

# Distilling Diffusion Models to Efficient 3D LiDAR Scene Completion

Shengyuan Zhang<sup>1</sup> An Zhao<sup>1</sup> Ling Yang<sup>3</sup> Zejian Li<sup>2,\*</sup> Chenye Meng<sup>1</sup>  
Haoran Xu<sup>4</sup> Tianrun Chen<sup>1</sup> AnYang Wei<sup>4</sup> Perry Pengyun GU<sup>4</sup> Lingyun Sun<sup>1</sup>

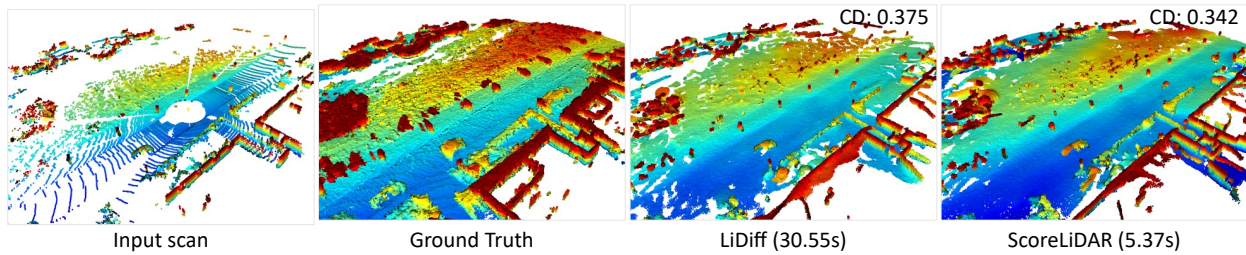
<sup>1</sup> College of Computer Science and Technology, Zhejiang University <sup>2</sup> School of Software Technology, Zhejiang University

<sup>3</sup> Peking University <sup>4</sup> Zhejiang Green Zhixing Technology co., ltd

<sup>1,2</sup>{zhangshengyuan, zhaoan040113, zejianlee, mengcy, tianrun.chen, sunly}@zju.edu.cn

<sup>3</sup>{yangling0818}@163.com <sup>4</sup>{Haoran.Xu5, weianyang, gupengyun}@geely.com \*Corresponding author

## SemanticKITTI



## KITT360

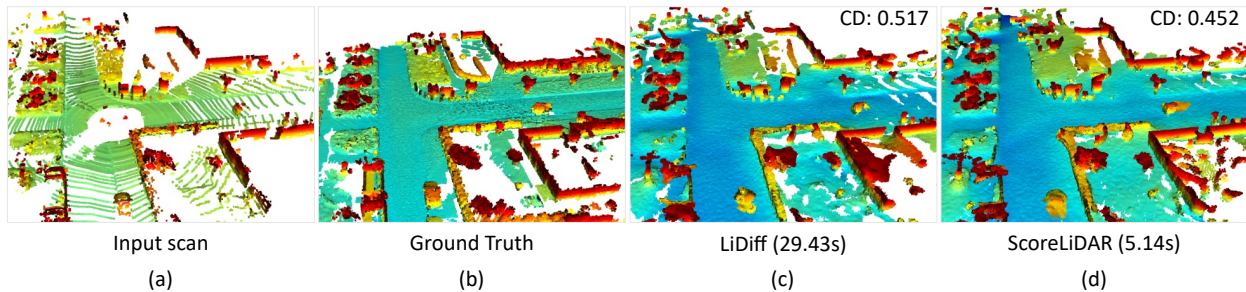


Figure 1. A demonstration of the LiDAR scene completion examples. Given a sparse LiDAR scan in (a), the model aims to recover the ground-truth dense scene as in (b). In these examples, scans are from SemanticKITTI [1] and KITT360 [16] dataset. In both cases, LiDiff [23], a SOTA LiDAR scene completion method, requires about 30 seconds as in (c). In comparison, our proposed ScoreLiDAR takes only about 5 seconds in (d), achieving over 5x speedup with improved completion quality indicated by lower Chamfer Distance (CD).

## Abstract

Diffusion models have been applied to 3D LiDAR scene completion due to their strong training stability and high completion quality. However, the slow sampling speed limits the practical application of diffusion-based scene completion models since autonomous vehicles require an efficient perception of surrounding environments. This paper proposes a novel distillation method tailored for 3D LiDAR scene completion models, dubbed **ScoreLiDAR**, which achieves efficient yet high-quality scene completion. **ScoreLiDAR** enables the distilled model to sample in significantly fewer steps after distillation. To improve completion quality, we also introduce a novel **Structural Loss**, which encourages the distilled model to capture the geometric structure of the 3D LiDAR scene. The loss contains

a scene-wise term constraining the holistic structure and a point-wise term constraining the key landmark points and their relative configuration. Extensive experiments demonstrate that **ScoreLiDAR** significantly accelerates the completion time from 30.55 to 5.37 seconds per frame ( $>5\times$ ) on SemanticKITTI and achieves superior performance compared to state-of-the-art 3D LiDAR scene completion models. Our model and code are publicly available on <https://github.com/happyw1nd/ScoreLiDAR>.

## 1. Introduction

Recognizing the surrounding environment accurately and efficiently using onboard sensors is crucial for the safe operation of autonomous vehicles [13, 14]. Among different types of sensors, 3D LiDAR has become one of the most

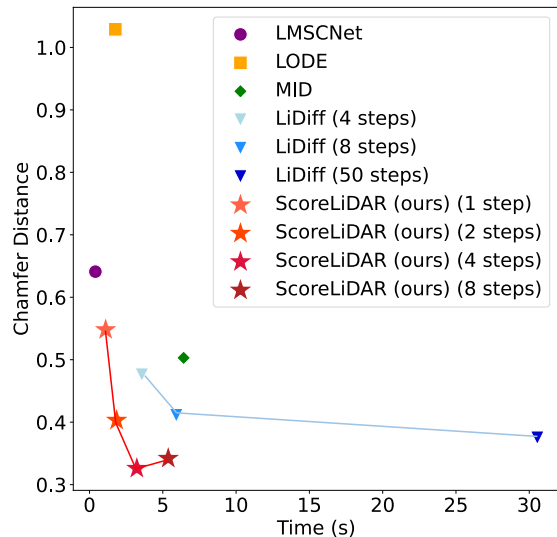


Figure 2. A visualization of LiDAR scene completion performances with different models on SemanticKITTI [1] dataset. Generally, our proposed ScoreLiDAR achieves better scene completion performance and speed trade-off.

widely adopted sensors due to its broader detection range and higher detection precision [21, 23]. However, driving scenarios are often complex, and the 3D point clouds collected by LiDAR are typically sparse, particularly in occluded areas [12, 31]. This sparsity causes a decline in the ability to understand 3D scenes [3, 23]. Thus, inferring and completing sparse 3D LiDAR scenes is necessary to provide a dense and more comprehensive scene representation.

Due to the advantages of strong training stability and high-generation quality, existing works utilize diffusion models to complete the 3D LiDAR scenes and achieve outstanding results [22, 23]. However, the diffusion model often requires multiple network iterations to obtain a dense, complete, and high-quality LiDAR scene, which is time-consuming [10, 45]. Autonomous vehicles require fast and efficient perception and recognition of surrounding environments, so the slow sampling speed limits the practical application of diffusion models. Although existing works have proposed acceleration methods for diffusion models [18, 28, 30, 33, 41], due to the differences between 3D LiDAR scenes and image data—where LiDAR scenes often contain complex geometric structure information—these techniques have not been explored in the acceleration of diffusion-based LiDAR scene completion model.

In this work, we propose **ScoreLiDAR**, a novel distillation method tailored for 3D LiDAR scene completion diffusion models, which enables efficient and high-quality scene completion (Fig. 1 and Fig. 2). ScoreLiDAR aims to tackle the unique 3D distribution alignment challenge in LiDAR scene completion. By exploiting a bidirectional gradient guidance mechanism, it allows the student model’s com-

pletion results under sparse point cloud conditions to progressively approach the multi-step iterative reconstruction quality of the teacher model. Given a completed scene generated by the student model, the teacher model predicts the scene’s score as a gradient to enhance realism, while a designed auxiliary model predicts the scene’s another score as a gradient to suppress unrealistic completion. The student model is then updated according to the difference between these two gradients. This update drives the completion process toward more structurally coherent LiDAR scenes. Finally, we introduce a **Structural Loss** consisting of a scene-wise term and a point-wise term constraining the key landmark points and their relative configuration. Prior studies [15, 19, 34, 35] demonstrated that the bidirectional gradient guidance mechanism can effectively accelerate 3D rendering speed. In this work, we argue that a similar approach can be directly applied to the distillation of diffusion-based LiDAR scene completion models.

Our contribution can be summarized as follows: (1) We propose **ScoreLiDAR**, a novel distillation method tailored for diffusion-based 3D LiDAR scene completion models, which achieves efficient scene completion. (2) We introduce a **Structural Loss** to effectively capture the geometric structure information of 3D point clouds during the distillation process, which ensures high-quality scene completion. (3) Extensive experiments show that ScoreLiDAR enables fast and efficient scene completion while achieving optimal generation quality compared to the existing models.

## 2. Related work

**3D LiDAR Scene completion** 3D LiDAR scene completion refers to recovering a complete scene from a sparse, incomplete LiDAR scan in applications such as autonomous driving [31, 38]. Current mainstream LiDAR scene completion methods include depth completion-based and Signed Distance Field (SDF)-based approaches. Depth completion-based methods aim to recover dense depth maps from sparse depth measurements [8, 36, 39]. These methods typically leverage deep learning techniques [4, 6] and can also incorporate guidance from RGB images to achieve higher-quality completion results [27, 42, 44]. SDF-based methods represent scenes as voxel grids, with the core idea of using signed distance fields to complete sparse LiDAR scenes [12, 31]. These methods are constrained by voxel resolution, making them prone to losing details within the scene [5, 23]. In addition, some methods introduce semantic information to enhance LiDAR scene completion [26, 37]. These methods can generate dense and complete scenes while providing semantic labels for each point, leading to broader application potential [32, 40].

**Diffusion-based 3D LiDAR scene completion** Due to the strong training stability and high generation quality of

diffusion models, many methods leverage diffusion models for LiDAR scene completion tasks [3, 11, 22–24]. The work of Lee *et al.* [11] is the first to apply diffusion models at the scene scale for LiDAR scene completion, enabling the generation of realistic scenes conditioned on partial observations from sparse point clouds. Similarly, R2DM [22] utilizes diffusion models based on distance and reflectance intensity image representations to generate various high-fidelity 3D LiDAR scenes. LiDiff [23] indicates that adding noise to point cloud data at the scene scale leads to a loss of detail. Therefore, LiDiff proposes operating directly on individual points and redefines the noise schedule and denoising processes to generate scenes with richer detail. Based on LiDiff, DiffSSC [3] further performs semantic scene completion tasks by implementing denoising and segmentation separately in both the point and semantic spaces. Moreover, LiDMs [24] constructs the pipeline from the perspectives of pattern realism, geometric realism, and object realism, achieving generation under different conditions.

Due to the slow sampling inherent in diffusion models, diffusion-based 3D LiDAR scene completion suffers from slow inference, hindering its application in autonomous vehicles. Thus, this paper proposes a distillation method for diffusion-based 3D LiDAR scene completion models to achieve faster and higher-quality completion.

### 3. Preliminary

#### 3.1. Brief introduction of diffusion models

The diffusion models have two processes: forward diffusion and reverse denoising process [9, 29]. In the forward diffusion process, given the data  $\mathbf{x}^0 \sim q(\mathbf{x})$  from the training distribution, the diffusion model adds different scales of noise to  $\mathbf{x}^0$  according to different timesteps  $t \in [1, T]$  to obtain noisy data  $\{\mathbf{x}^1, \mathbf{x}^2, \dots, \mathbf{x}^T\}$ . When  $T$  is large enough,  $\mathbf{x}^T$  approaches to standard Gaussian distribution, namely,  $q(\mathbf{x}^T) \approx \mathcal{N}(0, I)$ . This process is parameterized by a series of predefined noise factors  $\beta^t$ . By defining  $\alpha^t = 1 - \beta^t$ , the diffusion process is expressed as [9]:

$$\mathbf{x}^t = \sqrt{\bar{\alpha}^t} \mathbf{x}^0 + \sqrt{1 - \bar{\alpha}^t} \boldsymbol{\epsilon}^t \quad (1)$$

Here  $\bar{\alpha}^t = \prod_{s=1}^t \alpha^s$ ,  $p(\mathbf{x}^t | \mathbf{x}^0) = \mathcal{N}(\sqrt{\bar{\alpha}^t}, (1 - \bar{\alpha}^t)I)$ .

During the training, the diffusion model tries to predict the added noise at different timesteps  $t$ . Given the input  $\mathbf{x}^0$  and the condition  $c$  (optional), the noisy data  $\mathbf{x}^t$  can be calculated by Eq. (1). The diffusion model  $\epsilon_\theta$  predicts the noise according to  $\mathbf{x}^t, c, t$  and is then optimized by calculating the  $\ell_2$  loss between the predicted and the real noise.

$$\mathcal{L}_{DM} = \mathbb{E}_{t, \epsilon} [\|\epsilon - \epsilon_\theta(\mathbf{x}^t, c, t)\|^2] \quad (2)$$

Here  $\theta$  is the trainable parameter of  $\epsilon_\theta$ .

In the reverse denoising process, the diffusion model starts from the timestep  $T$  and progressively removes the

predicted noise to obtain a generated sample. The process of denoising  $\mathbf{x}^t$  to obtain  $\mathbf{x}^{t-1}$  can be written as in [9]

$$\mathbf{x}^{t-1} = \frac{1}{\sqrt{\alpha^t}} \left( \mathbf{x}^t - \frac{1 - \alpha^t}{\sqrt{1 - \bar{\alpha}^t}} \epsilon_\theta(\mathbf{x}^t, c, t) \right) + \sigma^t \mathbf{z} \quad (3)$$

Here  $\mathbf{z} \sim \mathcal{N}(0, I)$ . In this process, the number of required inference steps varies depending on different sampling methods. For instance, DDPM [9] requires 1000 timesteps, while DDIM [29] and DPM solver [17] can reduce this to no more than 100 timesteps.

#### 3.2. 3D LiDAR scene completion diffusion models

The 3D LiDAR scene completion diffusion models take the incomplete scan  $\mathcal{P} = \{\mathbf{p}_1, \mathbf{p}_2, \dots, \mathbf{p}_N\}$  ( $p_i \in \mathbb{R}^3$ ) and try to generate the complete scene  $\mathcal{G}^0 = \{\mathbf{p}_1^0, \mathbf{p}_2^0, \dots, \mathbf{p}_M^0\}$ . Given the input LiDAR scan  $\mathcal{P}$  and ground truth  $\mathcal{G}$ , a diffusion model can be trained to perform 3D LiDAR scene completion. The noisy scene  $\mathcal{G}^t$  at timestep  $t$  is calculated from the ground truth  $\mathcal{G}$  at point level [3, 23],

$$\mathbf{p}_m^t = \sqrt{\bar{\alpha}^t} \mathbf{p}_m + \sqrt{1 - \bar{\alpha}^t} \boldsymbol{\epsilon}^t, \forall \mathbf{p}_m \in \mathcal{G} \quad (4)$$

Here  $\mathcal{G}^t = \{\mathbf{p}_1^t, \mathbf{p}_2^t, \dots, \mathbf{p}_M^t\}$ . Because the LiDAR point cloud is sparse, the noisy data retains very little information about the original data. To generate more realistic point cloud scenes, the LiDAR scan  $\mathcal{P}$  can be used as a condition of the diffusion model [23]. In this case, the training loss of the diffusion model is given by:

$$\mathcal{L}_{DM} = \mathbb{E}_{t, \epsilon} [\|\epsilon - \epsilon_\theta(\mathcal{G}^t, \mathcal{P}, t)\|^2] \quad (5)$$

Then, as described in Sec. 3.1, the completed scene  $\mathcal{G}^0$  can be generated by progressive denoising from  $\mathcal{G}^T$ . Because the scale of the LiDAR scene is large and the data range is different across different point cloud axes, directly normalizing the entire scene compresses the data into a smaller range, which potentially leads to the loss of critical details [3, 23]. To solve this issue, LiDiff [23] modifies the diffusion process by adding a local noise offset to each point  $\mathbf{p}_m$ , gradually perturbing the point cloud at each timestep. For Eq. (1),  $\mathbf{x}^0$  is set to 0, and  $\mathbf{x}^t$  is added to each point  $\mathbf{p}_m$ ,

$$\mathbf{p}_m^t = \mathbf{p}_m + \left( \sqrt{\bar{\alpha}^t} \mathbf{0} + \sqrt{1 - \bar{\alpha}^t} \boldsymbol{\epsilon}^t \right) = \mathbf{p}_m + \sqrt{1 - \bar{\alpha}^t} \boldsymbol{\epsilon}^t \quad (6)$$

Due to this special case, the initial noisy scene  $\mathcal{G}^T$  cannot directly start from standard Gaussian noise in the sampling process. Instead, the LiDAR scan  $\mathcal{P}$  is used to obtain  $\mathcal{G}^T$  [23]. Firstly, given the initial incomplete scan  $\mathcal{P}$ , the number of the point clouds is increased by duplicating the original points  $K$  times and getting the pseudo dense scan  $\mathcal{P}^* = \{\mathbf{p}_1^*, \mathbf{p}_2^*, \dots, \mathbf{p}_M^*\}$ , where we assume  $M = KN$ . Then, we calculate the noisy point cloud  $\mathcal{P}^T$  by Eq. (6). As  $\mathcal{P}^T$  is noisy enough, it can be regarded as  $\mathcal{G}^T$  during the training. After that, a step-by-step denoising process is applied to obtain the completed scene  $\mathcal{G}^0$ .

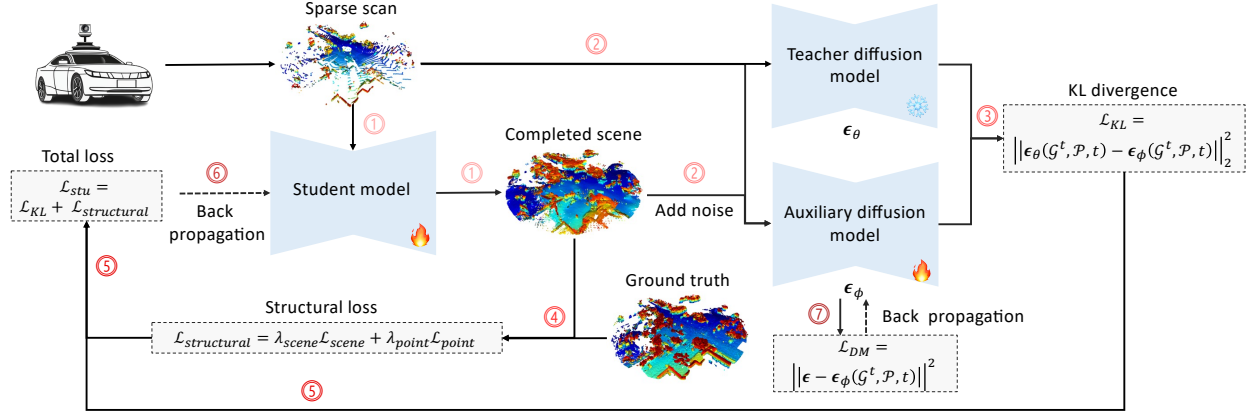


Figure 3. The overall structure of ScoreLiDAR. (1) The student model generates the completed scene based on the sparse scan. (2) The sparse scan and noisy completed scene are input to  $\epsilon_\theta$  and  $\epsilon_\phi$ . (3) The predicted score of  $\epsilon_\theta$  and  $\epsilon_\phi$  are used to calculate the KL divergence. (4) Structural loss is calculated based on the completed scene and the ground truth. (5) The total loss is calculated with KL divergence and structural loss. (6) The student model is optimized according to the total loss. (7) The diffusion model  $\epsilon_\phi$  is updated with the completed scene.

## 4. Method

Our goal is to distill a pre-trained 3D LiDAR scene completion diffusion model into a student model with significantly fewer sampling steps, enabling efficient and high-quality scene completion. Firstly, we introduce the distillation method tailored for 3D LiDAR scene completion diffusion models in Sec. 4.1. Then, we introduce the structural loss to improve the distillation process with both scene-wise loss and point-wise loss in Sec. 4.2. Finally, we describe the optimization procedure of ScoreLiDAR in Sec. 4.3. The overall structure of ScoreLiDAR is shown in Fig. 3.

### 4.1. Distillation for 3D LiDAR scene completion

Ideally, the final student model would achieve completion results comparable to those of the teacher model at a faster speed. In the 3D LiDAR scene completion scenario, let  $q^0$  be the distribution of the ground truth  $\mathcal{G}$ , and  $\epsilon_\theta$  be the pre-trained scene completion diffusion model whose multi-step generated distribution approximates  $q^0$ . Let  $G_{stu}$  be the student model that can perform efficient LiDAR scene completion with the generated distribution  $p_G^0$ . ScoreLiDAR aims to minimize the KL divergence between the distribution of the teacher model and the generated distribution of the student model [35, 43].

$$\min_{\eta} D_{KL}(p_G^0(\mathcal{G}^0; \eta) \| q^0(\mathcal{G}^0)) \quad (7)$$

Here  $\mathcal{G}^0$  is the completed scene generated by one-step sampling of  $G_{stu}$  conditioned on  $\mathcal{P}$ , for simplicity, we omit  $\mathcal{P}$  when representing the distribution,  $\eta$  is the trainable parameter of  $G_{stu}$ . However, the high-density regions of  $q^0$  are sparse in the data space, so it is hard to directly solve Eq. (7). According to Theorem 1 in [35], we expand the optimization problems in Eq. (7) by minimizing the KL divergence

between two distributions at different noise levels as:

$$\min_{\eta} \mathcal{L}_{KL} = \mathbb{E}_{t, \epsilon} [D_{KL}(p_G^t(\mathcal{G}^t) \| q^t(\mathcal{G}^t))] \quad (8)$$

Here  $t$  is the timestep controlling the noise level,  $\epsilon$  is random noise, and  $\mathcal{G}^t = \{\mathbf{p}_1^t, \mathbf{p}_2^t, \dots, \mathbf{p}_M^t\}$  is the noisy version of the completed scene  $\mathcal{G}^0$  at timestep  $t$ . The gradient of  $G_{stu}$  in Eq. (8) is approximated by

$$\begin{aligned} & \nabla_{\eta} D_{KL}(p_G^t(\mathcal{G}^t) \| q^t(\mathcal{G}^t)) \\ &= \mathbb{E}_{t, \epsilon} [\nabla_{\mathcal{G}^t} \log p_G^t(\mathcal{G}^t) - \nabla_{\mathcal{G}^t} \log q^t(\mathcal{G}^t)] \frac{\partial \mathcal{G}^t}{\partial \eta} \quad (9) \end{aligned}$$

We use the pre-trained diffusion model  $\epsilon_\theta$  to approximate  $-\sqrt{1 - \bar{\alpha}^t} \nabla_{\mathcal{G}^t} \log q^t(\mathcal{G}^t)$  as discussed in Section 5 of Supplementary Materials (Sec S5). Similarly,  $-\sqrt{1 - \bar{\alpha}^t} \nabla_{\mathcal{G}^t} \log p_G^t(\mathcal{G}^t)$  is approximated by an auxiliary diffusion model  $\epsilon_\phi$ , which is independently trained with the denoising loss Eq. (5) on generated samples  $\mathcal{G}^0$ . Then, with the simplification as in [9, 35], the  $\mathcal{L}_{KL}$  is estimated by

$$\mathcal{L}_{KL} \approx \mathbb{E}_{t, \epsilon} [\|\epsilon_\theta(\mathcal{G}^t, \mathcal{P}, t) - \epsilon_\phi(\mathcal{G}^t, \mathcal{P}, t)\|_2^2] \quad (10)$$

Thus, the gradient in Eq. (9) is approximated by

$$\begin{aligned} & \nabla_{\eta} D_{KL}(p_G^t(\mathcal{G}^t) \| q^t(\mathcal{G}^t)) \\ & \approx \mathbb{E}_{t, \epsilon} [\epsilon_\theta(\mathcal{G}^t, \mathcal{P}, t) - \epsilon_\phi(\mathcal{G}^t, \mathcal{P}, t)] \frac{\partial \mathcal{G}^t}{\partial \eta} \quad (11) \end{aligned}$$

Intuitively, the orientation of  $-\epsilon_\theta(\mathcal{G}^t, \mathcal{P}, t)$  points to the pre-trained distribution, while that of  $-\epsilon_\phi(\mathcal{G}^t, \mathcal{P}, t)$  points to the student model's generative distribution. Thus, decending along the bidirectional gradient  $[\epsilon_\theta(\mathcal{G}^t, \mathcal{P}, t) - \epsilon_\phi(\mathcal{G}^t, \mathcal{P}, t)]$  updates the student model's generative distribution toward the pre-trained distribution, achieving a more accurate completion. Sec S2.1 discusses the efficiency of the distillation process.

## 4.2. Structural loss

Although the distillation process in Sec. 4.1 is effective in training models [18, 45], we found that directly applying it to LiDAR scene completion diffusion models leads to loss of local details and reduced realism. This is because the point cloud in LiDAR scenes includes complex geometric information that is not explicitly captured by diffusion models. Thus, we introduce a structural loss to further refine the distillation process and improve the completion quality. This structural loss includes scene-wise loss and point-wise loss and can help the student model effectively capture geometric structure information of the 3D point clouds.

**Scene-wise loss.** In the distillation process mentioned in Sec. 4.1, the gradient  $\nabla_{\eta} D_{\text{KL}}$  in Eq. (11) is well-defined when  $t \gg 0$ , *i.e.* the generated samples are totally disturbed by Gaussian noise. However,  $\nabla_{\eta} D_{\text{KL}}$  becomes unreliable when  $t$  is small [43, 45]. This is because the student model often generates subpar results at the early stage due to the complexity of the point cloud data. It is easy for the noisy generated samples to lie outside the training distribution of the teacher model, causing the unreliable network prediction of the teacher model [43, 45].

To solve this issue, we introduce the scene-wise loss, which minimizes the distance between the ground truth scene  $\mathcal{G}$  and the completed scene  $\mathcal{G}^0$ ,

$$\mathcal{L}_{\text{scene}} = \frac{1}{|\mathcal{G}^0|} \sum_{\mathbf{p}_i^0 \in \mathcal{G}^0} \min_{\mathbf{p} \in \mathcal{G}} \|\mathbf{p}_i^0 - \mathbf{p}\|^2 \quad (12)$$

This loss calculates the mean squared error between each point  $\mathbf{p}_i^0$  in the generated scene  $\mathcal{G}^0$  and its closest corresponding point  $\mathbf{p}$  in the ground truth  $\mathcal{G}$ . It helps the student model capture the holistic structure, which prevents the optimization direction from deviating in the early stages and enhances training stability. The scene-wise loss enables the generated scenes to be closer to the ground truth globally, thereby enhancing the completion quality and fidelity.

**Point-wise loss.** As seen in Eq. (11), the distillation process only constrains the overall distribution of the completed scene, ignoring the relative positions between different points. Directly using the gradient in Eq. (11) to optimize the student model may lead to loss of local details.

Thus, we introduce the point-wise loss to capture the relative structural information between different points in the 3D LiDAR scene. The point-wise loss calculates the difference between the inter-point distance matrices of the completed scene and the ground truth. Due to the large number of points in the scene, calculating the distance matrix for all points is computationally intensive. Therefore, we select  $n$  key points to compute the distance matrix with  $n \ll M$ . Based on the local geometric features of each point, we

choose key points that are critical for representing the structure of the 3D LiDAR scene. For each point  $\mathbf{p}_i^0$  in the completed scene  $\mathcal{G}^0$ , we find its  $K$ -nearest neighbor, denoted as the set  $\mathcal{K}_i$ . Then we select the key points by calculating their curvature  $\kappa_i$ . The specific steps are as follows:

- Calculate the centroid  $\bar{\mathbf{p}}_i^0$  of the neighborhood  $\mathcal{K}_i$

$$\bar{\mathbf{p}}_i^0 = \frac{1}{K} \sum_{\mathbf{p}_j^0 \in \mathcal{K}_i} \mathbf{p}_j^0 \quad (13)$$

- Calculate the neighborhood covariance matrix  $\mathcal{C}_i$  for  $\bar{\mathbf{p}}_i^0$

$$\mathcal{C}_i = \frac{1}{K} \sum_{\mathbf{p}_j^0 \in \mathcal{K}_i} (\mathbf{p}_j^0 - \bar{\mathbf{p}}_i^0)(\mathbf{p}_j^0 - \bar{\mathbf{p}}_i^0)^T \quad (14)$$

- Perform eigen-decomposition on the covariance matrix  $\mathcal{C}_i$  to obtain the eigenvalues  $\lambda_1 < \lambda_2 < \dots < \lambda_m$ .
- Curvature  $\kappa_i$  can be calculated using the eigenvalues

$$\kappa_i = \frac{\lambda_1}{\sum_{j=1}^m \lambda_j} \quad (15)$$

A larger curvature  $\kappa_i$  indicates greater local shape variation. Those points with great local variation are typically located at corners, edges, or endpoints, which tend to shape the main structure of the scene. Therefore, the top  $n$  points with the highest curvature values are selected as key points.

Given the ground truth  $\mathcal{G} = \{\mathbf{p}_1, \mathbf{p}_2, \dots, \mathbf{p}_M\}$ , we select  $n$  key points ( $n \ll M$ ) from the point cloud to construct the  $n \times n$  distance matrix  $\mathcal{D}$ . The  $d_{ij} \in \mathcal{D}$  represents the pairwise Euclidean Distance between points  $i$  and  $j$ . Then, for completed scene  $\mathcal{G}^0$ , we select  $n$  points that are closest to the key points in  $\mathcal{G}$  as the corresponding key points to obtain the distance matrix  $\mathcal{D}_G$ . Thus, the point-wise loss is calculated by

$$\mathcal{L}_{\text{point}} = \mathbb{E}_{t, \epsilon} [\|\mathcal{D} - \mathcal{D}_G\|_2^2] \quad (16)$$

The point-wise loss can help the student model capture the relative configuration of key points and further enhance the geometric accuracy and detail retention of the completed scene. This ensures that key objects like cars, traffic cones, and walls are better completed, which is crucial for autonomous vehicles to recognize surroundings accurately.

**Structural loss.** The final structural loss of  $G_{stu}$  is

$$\mathcal{L}_{\text{structural}} = \lambda_{\text{scene}} \mathcal{L}_{\text{scene}} + \lambda_{\text{point}} \mathcal{L}_{\text{point}} \quad (17)$$

Here  $\lambda_{\text{scene}}$  and  $\lambda_{\text{point}}$  are the weight of scene-wise loss and point-wise loss.

Model	CD ↓	JSD ↓	EMD ↓	Times (s) ↓
LMSCNet [25]	0.641	0.431	-	0.40
LODE [12]	1.029	0.451	-	1.76
MID [31]	0.503	0.470	-	6.42
PVD [46]	1.256	0.498	-	262.54
LiDiff [23]	0.434	0.444	22.15	30.38
LiDiff (Refined) [23]	0.375	0.416	23.16	30.55
ScoreLiDAR	0.406	0.425	23.14	5.16
ScoreLiDAR (Refined)	0.342	0.399	23.26	5.37

Table 1. The completion performance on the SemanticKITTI dataset. Colors denote the 1st, 2nd, and 3rd best-performing model. The sampling time is estimated based on the official code and the provided checkpoints.

Model	CD ↓	JSD ↓	EMD ↓	Times (s) ↓
LMSCNet [25]	0.979	0.496	-	0.38
LODE [12]	1.565	0.483	-	1.64
MID [31]	0.637	0.476	-	6.23
LiDiff [23]	0.564	0.459	21.98	29.18
LiDiff (Refined) [23]	0.517	0.446	22.96	29.43
ScoreLiDAR	0.472	0.444	22.80	4.98
ScoreLiDAR (Refined)	0.452	0.437	23.02	5.14

Table 2. The completion performance on the KITTI-360 dataset. The meaning of notations is the same as those in Tab. 1.

Model	SemanticKITTI			KITTI360		
	CD ↓	JSD ↓	EMD ↓	CD ↓	JSD ↓	EMD ↓
ScoreLiDAR (Refined)	0.342	0.399	23.26	0.452	0.437	23.02
w/o Structural Loss (Refined)	0.419	0.430	24.61	0.549	0.445	24.56

Table 3. Ablation study of the structural loss.

### 4.3. Optimization procedure

During the training,  $G_{stu}$  and  $\epsilon_\phi$  are initialized with the teacher model  $\epsilon_\theta$  and optimized alternately. The auxiliary diffusion model  $\epsilon_\phi$  is trained on the completed scene of the student model with Eq. (5). As for  $G_{stu}$ , we follow the proposed method to select  $\frac{1}{30}$  of the points from the entire point cloud as key points for calculating the point distance matrix. Then,  $G_{stu}$  is optimized with the following objective

$$\mathcal{L}_{stu} = \mathcal{L}_{KL} + \mathcal{L}_{structural} \quad (18)$$

We set  $\lambda_{scene} = 0.5$  and  $\lambda_{point} = 0.01$  defaultly. The implementation details are provided in Sec S1.

## 5. Experiment

In this part, we conduct a series of experiments to evaluate the effectiveness of the proposed ScoreLiDAR. We compare ScoreLiDAR with advanced models including LMSCNet [25], LODE [12], MID [31], PVD [46] and LiDiff [23]. We first evaluate the performance of ScoreLiDAR in scene

Model	CD ↓	JSD ↓	EMD ↓	Time (s) ↓
LiDiff (50 steps) [23]	0.434	0.444	22.15	30.38
LiDiff (50 steps Refined) [23]	0.375	0.416	23.16	30.55
LiDiff (8 steps) [23]	0.447	0.432	24.90	5.69
LiDiff (8 steps Refined) [23]	0.411	0.406	25.74	5.92
ScoreLiDAR (8 Steps Refined)	0.342	0.399	23.14	5.37
ScoreLiDAR (4 Steps Refined)	0.326	0.386	23.98	3.23
ScoreLiDAR (2 Steps Refined)	0.403	0.379	-	1.85
ScoreLiDAR (1 Steps Refined)	0.548	0.384	-	1.10

Table 4. Ablation study of different sampling steps on the SemanticKITTI dataset.

completion tasks (Sec. 5.1). Secondly, we present the results of ablation studies showing the effectiveness of the structural loss and the performances of ScoreLiDAR given different sampling steps (Sec. 5.2). Finally, we further evaluate ScoreLiDAR with the qualitative analysis (Sec. 5.3).

### 5.1. Scene completion

We validate ScoreLiDAR on SemanticKITTI [1] and KITTI-360 [16] datasets. The existing SOTA LiDAR scene completion model LiDiff [23] is chosen as the teacher model. The student model shares the network architecture with the teacher model and is initialized by the teacher model. Moreover, we also use the refinement network in LiDiff [23] to refine the completed scene generated by the student model. We calculate the Chamfer Distance (CD) [2], the Jensen-Shannon Divergence (JSD) [20] and the Earth Mover’s Distance (EMD) [7] to evaluate the similarity between the completed scene and the ground truth. The smaller the value of these metrics, the closer the completed scene is to the ground truth.

Tab. 1 shows that ScoreLiDAR achieves the optimal performance compared to the existing models in most cases on the SemanticKITTI dataset. Compared to the SOTA method LiDiff [23] with refinement, which takes 30.55 seconds to complete a scene, ScoreLiDAR completes a scene in just 5.47 seconds (fivefold speedup) yet with 8% improvement in CD, 4% in JSD, and comparable results in EMD. The reason for the increase in EMD after refinement is the mismatch in the points numbers between  $\mathcal{G}^0$  and  $\mathcal{G}$ , leading to a higher matching cost. The refinement process optimizes local details but alters the global point cloud distribution, extending the overall transfer path. Although LMSCNet [25] and LODE [12] have faster completion speeds, their completion quality is significantly lower. The performance of ScoreLiDAR outperforms the teacher model LiDiff [23]. This is because ScoreLiDAR introduces a structural loss with scene-wise term and point-wise term, enabling the student model to effectively capture geometric structure information within LiDAR point cloud data during training. Results on KITTI-360 are shown in Tab. 2. ScoreLiDAR also achieves optimal performance in most cases and boasts

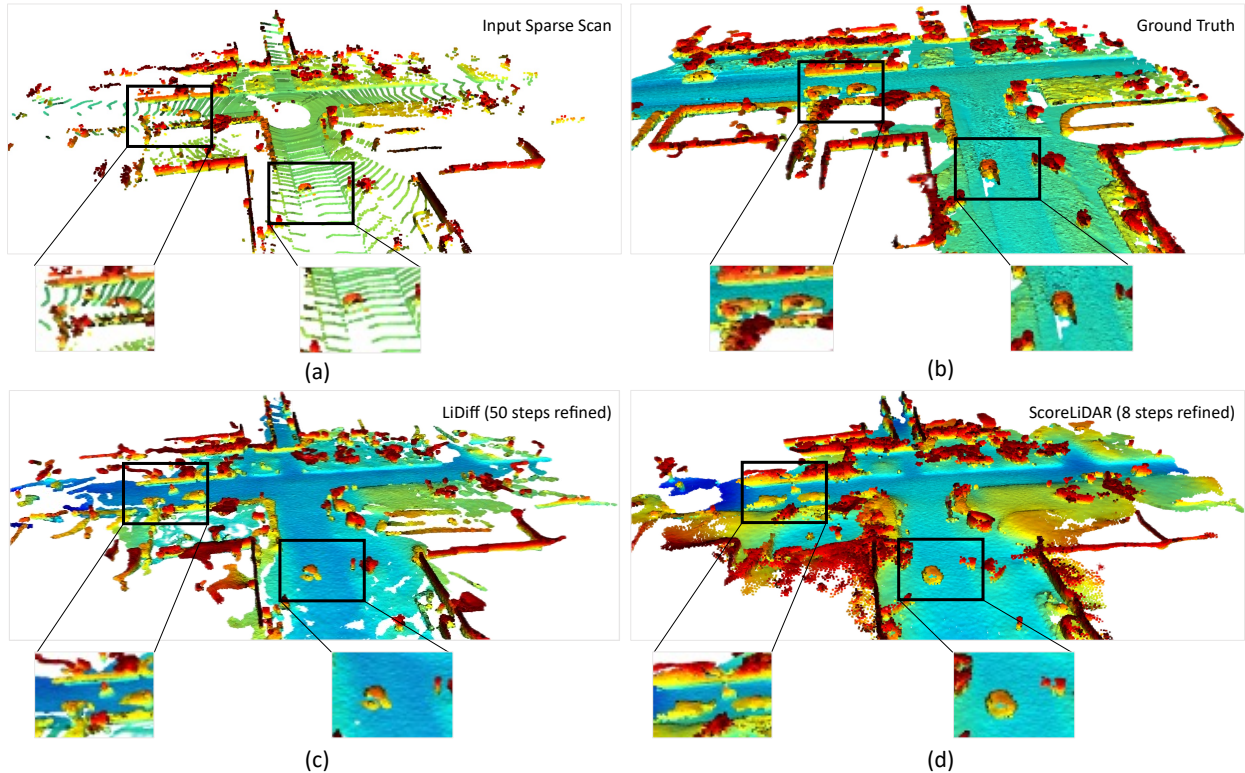


Figure 4. Qualitative results on KITTI-360. ScoreLiDAR achieves better completion than LiDiff [24] with fewer sampling steps.

a fivefold speedup with 12% improvement in CD and 2% in JSD compared to LiDiff [23].

## 5.2. Ablation study

In this part, we conduct the ablation study to verify the effectiveness of the structural loss in the training of the proposed ScoreLiDAR. We compared the scene completion performances of the proposed ScoreLiDAR with a variant that does not incorporate structural loss. The results are shown in Tab. 3. The results show that the variant without structural loss exhibits lower performance in scene completion on both datasets. However, after considering the structural loss, the performance of ScoreLiDAR improves significantly, which achieves better performance on all metrics. This supports our discussion in Sec. 4.3, incorporating structural loss enables the student model to capture the geometric structure feature of 3D point clouds, thereby facilitating the effective distillation of the student model.

Furthermore, we compared the scene completion performance of ScoreLiDAR with different sampling steps, and the results are shown in Tab. 4. It can be observed that as the sampling steps decrease from 8 to 1, the time required for ScoreLiDAR to complete a scene also decreases, with single-step sampling allowing a scene to be completed in only 1.1 seconds. With 8-step and 4-step sampling, ScoreLiDAR performs better than LiDiff. All metrics decay at

2-step and 1-step sampling, but in JSD ours still performs better than LiDiff. In summary, although the quality of scene completion decreases as the sampling steps are reduced, it still maintains performance comparable to or better than the existing model, achieving better performance and speed trade-off as in Fig. 2.

In addition, we compared the ablation results about the terms of structural loss, different keypoint selection methods, varying numbers of keypoints, and different values of  $\lambda_{scene}$  and  $\lambda_{point}$  and so on, which are presented in Sec. S4.

## 5.3. Qualitative analysis

Fig. 4 shows the completed scenes by our proposed ScoreLiDAR and LiDiff [23] on KITTI-360. ScoreLiDAR achieves completion results with higher quality and greater fidelity in less time compared to LiDiff. For example, as shown in Fig. 4(c), in the left region, there should be two vehicles as in Fig. 4(b), but the scene completed by LiDiff only contains one. In contrast, the scene completed by ScoreLiDAR with only 8 steps in Fig. 4(d) completes two vehicles and has clearer and more complete vehicle structures, making it closer to the ground truth.

To further demonstrate the effectiveness of ScoreLiDAR and the structural loss, we calculate the distance between the points in the completed scene and their corresponding points in the ground truth to evaluate the overall differ-

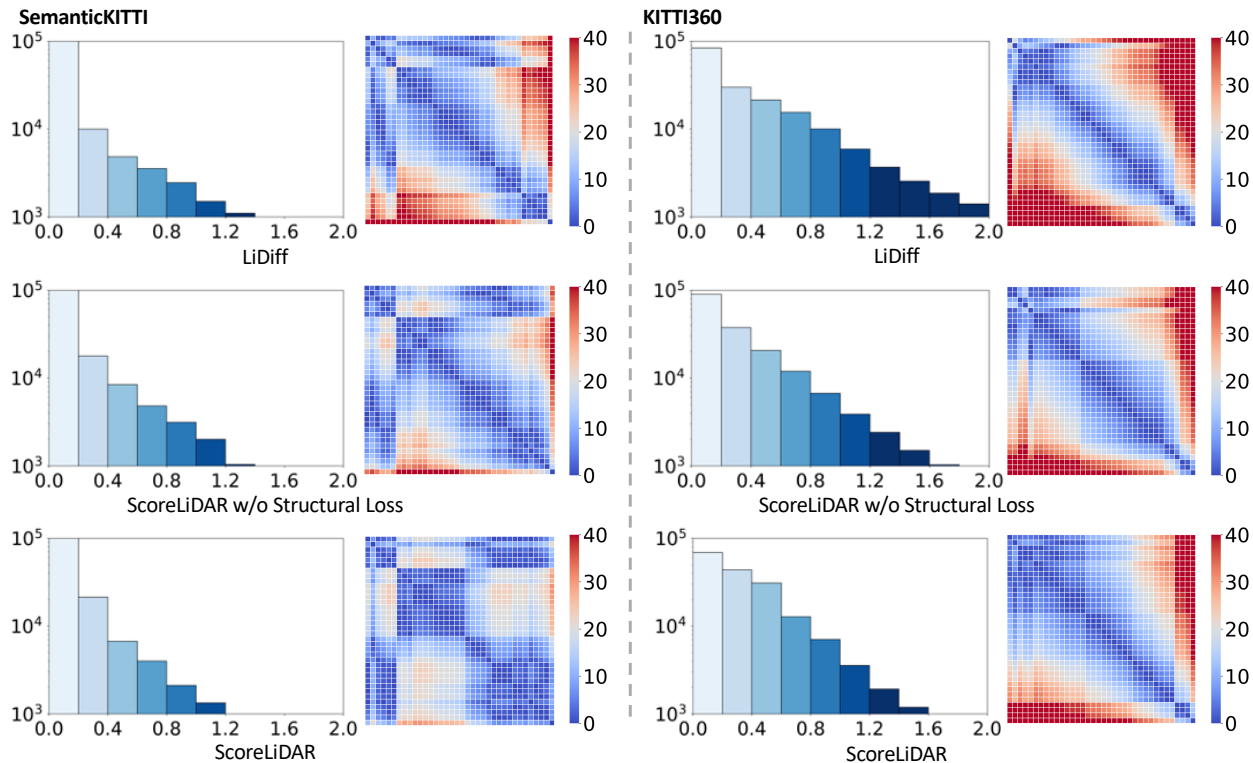


Figure 5. The qualitative analysis of structural loss. The bar chart shows the distribution of distances between corresponding points in the completed and ground truth scenes. A higher number of points with smaller distances demonstrates that the completed scene is closer to the ground truth. The heatmap represents the difference in distance matrices between the completed scene and the ground truth scene. Smaller values on the heatmap indicate that the completed scene is closer to the ground truth.

ence. We display the calculated results in bar chart in Fig. 5. ScoreLiDAR has the highest number of points with smaller distances to their corresponding points in the ground truth. The results show that the scenes completed by ScoreLiDAR are closer to the ground truth overall, demonstrating higher fidelity. Moreover, we selected 36 corresponding key points from the ground truth and the completed scene using the method described in Sec. 4.2 and calculated the point distance matrices  $\mathcal{D}$  and  $\mathcal{D}_G$ . We then visualized the difference between  $\mathcal{D}$  and  $\mathcal{D}_G$  as a heatmap. As shown in Fig. 5, on both datasets, the difference of point distance matrix between the completed scene of LiDiff [23] and the ground truth is the largest, followed by the ScoreLiDAR variant without the structural loss and the smallest difference is achieved by ScoreLiDAR. This also indicates that the scene completed by ScoreLiDAR is closer to the ground truth.

We also conduct a user study to evaluate the performance of ScoreLiDAR. We used ScoreLiDAR and LiDiff [23] to complete scenes based on the same input scans and asked users to choose the scene they believed was closer to the ground truth. ScoreLiDAR received a 65% user preference over LiDiff [23]. This indicates that the detail and fidelity of the scenes completed by ScoreLiDAR more closely resemble the ground truth for most users. The details of the

user study are shown in SM.

## 6. Conclusion

**Summary.** This paper proposes ScoreLiDAR, a novel distillation method tailored for 3D LiDAR scene completion diffusion models based on a bidirectional gradient guidance mechanism. By introducing the structural loss with scene-wise term and point-wise term, ScoreLiDAR trains the student model to effectively capture the holistic structure and the relative configuration of key points and achieve efficient and high-quality scene completion.

**Limitations.** While ScoreLiDAR achieves efficient, high-quality LiDAR scene completion, its performance is constrained by the teacher model. As the performance of the teacher model improves, so does the capability of the student model. Although we conducted a preliminary experiment on the semantic scene completion in the Supplementary Material, a more comprehensive exploration is still required. Thus, further exploration is required to find a more effective method to improve the training process of ScoreLiDAR and avoid the limitations of the teacher model, achieving more efficient semantic LiDAR scene completion.

## Acknowledgement

This paper is supported by Provincial Key Research and Development Plan of Zhejiang Province under No. 2024C01250(SD2), National Natural Science Foundation of China (Grant No. 62006208) and Dream Set Off - Kunpeng & Ascend Seed Program.

## References

- [1] 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*, pages 9297–9307, 2019. 1, 2, 6
- [2] M Akmal Butt and Petros Maragos. Optimum Design of Chamfer Distance Transforms. *IEEE TIP*, 7(10):1477–1484, 1998. 6
- [3] Helin Cao and Sven Behnke. DiffSSC: Semantic LiDAR Scan Completion Using Denoising Diffusion Probabilistic Models. *arXiv preprint arXiv:2409.18092*, 2024. 2, 3
- [4] Nathaniel Chodosh, Chaoyang Wang, and Simon Lucey. Deep Convolutional Compressed Sensing For Lidar Depth Completion. In *ACCV*, pages 499–513, 2019. 2
- [5] Angela Dai, Daniel Ritchie, Martin Bokeloh, Scott Reed, Jürgen Sturm, and Matthias Nießner. Scancomplete: Large-Scale Scene Completion And Semantic Segmentation For 3d Scans. In *CVPR*, pages 4578–4587, 2018. 2
- [6] Abdelrahman Eldesokey, Michael Felsberg, and Fahad Shahbaz Khan. Confidence Propagation Through CNNs For Guided Sparse Depth Regression. *IEEE TPAMI*, 42(10):2423–2436, 2019. 2
- [7] Haoqiang Fan, Hao Su, and Leonidas J Guibas. A Point Set Generation Network for 3d Object Reconstruction From a Single Image. In *CVPR*, pages 605–613, 2017. 6
- [8] Chen Fu, Christoph Mertz, and John M Dolan. Lidar and Monocular Camera Fusion: On-road Depth Completion For Autonomous Driving. In *IEEE Intell. Transp. Syst. Conf.*, pages 273–278, 2019. 2
- [9] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising Diffusion Probabilistic Models. *NeurIPS*, 33:6840–6851, 2020. 3, 4
- [10] Dongjun Kim, Chieh-Hsin Lai, Wei-Hsiang Liao, Naoki Murata, Yuhta Takida, Toshimitsu Uesaka, Yutong He, Yuki Mitsufuji, and Stefano Ermon. Consistency Trajectory Models: Learning Probability Flow ODE Trajectory of Diffusion. In *ICLR*, 2024. 2
- [11] 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. 3
- [12] Pengfei Li, Ruowen Zhao, Yongliang Shi, Hao Zhao, Jirui Yuan, Guyue Zhou, and Ya-Qin Zhang. Lode: Locally Conditioned Eikonal Implicit Scene Completion From Sparse Lidar. In *IEEE Intl. Conf. on Robotics and Automation*, pages 8269–8276, 2023. 2, 6
- [13] You Li, Julien Moreau, and Javier Ibanez-Guzman. Emergent Visual Sensors For Autonomous Vehicles. *IEEE Transactions on Intelligent Transportation Systems*, 24(5):4716–4737, 2023. 1
- [14] Yiming Li, Zhiding Yu, Christopher Choy, Chaowei Xiao, Jose M Alvarez, Sanja Fidler, Chen Feng, and Anima Anandkumar. Voxformer: Sparse Voxel Transformer For Camera-based 3d Semantic Scene Completion. In *CVPR*, pages 9087–9098, 2023. 1
- [15] Yixun Liang, Xin Yang, Jiantao Lin, Haodong Li, Xiaogang Xu, and Yingcong Chen. Luciddreamer: Towards High-fidelity Text-to-3d Generation Via Interval Score Matching. In *CVPR*, pages 6517–6526, 2024. 2
- [16] Yiyi Liao, Jun Xie, and Andreas Geiger. Kitti-360: A Novel Dataset And Benchmarks For Urban Scene understanding in 2d And 3d. *IEEE TPAMI*, 45(3):3292–3310, 2022. 1, 6
- [17] Cheng Lu, Yuhao Zhou, Fan Bao, Jianfei Chen, Chongxuan Li, and Jun Zhu. Dpm-solver: A Fast Ode Solver For Diffusion Probabilistic Model Sampling In Around 10 Steps. *NeurIPS*, 35:5775–5787, 2022. 3
- [18] Weijian Luo, Tianyang Hu, Shifeng Zhang, Jiacheng Sun, Zhenguo Li, and Zhihua Zhang. Diff-Instruct: A Universal Approach for Transferring Knowledge From Pre-trained Diffusion Models. *NeurIPS*, 36:76525–76546, 2023. 2, 5
- [19] Zhiyuan Ma, Yuxiang Wei, Yabin Zhang, Xiangyu Zhu, Zhen Lei, and Lei Zhang. Scaledreamer: Scalable Text-to-3d Synthesis With Asynchronous Score Distillation. In *ECCV*, pages 1–19, 2024. 2
- [20] María Luisa Menéndez, JA Pardo, L Pardo, and MC Pardo. The Jensen-Shannon Divergence. *Journal of the Franklin Institute*, 334(2):307–318, 1997. 6
- [21] Amir Meydani. State-of-the-Art Analysis Of The Performance Of The Sensors Utilized In Autonomous Vehicles In Extreme Conditions. In *Proceedings of the International Conference on Artificial Intelligence and Smart Vehicles*, pages 137–166, 2023. 2
- [22] Kazuto Nakashima and Ryo Kurazume. Lidar Data Synthesis With Denoising Diffusion Probabilistic Models. In *IEEE Intl. Conf. on Robotics and Automation*, pages 14724–14731, 2024. 2, 3
- [23] Lucas Nunes, Rodrigo Marcuzzi, Benedikt Mersch, Jens Behley, and Cyrill Stachniss. Scaling Diffusion Models To Real-World 3D LiDAR Scene Completion. In *CVPR*, pages 14770–14780, 2024. 1, 2, 3, 6, 7, 8
- [24] Haoxi Ran, Vitor Guizilini, and Yue Wang. Towards Realistic Scene Generation With LiDAR Diffusion Models. In *CVPR*, pages 14738–14748, 2024. 3, 7
- [25] Luis Roldao, Raoul de Charette, and Anne Verroust-Blondet. Lmscnet: Lightweight Multiscale 3d Semantic Completion. In *Proceedings of International Conference on 3D Vision*, pages 111–119, 2020. 6
- [26] Luis Roldao, Raoul De Charette, and Anne Verroust-Blondet. 3D Semantic Scene Completion: A Survey. *IJCV*, 130(8):1978–2005, 2022. 2
- [27] Kwonyoung Ryu, Kang-il Lee, Jegyeong Cho, and Kuk-Jin Yoon. Scanline Resolution-Invariant Depth Completion Using A Single Image And Sparse LiDAR Point Cloud. *IEEE Rob. Autom. Lett.*, 6(4):6961–6968, 2021. 2
- [28] Tim Salimans and Jonathan Ho. Progressive Distillation for Fast Sampling of Diffusion Models. In *ICLR*, 2021. 2

- [29] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising Diffusion Implicit Models. *arXiv preprint arXiv:2010.02502*, 2020. 3
- [30] Yang Song, Prafulla Dhariwal, Mark Chen, and Ilya Sutskever. Consistency Models. In *Int. Conf. Mach. Learn.*, pages 32211–32252, 2023. 2
- [31] Ignacio Vizzo, Benedikt Mersch, Rodrigo Marcuzzi, Louis Wiesmann, Jens Behley, and Cyrill Stachniss. Make It Dense: Self-Supervised Geometric Scan Completion of Sparse 3d Lidar Scans In Large Outdoor Environments. *IEEE Rob. Autom. Lett.*, 7(3):8534–8541, 2022. 2, 6
- [32] Fengyun Wang, Dong Zhang, Hanwang Zhang, Jinhui Tang, and Qianru Sun. Semantic Scene Completion With Cleaner Self. In *CVPR*, pages 867–877, 2023. 2
- [33] Fu-Yun Wang, Ling Yang, Zhaoyang Huang, Mengdi Wang, and Hongsheng Li. Rectified diffusion: Straightness is not your need in rectified flow. *arXiv preprint arXiv:2410.07303*, 2024. 2
- [34] Peihao Wang, Dejia Xu, Zhiwen Fan, Dilin Wang, Sreyas Mohan, Forrest Iandola, Rakesh Ranjan, Yilei Li, Qiang Liu, Zhangyang Wang, et al. Taming Mode Collapse In Score Distillation for Text-to-3d Generation. In *CVPR*, pages 9037–9047, 2024. 2
- [35] Zhengyi Wang, Cheng Lu, Yikai Wang, Fan Bao, Chongxuan Li, Hang Su, and Jun Zhu. ProlificDreamer: High-Fidelity And Diverse Text-to-3D Generation With Variational Score Distillation. *NeurIPS*, 36:8406–8441, 2023. 2, 4
- [36] Cho-Ying Wu and Ulrich Neumann. Scene Completeness-Aware Lidar Depth Completion For Driving Scenario. In *ICASSP*, pages 2490–2494, 2021. 2
- [37] Zhaoyang Xia, Youquan Liu, Xin Li, Xinge Zhu, Yuexin Ma, Yikang Li, Yuenan Hou, and Yu Qiao. Sepnet: Semantic Scene Completion On Point Cloud. In *CVPR*, pages 17642–17651, 2023. 2
- [38] Yuwen Xiong, Wei-Chiu Ma, Jingkang Wang, and Raquel Urtasun. Learning Compact Representations For Lidar Completion and Generation. In *CVPR*, pages 1074–1083, 2023. 2
- [39] Yan Xu, Xinge Zhu, Jianping Shi, Guofeng Zhang, Hujun Bao, and Hongsheng Li. Depth Completion from Sparse Lidar Data With Depth-Normal Constraints. In *ICCV*, pages 2811–2820, 2019. 2
- [40] Yujie Xue, Ruihui Li, Fan Wu, Zhuo Tang, Kenli Li, and Mingxing Duan. Bi-SSC: Geometric-Semantic Bidirectional Fusion for Camera-based 3D Semantic Scene Completion. In *CVPR*, pages 20124–20134, 2024. 2
- [41] Ling Yang, Zixiang Zhang, Zhilong Zhang, Xingchao Liu, Minkai Xu, Wentao Zhang, Chenlin Meng, Stefano Ermon, and Bin Cui. Consistency flow matching: Defining straight flows with velocity consistency. *arXiv preprint arXiv:2407.02398*, 2024. 2
- [42] Yanchao Yang, Alex Wong, and Stefano Soatto. Dense Depth Posterior (DDP) From Single Image And Sparse Range. In *CVPR*, pages 3353–3362, 2019. 2
- [43] Tianwei Yin, Michaël Gharbi, Richard Zhang, Eli Shechtman, Fredo Durand, William T Freeman, and Taesung Park. One-step Diffusion With Distribution Matching Distillation. In *CVPR*, pages 6613–6623, 2024. 4, 5
- [44] Qingyang Yu, Lei Chu, Qi Wu, and Ling Pei. Grayscale And Normal Guided Depth Completion With A Low-cost LiDAR. In *ICIP*, pages 979–983, 2021. 2
- [45] Shengyuan Zhang, Ling Yang, Zejian Li, An Zhao, Chenye Meng, Changyuan Yang, Guang Yang, Zhiyuan Yang, and Lingyun Sun. Distribution Backtracking Builds A Faster Convergence Trajectory for One-step Diffusion Distillation. In *ICLR*, 2025. 2, 5
- [46] Linqi Zhou, Yilun Du, and Jiajun Wu. 3d Shape Generation And Completion Through Point-voxel Diffusion. In *ICCV*, pages 5826–5835, 2021. 6