electronics

MDPI

Article

Discrete Diffusion-Based Generative Semantic Scene Completion

Yiqi Wu [1,2,3], Xuan Huang [2], Boxiong Yang [1,*], Yong Chen [4], Fadi Aburaid [5] and Dejun Zhang [2,3]

[1] School of Information & Intelligence Engineering, University of Sanya, Sanya 572022, China; wuyq@cug.edu.cn
[2] School of Computer Science, China University of Geosciences, Wuhan 430078, China; hx_1526@cug.edu.cn (X.H.); zhangdejun@cug.edu.cn (D.Z.)
[3] Hubei Key Laboratory of Intelligent Robot, Wuhan Institute of Technology, Wuhan 430205, China
[4] Geely Automobile Research Institute (Ningbo) Co., Ltd., Ningbo 315336, China; yong.chen@geely.com
[5] Department of Engineering, University of Technology and Applied Sciences, Mussanah 314, Oman; fadi.aburaid@utas.edu.om
* Correspondence: boxiongyang@sanyau.edu.cn

Abstract: Semantic scene completion through AI-driven content generation is a rapidly evolving field with crucial applications in 3D reconstruction and scene understanding. This task presents considerable challenges, arising from the intrinsic data sparsity and incomplete nature of input points generated by LiDAR. This paper proposes a generative semantic scene completion method based on a discrete denoising diffusion probabilistic model to tackle these issues. In the discrete diffusion phase, a weighted *K*-nearest neighbor uniform transition kernel is introduced based on feature distance in the discretized voxel space to control the category distribution transition processes by capturing the local structure of data, which is more in line with the diffusion process in the real world. Moreover, to mitigate the feature information loss during point cloud voxelization, the aggregated point features are integrated into the corresponding voxel space, thereby enhancing the granularity of the completion. Accordingly, a combined loss function is designed for network training that considers both the KL divergence for global completion and the cross-entropy for local details. The evaluation, which results from multiple public outdoor datasets, demonstrates that the proposed method effectively accomplishes semantic scene completion.

Keywords: semantic scene completion; diffusion model; generative model; point cloud

check for updates

Academic Editor: Ioannis Hatzilygeroudis

Received: 14 February 2025
Revised: 27 March 2025
Accepted: 1 April 2025
Published: 3 April 2025

Citation: Wu, Y.; Huang, X.; Yang, B.; Chen, Y.; Aburaid, F.; Zhang, D. Discrete Diffusion-Based Generative Semantic Scene Completion. *Electronics* **2025**, *14*, 1447. https://doi.org/10.3390/electronics14071447

Copyright: © 2025 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

Understanding 3D environments is a natural ability of humans, and extracting meaningful geometry and semantic information from incomplete scenes to infer the overall context is an important process in 3D environment understanding. Semantic features have been shown to improve various tasks, such as deraining [1] and desnowing [2]. Completing semantic scenes in outdoor settings presents significant challenges due to the inherent sparsity and incompleteness of point clouds. Achieving a comprehensive understanding and reconstruction of these scenes is crucial for enhancing scene analysis, thus supporting more effective monitoring and modeling of complex environments [3,4]. In practice, scene understanding is often achieved through semantic segmentation. When processing the inputs of scene point clouds, environment understanding is reflected not only in the semantic segmentation of points but also in the completion and prediction of missing areas.

Traditional segmentation and completion tasks for outdoor scene point clouds are usually studied independently [5,6]. Considering the potential intrinsic correlation between semantic and geometric structures in scenes, along with the availability of large-scale public

outdoor datasets like CarlaSC [7] and SemanticKITTI [8], related researchers have proposed the task of semantic scene completion, which leverage the combined learning of scene segmentation and completion to enhance the understanding of 3D environments [9–13]. For instance, S3CNet [10] jointly accomplishes 2D and 3D completion tasks and has demonstrated notable performance on the outdoor SemanticKITTI dataset. Similarly, JS3C-Net [12] initially conducts semantic segmentation and then provides the learned segmentation feature information to the subsequent completion network. It further proposes a hierarchical refinement module to improve the quality of completion.

Despite significant progress in this field, these methods lack the ability to infer and generate scenes, and they exhibit unsatisfactory completion performance on small objects and crowded scenes. Learnable generative modeling is a fundamental issue in deep learning, which is essential for the ability to capture the statistical characteristics of natural datasets and for generating content (such as 3D point clouds, images, or texts) for downstream applications. It has been proven effective for 3D scene and 2D image completion tasks [14–17]. The denoising diffusion probabilistic model (DDPM) [18] has emerged as a compelling alternative for content generation. The quality of samples produced by DDPM is comparable to that of Generative Adversarial Networks (GANs) [19], and their log-likelihood is on par with autoregressive models, all while requiring fewer inference steps.

While previous diffusion-based generation works mainly concentrate on Gaussian diffusion within continuous state spaces, diffusion models have also been developed for discrete spaces. Li et al. [20] introduced the discrete diffusion model applied to the task of semantic completion for 3D voxel data, which demonstrated its effectiveness in the generation and completion of the public outdoor scene dataset CarlaSC [7]. However, these works mainly focus on uniform diffusion in the discretized voxel space of the diffusion model, neglecting the transfer probabilities and its distance-related interactions between objects in real-world scenes. Additionally, voxelization neglects the fine geometric information of the original point cloud, which provides important clues about object shapes and scene layout, aiding in completing sparse scenes and small objects.

This paper proposes a generative semantic scene completion method that consumes sparse scene point clouds and simultaneously generates complete scenes and semantic layouts based on DDPM. Specifically, for the discrete diffusion process, a point feature compensation module is introduced to integrate point cloud geometric information into the corresponding voxel space, helping to complete sparse scenes and small objects. Meanwhile, a weighted K-nearest neighbor uniform transition kernel is designed to constrain the state transition process. In this way, the proposed method effectively captures the local structures of complex scenes, thus contributing to the quality and stability of the network's generation. The main contributions of the proposed method can be summarized as follows:

- A learnable generative network based on the denoising diffusion probabilistic model is proposed for accomplishing the task of semantic scene completion with a focus on incorporating fine details.
- The aggregated point features are extracted and fused into the corresponding voxel space for content generation, enabling granular completion of sparse scenes and small-volume objects.
- A weighted K-nearest neighbor uniform transition kernel based on feature distance is proposed. It controls the category distribution transition process by amplifying the influence of similar voxels within local regions.
- The designed network is evaluated on the public datasets. The visualized performance and quantitative comparisons demonstrate its superiority.

The remainder of the paper is structured as follows: Section 2 provides a review of related work on semantic scene completion, while Section 3 provides a detailed overview of

the learnable generative network designed in the paper. The proposed network is evaluated on public datasets, as detailed in Section 4. Finally, Section 5 comprehensively summarizes the designed network and provides an outlook for future research.

2. Related Works

This paper primarily focuses on the study of a learnable generative network for semantic scene completion. Subsequently, this section briefly introduces and discusses three related aspects: the background of semantic scene completion, generative models, and discrete diffusion generative models.

2.1. Semantic Scene Completion

Early work on scene completion primarily focused on indoor scenarios, where point clouds are dense, small-scale, and uniformly distributed. In contrast, outdoor scenes present significant challenges for semantic scene completion algorithms, largely attributable to the sparsity, large scale, and variations in density characteristic of the point clouds [13,21].

Recent studies have delved into the field of semantic scene completion, highlighting its emergence as a burgeoning research direction. SSCNet [9] is the first to integrate scene segmentation and completion in an end-to-end manner, utilizing 3D Convolutional Neural Networks (CNNs). JS3C-Net [12] proposed a novel sparse LiDAR point cloud semantic completion framework, which first achieves initial semantic segmentation of the scene point cloud through the neural network, then uses it as input to the semantic scene completion module. Further research has also integrated additional semantic feature information from 2D image networks through projection [22–24]. Cai et al. [25] encoded LiDAR scans as spherical projections, enriching the neighboring information. Moreover, LMSCNET [11] utilizes the UNet network with skip connections as its backbone to extract rich feature information, resulting in remarkable performance in semantic scene completion. S3CNet [10] introduces a sparse convolution-based completion neural network, along with a 2D variant that incorporates a multi-view fusion method to enhance the 3D network, exhibiting significant robustness in the completion of occluded regions and sparse scenes at greater distances. While these methods achieve semantic scene completion, they are heavily dependent on voxel completion labels and exhibit limited effectiveness in handling small objects.

2.2. Generative Model

The task of filling in occlusions and sparse areas due to incomplete observations presents a significant challenge. Learnable generative models have been proven to be effective for both 2D/3D completion tasks [14,15]. Chen et al. [16] demonstrated that GANs [19] can be employed to enhance the rationality of completion results. On the other hand, GANs are recognized for experiencing mode collapse and having difficulty in accurately capturing complex scenes that contain multiple objects [26].

Diffusion models [18] are better at covering the entire data distribution to evade model collapse and generating complex scenes due to their likelihood-based ideas [27]. Recently, diffusion models have shown impressive performance on real-world 2D images. By introducing the diffusion process, complex distributions can be learned using simple distributions, such as Gaussian distributions. In particular, compared to GANs, diffusion models have shown impressive results in high-resolution image generation [26] and conditional generation [28]. The latest advancements [29,30] in diffusion models suggest that a more diverse data distribution can be learned through the diffusion process. Luo et al. [29] proposed a correspondence between the noisy point cloud and the original point cloud, enabling a simple yet effective mean square error loss function for training, resulting in

uniform and high-quality point cloud generation. LION [30] combines global shape latent representations with point structure latent space, forming a hierarchical latent space. In these latent spaces, two hierarchical denoising diffusion models are trained. The use of generative models holds promising prospects due to their potential and capability to handle high-dimensional data. In this paper, the application of discrete diffusion models in the domain of semantic scene completion is further explored.

2.3. Discrete Diffusion Model

Previous research on diffusion models has mostly focused on continuous data spaces. However, they perform poorly when processing discrete data scenarios (such as textual labels and time series) as continuous representations do not fully cover the discreteness of the data. To address this issue, discrete diffusion models have been investigated for a range of applications, including the generation of text and the segmentation maps in lower-dimensional space [31,32].

The task of semantic scene completion simultaneously infers the completion of an entire scene and its semantic labels. Given the discreteness of semantic labels, applying discrete diffusion models to this task is reasonable and effective. Sohl-Dickstein et al. [33] were the first to introduce diffusion models with discrete state spaces, focusing on the noise in binary random variables during the diffusion process. Hoogeboom et al. [32] extended the binary random variables model to include categorical random variables characterized by uniform transition probabilities. Song et al. [34] unified various variations of denoising diffusion models and explored the extension of the diffusion model to discrete latent spaces. D3PMs [31] described a more general framework of a discrete diffusion model with indiscriminate uniform diffusion in discrete space. Given the advantages of applying discrete diffusion models to semantic scene completion tasks, this paper proposes a learnable scene generation network based on a discrete diffusion model with a designed transition kernel.

3. Method

Semantic scene completion aims to deduce a full geometric and semantic structure of a scene from partial and sparse input data. A semantic scene completion network is designed in this paper, which maps a given sparse input point cloud to a regular voxel space. Then, the network generates a label for each voxel within a voxel space of dimensions $L \times W \times H$, indicating whether it is empty or assigned to a specific class $c_i \in C = \{c_0, c_1, \ldots, c_n\}$, with $n + 1$ categories. L, W, H representing the length, width, and height of the voxel space, respectively. The output of the completed scene with semantic information is represented as $\mathbf{V} \in \mathbb{R}^{L \times W \times H \times C}$. Specifically, the proposed network is a learnable generative model based on a discrete diffusion probabilistic model for scene completion, as depicted in Figure 1. The network operates in a discretized voxel space and learns the semantic scene data distribution of 3D scenes through a weighted K-nearest neighbor uniform transition kernel based on feature distance (Sections 3.2 and 3.3). With the consideration of the geometric information loss caused by the voxelization process, the network adopts a point feature compensation module (Section 3.1). This module learns the detailed geometric feature information from the LiDAR data and integrates the learned features into the corresponding voxel space, thereby enhancing the granularity of the completion. Meanwhile, a combined loss function consisting of diffusion and semantic loss is introduced for the network training, which not only considers the overall layout but also emphasizes the local details (Section 3.4).

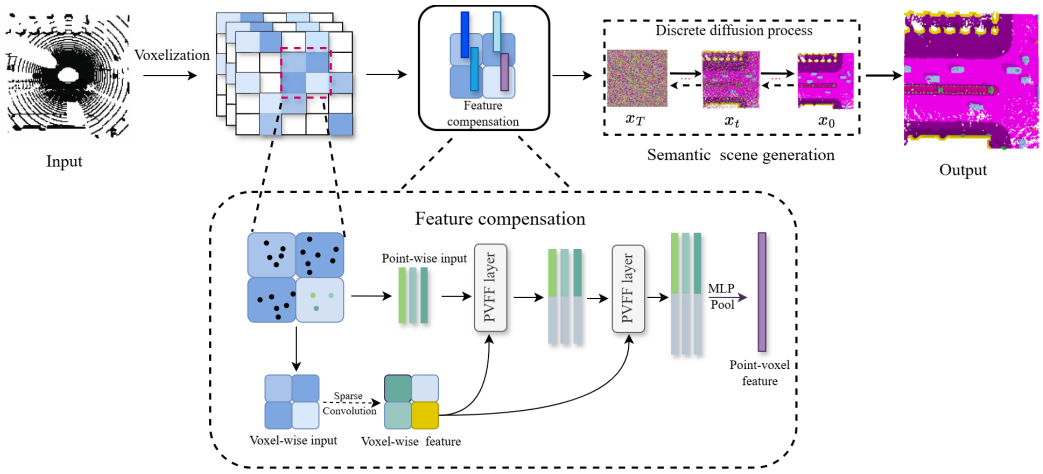


Figure 1. Pipeline of the proposed semantic scene completion network. The network uses sparse point clouds of outdoor scenes as input and generates a regular voxel tensor through point cloud voxelization. To address the information loss caused by voxelization, a feature compensation module is proposed to fuse voxel and corresponding spatial point cloud features. Finally, a weighted *K*-nearest neighbor uniform transition kernel based on feature distance is designed for a discrete diffusion process, amplifying the influence of similar voxels within local regions to learn the distribution of scene and semantic labels.

3.1. Point Feature Compensation

Existing research on semantic scene completion for point clouds, such as LMSCNET [11] and SCPNET [35], mostly involves transforming the data into voxel representations to handle large-scale scene data and efficiently utilize the convolution. However, due to voxel discretization, fine geometric information of the point cloud is often lost, and multiple points or even small objects may be merged into a single grid or fragmented into various grids, making them indistinguishable. In this case, smaller objects (e.g., pedestrians and cyclists) are disadvantaged compared to larger objects (e.g., cars). To address these issues, a point feature compensation module is designed before diffusion training, as depicted in the lower part of Figure 1.

Point Cloud Voxelization: Based on the size of a single scene in the given dataset, the point cloud is converted into a corresponding regular space by dividing the 3D space into evenly spaced voxels. Specifically, the coordinate value of each point is divided by the voxel's spatial size and then rounded to ensure that each point is mapped to the nearest voxel grid. The point's coordinates are thus converted into the index within the voxel grid, mapping the continuous space to a discrete voxel space. If a point's mapping result exceeds the grid boundaries, it is clamped to the grid's limits. During voxelization, the value of each voxel does not simply indicate whether it is occupied but rather records the number of points within the voxel, serving as the initial voxel input.

Point Grouping: Points are grouped based on the voxel they belong to. Due to factors such as distance, occlusion, uneven sampling, and the highly variable point density across the entire space, a voxel after grouping may contain a varying number of points. As shown in the feature compensation section of Figure 1, the number of points in the four voxel grids differs. Points within the same voxel space are considered as a group, and local features are aggregated and extracted to serve as the point feature compensation for that voxel.

Point-Voxel Feature Fusion: The feature compensation module in Figure 1 demonstrates a hierarchical fusion process of point-voxel features, with the key innovation being the point-voxel feature fusion (PVFF) layer. Specifically, the voxel $V = \{p_i = [x_i, y_i, z_i]^T \in \mathbb{R}^3\}_{i=1...t}$ represents a voxel space containing t points, where t is the initial voxel representation of V; each point p_i represents the coordinates of the i-th point. To obtain the features of each

voxel in the initial voxel input, a sparse convolution approach is employed. Sparse convolution reduces computational load and memory consumption by performing convolution operations only on non-zero voxels while maintaining a high level of feature extraction capability. This method leverages the sparsity of the voxel grid, targeting important regions for convolution to extract feature information for each voxel.

Then, the points within the voxel are input into the PVFF layer, where the points are aggregated into a local feature f_p, which is then fused with the voxel feature f_v. The output, which is a point feature with voxel information, serves as input for the next PVFF layer. After two rounds of point feature processing through the PVFF layers, the features are passed through a Multilayer Perceptron (MLP) and Max Pooling, selecting the most prominent features as the final voxel feature.

Point-Voxel Feature Fusion Layer: The PVFF layer is crucial for the fusion of point features and voxel features, as shown in Figure 2. The original points undergo multiple layers of MLP, including a linear layer, a batch normalization (BN) layer, and a ReLU activation layer. After obtaining the per-point feature representations, max pooling is applied to extract the point aggregation feature f_p from the voxel space V. The fusion of the point aggregation feature f_p and the voxel feature f_v first involves converting both through a linear layer to match the same dimensionality. The voxel feature f_v is then passed through a sigmoid activation and multiplied by the aggregated feature f_p to obtain the point-voxel fused feature. Finally, they are concatenated with the previous per-point feature to yield the final point-voxel feature.

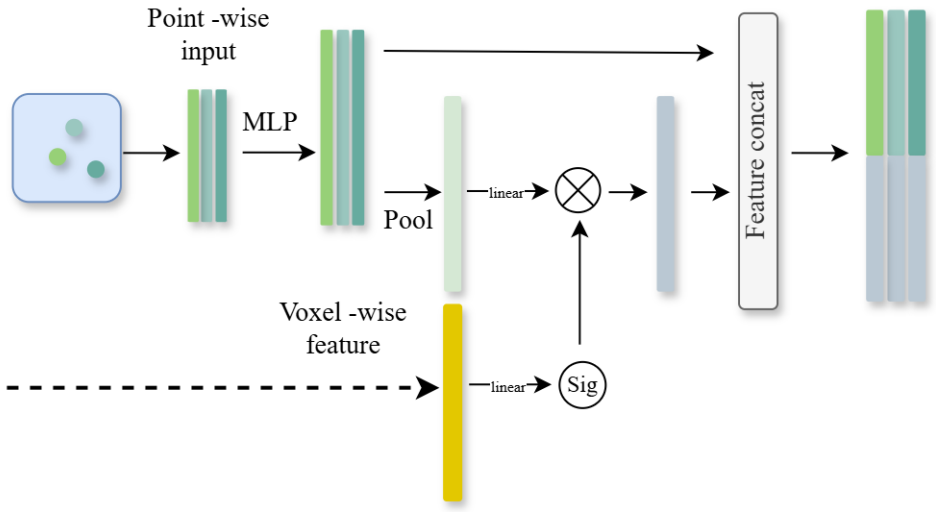


Figure 2. The design of the point-voxel feature fusion layer. This layer first extracts significant point features within the voxel space and then aggregates them with the voxel features to obtain the point-voxel fused feature.

3.2. Discrete Diffusion for Semantic Scene Completion

The discrete diffusion process for semantic scene completion is illustrated in Figure 3. It consists of the forward diffusion and the reverse generation. During the forward diffusion process, the original data distribution is gradually disrupted until the data become uniform noise. In contrast, the reverse generation process learns to recover the original data distribution from noise. Semantic scene completion is considered as a conditional generation task. In the reverse generation process, the method can infer a complete scene with corresponding semantic labels, given the sparse input s as the condition.

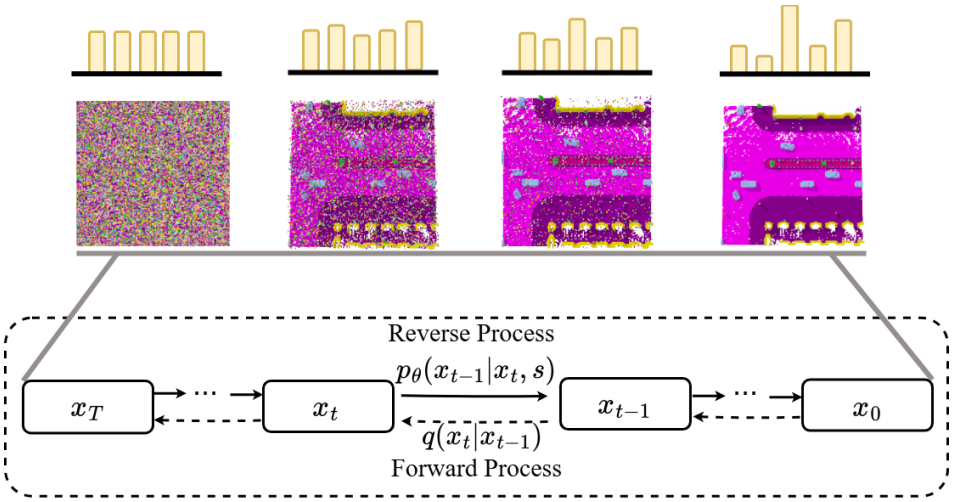


Figure 3. The process of discrete diffusion. From left to right, the reverse process $p_\theta(x_{t-1}|x_t, s)$ learns to gradually denoise, where s is the given voxel input with the point feature. From right to left, the forward diffusion process $q(\boldsymbol{x}_t|\boldsymbol{x}_{t-1})$ progressively adds noise, disrupting the original scene. The diffusion process is conducted in a discrete voxel space and class labels. The histograms in the figure intuitively show the multi-class distribution in the discrete diffusion process.

In the discrete diffusion process, the semantic variables at x_t belong to the semantic class space $c_i \in C = \{c_0, c_1, \ldots, c_n\}$. The forward process can be represented by a transition matrix Q_t, where each $[Q_t]_{ij}$ denotes the probability of transitioning from state $x_{t-1} = c_i$ to $x_t = c_j$. The state vector x is one-hot encoded, which ensures that each state is mutually exclusive. The forward transition probability is formulated as follows:

$$q(x_t|x_{t-1}) = \mathcal{C}(x_t|p = x_{t-1}Q_t), \tag{1}$$

where $\mathcal{C}(x|p)$ [31] is a categorical distribution over the one-hot row vector x with probabilities given by the row vector p, and $p = x_{t-1}Q_t$ is to be understood as a row vector-matrix product. This means that at time t, the transition from state x_{t-1} to state x_t is determined by the corresponding entry in the matrix Q_t.

In the forward process, the original x_0 is gradually decomposed into the noisy scene x_t; each forward step is represented by Q_t and is defined as $x_t = x_{t-1}Q_t$. Based on the Markov chain [36], the t-step noisy scene x_t can be derived from the original scene x_0, the cumulative transition matrix [31] $\bar{Q}_t = Q_1 Q_2 \cdots Q_t$, which converges to a known stationary distribution, and $q(x_t|x_0)$:

$$q(x_t|x_0) = \mathcal{C}(x_t|p = x_0\bar{Q}_t). \tag{2}$$

In the reverse process, a learnable parameter set θ is used to denoise the noisy scene and reconstruct the original semantic scene. Specifically, the learnable model $p_\theta(x_{t-1}|x_t, s)$ utilizes the sparse input s and to guide the denoising process, to guide the denoising process. Specifically, a reparameterization technique [18] is adopted to enable the network to infer the denoised mapping $\tilde{x}_0$, and then obtain the reverse probability distribution $p_\theta(x_{t-1}|x_t, s)$:

$$p_\theta(x_{t-1}|x_t, s) = q(x_{t-1}|x_t, \tilde{x}_0)p_\theta(\tilde{x}_0|x_t, s), \tag{3}$$

here, $q(x_{t-1}|x_t, \tilde{x}_0)$ can be expressed as follows using Bayes' theorem:

$$q(x_{t-1}|x_t, \tilde{x}_0) = \frac{q(x_t|x_{t-1}, \tilde{x}_0)q(x_{t-1}|\tilde{x}_0)}{q(x_t|\tilde{x}_0)}. \tag{4}$$

3.3. Weighted K-Nearest Neighbor Uniform Transition Kernel

The discrete diffusion model is able to control data corruption and denoising processes by selecting the transition matrix Q_t [6], thereby altering the Markov transition kernel. This stands in stark contrast with continuous diffusion, which primarily focuses on additive Gaussian noise. Hoogeboom et al. [32] later extended the binary random variables to categorical variables and employed a uniform diffusion transition matrix $Q_t = 1/C$, where the probability of transitioning from each state to any other state is the same, and therefore, the forward diffusion process can be presented as follows:

$$q(\boldsymbol{x}_t|\boldsymbol{x}_{t-1}) = \mathcal{C}(\boldsymbol{x}_t|(1-\boldsymbol{\beta}_t)\boldsymbol{x}_{t-1} + \boldsymbol{\beta}_t/C). \tag{5}$$

In the uniform diffusion process, global uniform interpolation is typically used to update the state. This method relies on a simple average-weighting strategy, estimating the value x_t by globally linearly weighting the current state x_{t-1}. This update assumes that all voxels contribute equally to the state transition, resulting in a smooth state transition. However, this uniform diffusion assumption has limitations, especially when dealing with scene voxel data that exhibit spatial correlations and local patterns. It may fail to capture the complex local structure inherent in the data. For instance, in some real-world scenarios, the relationships between voxels are often not uniform, and a single object may be split across multiple adjacent voxel spaces, influenced by similar voxels within local regions. Therefore, employing a uniform update strategy may not fully reflect the true relationships between voxels. To better capture the local structure of the data, a diffusion transfer method based on weighted *K*-nearest neighbor is proposed. In this method, the update of a voxel does not rely solely on a global linear combination but rather on the feature similarity between the voxel and its neighboring voxels. Specifically, the feature distance between each voxel and the selection of its *K*-nearest neighbors are computed first. Then, the update is weighted based on the feature distances to these nearest neighbor voxels, where voxels with smaller feature distances have a greater influence on the current voxel's state.

Weighted K-Nearest Neighbor: In weighted *K*-nearest neighbor diffusion, the first step is to calculate the feature distance between voxel x_{t-1} and other voxels to assess their similarity. Next, the *K*-nearest voxels are selected, and a weighted scheme is applied based on the feature distances between them. There are various designs for the weighting function, all of which emphasize the inverse relationship between feature distance and influence. The weighting function designed in this paper takes the form of a Gaussian kernel function:

$$w^k = \exp\left(-\frac{d(x_{t-1}, x_{t-1}^k)}{2\sigma^2}\right), 1 \le k \le K, \tag{6}$$

in this equation, $d(x_{t-1}, x_{t-1}^k)$ represents the feature distance between voxel x_{t-1} and its neighbor x_{t-1}^k, and σ is a hyperparameter that controls the smoothness of the weighting function.

Transition Matrix: Using the weighted *K*-nearest neighbor method, the state of x_{t-1} at time $t-1$ can be updated as follows:

$$x_t = x_{t-1} \cdot Q_t = \frac{1}{C}\sum_{k\in K} \omega^k \cdot x_{t-1}^k, \tag{7}$$

where *K* represents the *K*-nearest neighbors most similar to x_{t-1}, ω^k is the corresponding weight coefficient, and *C* denotes the semantic category space. Through this method, the state of the current voxel is strongly influenced by similar voxels within its neigh-

borhood and adjusted according to the feature distances, thereby better reflecting the local patterns.

Building upon the weighted K-nearest neighbor transition kernel, the state transition process can be further represented in the discrete diffusion probability model. Specifically, the forward diffusion process distribution $q(x_t|x_{t-1})$ for the transition from x_{t-1} at time $t-1$ to x_t at time t can be expressed as follows:

$$q(x_t|x_{t-1}) = \mathcal{C}(x_t|(1-\beta_t)x_{t-1} + \frac{1}{C}\beta_t \sum_{k\in K} \omega^k x_{t-1}^k). \tag{8}$$

where $\mathcal{C}(\cdot|\cdot)$ is multivariate categorical distribution, and β_t is a hyperparameter. Through this transition model, each voxel not only depends on its state at the previous time x_{t-1}, but also incorporates local structural information via the weighted K-nearest neighbor strategy, thereby more accurately capturing the spatial relationships and local patterns between voxels.

3.4. Training Objective

When training the proposed semantic scene completion network, a combined loss function is designed for the network. It takes into account the similarity of the semantic scene global distribution during the diffusion process, as well as the cross-entropy loss for detail optimization. Specifically, in the discrete diffusion generation process, the loss function consists of the Kullback–Leibler (KL) divergence [20] between the forward process $q(x_{t-1}|x_t, x_0)$ and the reverse process $p_\theta(x_{t-1}|x_t, s)$, as well as the KL divergence between the original scene $q(x_0)$ and the reconstructed scene $p_\theta(\tilde{x}_0|x_t, s)$, which is formalized as follows:

$$\mathcal{L}_{diffusion} = \mathcal{D}_{KL}(q(x_{t-1}|x_t, x_0) \parallel p_\theta(x_{t-1}|x_t, s)) + \lambda \mathcal{D}_{KL}(q(x_0) \parallel p_\theta(\tilde{x}_0|x_t, s)), \tag{9}$$

where λ is an auxiliary loss weight. Then, the cross-entropy loss [37] $\mathcal{L}_{ce}$ is employed to constrain the divergence between the generated semantic scene and the true semantic scene, optimizing the completion details. Therefore, the overall training objective is represented as follows:

$$\mathcal{L} = \mathcal{L}_{diffusion} + \mathcal{L}_{ce}. \tag{10}$$

4. Experiments

In this section, the qualitative and quantitative experimental results of the proposed method are presented and analyzed. Section 4.1 introduces the datasets employed for the experiments and provides details of the settings for training the network. In Section 4.2, the quantitative comparisons and analysis of the proposed method with other existing methods in terms of IoU are provided. The visualization results of semantic scene completion on CarlaSC [7], SemanticKITT [8], KITTI-360 [38], and SemanticPOSS [39] are shown in Section 4.3. Then, the ablation studies of the main components designed in this paper are presented in Section 4.4. Finally, Section 4.5 presents the robustness studies of the proposed method.

4.1. Dataset and Experimental Settings

The training and testing of the proposed network are conducted on four public scene datasets, CarlaSC [7], SemanticKITTI [8], KITTI-360 [38], and SemanticPOSS [39], which are also used in current semantic context completion works [11,20]. The CarlaSC dataset [20] comprises 24 scenes from 8 dynamic maps, featuring various traffic conditions ranging from low to high. For segmentation purposes, the dataset is divided into 18 scenes for

training, 3 for validation, and 3 for testing, each annotated with 10 semantic categories and an additional free label. The resolution of each scene is 128 × 128 × 8, which covers an area of 25.6 m in both front and rear directions, 25.6 m on either side, and a vertical height of 3 m. SemanticKITTI dataset [8] contains 8720 LiDAR point cloud scans and corresponding ground truth labels, divided into 22 point cloud sequences. Sequences 00–10 and 08 are utilized for training and validation, while sequences 11–21 are allocated for testing. The large-scale KITTI-360 dataset [38] comprises 320,000 images and 100,000 LiDAR scans, covering a driving distance of 73.7 km. It provides coarse bounding box annotations for both static and dynamic 3D scene elements and maps this information to the image domain, enabling dense semantic and instance annotations in both 3D point clouds and 2D images. Additionally, the SemanticPOSS dataset [39] for 3D semantic segmentation consists of 2988 LiDAR scans, capturing various complex scenes with a large number of dynamic instances. The data were collected at Peking University. KITTI-360 and SemanticPOSS follow the same format as SemanticKITTI.

For semantic scene completion, the completion dimensions on CarlaSC are 128 × 128 × 8, while the completion dimensions on SemanticKITTI, KITTI-360, and SemanticPOSS are 256 × 256 × 32. The network is trained on a Ubuntu 20.04 and an NVIDIA GeForce RTX A4000 GPU with CUDA 10.1 and PyTorch 1.11.0 environment.

4.2. Quantitative Comparisons

The qualitative results obtained from the CarlaSC [7] demonstrate the superiority of the method introduced in this paper. Comparisons are made between the proposed method and other state-of-the-art semantic completion methods on the CarlaSC dataset, including LMSCNET [11], SSCNET [9], MotionSC [7], Occsora [40] and Lee et al. [20].

The intersection-over-union (IoU) is employed as the evaluation metric, which is calculated as follows: $IoU = \frac{TP}{TP+FP+FN}$, where TP, FP, and FN represent true positive, false positive, and false negative. The IoU for each class is also adopted to elaborate detailed completion results for each class. Moreover, the mIoU, the mean of IoU for all classes, is also a key metric. It not only reflects the accuracy of semantic scene segmentation but also serves as an indicator of the completion effect.

Table 1 shows the IoU and mIoU comparisons with existing methods. The proposed method achieves the highest mIoU performance, which is the main evaluation indicator of the semantic scene completion task. The performance of IoU is also on par with other methods. In summary, the proposed method is capable of inferring complete and plausible semantic scenes from sparse point clouds. In addition, Table 2 provides a detailed presentation of the completion results for each class, allowing for a more intuitive analysis and comparison of semantic completion details. The proposed method has the highest IoU on small-volume objects, such as 'barriers', 'pedestrians', and 'vehicles', which shows that the network performs better in detailed geometric information learning. Moreover, it demonstrates superiority in completing detailed parts of the scene, especially in the completion of small-volume objects.

Table 3 presents the training time and sampling time for different generation steps. In this experiment, the generation resolution is uniformly set to 128 × 128 × 8, with a batch size of 4. The reported sampling time corresponds to the time required to generate a single sample. Experimental results indicate that as the number of generation steps increases, the training time remains relatively stable, whereas the sampling time increases significantly. This occurs because, during training, the model is optimized by randomly sampling time steps, while the sampling process requires sequential denoising operations. Under standard computational resources and efficiency constraints, our method can accomplish the generation task within a short time (e.g., only 2.45 s for 20 steps). For high-precision applications,

increasing the number of generation steps can be considered. These results demonstrate the method's strong potential for real-world deployment.

Table 1. Comparisons of the semantic scene completion results with other existing methods on the CarlaSC [7] dataset. Lee et al. [20] (A) represents the method without diffusion, while Lee et al. [20] (B) represents the method with diffusion. **Bold**—best in column for all methods.

Methods	IoU (%)	mIoU (%)
LMSCNet SS [11]	85.98	42.53
SSCNet Full [9]	80.69	41.91
MotionSC (T = 1) [7]	**86.46**	46.31
Lee et al. [20] (A)	80.70	39.94
Lee et al. [20] (B)	80.61	45.83
OccSora [40]	-	41.01
Ours	80.83	**48.08**

Table 2. Quantitative results of semantic scene completion method on CarlaSC [7] test set. **Bold**—best in column for all methods. Lee et al. [20] (A) represents the method without diffusion, while Lee et al. [20] (B) represents the method with diffusion.

Methods	Free	Building	Barrier	Other	Pedestrian	Pole	Road	Ground	Sidewalk	Vegetation	vehicles
LMSCNet SS [11]	**97.41**	25.61	3.35	11.31	33.76	**43.54**	85.96	**21.15**	52.64	39.99	53.09
SSCNET Full [9]	96.02	27.04	1.82	13.65	29.69	27.02	88.45	25.87	65.36	33.29	52.78
MotionSC (T = 1) [7]	97.42	31.59	2.63	14.77	39.87	42.11	**90.57**	**25.89**	60.77	42.41	61.37
Lee et al. [20] (A)	96.40	27.72	3.15	8.77	22.15	37.14	89.02	18.22	59.25	29.74	47.72
Lee et al. [20] (B)	96.00	**31.75**	3.42	**25.43**	46.22	43.32	84.57	13.01	**67.50**	37.45	55.46
Ours	95.81	23.31	**7.02**	19.28	**53.81**	42.52	86.28	16.17	62.42	**55.81**	**75.49**

Table 3. The computation time comparison between different generation steps during the inference phase on CarlaSC [7].

Steps	Training (time/epoch)	Sampling (time/img)
20	45 m 27 s	2.45 s
50	45 m 32 s	3.25 s
100	45 m 39 s	4.38 s
200	45 m 51 s	5.53 s

4.3. Visualization Results

The visualization results of semantic scene completion on the four outdoor scene datasets. Figure 4 shows the effectiveness of the proposed method on the CarlaSC dataset [7], as well as the visual comparison with existing method [20] and the ground truth (GT). As seen in the figure, our method illustrates outstanding performance in scene completion and notably outperforms the comparison methods in completing small-volume objects highlighted by the red circles, such as 'barriers', 'pedestrians', 'vehicles', etc.

To further evaluate the effectiveness of the proposed method, extensive experiments are also conducted on the SemanticKITTI [8], KITTI-360 [38], and SemanticPOSS [39] datasets. As shown in Figures 5–7, the completion visualization results on these three datasets demonstrate that the proposed method not only effectively infers complete and semantically plausible scenes from sparse inputs but also exhibits decent generalization ability, adapting to different environments, scene structures, and data distributions. This further validates the robustness of the proposed method in handling diverse inputs and complex scenes.

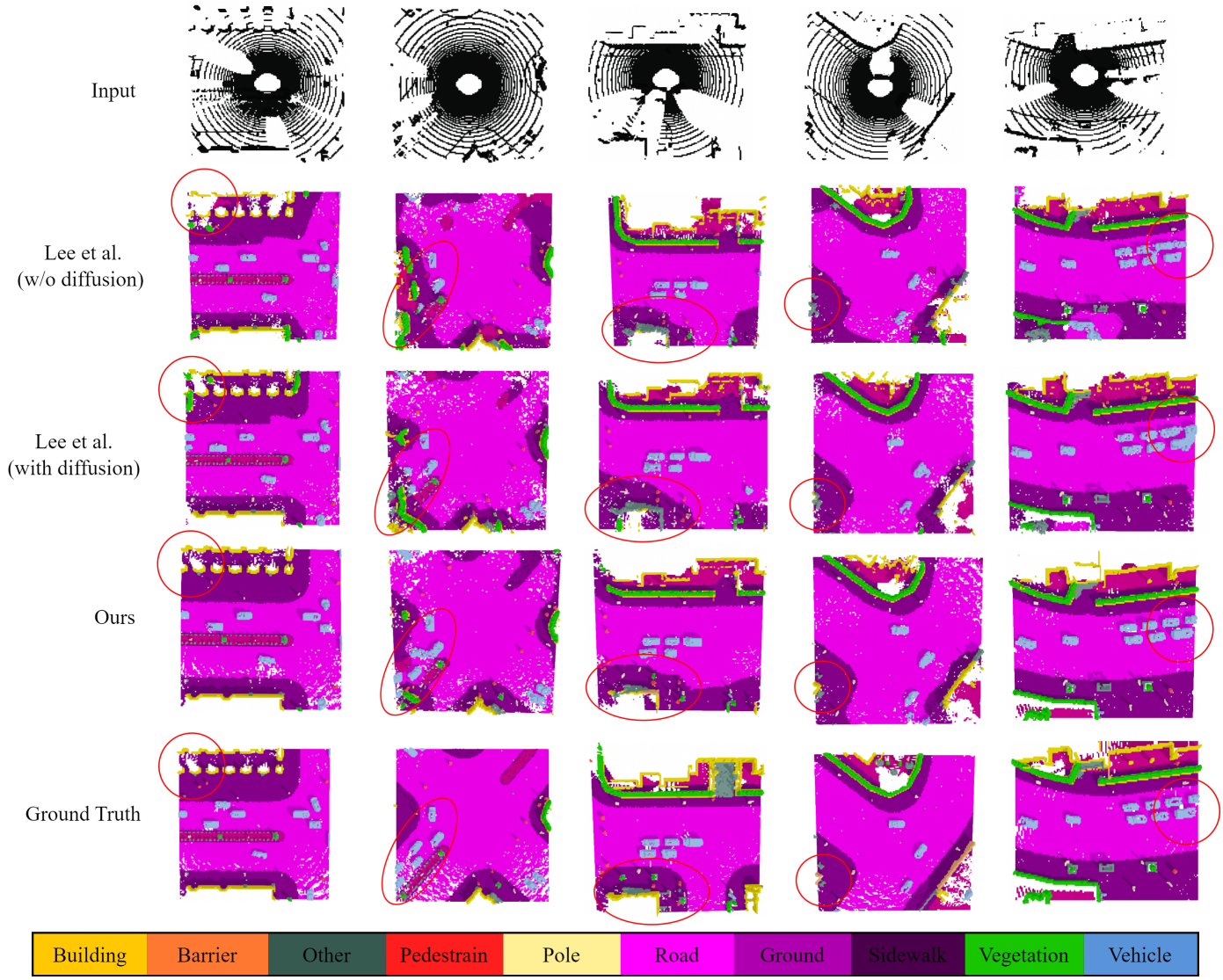


Figure 4. Semantic completion visualization results on the CarlaSC [7] dataset. The first line represents multiple sparse input scenes within the dataset. The second and third lines are the results of Lee et al.'s [20] method with and without diffusion, respectively. The fourth line is the visualization results of the proposed method in this paper. The last line is the corresponding ground truth. The comparison in the red circle marked in the results shows that our proposed method performs well on small-volume objects. The table at the bottom corresponds to the colors of each object class.

4.4. Ablation Studies

To assess the impact of the crucial components of the proposed method, ablation studies are conducted on the CarlaSC dataset [7]. As presented in Table 4, the results demonstrate that all key components involved in the proposed method contribute to the improvement of semantic scene completion.

Effect of discrete diffusion: The proposed method is based on a diffusion process in the discrete semantic domain. In the ablation study, we compared the completion performance of the continuous diffusion process. Under the continuous diffusion approach, discrete semantic data are first encoded into a continuous space for processing. As shown in the table, when applying continuous diffusion, both IoU and mIoU exhibit a significant decline. This suggests that semantic features are better leveraged in the discrete diffusion process, whereas encoding them into a continuous space may lead to semantic information loss.

Effect of Point feature compensation: To compensate for the information loss caused by point cloud voxelization, the point feature is extracted and fused into corresponding scene

voxels. This compensation process helps to preserve the fine-grained details that might otherwise be lost during voxelization, improving the overall representation of the scene. As can be seen from Table 4, not performing feature compensation will result in a decrease of 2.22 and 2.34 in IoU and mIoU, respectively.

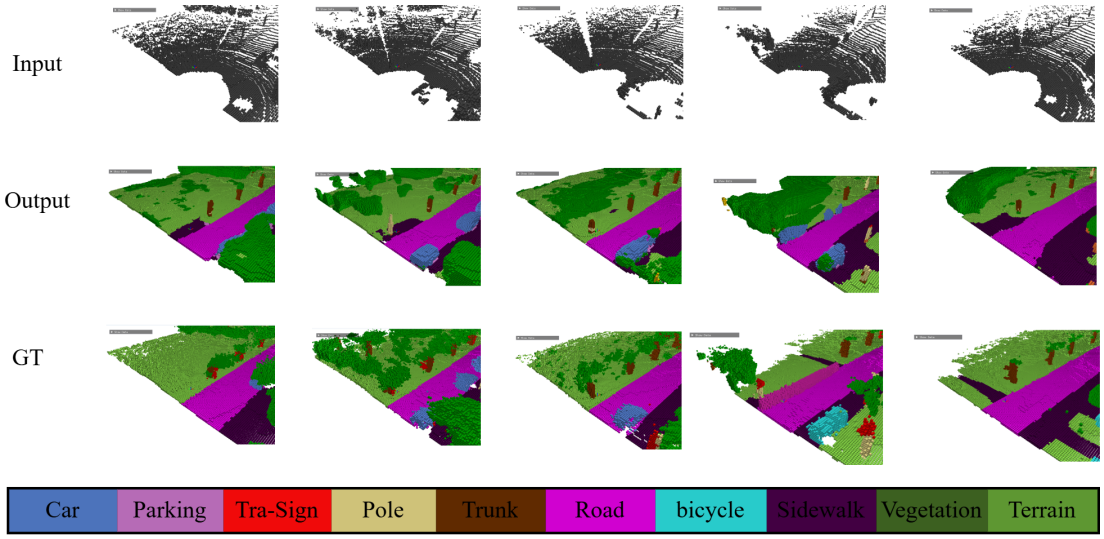


Figure 5. The visualization results for semantic scene completion on the SemanticKITTI [8] dataset. The first row represents the input of sparse scenes, the second row illustrates the outcomes of semantic scene completion using the method proposed in this paper, and the final row displays the corresponding ground truth. The table at the bottom corresponds to the colors of each object class.

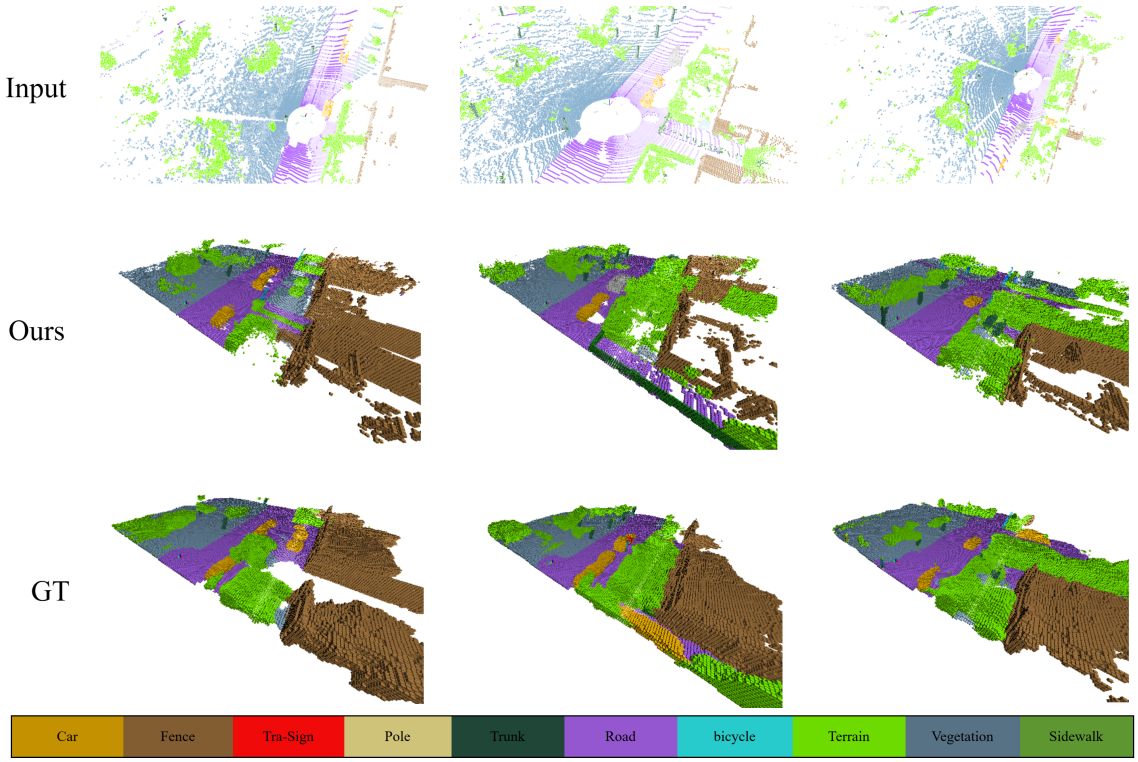


Figure 6. The visualization results for semantic scene completion on the KITTI-360 [38] dataset. The first row represents the input of sparse scenes, the second row illustrates the outcomes of semantic scene completion using the method proposed in this paper, and the final row displays the corresponding ground truth. The table at the bottom corresponds to the colors of each object class.

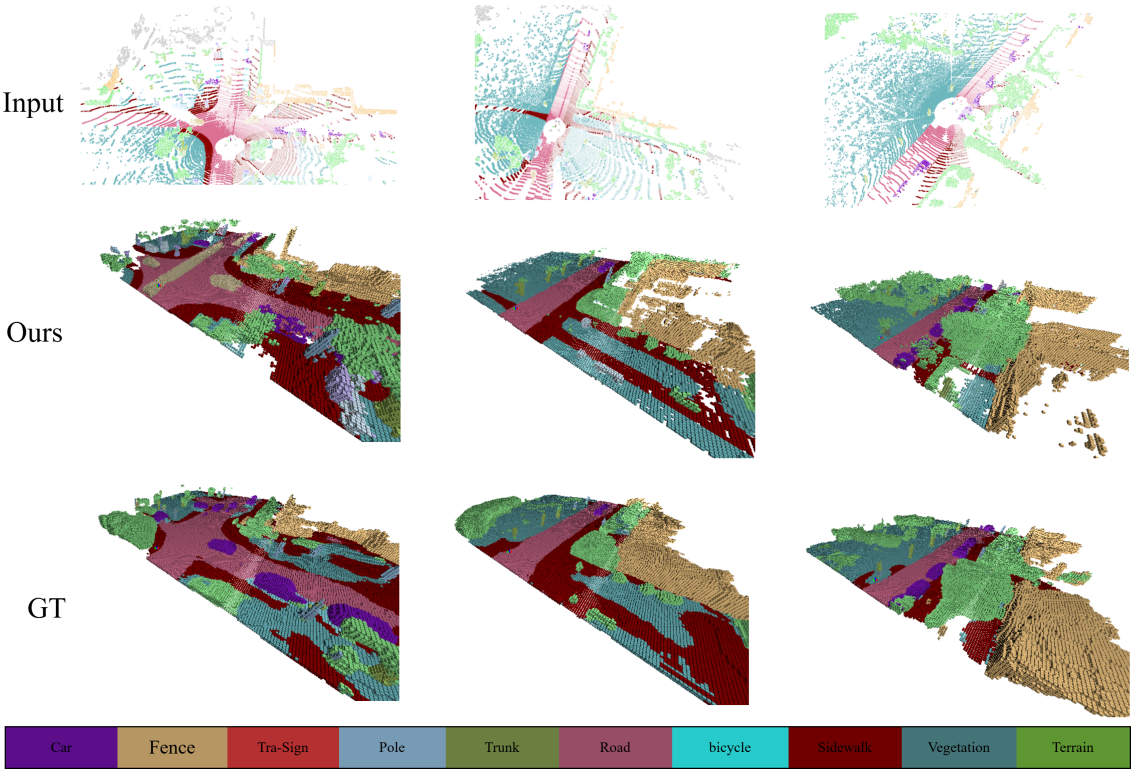


Figure 7. The visualization results for semantic scene completion on the SemanticPOSS [38] dataset. The first row represents the input of sparse scenes, the second row illustrates the outcomes of semantic scene completion using the method proposed in this paper, and the final row displays the corresponding ground truth. The table at the bottom corresponds to the colors of each object class.

Effect of K-NN transition kernel: To aggregate the local semantic features of the scene data and ensure a more reasonable distribution of semantic data during the discrete diffusion process, a weighted K-nearest neighbor uniform transition kernel is proposed based on feature distance. Its benefits are evaluated by using only the uniform transition kernel, excluding the weighted K-nearest neighbor kernel. As reported in Table 4, the IoU and mIoU of semantic scene completion decrease by 1.29 and 0.77, respectively.

Effect of K value: In addition, based on the proposed method, we further explored the impact of the choice of K value in the weighted K-nearest neighbor uniform transition kernel on method performance, aiming to gain a deeper understanding of the method's sensitivity to this hyperparameter. Through experiments, we tested different K values within the range of 10 to 100 and recorded the corresponding IoU and mIoU in Table 5. The experimental results indicate that as the K value increases, the proposed method's performance initially improves and then declines. This trend suggests that the choice of K value has a significant impact on method performance, and there exists an optimal range (around $K = 50$), beyond which performance may degrade. This finding provides important guidance for subsequent hyperparameter tuning.

Table 4. Ablation studies on the CarlaSC [7] dataset.

Methods	IoU (%)	mIoU (%)
complete network	80.83	48.08
with continuous diffusion	73.90	39.08
w/o point feature compensation	78.61	45.74
w/o K-NN transition kernel	79.54	47.31

Table 5. The impact of the K value in the weighted K-nearest neighbor uniform transition kernel on the proposed method.

K	IoU (%)	mIoU (%)
10	80.15	47.13
50	80.83	48.08
100	80.06	47.43

4.5. Robustness Studies

To validate the robustness of the proposed method in real-world environments, particularly under common noise conditions and varying resolutions, this section designs two kinds of robustness experiments: the Gaussian noise experiments and the resolution experiments. These experiments aim to evaluate the performance of the proposed method under different noise levels and LiDAR resolutions, further exploring its adaptability in practical applications.

The Gaussian noise experiments: In practical applications, 3D point cloud data are often affected by various environmental noises. To comprehensively evaluate the robustness of our method under complex noise conditions, we designed a noise perturbation experiment. Specifically, Gaussian random noise with a mean of 0 and standard deviations of 0.1, 0.5, and 1 was injected into the original point cloud data. The noise level of 0 corresponds to the original input data without any added noise.

The quantitative and visual experimental results are shown in Table 6 and Figure 8, respectively. As reported in the table, when the noise level is 0 (i.e., no noise), the highest mIoU of 48.08 is achieved. As the noise level increases, mIoU exhibits a downward trend. Despite the presence of noise interference, our method maintains a notable level of robustness against noise. Even at the highest noise level of 1, the mIoU remains at 41.45 without catastrophic degradation. For the visualization results, despite the impact of noise, our method can still effectively complete the scene and restore semantic information while maintaining a relatively complete structure and reasonable semantics. The experimental results demonstrate that the proposed method exhibits strong robustness and adaptability even in complex, noisy environments.

Table 6. Noise studies on the CarlaSC [7] dataset. The noise level of 0 indicates the original data without added noise.

Noise	IoU (%)	mIoU (%)
0	80.83	48.08
0.1	78.90	46.95
0.5	76.21	44.68
1	74.01	41.45

The resolution experiments: The resolution of LiDAR has a significant impact on the method's performance, especially in complex scenes. To investigate the method's adaptability to different resolutions, we further validate its performance at two resolutions on CarlaSC [7]: $64 \times 64 \times 4$ and $32 \times 32 \times 2$.

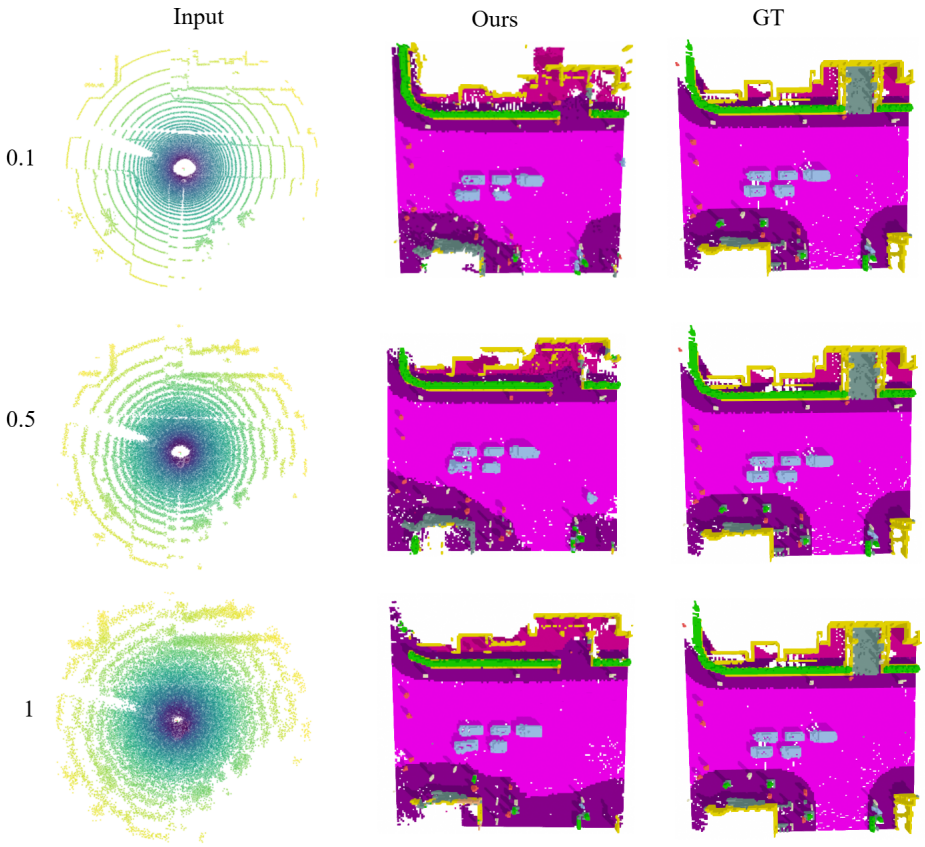


Figure 8. The Gaussian noise experiment on the CarlaSC [7] dataset. The first row presents the input, completion results, and ground truth under a noise standard deviation of 0.1. The second and third rows visualize the experimental results for noise standard deviations of 0.5 and 1, respectively.

The quantitative and visual experimental results are presented in Table 7 and Figure 9, respectively. As reported in the table, the proposed method effectively preserves finer details even at lower resolutions (64 × 64 × 4 and 32 × 32 × 2). Although the information content is reduced under these resolutions, the method maintains stable completion performance, demonstrating robust adaptability to resolution variations. Additionally, Figure 9 presents the comparison of visualization results under different resolutions. Even under low-resolution conditions, the proposed method can still generate complete and reasonable semantic scenes while maintaining high completion quality. Through these experiments, the proposed method shows robust performance across different resolutions, providing strong support for its application in real-world environments.

Table 7. Resolution studies on the CarlaSC [7] dataset; 128 × 128 × 8 is the standard resolution setting for this dataset.

Resolutions	IoU (%)	mIoU (%)
128 × 128 × 8	80.83	48.08
64 × 64 × 4	78.80	46.56
32 × 32 × 2	76.61	45.03

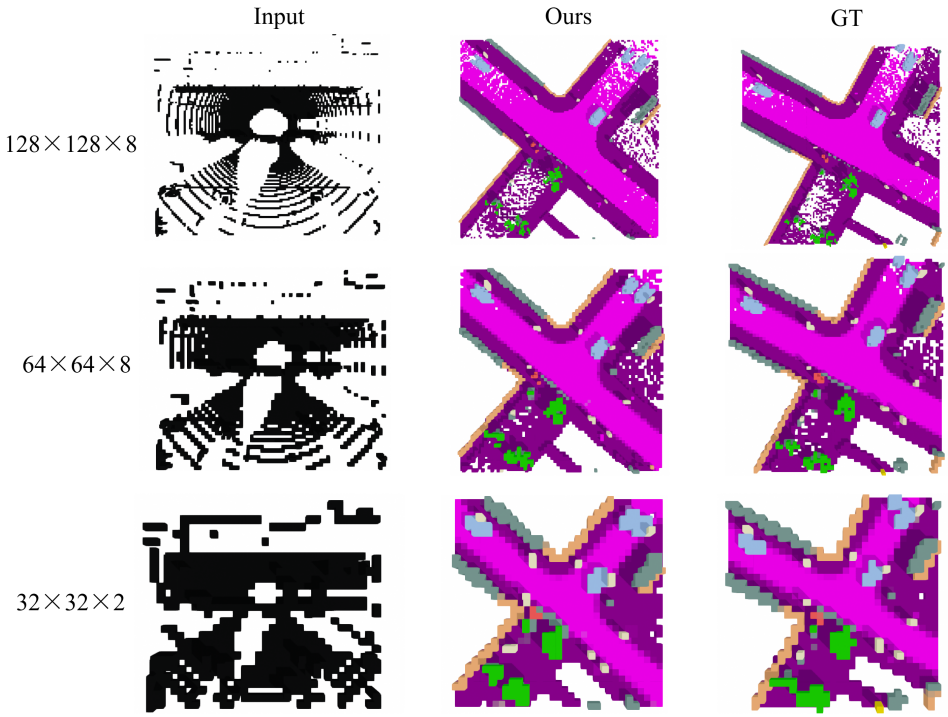


Figure 9. Visualization results under different resolutions on the CarlaSC dataset [7]. The first row presents the sparse input, the completion results of our method, and the ground truth at the original resolution. The second and third rows show the completion results and ground truth at different resolutions, respectively.

5. Conclusions

Semantic scene completion techniques enhance detail inference in sparse environments while achieving semantic segmentation, significantly contributing to the accuracy and richness required for advanced modeling and scene representation. AI-based generative networks relying on probabilistic models inherently offer solutions for this task. This paper introduces a semantic scene completion network based on a discrete diffusion probabilistic model. A weighted *K*-nearest neighbor uniform transition kernel-based feature distance is designed to capture local features and control the category distribution transition process. Simultaneously, the network utilizes point-voxel fused features to enhance the details of scene completion. The proposed method is validated as effective through qualitative and quantitative experiments on public datasets. Future research can be directed toward adapting and deploying the network on mobile terminals.

Author Contributions: Conceptualization, Y.W.; Data curation, F.A.; Funding acquisition, B.Y.; Methodology, Y.W. and X.H.; Project administration, B.Y.; Resources, Y.C.; Software, X.H.; Supervision, B.Y.; Visualization, D.Z.; Writing—original draft, X.H.; Writing—review and editing, Y.W. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by Hainan Province Science and Technology Special Fund (No. ZDYF2022GXIS005) and the Open Research Project of Hubei Key Laboratory of Intelligent Robot (HBIR 202406).

Data Availability Statement: The source code of the method is available at: https://github.com/hx123-cmd/DiffusionSSC (accessed on 13 February 2025).

Conflicts of Interest: Author Yong Chen was employed by the company Geely Automobile Research Institute (Ningbo) Co., Ltd. The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

References

1. Zhang, K.; Luo, W.; Yu, Y.; Ren, W.; Zhao, F.; Li, C.; Ma, L.; Liu, W.; Li, H. Beyond monocular deraining: Parallel stereo deraining network via semantic prior. *Int. J. Comput. Vis.* **2022**, *130*, 1754–1769.
2. Zhang, K.; Li, R.; Yu, Y.; Luo, W.; Li, C. Deep dense multi-scale network for snow removal using semantic and depth priors. *IEEE Trans. Image Process.* **2021**, *30*, 7419–7431. [CrossRef] [PubMed]
3. Wang, R.; Peethambaran, J.; Chen, D. LiDAR Point Clouds to 3-D Urban Models: A Review. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2018**, *11*, 606–627. [CrossRef]
4. Zhang, Y.; Yan, S.; Zhang, L.; Du, B. Fast Projected Fuzzy Clustering with Anchor Guidance for Multimodal Remote Sensing Imagery. *IEEE Trans. Image Process.* **2024**, *33*, 4640–4653. [CrossRef]
5. Gupta, S.; Arbelaez, P.; Malik, J. Perceptual organization and recognition of indoor scenes from RGB-D images. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Online, 13 December 2013; pp. 564–571.
6. Firman, M.; Mac Aodha, O.; Julier, S.; Brostow, G.J. Structured prediction of unobserved voxels from a single depth image. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Online, 5 July 2016; pp. 5431–5440.
7. Wilson, J.; Song, J.; Fu, Y.; Zhang, A.; Capodieci, A.; Jayakumar, P.; Barton, K.; Ghaffari, M. MotionSC: Data set and network for real-time semantic mapping in dynamic environments. *IEEE Robot. Autom. Lett.* **2022**, *7*, 8439–8446. [CrossRef]
8. Behley, J.; Garbade, M.; Milioto, A.; Quenzel, J.; Behnke, S.; Stachniss, C.; Gall, J. Semantickitti: A dataset for semantic scene understanding of lidar sequences. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Online, 24 June 2019; pp. 9297–9307.
9. Song, S.; Yu, F.; Zeng, A.; Chang, A.X.; Savva, M.; Funkhouser, T. Semantic scene completion from a single depth image. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Online, 31 October 2017; pp. 1746–1754.
10. Cheng, R.; Agia, C.; Ren, Y.; Li, X.; Liu, B. S3cnet: A sparse semantic scene completion network for lidar point clouds. *Proc. Mach. Learn. Res.* **2021**, *155*, 2148–2161.
11. Roldao, L.; de Charette, R.; Verroust-Blondet, A. Lmscnet: Lightweight multiscale 3d semantic completion. In Proceedings of the 2020 International Conference on 3D Vision (3DV), Fukuoka, Japan, 25–28 November 2020; pp. 111–119.
12. Zou, H.; Yang, X.; Huang, T.; Zhang, C.; Liu, Y.; Li, W.; Wen, F.; Zhang, H. Up-to-Down Network: Fusing Multi-Scale Context for 3D Semantic Scene Completion. In Proceedings of the 2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Prague, Czech Republic, 27 September–1 October 2021; pp. 16–23.
13. Yang, X.; Zou, H.; Kong, X.; Huang, T.; Liu, Y.; Li, W.; Wen, F.; Zhang, H. Semantic segmentation-assisted scene completion for lidar point clouds. In Proceedings of the 2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Prague, Czech Republic, 27 September–1 October 2021; pp. 3555–3562.
14. Jo, C.; Im, W.; Yoon, S.E. In-n-out: Towards good initialization for inpainting and outpainting. *arXiv* **2021**, arXiv:2106.13953.
15. Lugmayr, A.; Danelljan, M.; Romero, A.; Yu, F.; Timofte, R.; Van Gool, L. Repaint: Inpainting using denoising diffusion probabilistic models. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Online, 1 September 2022; pp. 11461–11471.
16. Chen, Y.T.; Garbade, M.; Gall, J. 3d semantic scene completion from a single depth image using adversarial training. In Proceedings of the 2019 IEEE International Conference on Image Processing (ICIP), Taipei, Taiwan, 22–25 September 2019; pp. 1835–1839.
17. Wu, Y.; Chen, X.; Huang, X.; Song, K.; Zhang, D. Unsupervised distribution-aware keypoints generation from 3D point clouds. *Neural Netw.* **2024**, *173*, 106158. [CrossRef]
18. Ho, J.; Jain, A.; Abbeel, P. Denoising diffusion probabilistic models. *Adv. Neural Inf. Process. Syst.* **2020**, *33*, 6840–6851.
19. Achlioptas, P.; Diamanti, O.; Mitliagkas, I.; Guibas, L. Learning representations and generative models for 3d point clouds. *Proc. Mach. Learn. Res.* **2018**, *80*, 40–49.
20. Lee, J.; Im, W.; Lee, S.; Yoon, S.E. Diffusion Probabilistic Models for Scene-Scale 3D Categorical Data. *arXiv* **2023**, arXiv:2301.00527.
21. Yan, X.; Gao, J.; Li, J.; Zhang, R.; Li, Z.; Huang, R.; Cui, S. Sparse single sweep lidar point cloud segmentation via learning contextual shape priors from scene completion. *Proc. AAAI Conf. Artif. Intell.* **2021**, *35*, 3101–3109.
22. Garbade, M.; Chen, Y.T.; Sawatzky, J.; Gall, J. Two stream 3d semantic scene completion. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, Online, 26 November 2019.
23. Li, J.; Liu, Y.; Gong, D.; Shi, Q.; Yuan, X.; Zhao, C.; Reid, I. Rgbd based dimensional decomposition residual network for 3d semantic scene completion. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Online, 26 November 2019; pp. 7693–7702.
24. Liu, Y.; Li, J.; Yan, Q.; Yuan, X.; Zhao, C.; Reid, I.; Cadena, C. 3D gated recurrent fusion for semantic scene completion. *arXiv* **2020**, arXiv:2002.07269.
25. Cai, Y.; Chen, X.; Zhang, C.; Lin, K.Y.; Wang, X.; Li, H. Semantic scene completion via integrating instances and scene in-the-loop. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Online, 18 October 2021; pp. 324–333.
26. Dhariwal, P.; Nichol, A. Diffusion models beat gans on image synthesis. *Adv. Neural Inf. Process. Syst.* **2021**, *34*, 8780–8794.

27. Fan, W.C.; Chen, Y.C.; Chen, D.; Cheng, Y.; Yuan, L.; Wang, Y.C.F. Frido: Feature pyramid diffusion for complex scene image synthesis. *Proc. AAAI Conf. Artif. Intell.* **2023**, *37*, 579–587. [CrossRef]
28. Gu, S.; Chen, D.; Bao, J.; Wen, F.; Zhang, B.; Chen, D.; Yuan, L.; Guo, B. Vector quantized diffusion model for text-to-image synthesis. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Online, 1 September 2022; pp. 10696–10706.
29. Luo, S.; Hu, W. Diffusion probabilistic models for 3d point cloud generation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Online, 18 October 2021; pp. 2837–2845.
30. Zeng, X.; Vahdat, A.; Williams, F.; Gojcic, Z.; Litany, O.; Fidler, S.; Kreis, K. LION: Latent point diffusion models for 3D shape generation. *arXiv* **2022**, arXiv:2210.06978.
31. Austin, J.; Johnson, D.D.; Ho, J.; Tarlow, D.; Van Den Berg, R. Structured denoising diffusion models in discrete state-spaces. *Adv. Neural Inf. Process. Syst.* **2021**, *34*, 17981–17993.
32. Hoogeboom, E.; Nielsen, D.; Jaini, P.; Forré, P.; Welling, M. Argmax flows and multinomial diffusion: Learning categorical distributions. *Adv. Neural Inf. Process. Syst.* **2021**, *34*, 12454–12465.
33. Sohl-Dickstein, J.; Weiss, E.; Maheswaranathan, N.; Ganguli, S. Deep unsupervised learning using nonequilibrium thermodynamics. *Proc. Mach. Learn. Res.* **2015**, *37*, 2256–2265.
34. Song, J.; Meng, C.; Ermon, S. Denoising diffusion implicit models. *arXiv* **2020**, arXiv:2010.02502.
35. Xia, Z.; Liu, Y.; Li, X.; Zhu, X.; Ma, Y.; Li, Y.; Hou, Y.; Qiao, Y. SCPNet: Semantic Scene Completion on Point Cloud. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Online, 27 June 2023; pp. 17642–17651.
36. Jarzynski, C. Equilibrium free-energy differences from nonequilibrium measurements: A master-equation approach. *Phys. Rev. E* **1997**, *56*, 5018.
37. Zhang, Z.; Sabuncu, M. Generalized cross entropy loss for training deep neural networks with noisy labels. *Adv. Neural Inf. Process. Syst.* **2018**, *31*. Available online: https://proceedings.neurips.cc/paper_files/paper/2018 (accessed on 13 February 2025)
38. Liao, Y.; Xie, J.; Geiger, A. Kitti-360: A novel dataset and benchmarks for urban scene understanding in 2d and 3d. *IEEE Trans. Pattern Anal. Mach. Intell.* **2022**, *45*, 3292–3310. [CrossRef] [PubMed]
39. Pan, Y.; Gao, B.; Mei, J.; Geng, S.; Li, C.; Zhao, H. Semanticposs: A point cloud dataset with large quantity of dynamic instances. In Proceedings of the 2020 IEEE Intelligent Vehicles Symposium (IV), Las Vegas, NV, USA, 19 October–13 November 2020; pp. 687–693.
40. Wang, L.; Zheng, W.; Ren, Y.; Jiang, H.; Cui, Z.; Yu, H.; Lu, J. Occsora: 4d occupancy generation models as world simulators for autonomous driving. *arXiv* **2024**, arXiv:2405.20337.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.