OLiDM: Object-aware LiDAR Diffusion Models for Autonomous Driving

Tianyi Yan[1,2*], **Junbo Yin**[3*], **Xianpeng Lang**[2], **Ruigang Yang**[4], **Cheng-Zhong Xu**[1], **Jianbing Shen**[1†]

[1]SKL-IOTSC, Computer and Information Science, University of Macau
[2]Li Auto Inc
[3]CEMSE Division, King Abdullah University of Science and Technology
[4]Shanghai Jiao Tong University
{tianyi.yan123, yinjunbocn}@gmail.com, jianbingshen@um.edu.mo

Abstract

To enhance autonomous driving safety in complex scenarios, various methods have been proposed to simulate LiDAR point cloud data. Nevertheless, these methods often face challenges in producing high-quality, diverse, and controllable foreground objects. To address the needs of object-aware tasks in 3D perception, we introduce *OLiDM*, a novel framework capable of generating high-fidelity LiDAR data at both the object and the scene levels. *OLiDM* consists of two pivotal components: the Object-Scene Progressive Generation (OPG) module and the Object Semantic Alignment (OSA) module. OPG adapts to user-specific prompts to generate desired foreground objects, which are subsequently employed as conditions in scene generation, ensuring controllable outputs at both the object and scene levels. This also facilitates the association of user-defined object-level annotations with the generated LiDAR scenes. Moreover, OSA aims to rectify the misalignment between foreground objects and background scenes, enhancing the overall quality of the generated objects. The broad effectiveness of *OLiDM* is demonstrated across various LiDAR generation tasks, as well as in 3D perception tasks. Specifically, on the KITTI-360 dataset, *OLiDM* surpasses prior state-of-the-art methods such as UltraLiDAR by 17.5 in FPD. Additionally, in sparse-to-dense LiDAR completion, *OLiDM* achieves a significant improvement over LiDARGen, with a 57.47% increase in semantic IoU. Moreover, *OLiDM* enhances the performance of mainstream 3D detectors by 2.4% in mAP and 1.9% in NDS, underscoring its potential in advancing object-aware 3D tasks.

Code — https://yanty123.github.io/OLiDM/

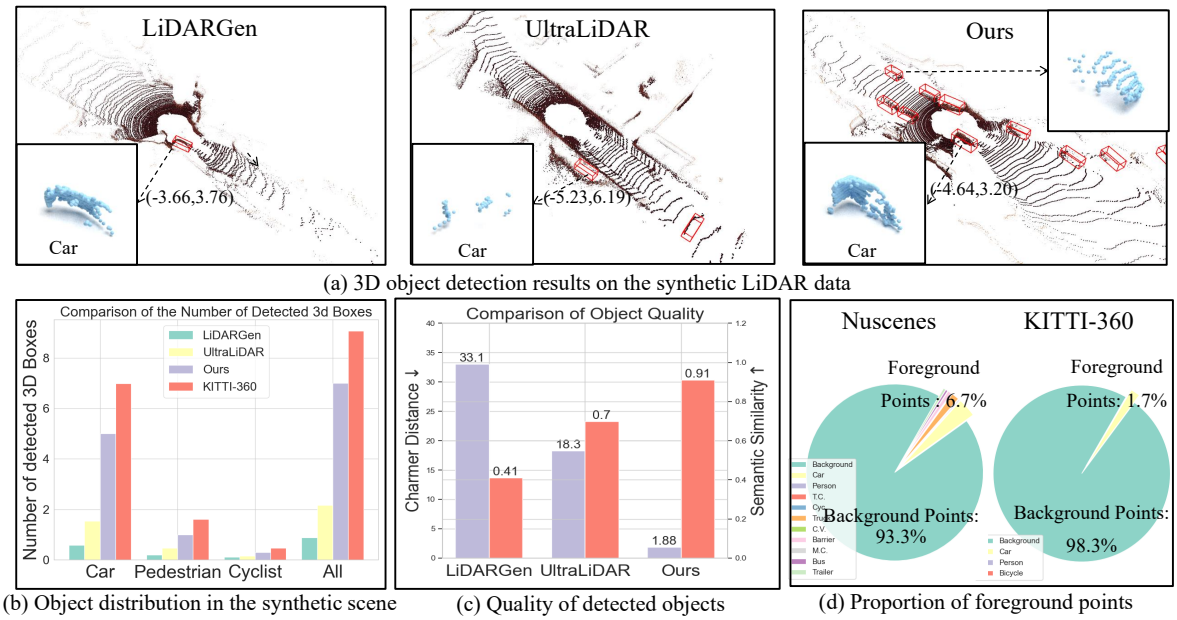
1 Introduction

LiDAR sensors are crucial for autonomous driving, as they can provide precise 3D information of the surroundings across various challenging conditions (Li et al. 2024, 2023a; Yan, Mao, and Li 2018; Cheng et al. 2023a; Yin et al. 2024). However, acquiring and annotating high-quality LiDAR data is costly and labor-intensive due to the sparse and noisy nature of LiDAR point cloud (Meng et al. 2020; Wu et al. 2020). This highlights the necessity for advanced generation algorithms to produce diverse, controllable, and scalable LiDAR point cloud data.

Recent breakthroughs in 3D generative models (Zyrianov, Zhu, and Wang 2022; Nakashima and Kurazume 2023) have enabled promising applications such as LiDAR data generation, which are critical for self-driving vehicles. To generate novel LiDAR data from scratch, LiDARGen (Zyrianov, Zhu, and Wang 2022) and R2DM (Nakashima and Kurazume 2023) transform 3D point clouds into structured 2D range images and leverage existing 2D diffusion models (Ho, Jain, and Abbeel 2020). Later, UltraLiDAR (Xiong et al. 2023) utilizes voxelized LiDAR point clouds in conjunction with VQ-VAE (Yu et al. 2021) to facilitate the densification of sparse LiDAR data. Despite their effectiveness in generating scene-level LiDAR data, we have observed significant discrepancies between these synthetic data and real-world data when applied to the 3D detection task. Firstly, as shown in fig. 1 (a) and (c), the quality of the generated objects is relatively low. In fig. 1 (b), significantly fewer foreground objects are detected in these synthetic data, highlighting the distribution discrepancy with real-world data. In practical applications, the focus is typically on foreground objects (Lang et al. 2019), yet existing models often generate LiDAR data without distinguishing between foreground and background, treating the scene as a whole. Additionally, foreground points are generally more sparse as shown in fig. 1 (d)). The challenge of developing a generative model capable of producing LiDAR data with high-quality 3D foreground objects still remains unresolved.

To address the above problems, we propose *OLiDM*, an **O**bject-aware **Li**DAR **D**iffusion **M**odel that aims to produce controllable and realistic LiDAR point cloud data *at both object and scene levels*. *OLiDM* comprises two key modules: the Object-Scene Progressive Generation (OPG) module and the Object Semantic Alignment (OSA) module. In particular, OPG advances the generation process by progressively modeling the foreground objects and background scenes. To effectively model the objects, OPG incorporates rich conditions, such as detailed textual descriptions and precise geometric specifications. These conditions provide both semantic and geometric context, explicitly enhancing the diversity of objects. Then, an object denoiser receives these condi-

*These authors contributed equally.
†Corresponding author: *Jianbing Shen*.



(a) 3D object detection results on the synthetic LiDAR data

(b) Object distribution in the synthetic scene (c) Quality of detected objects (d) Proportion of foreground points

Figure 1: To assess the quality of foreground objects, we utilize an off-the-shelf 3D detector (*i.e.*, SECOND (Yan, Mao, and Li 2018) trained on KITTI (Geiger et al. 2013)) to identify 3D objects in LiDAR data generated by various methods, including LiDARGen, UltraLiDAR and our *OLiDM*. **(a)** Visualization of some detected 3D objects in generated LiDAR scenes, where *OLiDM* enables the creation of LiDAR data with high-fidelity foreground objects. **(b-c)** We count the number and evaluate the quality of the detected objects, revealing our generated LiDAR data presents a similar distribution compared to the real data in KITTI-360 (Liao, Xie, and Geiger 2022). In contrast, LiDARGen and UltraLiDAR produce significantly fewer foreground objects with lower quality. **(d)** Foreground object points represent a minimal fraction of the total scene points, highlighting the challenges of generating high-quality foreground objects. Please zoom in for detailed visualization.

tions and generates high-quality 3D objects. Subsequently, these objects serve as a prior condition to assist the scene generation process. Here, a scene controller is introduced to embed the object condition into features, enforcing the scene denoiser to produce more meaningful LiDAR scenes with controllable 3D objects. Moreover, unlike previous approaches that generate simulated LiDAR data without specific 3D object information, OPG provides initial 3D annotations for the foreground objects, which can potentially benefit the downstream 3D perception tasks.

For generating scene-level LiDAR data, previous methods (Nakashima and Kurazume 2023; Zyrianov, Zhu, and Wang 2022) typically operate on spatially disordered range images, where range values of foreground and background can be mixed in a local window. In contrast, our OSA module tactfully partitions regions into various semantic subspaces. These semantic subspaces, defined by object category, serve as semantic masks to balance the diffusion loss, thereby facilitating model learning and refining object details within the scene. Through a meticulously designed learning process, *OLiDM* emerges as a highly practical and scalable LiDAR generator at both object and scene levels.

Our main contributions are summarized as follows:

- We introduce a new diffusion framework, *OLiDM*, to ensure the generation of high-quality LiDAR objects. To the best of our knowledge, *OLiDM* is the first effort that handles controllable and realistic LiDAR data at both object and scene levels.
- We propose an Object-Scene Progressive Generation (OPG) process. OPG integrates semantic and geometric conditions to achieve precise 3D object modeling. A new scene controller is also introduced to smoothly incorporate these 3D objects into the overall LiDAR scene.
- An Object Semantic Alignment (OSA) module is proposed to enforce the consistency optimization of foreground objects over their respective semantic subspaces during the diffusion process, which improves the overall quality of the generated LiDAR data.

Extensive experiments demonstrate that *OLiDM* generates high-quality LiDAR point clouds, achieving the best FPD and JSD on KITTI-360. Moreover, we are the first to evaluate the quality of foreground LiDAR objects generated by various methods, where *OLiDM* showing superior performance across all metrics. Additionally, *OLiDM* excels in conditional generation tasks such as sparse-to-dense LiDAR completion. Finally, our validation on the nuScenes dataset confirms that *OLiDM* effectively enhances the performance of downstream 3D perception tasks, *e.g.*, improving mAP by 2.4% of mainstream 3D detectors.

2 Related Work

2.1 Generative Models

Building upon DDPM's foundation (Ho, Jain, and Abbeel 2020), subsequent research (Li et al. 2023b; Lee et al. 2021; Lugmayr et al. 2022; Saharia et al. 2022) has explored integrating various forms of control into these models, demon-

strating remarkable versatility in applications such as text-to-image synthesis and inpainting. Shifting the focus to the 3D domain, initial research in point cloud generation focuses on Generative Adversarial Networks (GANs), with r-GAN (Achlioptas et al. 2018) setting the groundwork. Subsequent enhancements (Zhang et al. 2022; Shu, Park, and Kwon 2019) refined these models, while recent developments (Cheng et al. 2023b; Zhou, Du, and Wu 2021) incorporate diffusion processes into point cloud generation. DiT-3D (Mo et al. 2024) utilizes a diffusion transformer architecture for denoising voxelized point clouds. These advancements were primarily trained on comprehensive datasets with specific categories (Chang et al. 2015; Wu et al. 2015). Point-E (Nichol et al. 2022) represents a significant advancement in creating diverse 3D point clouds, leveraging large-scale text and 3D model pairings for training. Unlike general 3D object generation, this paper tackles the more challenging scenarios in autonomous driving, addressing issues like sparsity, occlusion, and uneven point distribution. Through tactful model design, *OLiDM* effectively overcomes these obstacles, ensuring the generation of realistic and controllable LiDAR objects and scenes.

2.2 LiDAR Data Generation

For the task of LiDAR data generation in autonomous driving, deep generative models such as GANs (Achlioptas et al. 2018; Caccia et al. 2019a; Sauer et al. 2021), VAEs (Caccia et al. 2019b) and their innovative hybrids have demonstrated significant advancements. Most recently, LiDARGen (Zyrianov, Zhu, and Wang 2022) proposes a new diffusion-based model for LiDAR data generation by transforming the input as the range image. R2DM (Nakashima and Kurazume 2023) further applies the DDPM to enhance the quality of generation. UltraLiDAR (Xiong et al. 2023) leverages voxelized LiDAR point clouds in conjunction with VQ-VAE (Yu et al. 2021), promoting efficient LiDAR data generation and completion. LiDM (Ran, Guizilini, and Wang 2024) leverages multi-modal conditions as input for LiDAR generation. Notably, these methods mainly focus on mimicking the data distribution of entire LiDAR scenes, where background areas inevitably constitute most of the points. As a result, they overlook the inherent differences between foreground objects and background areas, compromising the overall quality. Our work shifts more attention to foreground objects, which are practical and crucial in autonomous driving. By integrating detailed conditions into a progressive generation framework from object to scene levels, *OLiDM* provides a more effective solution for LiDAR data generation.

3 The Proposed *OLiDM* Framework

3.1 Overall Architecture

Formally, *OLiDM* (**O**bject-aware **Li**DAR **D**iffusion **M**odels) aims at generating both object-level point cloud $\hat{P}^o \in \mathbb{R}^{N^0 \times 4}$ and scene-level one $\hat{P}^s \in \mathbb{R}^{N^s \times 4}$, with input conditions $\mathcal{C} = \{T, B\}$, where N^o and N^s represent the maximum numbers of points, and $\hat{P}^o$ and $\hat{P}^s$ are characterized by 4-d attributes (*i.e.*, the 3D coordinates (x, y, z) and the LiDAR intensity i). T denotes textual descriptions of objects and B outlines their 3D bounding boxes to specify positions and orientations within the scene. This involves an progressive object-to-scene generation process, as shown in fig. 2:

Object-level: Given the condition $\mathcal{C}$ and the noised point cloud P_t^o, an object denoiser $\epsilon_\theta^o(\cdot)$ is designed to iteratively estimate and remove the object noise ϵ^o, where $\epsilon^o \sim \mathcal{N}(0, 1)$ is applied on the real object-level point cloud $P_0^o \in \mathbb{R}^{N \times 4}$ to obtain $P_t^o \in \mathbb{R}^{N \times 4}$ at timestep t. The intuition behind $\mathcal{C}$ is that the semantic conditions T can potentially handle long-tail classes at a semantic level, while geometric conditions B enhance the simulation of distant cars and rare poses at the instance level. Accordingly, the object denoiser $\epsilon_\theta^o(\cdot)$ gradually refines P_t^o towards P_0^o, yielding a series of synthetic LiDAR objects $\mathcal{P} = \left\{\hat{P}^o\right\}$ given various $\mathcal{C}$.

Scene-level: The above synthetic point clouds $\mathcal{P}$ captures the details of the foreground objects, which naturally serves as a foundation for inpainting a complete LiDAR scene $P^s \in \mathbb{R}^{N^s \times 4}$. Here, we formulate P^s as a range image $I \in [0, 1]^{H \times W \times 2}$ to ensure computation efficiency, where H and W are the sensor's resolution in azimuth and elevation, respectively. 2-d channels represent distance and intensity. To achieve high-fidelity scene-level generation, a new scene controller $\epsilon_\phi(\cdot)$ is introduced to assist the scene denoiser $\epsilon_\theta^s(\cdot)$ in refining the noisy scene input I_t at timestamp t. This is accomplished by carefully encoding the object-level information I^o, which is extracted from $\mathcal{P}$. The scene-level and object-level latent features effectively interact to produce a more accurate and coherent representation of the scene I_0, *i.e.*, enriched by high-quality 3D objects.

3.2 Object-Scene Progressive Generation Process

Previous LiDAR generation approaches (Nakashima and Kurazume 2023; Xiong et al. 2023) primarily generate point cloud at the *entire scene* level, overlooking the differences between foreground and background and thus inevitably degrading the quality of foreground objects. The rationale for handling the foreground points includes: (1) *Object-Scene Point Imbalance.* The foreground typically constitutes less than 10% of the points in an entire scene (see fig. 1 (d)), yet existing methods often fail to adequately address these sparse foreground points. As a result, the model may become biased toward the data distribution of background points. (2) *Object-Scene Semantic Discrepancy.* Foreground objects often include pedestrians and vehicles, which are more critical in autonomous driving, while background areas typically consist of roads, buildings, and vegetation. Consequently, points from the foreground and background exhibit distinct geometries that need to be addressed differently. To this end, we introduce the OPG, which employs a progressive framework—object-first, scene-later—to effectively address the inherent discrepancies between the two.

Object-level LiDAR Generation Unlike the point cloud objects in 3D Shapenet (Chang et al. 2015) that present relatively complete and smooth patterns, 3D LiDAR objects exhibit significantly different characteristics depending on their position relative to the ego-vehicle, with distant objects

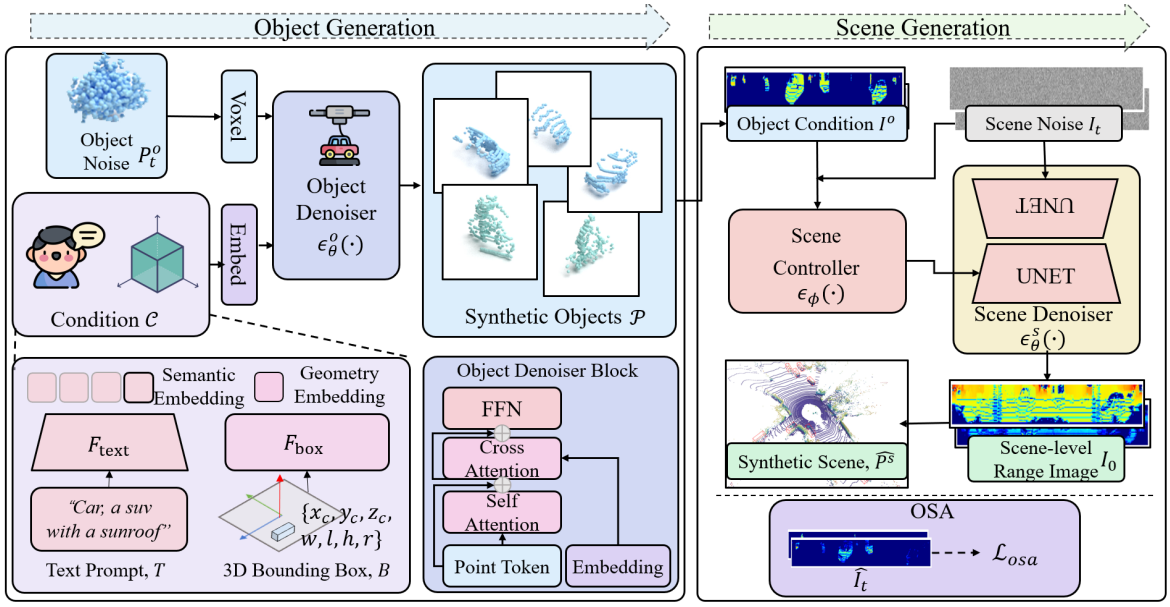


Figure 2: **Pipeline of *OLiDM*.** *OLiDM* is designed to generate diverse, controllable, and realistic LiDAR point clouds at both object and scene levels through the Object-Scene Progressive Generation (OPG) process. **Object Generation:** OPG carefully combines conditions such as text descriptions and 3D geometric context to accurately model LiDAR objects. **Scene Generation:** OPG then incorporates these generated objects as specific conditions during scene-level generation, supported by a scene controller and an object semantic alignment module.

often appearing sparser. As such, it is necessary to redesign current diffusion models for object-level LiDAR generation.

Given a real object point cloud $P_0^o \in \mathbb{R}^{N\times 3}$ (omitting intensity here for simplicity) from the database, where N is the number of points with x, y, z coordinates, we first add noise ϵ^o on P_0^o to obtain the noisy version $P_t^o \in \mathbb{R}^{N\times 3}$, which mainly follows DDPM (Ho, Jain, and Abbeel 2020). P_t^o is then voxelized and embedded as $\boldsymbol{V}_t$ by leveraging 3D convolution operations (Mo et al. 2024). Then, we define the conditions $\mathcal{C} = \{T, B\}$. Specifically, T is formulated as "*An object from class* ⟨category⟩, ⟨description⟩." Here, ⟨category⟩ refers to the object's category, as annotated in the datasets, while ⟨description⟩ offers detailed instance-level descriptions, such as "*a sports car that is on the street, side view*". During training, we employ an advanced caption model (Li et al. 2022) to provide rich descriptions for each 3D object. Then, we get the embeddings of T by:

$$[\boldsymbol{f}_{\text{CLS}}^T, \boldsymbol{f}_0^T, \cdots, \boldsymbol{f}_L^T] = F_{\text{text}}(T), \tag{1}$$

where L denotes the length of the sentence, $F_{\text{text}}(\cdot)$ is the pre-trained CLIP (Chen et al. 2020). After that, we let $\boldsymbol{f}^T = \boldsymbol{f}_{\text{CLS}}^T$ to summarize T since $\boldsymbol{f}_{\text{CLS}}^T$ has effectively integrated the context by attending to all positions in $[\boldsymbol{f}_{\text{CLS}}^T, \boldsymbol{f}_0^T, \cdots, \boldsymbol{f}_L^T]$. Meanwhile, we propose to utilize the 3D bounding box, $B = [x_c, y_c, z_c, w, l, h, r]$, to provide crucial object geometric information that indicates the distribution of points. To be specific, the 3D center and rotation reveal the distance and orientation relative to the ego-vehicle, which in turn determines the visibility of the object's points. Additionally, the dimensions of the box potentially offer a unique identifier for each instance within the same category. Here, we extract geometry-aware embeddings $\boldsymbol{f}^B$ by:

$$\boldsymbol{f}^B = F_{\text{box}}(B) = F_{\text{fe}}([x_c, y_c, z_c, w, l, h, r]), \tag{2}$$

where $F_{\text{fe}}(\cdot)$ denotes Fourier Embedder (Mildenhall et al. 2021). Furthermore, $\boldsymbol{f}^T$ and $\boldsymbol{f}^B$ are combined to obtain a comprehensive condition embedding $\boldsymbol{c}$:

$$\boldsymbol{c} = F_{\text{com}}([\boldsymbol{f}^B, \boldsymbol{f}^T]), \tag{3}$$

where $[\cdot, \cdot]$ is the concatenation operation and $F_{\text{com}}(\cdot)$ can be realized by linear layers.

Since we have the noised input embedding $\boldsymbol{V}_t$ and the condition embedding $\boldsymbol{c}$, an Object Denoiser (See fig. 2) is proposed to estimate the noise at timestep t. Inspired by existing diffusion models (Nichol et al. 2022; Luo and Hu 2021), we apply the cross-attention mechanism to enhance the interaction between the inputs. The query, key and value in the cross-attention layer are denoted as:

$$\mathbf{q} = \text{linear}(\boldsymbol{V}_t), \mathbf{k} = \text{linear}(\boldsymbol{c}+\boldsymbol{f}^t), \mathbf{v} = \text{linear}(\boldsymbol{c}+\boldsymbol{f}^t), \tag{4}$$

where $\boldsymbol{f}^t$ is the embedded timestep t. In this way, the noised voxel embedding $\boldsymbol{V}_t$ can integrate information from $\boldsymbol{f}^t$ and $\boldsymbol{c}$ to refine its 3D representation:

$$\hat{\boldsymbol{V}}_t = \boldsymbol{V}_t + \mathbf{v}^{\mathsf{T}} \cdot \text{softmax}(\mathbf{k} \cdot \mathbf{q}^{\mathsf{T}}) \tag{5}$$

Finally, a feed-forward network estimates an $N \times 3$ tensor based on the attention output $\hat{\boldsymbol{V}}_t$, representing the object noise towards the ground truth ϵ^o. The Object Denoiser $\epsilon_\theta^o(\cdot)$ can be optimized as:

$$\mathcal{L}_{\text{object}} = \mathbb{E}_{P_0^o, \boldsymbol{c}, \epsilon^o, t}[\|\epsilon^o - \epsilon_\theta^o(P_t^o, t, \boldsymbol{c})\|^2]. \tag{6}$$

During inference, the Object Denoiser iteratively processes the noised input object points with condition to generate a synthetic 3D object. This can be repeated to produce an unlimited number of point cloud objects $\mathcal{P}$ based on different $\mathcal{C}$. Notably, users have the flexibility to define semantic

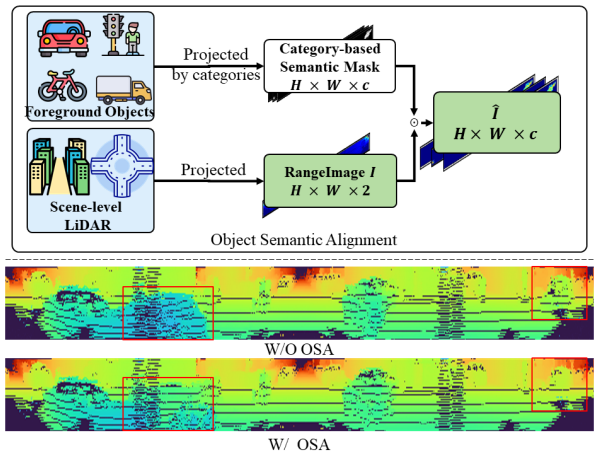


Figure 3: **The Object Semantic Alignment (OSA) module** aligns object features based on their semantic space.

and geometric conditions to account for different object distribution settings. The process can be simplified by applying uniform or random sampling for B on the road and specifying only the category description for each object as T.

Scene-level LiDAR Generation Building upon the object point clouds $\mathcal{P}$, our goal is to generate complete LiDAR scenes P^s that are compatible with these high-quality objects. The main challenge in scene-level point cloud generation lies in handling hundreds of thousands of points simultaneously. Following the methodologies in (Zyrianov, Zhu, and Wang 2022), we choose to represent the entire LiDAR scene as a range image to save computation.

Specifically, object points $\mathcal{P}$ are first projected into an object-level range image I^o. Then, a scene controller $\epsilon_\phi(\cdot)$ is introduced to maintain the contents and positions of the objects and guide the generation process. This is achieved by initially extracting a conditioning embedding and combining it with the noised scene input I_t to produce the intermediate latent feature $\boldsymbol{h_c^t}$:

$$\boldsymbol{h_c^t} = \text{zero}(\text{conv}(I^o)) + \text{zero}(\text{conv}(I_t)) \tag{7}$$

Here, zero denotes the zero convolution layer (Zhang, Rao, and Agrawala 2023). Following several layers of stacked U-Net blocks (Ronneberger, Fischer, and Brox 2015) and zero convolutions, we refine $\boldsymbol{h_c^t}$ and effectively control the detailed rendering of foreground objects.

The scene denoiser $\epsilon_\theta^s(\cdot)$ then processes the scene noise I_t along with the control embedding to extract the final latent feature. The optimization process of the above process can be denoted as:

$$\mathcal{L}_{\text{object-scene}} = \mathbb{E}_{I^o, I_0, \epsilon^s, t}[\|\epsilon^s - \epsilon_\theta^s(I_t, t, \epsilon_\phi(I^o))\|^2]. \tag{8}$$

By leveraging this progressive generation framework, *OLiDM* not only ensures the realism and fidelity of individual object details but also preserves the contextual integrity and coherence of the entire scene, thereby resulting in highly realistic and contextually accurate LiDAR scenes.

3.3 Object Semantic Alignment

Here, we introduce the proposed OSA module. While range images contain rich depth information, they suffer from *spatial misalignment* issues (Bai et al. 2024), where adjacent pixels in the range image can represent real-world distances greater than 30 meters. Previous approaches (Xiong et al. 2023; Nakashima and Kurazume 2023) directly feed entire range images into a 2D backbone, resulting in mixed features for near and distant objects. This hinders the extraction of accurate geometric information of the foreground objects and corrupts the generation quality.

To rectify the object-specific representations from the mixed features, we design the OSA module, which aligns features within the semantic subspace for each 3D object. By splitting the features into distinct spaces based on categories, we minimize interference between regions. As illustrated in fig. 17, this method sharpens edge information for foreground objects, resulting in the higher-quality generation.

Specifically, given a frame of range image $I \in \mathbb{R}^{H\times W\times 2}$, OSA first calculates binary mask $M \in \mathbb{R}^{H\times W\times c}$ according to foreground objects $\mathcal{P}$, where c is the number of categories readily defined in the object generation process, and each $M_i \in H \times W$ is a pixel-wise mask for range image I, indicating whether each point represents the object from c_i category. Next, OSA defines a tensor $\hat{I} \in \mathbb{R}^{H\times W\times c}$, which is a c-channel version of I, with the i-th channel defined by $\hat{I}_i = I \odot M_i$. Here, $\hat{I}_t$, the noisy version of $\hat{I}$ serves as input of the scene denoiser $\epsilon^s(\cdot)$ to calculate OSA loss:

$$\mathcal{L}_{\text{osa}} = \mathbb{E}_{\hat{I}_0, \hat{\epsilon^s}, t}[\left\|\hat{\epsilon^s} - \epsilon_\theta^s(\hat{I}_t, t)\right\|^2], \tag{9}$$

where $\hat{\epsilon^s} \in \mathbb{R}^{H\times W\times c}$ is t steps noise we added on $\hat{I}$. We adopt eq. (9) as an additional loss with eq. (8) to achieve alignment on the semantic feature subspace at the category level, thereby enhancing the quality of foreground object generation and ensuring a more consistent integration between foreground and background.

4 Experiments

4.1 Experimental Setups

Dataset and Setting. nuScenes (Caesar et al. 2020) and KITTI-360 (Liao, Xie, and Geiger 2022) are popular datasets widely used in autonomous driving research, featuring detailed annotations for evaluating different tasks. For the generation task, we employ the KITTI-360 dataset to demonstrate the effectiveness, while for 3D object detection and other tasks (Han et al. 2024; Tao et al. 2023), we use the nuScenes (Caesar et al. 2020) as the benchmark.

Evaluation Metrics. For the scene-level LiDAR generation, we use MMD, JSD on the BEV plane and FPD as the metrics, and we generate 10k samples using 0000 and 0002 seq as the validation split following (Zyrianov, Zhu, and Wang 2022; Xiong et al. 2023). For the quality of foreground objects, we use Chamfer Distance (CD), and Jensen-Shannon Divergence (JSD) to judge. We also define Semantic Similarity (SS) to evaluate semantic closeness using cosine sim-

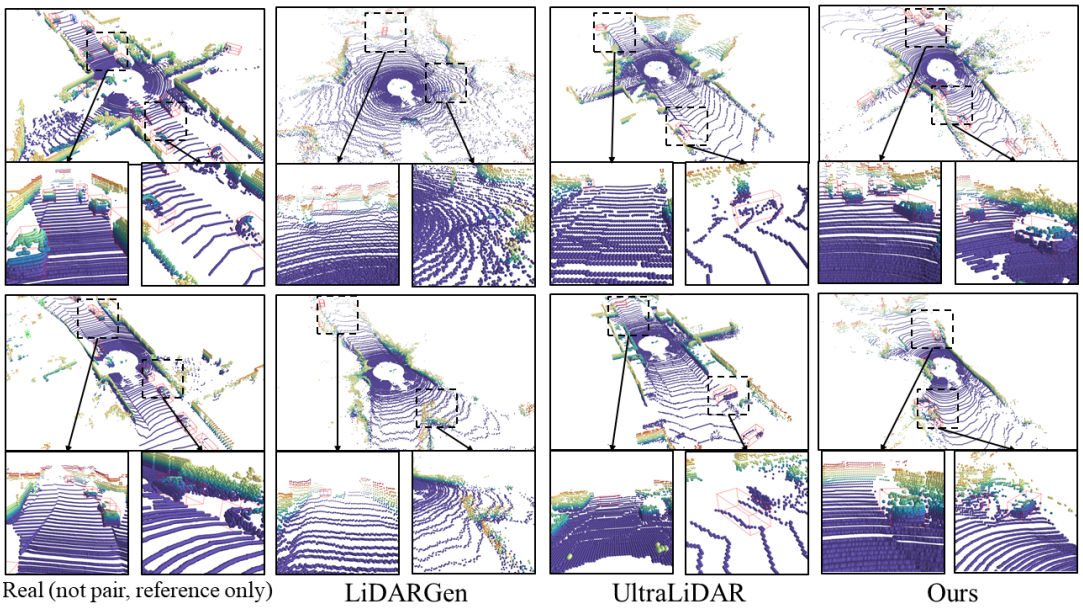


Figure 4: **Qualitative comparison against baselines on LiDAR generation.** We compare with LiDARGen, UltraLiDAR and include real LiDAR for reference. *OLiDM* generates LiDAR data with more realistic sparsity and beam patterns. Red, Blue and Green boxes are the detected objects (car, cyclist and pedestrian) from a 3D detector trained on KITTI.

Method	JSD$_{(\downarrow)}$	MMD$_{(\downarrow)}$	FPD$_{(\downarrow)}$
LiDARVAE	1.61	10.0	-
Proj.GAN	8.05	3.47	-
LiDARGen	0.67	3.87	90.3
UltraLiDAR	0.71	**1.96**	25.1
R2DM	0.49	4.35	14.6
OLiDM w/o OSA	0.47	3.89	9.20
OLiDM	**0.42**	3.45	**7.60**

Table 1: **Quantitative results of the LiDAR generation task on KITTI-360.** MMD and JSD are multiplied with 10^4 and 10, respectively.

Method	CD$_{(\downarrow)}$	SS$_{(\uparrow)}$	JSD$_{(\downarrow)}$	#Box
LiDARGen	33.1	0.41	0.97	0.88
UltraLiDAR	18.3	0.70	0.95	2.16
R2DM	17.6	0.86	0.91	4.66
OLiDM^{-}	14.3	0.84	0.93	5.02
OLiDM^{T}	9.21	0.86	0.87	5.92
OLiDM^{B}	4.32	0.85	0.82	6.77
OLiDM	**1.88**	**0.91**	**0.74**	**7.10**

Table 2: **Quantitative results of LiDAR objects on KITTI-360.** We sample 1000 objects for evaluation.

ilarity of embeddings from the pre-trained Openshape (Liu et al. 2023).

4.2 Object-Aware LiDAR Generation

Scene-level Generation. *OLiDM* is compared against several state-of-the-art LiDAR data generation methods, including LiDARGen (Zyrianov, Zhu, and Wang 2022), R2DM (Nakashima and Kurazume 2023), and UltraLidar (Xiong et al. 2023), as detailed in table 1. *OLiDM* achieves superior performance in terms of JSD and FPD metrics, indicating that our *OLiDM* produces the most realistic point clouds in both BEV and point spaces. Although UltraLiDAR shows better outcomes on the MMD metric due to its voxel-based BEV modeling, it falls short in the FPD metric, which assesses point-level accuracy. *OLiDM* exceeds UltraLiDAR for 17.5 at FPD. The enhancements stem from *OLiDM*'s innovative strategy, which ensure a more faithful and robust synthesis of LiDAR data.

Object-level Generation. Previous LiDAR point cloud generation methods often overlook quality control at the object level, potentially limiting their applicability in autonomous driving. In response, we propose to evaluate the quality of foreground objects within the generated LiDAR data, as demonstrated by table 2. We utilize a well-trained 3D detector (Yan, Mao, and Li 2018) to identify foreground objects within LiDAR scenes. A critical metric in our analysis, #Box, represents the average count of detected 3D boxes per scene, with values approaching the KITTI-360 dataset's standard of 9.06 reflecting superior performance. *OLiDM* notably aligns with the distribution of real foreground objects, significantly reducing the CD metric to 1.88 and JSD metric to 0.74, demonstrating its effectiveness in generating high-fidelity LiDAR objects. UltraLiDAR struggles due to its voxelization approach, which excessively sparsifies the dense point clouds of objects, resulting in a mere 2.16 detected boxes. Meanwhile, our method captures 7.1 boxes, significantly closer to the real data benchmark of 9.06. These experiments indicate the critical role of *OLiDM* in generating high-fidelity LiDAR objects.

Qualitative Comparison. The qualitative comparison is presented in fig. 4, where LiDARGen demonstrates accept-

Method	Depth	Reflectence	Semantics
	MSE↓	MSE↓	IoU%↑
Nearest-neighbor	2.069	0.100	20.00
Bilinear	2.193	0.106	18.19
Bicubic	2.314	0.110	17.26
LiDARGen	1.551	0.080	22.46
R2DM	0.923	0.050	34.44
OLiDM	**0.702**	**0.048**	**79.93**

Table 3: **Quantitative results of sparse-to-dense LiDAR generation on KITTI-360,** where we input 25% beams.

3D Detector	Aug. Method	Paradigm	mAP	NDS
PointPillars	Baseline	None	44.8	58.3
	+ GT-Aug	Real	45.1	58.9
	+LiDAR-Aug	Synthetic	45.6	58.4
	+ *OLiDM*	Synthetic	**46.7**	**61.1**
CenterPoint	Baseline	None	59.0	65.8
	+ GT-Aug	Synthetic	59.2	66.5
	+ LiDAR-Aug	Synthetic	60.5	67.3
	+ *OLiDM*	Synthetic	**61.9**	**68.5**

Table 4: **Evaluation of the effectiveness in augmenting mainstream 3D detectors** on the nuScenes. Compared with other data augmentation techniques, *OLiDM* achieves superior performance across various 3D detectors.

able performance in foreground patterns but struggles with scene-level fidelity. Although UltraLiDAR shows impressive scene-level patterns through its BEV representation, it sacrifices object-level quality, probably due to the BEV representation's inability to adequately capture 3D object details. In contrast, our method maintains high quality in both background structures and foreground objects, highlighting the effectiveness of the OPG and OSA designs.

4.3 Conditional LiDAR Generation

Sparse-to-Dense LiDAR Completion. Following the settings of LiDARGen and R2DM, we conduct LiDAR upsampling experiments on the KITTI-360, using 16-beam LiDAR as input and producing 64-beam LiDAR. In table 3 and fig. 5, *OLiDM* achieves the best results across all metrics, with a significant improvement in semantic IoU from 34% to 79%. The success can be attributed to the OPG's ability to stably and reliably encode conditions and facilitate feature interaction, thus controlling the overall content generation.

Controllable LiDAR Generation. To verify our method's capability to control the generation of LiDAR data, we conduct validations from both object-level and scene-level control perspectives in fig. 6. For the object level, we manipulate the textual descriptions and spatial geometric positions to control the generated object's semantic content and LiDAR scanning patterns. For the scene level, we leverage foreground objects as conditions for the generation of desired LiDAR scenes, including extremely crowded scenarios. This advantage demonstrates our ability to generate LiDAR data that aligns with user expectations and adheres to real-world physical rules, which is crucial for the autonomous driving industry in creating specific corner-case scenarios.

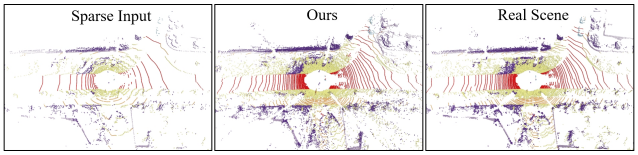


Figure 5: **Sparse-to-dense Completion.** The semantic results are predicted by RangeNet-53 (Milioto et al. 2019).

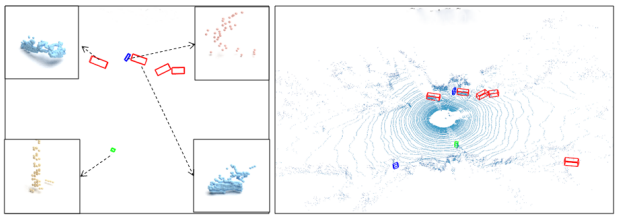


Figure 6: ***OLiDM* controls contents at both object and scene levels.** *OLiDM* guarantees that the desired 3D objects are accurately represented in the final LiDAR scenes.

4.4 Downstream Task

The above experiments demonstrate that *OLiDM* can produce high-quality 3D LiDAR objects, which can be naturally leveraged to enhance downstream 3D detection task (Yin et al. 2021). To further validate our approach, we employ CenterPoint (Yin, Zhou, and Krahenbuhl 2021) and Pointoillars (Lang et al. 2019) as baselines to examine improvements in detection performance. Specifically, we augment the detectors using the 3D objects generated by *OLiDM*, and compare with the widely used GT-Aug (Yan, Mao, and Li 2018) technique (GT-Aug introduces randomly sampled ground truth from other scenes into current scene to facilitate the training). As shown in table 4, *OLiDM* improves the detector performance by 2.7% compared to GT-Aug. These results demonstrate that high-quality 3D objects potentially enhance the 3D perception task. It is worth noting that other current LiDAR generation methods are unable to produce controllable and high-fidelity 3D LiDAR objects, which limits their applicability in downstream perception tasks.

5 Conclusion

In this paper, we introduced *OLiDM*, a novel framework for generating realistic and controllable LiDAR data at both object and scene levels. By leveraging the Object-Scene Progressive Generation (OPG) and Object Semantic Alignment (OSA), *OLiDM* not only captures data distribution similar to real-world LiDAR data, but also enhances the performance of downstream perception tasks. Our evaluations across diverse benchmarks highlight the model's ability to produce high-fidelity LiDAR data.

Acknowledgements

This work was supported in part by the FDCT grants 0102/2023/RIA2 and 0154/2022/A3, the MYRG-CRG2022-00013-IOTSC-ICI grant, and the SRG2022-00023-IOTSC grant.

References

Achlioptas, P.; Diamanti, O.; Mitliagkas, I.; and Guibas, L. 2018. Learning representations and generative models for 3d point clouds. In *International conference on machine learning*, 40–49. PMLR.

Bai, Y.; Fei, B.; Liu, Y.; Ma, T.; Hou, Y.; Shi, B.; and Li, Y. 2024. RangePerception: Taming LiDAR Range View for Efficient and Accurate 3D Object Detection. *Advances in Neural Information Processing Systems*, 36.

Caccia, L.; Van Hoof, H.; Courville, A.; and Pineau, J. 2019a. Deep generative modeling of lidar data. In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 5034–5040. IEEE.

Caccia, L.; Van Hoof, H.; Courville, A.; and Pineau, J. 2019b. Deep generative modeling of lidar data. In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 5034–5040. IEEE.

Caesar, H.; Bankiti, V.; Lang, A. H.; Vora, S.; Liong, V. E.; Xu, Q.; Krishnan, A.; Pan, Y.; Baldan, G.; and Beijbom, O. 2020. nuscenes: A multimodal dataset for autonomous driving. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 11621–11631.

Chang, A. X.; Funkhouser, T.; Guibas, L.; Hanrahan, P.; Huang, Q.; Li, Z.; Savarese, S.; Savva, M.; Song, S.; Su, H.; et al. 2015. Shapenet: An information-rich 3d model repository. *arXiv preprint arXiv:1512.03012*.

Chen, T.; Kornblith, S.; Norouzi, M.; and Hinton, G. 2020. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, 1597–1607. PMLR.

Cheng, W.; Yin, J.; Li, W.; Yang, R.; and Shen, J. 2023a. Language-guided 3d object detection in point cloud for autonomous driving. *arXiv preprint arXiv:2305.15765*.

Cheng, Y.-C.; Lee, H.-Y.; Tulyakov, S.; Schwing, A. G.; and Gui, L.-Y. 2023b. Sdfusion: Multimodal 3d shape completion, reconstruction, and generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 4456–4465.

Geiger, A.; Lenz, P.; Stiller, C.; and Urtasun, R. 2013. Vision meets robotics: The kitti dataset. *The International Journal of Robotics Research*, 32(11): 1231–1237.

Han, W.; Tao, R.; Ling, H.; and Shen, J. 2024. Weakly Supervised Monocular 3D Object Detection by Spatial-Temporal View Consistency. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Ho, J.; Jain, A.; and Abbeel, P. 2020. Denoising diffusion probabilistic models. In *Advances in neural information processing systems*, volume 33, 6840–6851.

Lang, A. H.; Vora, S.; Caesar, H.; Zhou, L.; Yang, J.; and Beijbom, O. 2019. Pointpillars: Fast encoders for object detection from point clouds. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 12697–12705.

Lee, S.-g.; Kim, H.; Shin, C.; Tan, X.; Liu, C.; Meng, Q.; Qin, T.; Chen, W.; Yoon, S.; and Liu, T.-Y. 2021. PriorGrad: Improving conditional denoising diffusion models with data-dependent adaptive prior. *arXiv preprint arXiv:2106.06406*.

Li, J.; Li, D.; Xiong, C.; and Hoi, S. 2022. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *International Conference on Machine Learning*, 12888–12900. PMLR.

Li, X.; Yin, J.; Li, W.; Xu, C.; Yang, R.; and Shen, J. 2024. Di-v2x: Learning domain-invariant representation for vehicle-infrastructure collaborative 3d object detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, 3208–3215.

Li, X.; Yin, J.; Shi, B.; Li, Y.; Yang, R.; and Shen, J. 2023a. Lwsis: Lidar-guided weakly supervised instance segmentation for autonomous driving. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, 1433–1441.

Li, Y.; Liu, H.; Wu, Q.; Mu, F.; Yang, J.; Gao, J.; Li, C.; and Lee, Y. J. 2023b. Gligen: Open-set grounded text-to-image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 22511–22521.

Liao, Y.; Xie, J.; and Geiger, A. 2022. KITTI-360: A novel dataset and benchmarks for urban scene understanding in 2d and 3d. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(3): 3292–3310.

Liu, M.; Shi, R.; Kuang, K.; Zhu, Y.; Li, X.; Han, S.; Cai, H.; Porikli, F.; and Su, H. 2023. OpenShape: Scaling Up 3D Shape Representation Towards Open-World Understanding. *arXiv preprint arXiv:2305.10764*.

Lugmayr, A.; Danelljan, M.; Romero, A.; Yu, F.; Timofte, R.; and Van Gool, L. 2022. Repaint: Inpainting using denoising diffusion probabilistic models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 11461–11471.

Luo, S.; and Hu, W. 2021. Diffusion probabilistic models for 3d point cloud generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2837–2845.

Meng, Q.; Wang, W.; Zhou, T.; Shen, J.; Van Gool, L.; and Dai, D. 2020. Weakly supervised 3d object detection from lidar point cloud. In *European Conference on computer vision*, 515–531. Springer.

Mildenhall, B.; Srinivasan, P. P.; Tancik, M.; Barron, J. T.; Ramamoorthi, R.; and Ng, R. 2021. Nerf: Representing scenes as neural radiance fields for view synthesis. *Communications of the ACM*, 65(1): 99–106.

Milioto, A.; Vizzo, I.; Behley, J.; and Stachniss, C. 2019. Rangenet++: Fast and accurate lidar semantic segmentation. In *2019 IEEE/RSJ international conference on intelligent robots and systems (IROS)*, 4213–4220. IEEE.

Mo, S.; Xie, E.; Chu, R.; Hong, L.; Niessner, M.; and Li, Z. 2024. Dit-3d: Exploring plain diffusion transformers for 3d shape generation. *Advances in Neural Information Processing Systems*, 36.

Nakashima, K.; and Kurazume, R. 2023. LiDAR Data Synthesis with Denoising Diffusion Probabilistic Models. *arXiv preprint arXiv:2309.09256*.

Nichol, A.; Jun, H.; Dhariwal, P.; Mishkin, P.; and Chen, M. 2022. Point-e: A system for generating 3d point clouds from complex prompts. *arXiv preprint arXiv:2212.08751*.

Ran, H.; Guizilini, V.; and Wang, Y. 2024. Towards Realistic Scene Generation with LiDAR Diffusion Models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 14738–14748.

Ronneberger, O.; Fischer, P.; and Brox, T. 2015. U-net: Convolutional networks for biomedical image segmentation. In *Medical image computing and computer-assisted intervention–MICCAI 2015: 18th international conference, Munich, Germany, October 5-9, 2015, proceedings, part III 18*, 234–241. Springer.

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. 2022. Photorealistic text-to-image diffusion models with deep language understanding. *Advances in Neural Information Processing Systems*, 35: 36479–36494.

Sauer, A.; Chitta, K.; Müller, J.; and Geiger, A. 2021. Projected gans converge faster. *Advances in Neural Information Processing Systems*, 34: 17480–17492.

Shu, D. W.; Park, S. W.; and Kwon, J. 2019. 3d point cloud generative adversarial network based on tree structured graph convolutions. In *Proceedings of the IEEE/CVF international conference on computer vision*, 3859–3868.

Tao, R.; Han, W.; Qiu, Z.; Xu, C.-Z.; and Shen, J. 2023. Weakly supervised monocular 3d object detection using multi-view projection and direction consistency. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 17482–17492.

Wu, Y.; Wang, Y.; Zhang, S.; and Ogai, H. 2020. Deep 3D object detection networks using LiDAR data: A review. *IEEE Sensors Journal*, 21(2): 1152–1171.

Wu, Z.; Song, S.; Khosla, A.; Yu, F.; Zhang, L.; Tang, X.; and Xiao, J. 2015. 3d shapenets: A deep representation for volumetric shapes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 1912–1920.

Xiong, Y.; Ma, W.-C.; Wang, J.; and Urtasun, R. 2023. Learning Compact Representations for LiDAR Completion and Generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 1074–1083.

Yan, Y.; Mao, Y.; and Li, B. 2018. Second: Sparsely embedded convolutional detection. *Sensors*, 18(10): 3337.

Yin, J.; Shen, J.; Chen, R.; Li, W.; Yang, R.; Frossard, P.; and Wang, W. 2024. Is-fusion: Instance-scene collaborative fusion for multimodal 3d object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 14905–14915.

Yin, J.; Shen, J.; Gao, X.; Crandall, D. J.; and Yang, R. 2021. Graph neural network and spatiotemporal transformer attention for 3D video object detection from point clouds. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(8): 9822–9835.

Yin, T.; Zhou, X.; and Krahenbuhl, P. 2021. Center-based 3d object detection and tracking. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 11784–11793.

Yu, J.; Li, X.; Koh, J. Y.; Zhang, H.; Pang, R.; Qin, J.; Ku, A.; Xu, Y.; Baldridge, J.; and Wu, Y. 2021. Vector-quantized image modeling with improved vqgan. *arXiv preprint arXiv:2110.04627*.

Zhang, L.; Rao, A.; and Agrawala, M. 2023. Adding conditional control to text-to-image diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 3836–3847.

Zhang, X.; Zheng, Z.; Gao, D.; Zhang, B.; Pan, P.; and Yang, Y. 2022. Multi-view consistent generative adversarial networks for 3d-aware image synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 18450–18459.

Zhou, L.; Du, Y.; and Wu, J. 2021. 3d shape generation and completion through point-voxel diffusion. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 5826–5835.

Zyrianov, V.; Zhu, X.; and Wang, S. 2022. Learning to generate realistic lidar point clouds. In *European Conference on Computer Vision*, 17–35.