DriveLiDAR4D: Sequential and Controllable LiDAR Scene Generation for Autonomous Driving

Kaiwen Cai[1*], Xinze Liu[1*], Xia Zhou[1*], Hengtong Hu[1], Jie Xiang[1], Luyao Zhang[1], Xueyang Zhang[1], Kun Zhan[1], Yifei Zhan[1], Xianpeng Lang[1]

[1]Li Auto Inc.
caikaiwen1@lixiang.com

Abstract

The generation of realistic LiDAR point clouds plays a crucial role in the development and evaluation of autonomous driving systems. Although recent methods for 3D LiDAR point cloud generation have shown significant improvements, they still face notable limitations, including the lack of sequential generation capabilities and the inability to produce accurately positioned foreground objects and realistic backgrounds. These shortcomings hinder their practical applicability. In this paper, we introduce DriveLiDAR4D, a novel LiDAR generation pipeline consisting of multimodal conditions and a novel sequential noise prediction model LiDAR4DNet, capable of producing temporally consistent LiDAR scenes with highly controllable foreground objects and realistic backgrounds. To the best of our knowledge, this is the first work to address the sequential generation of LiDAR scenes with full scene manipulation capability in an end-to-end manner. We evaluated DriveLiDAR4D on the nuScenes and KITTI datasets, where we achieved an FRD score of 743.13 and an FVD score of 16.96 on the nuScenes dataset, surpassing the current state-of-the-art (SOTA) method, UniScene, with an performance boost of 37.2% in FRD and 24.1% in FVD, respectively.

1 Introduction

Data is a foundational element driving artificial intelligence advances. Within autonomous driving research, high-quality data is particularly crucial due to: i) the inherent data-intensive requirements of deep learning models, and ii) the necessity of capturing corner cases — rare driving behaviours and uncommon road environments — which are essential for developing safety-critical systems. However, collecting and annotating diverse multi-modal datasets (e.g., camera and LiDAR) remains time-consuming and resource-intensive. While recent generative models have demonstrated promising capabilities for synthesizing visual data, LiDAR scene generation—despite its critical role in providing geometric awareness — remains comparatively underdeveloped. In this work, we aim to advance existing LiDAR scene generation techniques to better address real-world autonomous driving requirements.

To synthesize realistic LiDAR data that accurately captures diverse real-world traffic scenarios, a LiDAR scene generation method should suppport flexible customization of road layouts and dynamic object placements. Recent studies, such as LiDARGen (Zyrianov, Zhu, and Wang 2022), UltraLiDAR (Xiong et al. 2023), R2DM (Nakashima and Kurazume 2024), and RangeLDM (Hu, Zhang, and Hu 2024), have made considerable advancements in producing realistic LiDAR data. However, these techniques predominantly generate data in an unconditional manner, lacking the capability to manipulate specific scene elements.

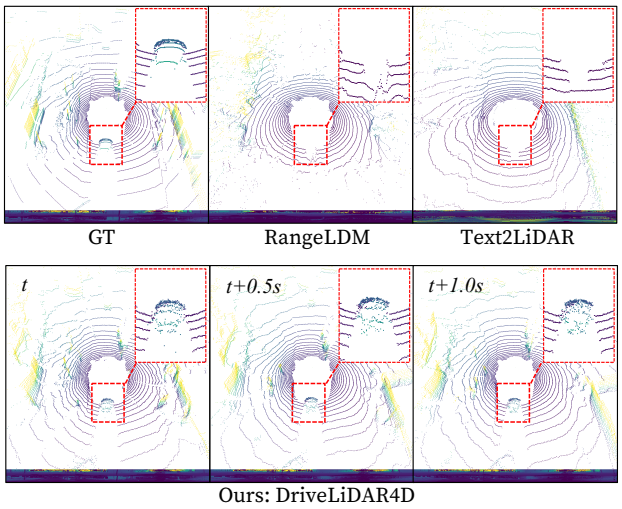
The quality of background LiDAR point clouds is as critical as that of foreground objects to ensure realism. To this end, Text2LiDAR (Wu et al. 2024) utilizes textual descriptions as conditioning input. Nonetheless, this method is confined to coarse descriptions encompassing weather conditions, time of day, and object names. The omission of detailed background information, such as specific descriptions of trees or buildings, compromises the realism of the generated LiDAR data.

In addition, existing 3D LiDAR scene generation methods are inadequate for accurately modeling the dynamic behaviors of objects. To mitigate this issue, LidarDM (Zyrianov et al. 2024) proposes to generate LiDAR sequences by separately modeling static scenes and dynamic objects. However, it lacks background control, and its two-stage compositional strategy potentially compromises the realism and coherence of the resulting point cloud distributions.

In summary, current LiDAR scene generation methods are notably deficient when it comes to integrating all critical capbalities: i) sequential scene generation with detailed control over both ii) foreground and iii) background components. To bridge this gap, we propose DriveLiDAR4D, an end-to-end 4D LiDAR scene generation pipeline that facilitates the generation of sequential LiDAR scenes with comprehensive scene manipulation capabilities. DriveLiDAR4D is distinguished by two principal features: i) the incorporation of multimodal conditions, including scene captions, road sketches and object priors, and ii) a meticulously designed equirectangular spatial-temporal noise prediction model, LiDAR4DNet, which ensures spatial and temporal consistency throughout denoising processes.

Fig. 1 displays LiDAR scenes generated by different methods on the nuScenes val split. It is evident that

*These authors contributed equally.



	Fore-ground control	Back-ground control	Object-fidelity enhancement	Sequential generation
RangeLDM (ECCV2024)	✗	✗	✗	✗
Text2LiDAR (ECCV2024)	✗	✓ (Coarse)	✗	✗
Ours: DriveLiDAR4D	✓	✓ (Fine-grained)	✓	✓

Figure 1: LiDAR scenes generated by different methods on the nuScenes val split. DriveLiDAR4D is the first work to achieve sequential LiDAR scene generation with highly controllable scene manipulation abilities, including foreground control, background control and object-fidelity enhancement.

RangeLDM (Hu, Zhang, and Hu 2024) and Text2LiDAR (Wu et al. 2024) are unable to generate the vehicle accurately, and backgrounds do not align with those observed in the Ground Truth(GT) scene. In contrast, the vehicles' positions and structures, and the background generated by DriveLiDAR4D closely align with those in the GT scene. Furthermore, it is noteworthy that DriveLiDAR4D is capable of generating a sequence of LiDAR scenes that maintain temporal consistency, whereas RangeLDM (Hu, Zhang, and Hu 2024) and Text2LiDAR (Wu et al. 2024) can only create individual LiDAR scenes in isolation.

To summarize, our specific contributions are as follows:

- This is the first work to achieve precise control over foreground objects, including manipulation of their positions and sizes, and fine-grained control over background elements in LiDAR scene generation.
- We propose a novel equirectangular spatio-temporal diffusion model, LiDAR4DNet, which achieves end-to-end *sequential* LiDAR scene generation while ensuring consistency over foreground and background elements.
- We demonstrate the effectiveness of our proposed DriveLiDAR4D on the KITTI and nuScenes datasets, where it outperforms the current SOTA methods.

2 Related Work

2.1 Diffusion Models

Diffusion models have been the de-facto choice in image and video generation tasks (Podell et al. 2023; Gupta et al. 2024; Yang et al. 2024).

Research in this field encompasses various aspects of improvement in diffusion policy (Croitoru et al. 2023; Liu, Gong, and Liu 2023), model architecture (Bar-Tal et al. 2024; Peebles and Xie 2023), conditioning strategies (Zhang, Rao, and Agrawala 2023; Khachatryan et al. 2023), among others. For a comprehensive survey of image and video generation, we refer readers to (Cao et al. 2024). However, these methods are primarily designed for visual modality data. LiDAR data presents distinct challenges compared to visual data, as it describes a scene with unordered and unevenly distributed points. This fundamental difference makes it challenging to effectively leverage diffusion models for the LiDAR modality.

2.2 LiDAR Scene Generation

Recent LiDAR data generation approaches can be categorized as either unconditional or conditional generation. In the unconditional category, methods such as LiDARGen (Zyrianov, Zhu, and Wang 2022), R2DM (Nakashima and Kurazume 2024), and LidarGRIT (Haghighi et al. 2024) train diffusion models directly on the pixel space, while RangeLDM additionally employs VAE (Kingma, Welling et al. 2013) to compress equirectangular images. In the conditional category, Text2LiDAR (Wu et al. 2024) conditions the diffusion process on coarse textual scene descriptions. LiDM (Ran, Guizilini, and Wang 2024) investigates various conditioning inputs but applies each condition independently, limiting its comprehensive manipulation capabilities. Our object priors condition shares similarities with OLiDM (Yan et al. 2024), which utilizes synthetic objects as conditions for scene generation. However, OLiDM (Yan et al. 2024) is designed to generate single LiDAR scenes, whereas our approach incorporates multiple multimodal conditions to generate *sequential* LiDAR scenes—a non-trivial task requiring careful pipeline design. LidarDM (Zyrianov et al. 2024) decomposes the generation task into two stages, static and dynamic object synthesis. But this approach compromises the realism of object point clouds due to its simplistic modeling of 3D object structures. Moreover,

We also see a growing research on jointly generating LiDAR point clouds and images. XDrive (Xie et al. 2025) uses two diffusion models to generate image and LiDAR point clouds simultaneously, where latent LiDAR features and latent image features attend to each other via epipolar guidance. Albeit, XDrive (Xie et al. 2025) is limited to single LiDAR scene generation. UniScene (Li et al. 2024) indirectly enforces temporal consistency across sequential frames by mediating cross-modal interactions through implicit conditioning on 3D occupancy priors. Nevertheless, it exhibits limitations in synthesizing high-fidelity objects.

In summary, current methodologies in LiDAR generation field predominantly focus on 3D scene generation with coarse conditioning, neglecting the integration of complete

temporal coherence with fine-grained controllability. To the best of our knowledge, DriveLiDAR4D is the first unified framework to resolve the aforementioned limitations by integrating a equirectangular spatio-temporal diffusion model with multimodal conditioning capabilities.

3 DriveLiDAR4D

Fig. 3 illustrates the pipeline of DriveLiDAR4D: During training, we first derive the three multimodal conditions: road sketches, scene captions and object priors. Then, LiDAR4DNet takes as input a sequence of noised equirectangular images, conditioned on the three multimodal conditions, and predicts the added noises. During inference, LiDAR4DNet reconstructs the sequence of equirectangular images, again utilizing the three multimodal conditions. In this section, we first detail the multimodal conditions and then elaborate on the specifics of LiDAR4DNet.

3.1 Preliminary

Our method is based on diffusion models, and we briefly introduce its principle in this section. We employ the denoising diffusion probabilistic model (DDPM). In DDPM, a forward diffusion process will gradually destroy a sample $\boldsymbol{x}$ by adding Gaussian noise at each timestep t: $\boldsymbol{x}_t = \alpha_t \boldsymbol{x} + \sigma_t \epsilon$. The variable $\boldsymbol{x}_t$ will finally arrive at Gaussian distribution, which can be written by

$$q(\boldsymbol{x}_t|\boldsymbol{x}) = \mathcal{N}(\alpha_t \boldsymbol{x}, \sigma_t^2 \boldsymbol{I}), \tag{1}$$

where α_t and σ_t are hyperparameters that determine the noise scheduling. We follow the α-cosine schedule (Saharia et al. 2022) where $\alpha_t = \cos(\pi t/2)$, $\sigma_t = \sin(\pi t/2)$. Consequently, the intermediate transition process of variable from timestep s to t ($0 < s < t < 1$) can be formulated as

$$q(\boldsymbol{x}_t|\boldsymbol{x}_s) = \mathcal{N}(\alpha_{t|s} \boldsymbol{x}_s, \sigma_{t|s}^2 \boldsymbol{I}), \tag{2}$$

where $\alpha_{t|s} = \alpha_t/\alpha_s$, $\sigma_{t|s}^2 = \sigma_t^2 - \alpha_{t|s}^2 \sigma_s^2$.

A reverse diffusion process is used to denoise the sample in the latent space, which can be represented as

$$p(\boldsymbol{x}_s|\boldsymbol{x}_t) = \mathcal{N}(\boldsymbol{\mu}_t(\boldsymbol{x}, \boldsymbol{x}_t), \Sigma_t^2 \boldsymbol{I}), \tag{3}$$

where $\boldsymbol{\mu}_t(\boldsymbol{x}, \boldsymbol{x}_t) = (\alpha_{t|s}\sigma_s^2/\sigma_t^2)\cdot\boldsymbol{x}_t + (\alpha_s \sigma_{t|s}^2/\sigma_t^2)\cdot\boldsymbol{x}$, $\Sigma_t^2 = \sigma_{t|s}^2 \sigma_s^2/\sigma_t^2$.

During training, with a noised sample $\boldsymbol{x}_t$, the model is learning to estimate the unknown $\boldsymbol{x}_s$ via estimating $\boldsymbol{x}$ in (3). Practically, we adopt ϵ-prediction (Saharia et al. 2022) by reparameterizing $\boldsymbol{x}$ as a function of $\boldsymbol{x}_t$ and ϵ. The loss function is denoted as

$$\mathcal{L} = \mathbb{E}_{\boldsymbol{x}, \boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}, t)} \left[\|\boldsymbol{\epsilon} - \hat{\boldsymbol{\epsilon}}(\boldsymbol{x}_t, t, \boldsymbol{c})\|^2 \right], \tag{4}$$

where $\boldsymbol{c}$ means conditions (will be described in Sec. 3).

Once the model is trained, we evaluate (3) iteratively as $t : 1 \rightarrow 0$ and the final sample is expected to approximate the data distribution. We set the number of iterations as 256 in denoising process.

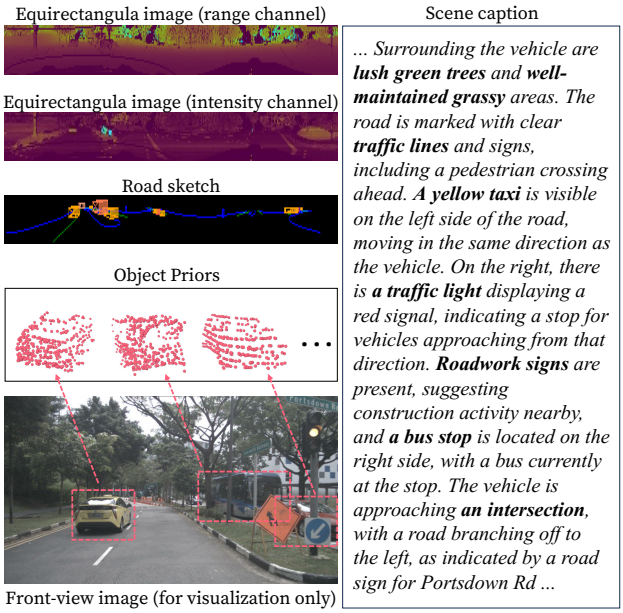


Figure 2: Visualization of the multimodal conditions of an example from the nuScenes dataset (Images have been resized for better visualization).

3.2 Multimodal Conditions

Road Sketch: The road sketch incorporates road layouts and object-specific information. Firstly, road layouts are delineated through curbs and lane lines. This representation provides a pixel-to-pixel control of the spatial organization of the road. Secondly, object-specific information is instantiated as 3D bounding boxes, which encapsulate the size, location, and heading direction of each object.

To integrate the two components into a unified representation, both the road layouts and 3D bounding boxes are projected onto the equirectangular image (will be detailed soon) plane of the LiDAR sensor. This results in a road sketch that has a same size as the input equirectangular image. Fig. 2 depicts an example of road sketch condition from the nuScenes dataset.

Scene Caption: In addition to foreground control by road sketches, we propose to use scene captions that provide a comprehensive description of the background. Unfortunately, existing LiDAR datasets lack high-quality scene captions: the KITTI-360 dataset (Liao, Xie, and Geiger 2022) does not include scene captions, while the nuScenes dataset (Caesar et al. 2020) provides only brief, one-sentence descriptions of the weather or time of day. Therefore, we leverage capabilities of the powerful vision-language model GPT-4V (Hurst et al. 2024) to generate detailed scene captions. Specifically, we input the surrounding images into GPT-4V and request descriptive text for the depicted scene. Fig. 2 depicts an example of the scene caption condition from the nuScenes dataset. In this example, the scene caption encompasses not only the foreground object "taxi", but also the background elements such as "trees" and "grassy areas".

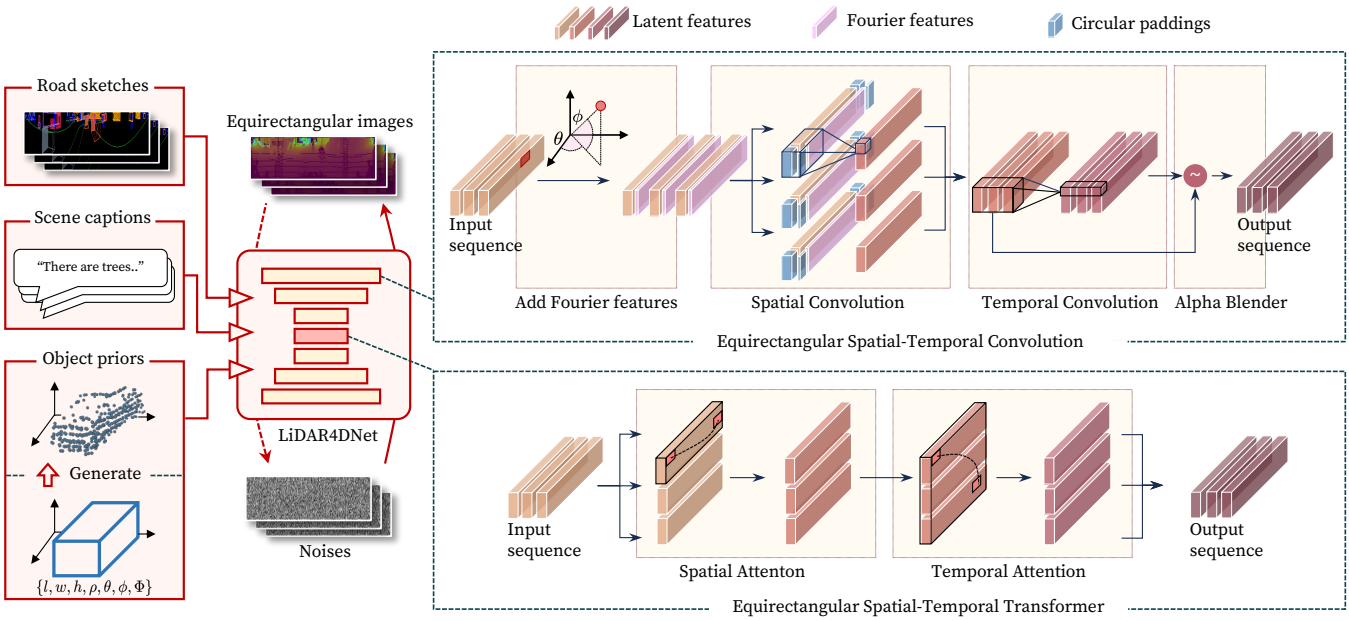


Figure 3: The pipeline of DriveLiDAR4D. We first derive multi-modal conditions, including road sketches, scene captions and object priors from a given road scene (see Sec. 3.2). Then, the proposed LiDAR4DNet predicts the sequential noises based on the multimodal conditions, where Equirectangular Spatial-Temporal Convolution (EST-Conv) and Equirectangular Spatial-Temporal Convolution (EST-Trans) enforce spatial and temporal consistency (see Sec. 3.3).

Object Priors: In autonomous driving scenarios, LiDAR measurements of different elements are often imbalanced; for example, a vehicle may be represented by hundreds of points, whereas the ground may encompass thousands. Diffusion models typically attempt to approximate the entire scene without addressing this imbalance, which can result in suboptimal generation quality for objects. To mitigate this issue, we propose initially synthesizing the point clouds of objects, which then serve as a condition to guide the model in generating the entire scene. The underlying intuition is that while road sketches indicate the locations of objects, the distribution of points for those objects is not adequately guided. Incorporating synthetic object point clouds as priors provides a stronger guidance during the denosing process.

We train an object generation model, DiT-3D (Mo et al. 2023), conditioned on the object's category, size, polar coordinates relative to the LiDAR sensor, and heading direction. We denote these conditions as $\{category, l, w, h, \rho, \theta, \phi, \Phi\}$. The object priors are obtained by generating the point clouds of the objects using the pretrained object generation model. We then project these object points onto the equirectangular image plane of the LiDAR sensor.

3.3 LiDAR4DNet

We follow (Nakashima and Kurazume 2024; Wu et al. 2024) and adopt equirectangular representation due to its efficiency in describing large scenes. A LiDAR sensor with $H \times W$ spatial resolution captures range and reflectance measurements at W azimuth angles and H elevation angles, resulting in an equirectangular image $\boldsymbol{x} \in \mathbb{R}^{2\times H\times W}$. We scale a range variable using $\hat{d} = {\log(d+1)}/{\log(d_{max}+1)}$ and subsequently normalize it to the range $[-1, 1]$.

As is depicted in Fig. 3, the proposed LiDAR4DNet is a UNet-like encoder-decoder model (Saharia et al. 2022), incorporating stacked Equirectangular Spatial-Temporal Convolution (EST-Conv) modules across four scales, along with an Equirectangular Spatial-Temporal Transformer (EST-Trans) module at the bottleneck.

EST-Conv: Research in RGB image generation has explored 3D convolutions for learning temporal features. However, equirectangular images possess distinct characteristics compared to standard RGB images: 1) the pixel coordinates (θ, ϕ) exhibit strong correlation with LiDAR data patterns (e.g., regions that are in close proximity and are positioned low tend to represent the ground), 2) the left and right boundaries of equirectangular images represent a continuous region in LiDAR measuring space (particularly for LiDAR sensors with 360-degree horizontal field-of-view (FOV)), and 3) the pixel value distribution differs significantly from that of RGB images. The unique properties of equirectangular images make the optimum solution for processing sequential equirectangular images unknown.

Therefore, we propose EST-Conv, a novel approach designed to learn spatial and temporal features from equirectangular images. In EST-Conv, to enhance spatial consistency, we leverage the correlation between pixel coordinates and LiDAR measurements by extracting Fourier features (Tancik et al. 2020) for each pixel coordinate (θ, ϕ) and concatenating these features with the input equirectangular im-

	Venues	KITTI-360			nuScenes		
		FRD↓	MMD[1]↓	JSD↓	FRD↓	MMD[1]↓	JSD↓
LiDARGen (Zyrianov, Zhu, and Wang 2022)	ECCV2022	2040.1	3.87	0.067	-	19.00	0.160
R2DM[2] (Nakashima and Kurazume 2024)	ICRA2024	275.67	5.55	0.049	-	-	-
Text2LiDAR[2] (Wu et al. 2024)	ECCV2024	831.64	4.36	0.044	-	-	-
RangeLDM[2] (Hu, Zhang, and Hu 2024)	ECCV2024	2022.71	**0.75**	**0.035**	492.49	2.75	0.054
DriveLiDAR4D		**244.25**	3.31	0.042	**210.55**	**1.84**	**0.045**

1: MMD has been multiplied by 10^4.

2: We reproduced the results with their released source codes.

Table 1: Unconditional generation results of different methods on the KITTI-360 test set and the nuScenes val split.

ages. The integration of Fourier features helps the model better capture the underlying geometric patterns in LiDAR data. Moreover, equirectangular images capture a 360-degree horizontal field of view, implying that the left and right boundaries represent continuous 3D space. To better model this spatial continuity, we replace standard zero-padding in 2D convolutions with circular padding, enhancing feature learning across horizontal image boundaries. Sequential equirectangular features are expaned along the batch dimension prior to being processed by 2D convolutional operations.

To enforce temporal consistency, we adopt 3D convolutions to directly process sequential equirectangular features. The unique geometry of equirectangular projections means adjacent pixels may represent distant objects in 3D space. Therefore, we implement 3D convolutions with small kernel sizes (d=3, h=1, w=1). Finally, we employ an Alpha-blender (Blattmann et al. 2023) to combine spatial and temporal equirectangular features. The Alpha-blender can be written as $\boldsymbol{y} = \alpha \cdot \boldsymbol{x}_S + (1-\alpha) \cdot \boldsymbol{x}_T$, where α is a learnable parameter, $\boldsymbol{x}_S$ and $\boldsymbol{x}_T$ are spatial and temporal features respectively.

EST-Trans: Convolutions efficiently process features with progressively increasing receptive fields but exhibit limitations in capturing long-range correlations. On the other hand, attention mechanisms can effectively model long-range dependencies, albeit with significant computational overhead. To effectively model long-range dependencies, we introduce EST-Trans and apply it exclusively at the bottleneck layer as illustrated in Fig. 3. We emperically found that this design strike a good balance between performance and efficiency. In EST-Trans, input sequences are stacked along the batch dimension before applying spatial attention and then expanded before applying temporal attention. Spatial attention captures relationships across individual equirectangular images, while temporal attention facilitates feature interactions among sequential equirectangular images. This architecture minimizes computational costs while simultaneously improving both spatial and temporal consistency.

Injecting Multimodal Conditions: Due to the distinct characteristics of the three conditioning types, we implement tailored strategies for each. For road sketches, we employ channel concatenation due to their precise pixel-to-pixel correspondence with equirectangular images. For object priors, which may object involve shape transformations, we leverage ControlNet (Zhang, Rao, and Agrawala 2023) conditioning strategy to provide enhanced learning capacity. For scene captions, in addition to applying cross-attention (Saharia et al. 2022), we fuse timestep variables with captions to prevent captions' influence from being overshadowed by the other two conditions. Additional architecture details are provided in the supplementary materials.

4 Experimental Results

4.1 Experimental Settings

Datasets. We validated the effectiveness of DriveLiDAR4D on the two real-world autonomous driving datasets, the KITTI-360 (Liao, Xie, and Geiger 2022) and nuScenes (Caesar et al. 2019) datasets. The KITTI-360 dataset features a Velodyne-64E LiDAR with 64 beams and 360-degree horizontal FOV. Following existing works (Nakashima and Kurazume 2024; Wu et al. 2024), we designated sequences 0 and 1 as the test set (26367 frames) and utilized the remaining sequences as training set (50348 frames). The nuScenes dataset provides 32-beam LiDAR measurements with 360-dergree horizontal FOV at 20Hz and keyframe annotations at 2Hz. Following common practice (Gao et al. 2024), we expanded the nuScenes bounding box labels to 12Hz and adhered to the official training and validation dataset splits (700 training scenes and 150 validation scenes), resulting in 165280 training samples and 35364 validation samples.

4.2 Unconditional LiDAR Scene Generation

We compare DriveLiDAR4D against SOTA LiDAR scene generation methods, LiDARGen (Zyrianov, Zhu, and Wang 2022), R2DM (Nakashima and Kurazume 2024), Text2LiDAR(Wu et al. 2024), and RangeLDM (Hu, Zhang, and Hu 2024). Tab. 1 illustrates the unconditional generation results on the KITTI-360 test set and the nuScenes val split. It can be seen that DriveLiDAR4D achieves comparable generation quality than RangeLDM on the KITTI-360 dataset, with superior FRD (Wu et al. 2024) and slightly inferior MMD (Nakashima and Kurazume 2024) and JSD (Nakashima and Kurazume 2024). However, DriveLiDAR4D surpasses RangeLDM on the nuScenes dataset in all three metrics, FRD, MMD, and JSD. Overall, these results demonstrates superiority of DriveLiDAR4D among all compared methods.

	Road sketch	Scene caption	FRD↓	MMD[1]↓	JSD↓
Config A			210.55	1.84	0.045
Config B	✓		178.24	0.85	0.041
Config C	✓	✓	**175.17**	**0.85**	**0.039**

1: MMD has been multiplied by 10^4.

Table 2: Conditional generation results of DriveLiDAR4D with different conditions on the nuScenes val split.

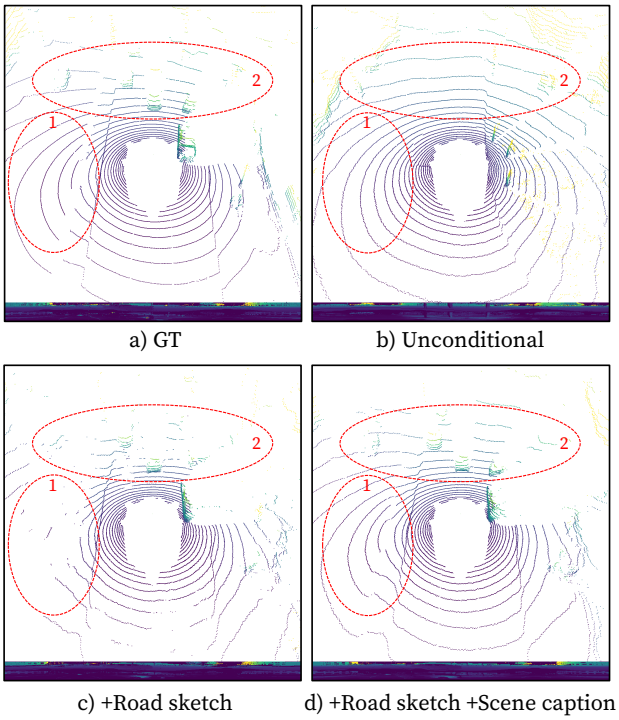


Figure 4: LiDAR scenes generated by DriveLiDAR4D with different conditions on the nuScenes val split.

4.3 Conditional LiDAR Scene Generation

We examine the impacts of road sketch and scene caption conditions by training LiDAR4DNet with subsets of the multimodal conditions. Tab. 2 presents results of DriveLiDAR4D with different conditions on the nuScenes val split. Obviously, the influence of road sketch conditioning is particularly significant, yielding a 15.34% improvement in FRD (from 210.55 to 178.24) and a 53.80% enhancement in MMD (from 1.84 to 0.85) compared to unconditional generation. This is attributed to the precise information about the road structure and the foreground objects embedded within the road sketches. Scene captions contribute to further enhancing the realism of the generated point clouds by providing complementary contextual information.

Fig. 4 shows the qualitative comparison of the LiDAR scenes generated by DriveLiDAR4D with different conditions on the nuScenes val split. Without any conditioning, the generated LiDAR scene lacks specific object structures (as shown in Fig. 4b). When conditioned solely on road sketches (as shown in Fig. 4c), DriveLiDAR4D successfully controls object locations in the generated scene, though the background area (red circle 1) is incomplete. Upon incorporating scene caption conditioning (shown in Fig. 4d), both foreground objects (red circle 2) and background elements (red circle 1) exhibit more complete structures, demonstrating the complementary benefits of this additional conditioning. This improvement stems from the comprehensive nature of scene captions, which encapsulate both foreground and background contextual information.

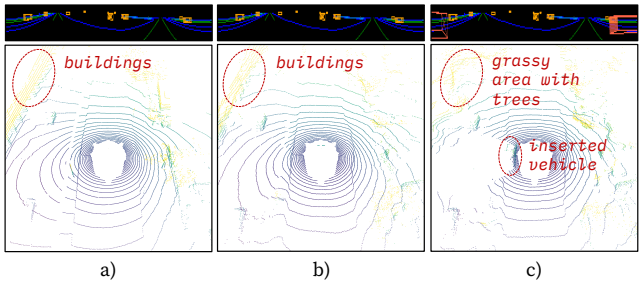


Figure 5: Visualization of fine-grained scene manipulation of DriveLiDAR4D on the nuScenes val split. Top row: road sketches. Bottom row: LiDAR scenes. a) GT scene, b) Generated scene with multimodal conditions, c) Generated scene with *edited* multimodal conditions.

Furthermore, Fig. 5 demonstrates an example of fine-grained manipulation of road elements in a generated LiDAR scene. In Fig. 5b, we re-simulated the real scene with same multimodal conditions as those of GT scene, where we can see that the generated scene closely align with GT scene (e.g., see buildings). In Fig. 5c, we inserted an vehicle and changed the background. This manipulation is achieved through edited multimodal conditions comprising 1) addition of a 3D box to the road sketch, 2) extraction of he corresponding object prior, and 3) modification of the scene caption from {... *building* ...} to {... *grassy area with trees*...}. The resulting scene successfully incorporates these manipulations while maintaining realism.

4.4 Sequential and Controllable LiDAR Scene Generation

	MMD[1]↓	JSD↓	FRD↓	FVD↓
LidarDM	25.53	0.155	1800.18	28.48
UniScene	21.66	0.143	1182.94	21.04
DriveLiDAR4D	**2.94**	**0.069**	**743.13**	**16.96**

1: MMD has been multiplied by 10^4.

Table 3: Sequential generation results of different methods on the nuScenes val split.

We compare DriveLiDAR4D with the SOTA sequential LiDAR generation methods, LidarDM (Zyrianov et al. 2024)

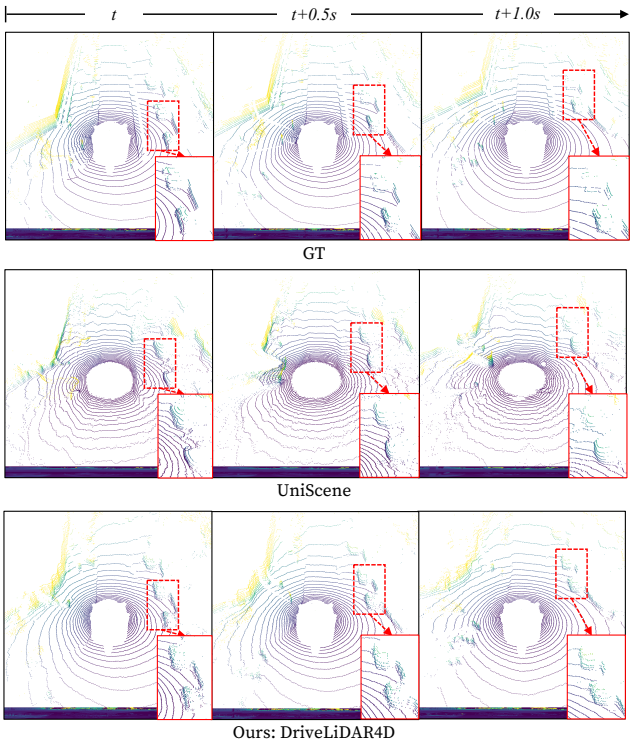


Figure 6: Sequential LiDAR scenes generated by different methods on the nuScenes val split. Note that DriveLiDAR4D generates sequences of 20 frames, of which we present three representative frames for a clear comparison. (Please refer to the supplementary material for more visual comparisons.)

and UniScene (Li et al. 2024). UniScene is a joint RGB-LiDAR generation approach, and we evaluate the quality of its generated LiDAR scenes. It is important to note the difference in sequence length capabilities: LidarDM and UniScene generate sequences of 5 and 8 frames respectively, whereas DriveLiDAR4D generates 20 frames at a time. The FVD metric is computed using the the first 5 frames of all the generated sequences to ensure a fair comparison.

Tab. 3 presents the results of different methods on the nuScenes val split. It can be seen that DriveLiDAR4D achieves lower FRD, MMD and JSD scores than LidarDM and UniScene, indicating improved spatial fidelity and diversity in the generated scenes. In addition, DriveLiDAR4D attains the lowest FVD (Li et al. 2024) score across the comparing methods, indicating superior temporal consistency in the generated sequences .

Fig. 6 illustrates the qualitative comparison of the sequential LiDAR scene generated by different methods on the nuScenes val split. It can be seen that the objects' structures are mostly broken in UniScene's generated scenes, and the objects' locations diverges from the GT locations (as evident in the zoomed-in views). In contrast, DriveLiDAR4D yields better objects structures and accurate locations, benefiting from the rich information provided by multimodal

	EST-Conv	EST-Trans	FRD↓	FVD↓
Config A			318.05	29.50
Config B	✓		311.38	17.28
Config C	✓	✓	**292.28**	**14.86**

Table 4: Sequential generation results of DriveLiDAR4D with different module configurations.

conditioning. Furthermore, DriveLiDAR4D exhibits better temporal consistency with coherent vehicle representations across frames, which is attributed to the EST-Conv and EST-Trans components of LiDAR4DNet.

4.5 The Impact of Model Configuration

We study the impact of EST-Conv and EST-Trans of LiDAR4DNet by evaluating the following model configurations: *Config A* refers to the model where we remove EST-Trans in DriveLiDAR4D and replace EST-Conv with regular 2D Convolution blocks. *Config B* refers to the model where we remove EST-Trans in DriveLiDAR4D. *Config C* is the DriveLiDAR4D model.

By comparing *Config A* and *Config B* in Tab. 4, it is evident that the EST-Conv is critical in enhancing the temporal consistency of the generated LiDAR scenes. By capturing and correlating local features across consecutive frames, the EST-Conv module enables the model to generate temporally coherent and high-quality point clouds. Consequently, both FRD and FVD scores are reduced. Furthermore, *Config C* yields additional FRD and FVD improvements compared to *Config B*, indicating that EST-Trans substantially enhances spatio-temporal consistency in the generated point clouds.

4.6 Real-world 3D Object Detection Evaluation

	mAP ↑	GT Agg.[1]↑
GT	0.804	100.0%
UniScene (Li et al. 2024)	0.078	9.7%
LidarDM (Zyrianov et al. 2024)	0.140	17.4%
DriveLiDAR4D w.o. object priors	0.123	15.3%
DriveLiDAR4D	**0.407**	**50.6%**

1: GT Agg. denotes aggrements with mAP of GT.

Table 5: 3D object detection results on LiDAR scenes generated by different method on the nuScenes val split. (Please refer to the supplementary material for a visualization of detected objects.)

The accuracy of object localization and the fidelity of objects' points are essential in real-world autonomous driving scenarios, as these factors significantly impact the performance of perception systems. Tab. 5 presents the 3D object detection results for *cars* on LiDAR scenes generated by different methods on the nuScenes val split. It can be observed that UniScene and LidarDM only achieved a mAP of 0.078 and 0.140, respectively, lagging significantly from

the results on the real-world GT data. In comparison, DriveLiDAR4D achieves a mAP of 0.407, indicating strongest agreement, 50.6%, with GT data. The ablated variant, DriveLiDAR4D w.o. object priors, yields a mAP of 0.123, showing that object priors plays an important role in enhancing object-level fidelities. This is because object priors provide direct and dense guide on objects' point distributions.

5 Conclusion

In this paper, we introduce DriveLiDAR4D, a novel 4D LiDAR scene generation pipeline that incorporates three multimodal conditions and a equirectangular spatial-temporal noise prediction model, LiDAR4DNet. The multimodal conditions enable precise scene manipulation, encompassing both foreground objects, background elements and object-level fidelity enhancement. Concurrently, LiDAR4DNet ensures spatial and temporal consistency in the generated sequential LiDAR scenes. We believe that the realistic 4D LiDAR scenes generated by DriveLiDAR4D would contribute significantly to the development and evaluation of real-world autonomous driving systems.

References

Bar-Tal, O.; Chefer, H.; Tov, O.; Herrmann, C.; Paiss, R.; Zada, S.; Ephrat, A.; Hur, J.; Liu, G.; Raj, A.; et al. 2024. Lumiere: A space-time diffusion model for video generation. In *SIGGRAPH Asia 2024 Conference Papers*, 1–11.

Blattmann, A.; Dockhorn, T.; Kulal, S.; Mendelevitch, D.; Kilian, M.; Lorenz, D.; Levi, Y.; English, Z.; Voleti, V.; Letts, A.; Jampani, V.; and Rombach, R. 2023. Stable Video Diffusion: Scaling Latent Video Diffusion Models to Large Datasets. *arXiv*, abs/2311.15127.

Caesar, H.; Bankiti, V.; Lang, A. H.; Vora, S.; Liong, V. E.; Xu, Q.; Krishnan, A.; Pan, Y.; Baldan, G.; and Beijbom, O. 2019. nuScenes: A multimodal dataset for autonomous driving. *arXiv preprint arXiv:1903.11027*.

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 *Computer Vision and Pattern Recognition (CVPR)*, 11618–11628.

Cao, H.; Tan, C.; Gao, Z.; Xu, Y.; Chen, G.; Heng, P.-A.; and Li, S. Z. 2024. A survey on generative diffusion models. *IEEE Transactions on Knowledge and Data Engineering*.

Croitoru, F.-A.; Hondru, V.; Ionescu, R. T.; and Shah, M. 2023. Diffusion models in vision: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(9): 10850–10869.

Gao, R.; Chen, K.; Xie, E.; Hong, L.; Li, Z.; Yeung, D.-Y.; and Xu, Q. 2024. MagicDrive: Street View Generation with Diverse 3D Geometry Control. In *International Conference on Learning Representations*.

Gupta, A.; Yu, L.; Sohn, K.; Gu, X.; Hahn, M.; Li, F.-F.; Essa, I.; Jiang, L.; and Lezama, J. 2024. Photorealistic video generation with diffusion models. In *European Conference on Computer Vision*, 393–411. Springer.

Haghighi, H.; Samadi, A.; Dianati, M.; Donzella, V.; and Debattista, K. 2024. Taming Transformers for Realistic Lidar Point Cloud Generation. *arXiv preprint arXiv:2404.05505*.

Hu, Q.; Zhang, Z.; and Hu, W. 2024. Rangeldm: Fast realistic lidar point cloud generation. In *European Conference on Computer Vision*, 115–135. Springer.

Hurst, A.; Lerer, A.; Goucher, A. P.; Perelman, A.; Ramesh, A.; Clark, A.; Ostrow, A.; Welihinda, A.; Hayes, A.; Radford, A.; et al. 2024. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*.

Khachatryan, L.; Movsisyan, A.; Tadevosyan, V.; Henschel, R.; Wang, Z.; Navasardyan, S.; and Shi, H. 2023. Text2video-zero: Text-to-image diffusion models are zero-shot video generators. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 15954–15964.

Kingma, D. P.; Welling, M.; et al. 2013. Auto-encoding variational bayes.

Li, B.; Guo, J.; Liu, H.; Zou, Y.; Ding, Y.; Chen, X.; Zhu, H.; Tan, F.; Zhang, C.; Wang, T.; Zhou, S.; Zhang, L.; Qi, X.; Zhao, H.; Yang, M.; Zeng, W.; and Jin, X. 2024. UniScene: Unified Occupancy-centric Driving Scene Generation. *arXiv*, abs/2412.05435.

Liao, Y.; Xie, J.; and Geiger, A. 2022. KITTI-360: A Novel Dataset and Benchmarks for Urban Scene Understanding in 2D and 3D. *Pattern Analysis and Machine Intelligence (PAMI)*.

Liu, X.; Gong, C.; and Liu, Q. 2023. Flow Straight and Fast: Learning to Generate and Transfer Data with Rectified Flow. In *International Conference on Learning Representations (ICLR)*.

Mo, S.; Xie, E.; Chu, R.; Hong, L.; Nießner, M.; and Li, Z. 2023. DiT-3D: Exploring Plain Diffusion Transformers for 3D Shape Generation. In *Conference on Neural Information Processing Systems (NeurIPS)*.

Nakashima, K.; and Kurazume, R. 2024. Lidar data synthesis with denoising diffusion probabilistic models. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, 14724–14731. IEEE.

Peebles, W.; and Xie, S. 2023. Scalable Diffusion Models with Transformers. In *IEEE International Conference on Computer Vision (ICCV)*, 4172–4182.

Podell, D.; English, Z.; Lacey, K.; Blattmann, A.; Dockhorn, T.; Müller, J.; Penna, J.; and Rombach, R. 2023. Sdxl: Improving latent diffusion models for high-resolution image synthesis. *arXiv preprint arXiv:2307.01952*.

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.

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.

Tancik, M.; Srinivasan, P.; Mildenhall, B.; Fridovich-Keil, S.; Raghavan, N.; Singhal, U.; Ramamoorthi, R.; Barron, J.; and Ng, R. 2020. Fourier features let networks learn high frequency functions in low dimensional domains. *Advances in neural information processing systems*, 33: 7537–7547.

Wu, Y.; Zhang, K.; Qian, J.; Xie, J.; and Yang, J. 2024. Text2LiDAR: Text-guided LiDAR Point Cloud Generation via Equirectangular Transformer. In *European Conference on Computer Vision*, 291–310. Springer.

Xie, Y.; Xu, C.; Peng, C.; Zhao, S.; Ho, N.; Pham, A. T.; Ding, M.; Zhan, W.; and Tomizuka, M. 2025. X-Drive: Cross-modality Consistent Multi-Sensor Data Synthesis for Driving Scenarios. In *The Thirteenth International Conference on Learning Representations*, volume abs/2411.01123.

Xiong, Y.; Ma, W.-C.; Wang, J.; and Urtasun, R. 2023. Ultralidar: Learning compact representations for lidar completion and generation. *arXiv preprint arXiv:2311.01448*.

Yan, T.; Yin, J.; Lang, X.; Yang, R.; Xu, C.-Z.; and Shen, J. 2024. OLiDM: Object-aware LiDAR Diffusion Models for Autonomous Driving. *arXiv preprint arXiv:2412.17226*.

Yang, Z.; Teng, J.; Zheng, W.; Ding, M.; Huang, S.; Xu, J.; Yang, Y.; Hong, W.; Zhang, X.; Feng, G.; et al. 2024. Cogvideox: Text-to-video diffusion models with an expert transformer. *arXiv preprint arXiv:2408.06072*.

Zhang, L.; Rao, A.; and Agrawala, M. 2023. Adding Conditional Control to Text-to-Image Diffusion Models. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, 3813–3824. IEEE.

Zyrianov, V.; Che, H.; Liu, Z.; and Wang, S. 2024. LidarDM: Generative LiDAR Simulation in a Generated World. *arXiv preprint arXiv:2404.02903*.

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