address these issues, our work introduces the Pyramid Discrete Diffusion model (PDD) for 3D scene generation. This novel approach employs a multi-scale model capable of progressively generating high-quality 3D scenes from coarse to fine. In this way, the PDD can generate high-quality scenes within limited resource constraints and does not require additional data sources. To the best of our knowledge, we are the first to adopt the simple but effective coarse-to-fine strategy for 3D large scene generation. Our experiments, covering both unconditional and conditional generation, have yielded impressive results, showcasing the model's effectiveness and robustness in generating realistic and detailed 3D scenes. Our code will be available to the public.

*corresponding author.

1. Introduction

In recent years, generative models have seen significant advancements in both 2D and 3D fields, primarily driven by the evolution of diffusion model techniques [4, 11, 21, 48]. While those generation tasks have made impressive visual effects, 3D large scene generation task stands out for its vast applicability in various cutting-edge applications such as autonomous driving [24, 43, 45], virtual reality [31, 33, 40], and robotic manipulation [9, 18, 47].

Although the diffusion model has benefited 2D high-quality image synthesis by latent design [34, 37] or multi-scale features [36, 38], transferring those techniques to 3D scene generation presents significant challenges due to two reasons. The first is the significant resolution reduction. For example, a diffusion model can handle 512×512 resolution in 2D but can generate only $128 \times 128 \times 16$ in 3D scenes. On the other hand, the scarcity of comprehensive real-world 3D scene datasets is insufficient for training robust diffusion models that should require a large number of data.

To tackle the challenges associated with low-resolution training in diffusion processes, several methods introduce auxiliary signals for guidance that include employing Scene

Graphs as outlined in [42], using classifier guidance as per [53], and integrating 2D maps as demonstrated in [30]. Albeit these techniques can remedy the lost high-resolution information by other signals, they tend to depend on extra data sources, accelerating the scarcity of collected 3D data.

Inspired by the coarse-to-fine pipeline widely used in image resolution [15, 32, 39], we introduce the Pyramid Discrete Diffusion model (PDD) for 3D scene generation. Specifically, PDD has several multi-scale models capable of progressively generating high-quality 3D scenes starting from more minor scales. Albeit simple, this innovative approach has been severely explored before. To the best of our knowledge, we are the first to extend the coarse-to-fine diffusion to 3D semantic scenes and incorporate a scene subdivision method with three advantages. At first, it enables the generation of high-quality scenes within limited resource constraints and facilitates the gradual refinement of scenes from coarse to high-resolution without the need for additional data sources. Secondly, PDD's structural flexibility yields impressive results in cross-data transfer applications using the SemanticKITTI dataset, significantly outperforming baseline models. Thirdly, PDD holds the potential to generate infinite outdoor scenes, demonstrating its scalability and adaptability in varied environmental contexts.

The main contributions of this work are as follows:

- We conduct extensive experiments on 3D diffusion across various pyramid scales, successfully demonstrating the generation of high-quality scenes with decent computational resources.
- We introduce and elaborate on metrics for evaluating the quality of 3D scene generation. These metrics are versatile and applicable across multiple 3D scene datasets.
- 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 [15, 32, 39] rise to prominence, especially in applications in 2D image creation [10, 36, 37]. In order to generate high-fidelity images via diffusion models, a multi-stage diffusion process is proposed and employed as per [16, 17, 38]. 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. In current practice, the majority of 3D generative models primarily focus on 3D point clouds, as 3D point clouds are more straightforward. It has been widely used in various computer vision applications such as digital human [29, 41, 50], autonomous driving [23], and 3D scene reconstruction [19]. Point clouds generation aims to synthesize a 3D point clouds from a random noise [6, 7], or scanned lidar points [20]. 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, 6, 7]. 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 generation [36–38]. Yet, applying diffusion models for 3D data generation has not been thoroughly explored hitherto. Point-Voxel Diffusion [51] proposes to generate a raw point cloud through the diffusion process while LION [49] and DPM [28] 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 [20], 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 largely 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 [5, 25, 26, 46] 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.

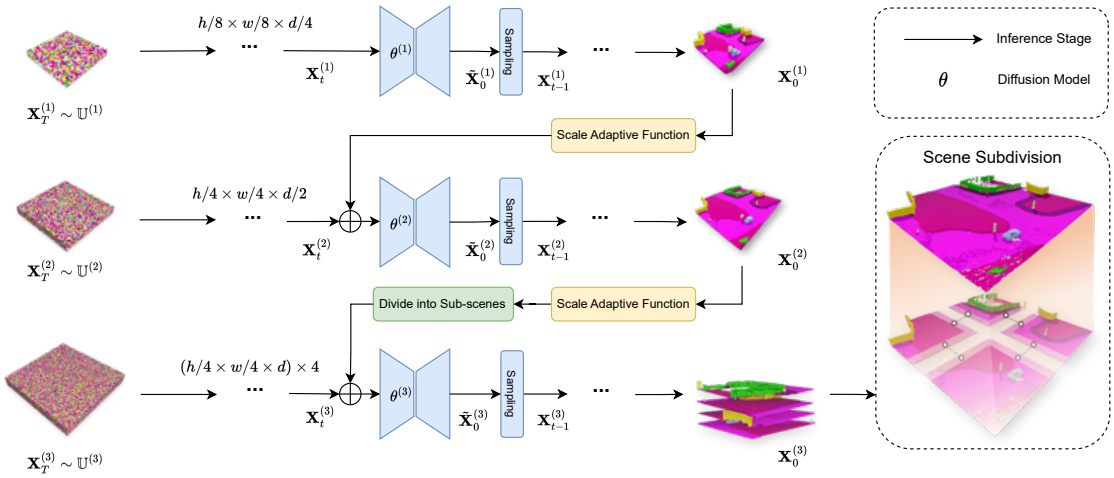

Figure 2. Framework of the proposed Pyramid Discrete Diffusion model. In our structure, there are three different scales. Scenes generated by a previous scale can serve as a condition for the current scale after processing through our scale adaptive function. Furthermore, for the final scale processing, 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.

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 $\bar{\mathbf{Q}}_t = \mathbf{Q}_1\mathbf{Q}_2\cdots\mathbf{Q}_t$:

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

where $\mathrm{Cat}(\mathbf{X};\mathbf{P})$ 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 $p_\theta\left(\mathbf{X}_{t-1} \mid \mathbf{X}_t\right)$:

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

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 content, 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:

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

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 trilinear 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,

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

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 subcomponents 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 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. 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 [52]. 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 [12] and urban modeling [22] 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 [14] 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 Appendix.

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 [13] and point-based PointNet++ [35], are implemented to perform the segmentation tasks. We report the mean Intersection over Union (mIoU) and Mean Accuracy (MAs) for evaluation.

F3D is a 3D adaption of the 2D Fréchet Inception Distance (FID) [14], 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

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 [20]	-	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

Table 2. Comparison of various diffusion models on 3D semantic scene generation of CarlaSC. DiscreteDiff [2], LatentDiff [20], 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.

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 [44] 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 with 3,834 scans, is used for our cross-dataset transfer experiment. Both datasets are adjusted to ensure consistency in semantic categories, with further details in the Appendix.

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 1. We implement 3D-UNets [8] 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 [20].

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 AdamW optimizer [27], 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 SAF and set the overlap ratio in scene subdivision, δ_l to 0.0625.

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 1. Different scales in the 3D scene pyramid.

4.3. Main Results

Generation Quality. We compare our approach with two baselines, the original Discrete Diffusion [2] and the Latent Diffusion [20]. The result reported in Table 2 highlights the superiority of our method across all metrics in both unconditional and conditional settings. 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 unlabeled point clouds and the same coarse scenes. Results in Table 2 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.

Figure 3. **Visualization of unconditional generation results on CarlaSC.** We compare with two baseline models – DiscreteDiff [2] and LatentDiff [20] 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.

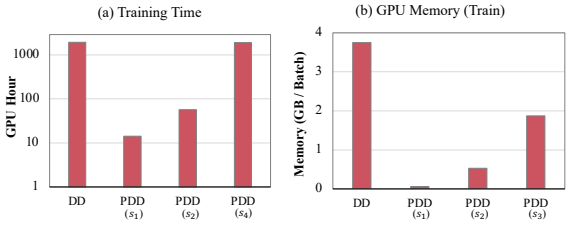

Figure 4. Training time and memory usage for training PDD and DD on CarlaSC dataset.

Computational Resources. Figure 4 depicts the GPU training time and memory requirements for our PDD on identical configurations. Using a logarithmic scale for training time emphasizes the efficiency gains of our method. The initial training stage, PDD (s_1), requires substantially less time—up to 100 times less—compared to training the full DD model. It also minimizes GPU memory usage, which broadens the potential for deployment on hardware with lower specifications. This enhanced efficiency extends to subsequent scales, with the final scale, PDD (s_4), only necessitating retraining at smaller scales. Such an approach significantly cuts down on total training time and memory usage, highlighting the pragmatic benefits of our pyramid training architecture.

4.4. Ablation Studies

Pyramid Diffusion. Our experiments explore the impact of varying refinement scales on the quality of generated scenes. According to Table 3, 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 substantially 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 lead to diminished outcomes. The lowest mask ratio test, 0.0625, achieves the best results across all metrics, suggesting a balance between detail retention and computational efficiency. Thus, we set a mask ratio of 0.0625 as the standard for

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

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.

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

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

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

Table 4. 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.

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 Pyramid Discrete Diffusion model shows enhanced quality in scene generation after finetuning with SemanticKITTI data, as indicated by the improved mIoU, F3D, and MMD metrics in Table 5. The fine-tuning process effectively adapts the model to the dataset's complex object distributions and scene dynamics, resulting in improved results for both generation scenarios. We also highlight that, despite the higher training efforts of the Discrete Diffusion (DD) approach, our method outperforms DD even without fine-tuning, simply by using coarse scenes from SemanticKITTI. This demonstrates the strong cross-data transfer capability of our approach.

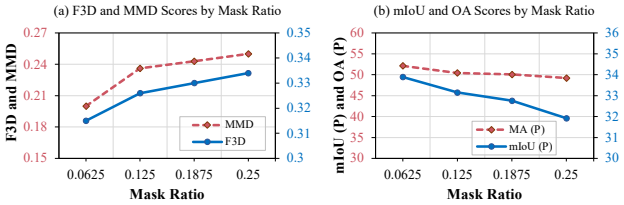

Figure 6. Effects of mask ratio on unconditional generation results.

Method	Finetuned Scales	Condition	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

Table 5. Generation results on SemanticKITTI. Setting *Finetuned Scales* to None stands for train-from-scratch and others stand for finetuning corresponding pre-trained CarlaSC model.

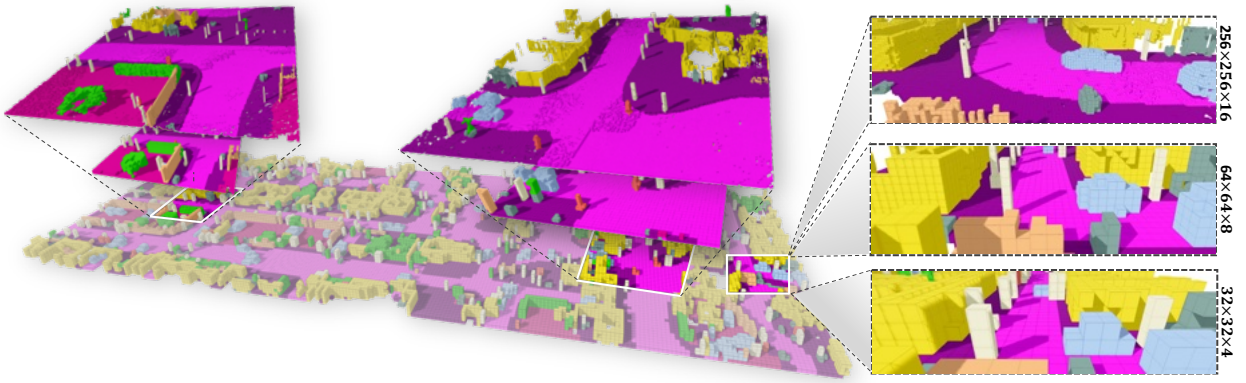

Figure 7. Infinite Scene Generation. Thanks to the pyramid representation, PDD can be readily applied for unbounded scene generation. This involves the initial efficient synthesis of a large-scale coarse 3D scene, followed by subsequent refinement at higher levels.

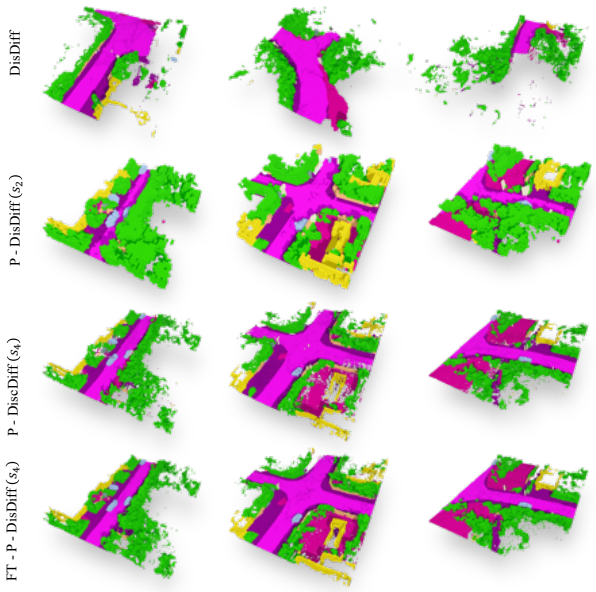

Figure 8. SemanticKITTI unconditional generation. *FT* stands for finetuning pre-trained model from CarlaSC.

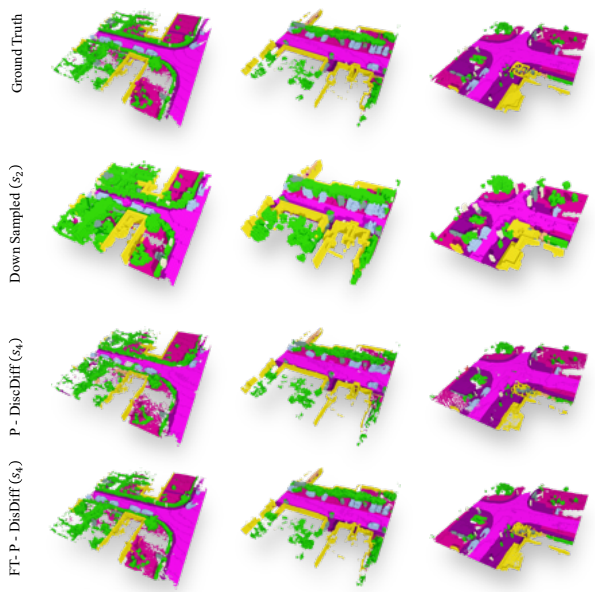

Figure 9. SemanticKITTI conditional generation. Our proposed PDD achieves results close to the groundtruth. Note that *FT* stands for finetuning from CarlaSC models.

Infinite Scene Generation. Figure 7 demonstrates our model's ability to generate large-scale, coarse-grained scenes beyond standard dataset dimensions. This initial scale precedes a refinement process that adds detail to these expansive outdoor scenes. Our model produces continuous cityscapes without needing additional inputs. Using our method, it is possible to generate infinite scenes. The figure shows the generation process in scales: beginning with a coarse scene, it focuses on refining a segment into detailed 3D scenes.

5. Conclusion

In this work, we introduce the Pyramid Discrete Diffusion model (PDD) to address the significant challenges in 3D large scene generation, particularly in the limitations of low-resolution and available datasets. The PDD demonstrates a novel approach in progressively generating high-quality 3D scenes from coarse to fine. Compared to the other methods, the PDD can generate high-quality scenes within limited resource constraints and does not require additional data sources. Our experiments highlight its impressive performance in both unconditional and conditional generation tasks, offering a robust solution for realistic and detailed scene creation. Looking forward, our 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.

References

[1] Tejas Anvekar, Ramesh Ashok Tabib, Dikshit Hegde, and Uma Mudengudi. Vg-vae: A venatus geometry point-cloud variational auto-encoder. In *CVPR*, 2022. 2

[2] Jacob Austin, Daniel D Johnson, Jonathan Ho, Daniel Tarlow, and Rianne Van Den Berg. Structured denoising diffusion models in discrete state-spaces. In *NeurIPS*, 2021. 3, 5, 6, 7

[3] Jens Behley, Martin Garbade, Andres Milioto, Jan Quenzel, Sven Behnke, Cyrill Stachniss, and Jurgen Gall. Semantickitti: A dataset for semantic scene understanding of lidar sequences. In *ICCV*, 2019. 5

[4] Hanqun Cao, Cheng Tan, Zhangyang Gao, Yilun Xu, Guangyong Chen, Pheng-Ann Heng, and Stan Z Li. A survey on generative diffusion model. *arXiv preprint arXiv:2209.02646*, 2022. 1

[5] Zhaoxi Chen, Guangcong Wang, and Ziwei Liu. Scenedreamer: Unbounded 3d scene generation from 2d image collections. In *arXiv preprint arXiv: 2302.01330*, 2023. 2

[6] An-Chieh Cheng, Xueting Li, Min Sun, Ming-Hsuan Yang, and Sifei Liu. Learning 3d dense correspondence via canonical point autoencoder. In *NeurIPS*, 2021. 2

[7] An-Chieh Cheng, Xueting Li, Sifei Liu, Min Sun, and Ming-Hsuan Yang. Learning 3d dense correspondence via canonical point autoencoder. In *ECCV*, 2022. 2

[8] Özgün Çiçek, Ahmed Abdulkadir, Soeren S Lienkamp, Thomas Brox, and Olaf Ronneberger. 3d u-net: learning dense volumetric segmentation from sparse annotation. In *MICCAI*, 2016. 5

[9] Yang Cong, Ronghan Chen, Bingtao Ma, Hongsen Liu, Dongdong Hou, and Chenguang Yang. A comprehensive study of 3-d vision-based robot manipulation. *IEEE Transactions on Cybernetics*, 2021. 1

[10] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. In *NeurIPS*, 2021. 2

[11] Wenqi Fan, Chengyi Liu, Yunqing Liu, Jiatong Li, Hang Li, Hui Liu, Jiliang Tang, and Qing Li. Generative diffusion models on graphs: Methods and applications. *arXiv preprint arXiv:2302.02591*, 2023. 1

[12] Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *CVPR*, 2012. 4

[13] Benjamin Graham, Martin Engelcke, and Laurens Van Der Maaten. 3d semantic segmentation with submanifold sparse convolutional networks. In *CVPR*, 2018. 4

[14] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. In *NeurIPS*, 2017. 4

[15] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *NeurIPS*, 2020. 2

[16] Jonathan Ho, William Chan, Chitwan Saharia, Jay Whang, Ruiqi Gao, Alexey Gritsenko, Diederik P Kingma, Ben Poole, Mohammad Norouzi, David J Fleet, et al. Imagen video: High definition video generation with diffusion models. *arXiv preprint arXiv:2210.02303*, 2022. 2

[17] Jonathan Ho, Chitwan Saharia, William Chan, David J Fleet, Mohammad Norouzi, and Tim Salimans. Cascaded diffusion models for high fidelity image generation. *The Journal of Machine Learning Research*, 2022. 2

[18] Wenlong Huang, Chen Wang, Ruohan Zhang, Yunzhu Li, Jiajun Wu, and Li Fei-Fei. Voxposer: Composable 3d value maps for robotic manipulation with language models. *arXiv preprint arXiv:2307.05973*, 2023. 1

[19] Ziquan Lan, Zi Jian Yew, and Gim Hee Lee. Robust point cloud based reconstruction of large-scale outdoor scenes. In *CVPR*, 2019. 2

[20] Jumin Lee, Woobin Im, Sebin Lee, and Sung-Eui Yoon. Diffusion probabilistic models for scene-scale 3d categorical data. *arXiv preprint arXiv:2301.00527*, 2023. 2, 5, 6

[21] Puheng Li, Zhong Li, Huishuai Zhang, and Jiang Bian. On the generalization properties of diffusion models. *arXiv preprint arXiv:2311.01797*, 2023. 1

[22] Xinke Li, Chongshou Li, Zekun Tong, Andrew Lim, Junsong Yuan, Yuwei Wu, Jing Tang, and Raymond Huang. Campus3d: A photogrammetry point cloud benchmark for hierarchical understanding of outdoor scene. In *ACM MM*, 2020. 4

[23] Ying Li, Lingfei Ma, Zilong Zhong, Fei Liu, Michael A Chapman, Dongpu Cao, and Jonathan Li. Deep learning for lidar point clouds in autonomous driving: A review. *NeurIPS*, 2020. 2

[24] Yingwei Li, Adams Wei Yu, Tianjian Meng, Ben Caine, Jiquan Ngiam, Daiyi Peng, Junyang Shen, Yifeng Lu, Denny Zhou, Quoc V Le, et al. Deepfusion: Lidar-camera deep fusion for multi-modal 3d object detection. In *CVPR*, 2022. 1

[25] Zhengqi Li, Qianqian Wang, Noah Snavely, and Angjoo Kanazawa. Infinitenature-zero: Learning perpetual view generation of natural scenes from single images. In *ECCV*, 2022. 2

[26] Chieh Hubert Lin, Hsin-Ying Lee, Willi Menapace, Menglei Chai, Aliaksandr Siarohin, Ming-Hsuan Yang, and Sergey Tulyakov. Infinicity: Infinite-scale city synthesis. *arXiv preprint arXiv:2301.09637*, 2023. 2

[27] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017. 5

[28] Shitong Luo and Wei Hu. Diffusion probabilistic models for 3d point cloud generation. In *CVPR*, 2021. 2

[29] Qianli Ma, Jinlong Yang, Siyu Tang, and Michael J. Black. The power of points for modeling humans in clothing. In *ICCV*, 2021. 2

[30] Ruben Mascaro, Lucas Teixeira, and Margarita Chli. Diffuser: Multi-view 2d-to-3d label diffusion for semantic scene segmentation. In *ICRA*, 2021. 2

[31] Satoshi Moro and Takashi Komuro. Generation of virtual reality environment based on 3d scanned indoor physical space. In *ISVC*, 2021. 1

[32] Alexander Quinn Nichol and Prafulla Dhariwal. Improved denoising diffusion probabilistic models. In *ICML*, 2021. 2

[33] Muhammed Nur Ögün, Ramazan Kurul, Mustafa Fatih Yaşar, Sule Aydin Turkoglu, Şebnem Avci, and Nebil Yildiz.

Effect of leap motion-based 3d immersive virtual reality usage on upper extremity function in ischemic stroke patients. *Arquivos de neuro-psiquiatria*, 2019. 1

[34] Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. Sdxl: Improving latent diffusion models for high-resolution image synthesis. *arXiv preprint arXiv:2307.01952*, 2023. 1

[35] Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. In *NeurIPS*, 2017. 4

[36] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 2022. 1, 2

[37] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *CVPR*, 2022. 1, 2

[38] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. Photorealistic text-to-image diffusion models with deep language understanding. In *NeurIPS*, 2022. 1, 2

[39] Jascha Sohl-Dickstein, Eric A. Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. *arXiv preprint arXiv:1503.03585*, 2015. 2

[40] Misha Sra, Sergio Garrido-Jurado, Chris Schmandt, and Pattie Maes. Procedurally generated virtual reality from 3d reconstructed physical space. In *Proceedings of the 22nd ACM Conference on Virtual Reality Software and Technology*, 2016. 1

[41] Shih-Yang Su, Timur Bagautdinov, and Helge Rhodin. Npc: Neural point characters from video. In *ICCV*, 2023. 2

[42] Jiapeng Tang, Yinyu Nie, Lev Markhasin, Angela Dai, Justus Thies, and Matthias Nießner. Diffuscene: Scene graph denoising diffusion probabilistic model for generative indoor scene synthesis. *arXiv preprint arXiv:2303.14207*, 2023. 2

[43] Yingjuan Tang, Hongwen He, Yong Wang, Zan Mao, and Haoyu Wang. Multi-modality 3d object detection in autonomous driving: A review. *Neurocomputing*, 2023. 1

[44] Joey Wilson, Jingyu Song, Yuewei Fu, Arthur Zhang, Andrew Capodieci, Paramsothy Jayakumar, Kira Barton, and Maani Ghaffari. Motionsc: Data set and network for real-time semantic mapping in dynamic environments. *IEEE Robotics and Automation Letters*, 7(3), 2022. 5

[45] Penghao Wu, Xiaosong Jia, Li Chen, Junchi Yan, Hongyang Li, and Yu Qiao. Trajectory-guided control prediction for end-to-end autonomous driving: A simple yet strong baseline. In *NeurIPS*, 2022. 1

[46] Haozhe Xie, Zhaoxi Chen, Fangzhou Hong, and Ziwei Liu. Citydreamer: Compositional generative model of unbounded 3d cities. *arXiv preprint arXiv:2309.00610*, 2023. 2

[47] Zhenjia Xu, Zhanpeng He, Jiajun Wu, and Shuran Song. Learning 3d dynamic scene representations for robot manipulation. *arXiv preprint arXiv:2011.01968*, 2020. 1

[48] Ling Yang, Zhilong Zhang, Yang Song, Shenda Hong, Runsheng Xu, Yue Zhao, Wentao Zhang, Bin Cui, and Ming-Hsuan Yang. Diffusion models: A comprehensive survey of methods and applications. *ACM Comput. Surv.*, 2023. 1

[49] 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 *NeurIPS*, 2022. 2

[50] Yufeng Zheng, Wang Yifan, Gordon Wetzstein, Michael J. Black, and Otmar Hilliges. Pointavatar: Deformable point-based head avatars from videos. In *CVPR*, 2023. 2

[51] Linqi Zhou, Yilun Du, and Jiajun Wu. 3d shape generation and completion through point-voxel diffusion. In *ICCV*, 2021. 2

[52] Fuzhen Zhuang, Zhiyuan Qi, Keyu Duan, Dongbo Xi, Yongchun Zhu, Hengshu Zhu, Hui Xiong, and Qing He. A comprehensive survey on transfer learning. *arXiv preprint arXiv:1911.02685*, 2020. 4

[53] Vlas Zyrianov, Xiyue Zhu, and Shenlong Wang. Learning to generate realistic lidar point cloud. In *ECCV*, 2022. 2