CymbaDiff: Structured Spatial Diffusion for Sketch-based 3D Semantic Urban Scene Generation

Li Liang[1] **Bo Miao**[2] **Xinyu Wang**[1] **Naveed Akhtar**[3] **Jordan Vice**[1] **Ajmal Mian**[1]

[1] The University of Western Australia
[2] AIML, The University of Adelaide
[3] The University of Melbourne

Abstract

Outdoor 3D semantic scene generation produces realistic and semantically rich environments for applications such as urban simulation and autonomous driving. However, advances in this direction are constrained by the absence of publicly available, well-annotated datasets. We introduce SketchSem3D, the first large-scale benchmark for generating 3D outdoor semantic scenes from abstract freehand sketches and pseudo-labeled annotations of satellite images. SketchSem3D includes two subsets, Sketch-based SemanticKITTI and Sketch-based KITTI-360 (containing LiDAR voxels along with their corresponding sketches and annotated satellite images), to enable standardized, rigorous, and diverse evaluations. We also propose Cylinder Mamba Diffusion (CymbaDiff) that significantly enhances spatial coherence in outdoor 3D scene generation. CymbaDiff imposes structured spatial ordering, explicitly captures cylindrical continuity and vertical hierarchy, and preserves both physical neighborhood relationships and global context within the generated scenes. Extensive experiments on SketchSem3D demonstrate that CymbaDiff achieves superior semantic consistency, spatial realism, and cross-dataset generalization. The code and dataset will be available at https://github.com/Lillian-research-hub/CymbaDiff.

1 Introduction

Generative modeling has demonstrated remarkable progress in the 2D and 3D domains, largely fueled by the rapid development of diffusion models [1, 2, 3, 4]. In 3D, diffusion approaches have significantly advanced 3D object synthesis [5, 6] and indoor scene generation [7, 8]. However, generating large-scale 3D outdoor environments remains widely underexplored [9, 10, 11], as outdoor urban scenes pose greater challenges due to their higher semantic diversity, complex spatial structures, and dynamic contextual dependencies. Despite these challenges, synthesizing realistic and scalable 3D urban scenes is increasingly critical, as it underpins a wide range of emerging applications, including city-scale simulation [12, 13] and autonomous driving [14, 15, 16, 17].

A few methods have recently surfaced for 3D outdoor scene generation [9, 10, 11, 18, 19], often relying on bird's-eye view (BEV) with only road data or multi-scale scene hierarchies to guide generation. BEV-based approaches suffer from insufficient 3D structural information, limiting both semantic richness and geometric fidelity. Meanwhile, modeling multi-scale scene hierarchies typically requires generative models to repeatedly synthesize scenes at multiple spatial resolutions, increasing both computational and structural complexity. Moreover, due to the lack of a public large-scale benchmark, current approaches typically use self-curated and heavily preprocessed datasets for evaluation [9], which fundamentally constrains rigorous benchmarking. Sketch-based methods [20, 21, 22, 23] have recently emerged as a promising paradigm for user-guided 3D generation, enabling intuitive control through freehand drawings. However, their applicability remains confined to the synthesis of isolated 3D objects or simple indoor scenes. Expanding sketch-based 3D reconstruction to outdoor scenes is currently widely open. Challenges in this novel pursuit arise from complex

39th Conference on Neural Information Processing Systems (NeurIPS 2025).

scene layouts, diverse object geometries, and the need to preserve spatio-semantic coherence across large-scale scenes.

This work takes a significant step towards extending sketch-based generation to outdoor environments. To that end, we build upon the growth of State Space Models (SSMs) [24], which have gained increased attention across image segmentation [25, 26] and point cloud processing [27, 28] for their ability to capture long-range dependencies while remaining efficient through selective computation.However, to enhance global contextual understanding, SSMs typically aggregate information from multiple scan directions, leading to substantial memory overhead. Moreover, the scanning order imposed by the Cartesian coordinate system can distort local neighborhood relationships, especially in scenes with limited spatial coherence.

To address the above-noted challenges for sketch-based 3D outdoor scene generation, we first present 'SketchSem3D', a large-scale dataset tailored for the task. SketchSem3D enables the synthesis of semantically rich outdoor 3D environments from freehand sketches and pseudo-labeled satellite image annotations. The annotation pipeline properly integrates CLIP-based textual guidance [29] with image embeddings from the Segment Anything Model (SAM) [30], enabling robust and automated semantic labeling. SketchSem3D comprises two subsets, Sketch-based SemanticKITTI and Sketch-based KITTI-360, designed to support standardized benchmarking and fair comparison. Building upon this dataset, we define the novel 'sketch-based 3D outdoor scene generation' research task. We also propose Cylinder Mamba Diffusion, the first approach to handle this task. As adjacent Cartesian-based voxel sequences may misrepresent spatial proximity in outdoor scenes, CymbaDiff is particularly tailored to handle voxel discrepancies. Our underlying model is a denoising network, combining an SSM architecture with generative diffusion in the latent space. We design cylinder mamba blocks to enhance spatial coherence during the generative process, imposing a structured spatial ordering to explicitly encode cylindrical continuity and vertical hierarchy, preserving spatial neighborhood relationships within scenes.

Our key contributions are summarized below:

- We introduce the novel task of 'sketch-based 3D outdoor scene generation', which enables intuitive and flexible user interaction through freehand sketches and pseudo-labeled satellite image annotations. By reducing the need for manual semantic annotation, this task offers an efficient solution to generate training data for applications such as urban-scale simulation and autonomous driving.
- We present SketchSem3D, the first public large-scale sketch-based benchmark for 3D outdoor semantic scene generation. It includes two subsets, Sketch-based SemanticKITTI and Sketch-based KITTI-360, and enables standardized benchmarking for the development and evaluation of generative models in complex outdoor settings.
- We propose CymbaDiff, a generative model that incorporates the proposed cylinder mamba blocks to enhance spatial coherence during the generation process. We also conduct extensive experiments on the Sketch-based SemanticKITTI and Sketch-based KITTI-360 benchmarks, demonstrating state-of-the-art performance in 3D semantic scene generation and completion.

2 Related Work

2.1 State Space Models

Recent studies have demonstrated the strong capability of State-Space Models (SSMs) in capturing long-range dependencies across sequential data [31, 32]. These models have been successfully applied in a variety of domains, including medical image segmentation [25, 33], image restoration [34, 35], natural language processing (NLP) [36, 37], and point cloud processing [38, 28]. Many of these approaches build upon foundational architectures such as VisionMamba [39], S4ND [40], and Mamba-ND [41]. Specifically, VisionMamba [39] integrates bidirectional SSMs for data-dependent global context modeling and employs positional embeddings to enhance location-aware visual recognition. S4ND [40] extends the SSM framework by incorporating local convolution operations, thereby enabling processing beyond one-dimensional inputs. Mamba-ND [41] further addresses multi-dimensional data by utilizing various scan patterns within a single block to enhance performance in discriminative tasks. Despite their strengths, these methods primarily focus on maximizing contextual information through multiple scanning directions, often neglecting structured spatial coherence across horizontal and vertical hierarchies, particularly under memory-constrained settings.

2.2 3D Semantic Scene Generation

Diffusion models have evolved from generating 2D images to addressing increasingly complex 3D data modeling tasks [2]. Compared to traditional generative models such as Generative Adversarial Networks (GANs) [42] and Variational Autoencoders [43], diffusion models follow a progressive denoising process [44], which enhances training stability and improves the capacity to capture complex data distributions. These advantages render diffusion models particularly suitable for 3D data generation tasks. While much of the existing research has focused on object-level synthesis [45, 46, 47, 48, 49, 50, 51, 52] and indoor scene generation [53, 54, 55, 56], there is a growing body of work exploring 3D outdoor semantic scene generation [57, 11, 10, 58, 47, 9] as it underpins a wide range of emerging applications, including autonomous driving [14, 15, 16, 17] and city-scale simulation [12, 13]. For instance, UrbanDiff [9] conditions generation on BEV maps to produce urban scenes in the form of semantic occupancy grids, integrating both geometry and semantic information. P-DiscreteDif [10] proposes a progressive multi-scale strategy that synthesizes large-scale 3D scenes by conditioning each stage on the output from the preceding resolution level, with the initial model conditioned solely on noise. Despite these advancements, the absence of standardized datasets for 3D outdoor semantic scene generation has led to the use of heterogeneous benchmarks with inconsistent scene conditions, thereby limiting fair comparison and hindering systematic progress in the field.

2.3 3D Semantic Scene Completion

3D semantic scene completion methods can be broadly categorized into four categories: image-based approaches [59, 60], point cloud-based methods [61, 62], voxel-based techniques [63, 64], and multi-modality-based frameworks [65, 66]. Most existing methods are built upon convolutional neural networks (CNNs) or Transformer-based architectures. For instance, Xia *et al.* [64] propose a CNN network (SCPNet), which enhances single-frame scene completion by incorporating dense relational semantic knowledge distillation along with a label rectification strategy to mitigate artifacts introduced by dynamic objects. CGFormer [59] enhances semantic scene completion by introducing a context- and geometry-aware voxel transformer, which initializes queries based on the contextual information from individual input images and extends deformable cross-attention mechanisms from 2D image space to 3D voxel space. While CNNs are computationally efficient, they are inherently limited by their receptive field size. Transformers address this limitation by enabling global context modeling but come with high memory costs. Recently, Segmamba [25] has emerged as a promising alternative, offering a favorable trade-off by supporting large receptive fields with improved memory efficiency, making it suitable for 3D semantic scene completion.

3 SketchSem3D Dataset

Sketch-based methods have recently gained increasing attention as a promising paradigm for user-guided 3D modeling, offering intuitive and flexible interaction through freehand drawing. While these approaches show great potential, they are constrained to generating isolated 3D objects and lack the capacity to model complex, semantically rich scenes. In a related direction, UrbanDiff [9] introduced BEV representations as conditional inputs for 3D semantic scene generation. By leveraging the spatial alignment between 2D projections and 3D structures, this approach promotes 2D-to-3D consistency. However, BEV-based supervision inherently constrains the diversity of the generated scenes. Moreover, acquiring BEV images that accurately reflect the semantic layout of complex 3D environments is particularly challenging in outdoor settings.

We propose a sketch-based framework for 3D outdoor semantic scene generation. It enables users to define scene layouts using coarse freehand sketches combined with pseudo-labeled satellite image annotations, facilitating a more natural and accessible interaction modality. By circumventing the need for labor-intensive annotations and large-scale sensor-based data collection, the framework significantly enhances scalability. We leverage this framework in the design of our SketchSem3D benchmark dataset.

3.1 Benchmark Construction

The benchmark comprises two distinct datasets, Sketch-based SemanticKITTI and Sketch-based KITTI-360, each constructed through a systematic three-stage pipeline discussed below.

Data Sourcing. We construct the two datasets using the 3D ground truth (GT) from SemanticKITTI [67] and SSCBench-KITTI-360 [68], respectively. Each scene is enriched with freehand sketches and pseudo-labeled satellite image annotations to enable conditioned 3D scene generation. Both datasets comprise five components: freehand (like) sketches, satellite images, pseudo-labeled

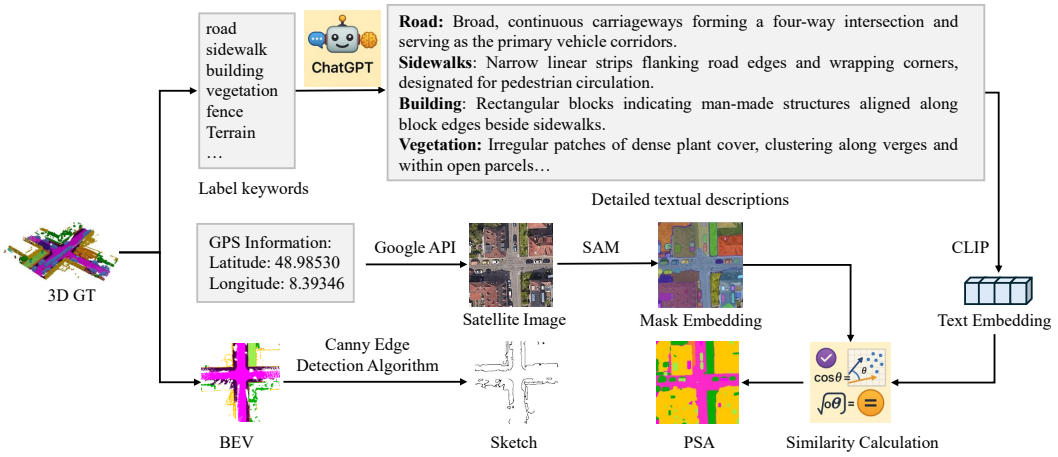

Figure 1: Pipeline for SketchSem3D construction. SAM and PSA denote Segment Anything Model and Pseudo-labeled Satellite Image Annotations, respectively.

Table 1: Comparing SketchSem3D (last two rows) with BEV-based NuScenes.

Dataset	Pairs	Condition	3D Geospatial Semantics	Classes	3D GT Voxels
BEV-based NuScenes [9]	34149	BEV	✗	17	$192 \times 192 \times 16$
Sketch-based SemanticKITTI	58987	Sketch / PSA	✓	20	$256 \times 256 \times 32$
Sketch-based KITTI-360	36057	Sketch / PSA	✓	19	$256 \times 256 \times 32$

annotations, semantic label keywords, and 3D GT (output). Figure 1 shows the dataset construction pipeline. The 3D GT is extended from the respective source datasets. Sketches are generated by applying the Canny edge detector [69] to BEV projections of 3D GT. These sketches closely resemble freehand drawings, which can be more easily produced at test time compared to BEV projections, providing abstract representations of scene geometry.

Semantic categories (e.g., road, tree, vehicle) are also available as GT and recorded as label keywords without spatial encoding. To enrich the semantic context, GPT-4 [70] is used to generate descriptive texts for each category, supporting alignment with visual features. We leverage the GPS information provided in KITTI [71] and KITTI-360 [72] to retrieve the corresponding satellite images. We then apply CLIP [29] to encode the enriched contextual descriptions and SAM [30] to obtain mask-level embeddings from the satellite images. By computing the cosine similarity between text and image embeddings, we infer the semantic composition of each scene from the satellite perspective, producing the pseudo-labeled annotations used in our SketchSem3D dataset.

Data Filtering and Formatting. To address any semantic labeling errors or inconsistencies in the automated alignment between CLIP [29] text embeddings and SAM [30] image mask embeddings, we perform a *manual review* of the resulting class distributions to ensure annotation accuracy and dataset reliability. Each sketch-based dataset consists of five components: (*i*) the sketch, (*ii*) satellite image, (*iii*) pseudo-labeled satellite image annotations, (*iv*) label keywords, and (*v*) 3D GT. The sketch is stored as a binary edge map in image format, capturing the structural outline of the scene. The satellite image is a geo-referenced RGB image of the same size, spatially aligned with the GPS coordinates of the corresponding scene. The pseudo-labeled satellite image annotations are single-channel semantic maps, where each pixel represents a semantic class ID. Although two-dimensional, these annotations provide coarse semantic cues that serve as important conditional guidance for reconstructing 3D voxel scenes. The label keywords for each scene are saved in a `.txt` file indexed by scene ID, listing the semantic class keywords present in the scene. Finally, 3D GT is provided as a volumetric label map, where each voxel is assigned a semantic class encoded as a 16-bit unsigned integer, following the format of SemanticKITTI [67].

3.2 Data Statistics Comparison and Evaluation Metrics

Table 1 compares our SketchSem3D dataset with the BEV-based NuScenes dataset [9]. We can see that our dataset is better in every aspect offering higher resolution, more classes, additional geospatial

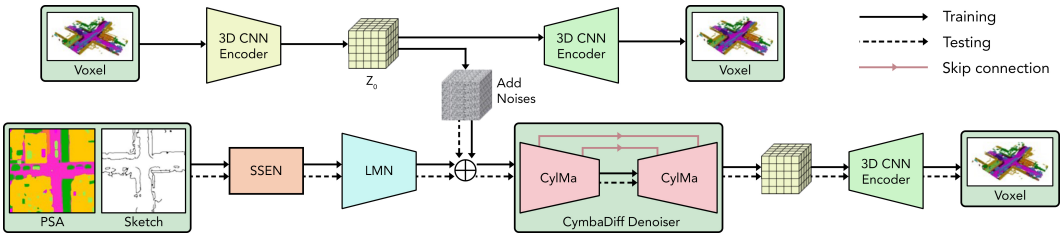

Figure 2: Architecture of our CymbaDiff generation network. The Scene Structure Estimation Network (SSEN) extracts abstract structural information from Pseudo-labeled Satellite Image Annotations (PSA) and the Sketch. The Latent Mapping Network (LMN) compresses the input conditions into a latent representation, which is then processed by the CymbaDiff denoiser, which utilizes the proposed cylinder mamba blocks (CylMa) to perform latent denoising.

semantics, two conditions instead of one and contains a much larger number of 3D scenes (total 95,044 compared to 34,149 in [9]). Notably, each subset of SketchSem3D contains more frames than [9]. Moreover, our conditions (sketch and PSA) are easier to obtain at test time, enhancing the practicality. Sketch-based SemanticKITTI includes 58,172 training and 815 validation frames, while Sketch-based KITTI-360 consists of 33,892 training and 2,165 validation frames. In SketchSem3D, all satellite images, sketches, and pseudo-labeled annotations are standardized to a resolution of 256×256 pixels, with corresponding 3D GT of $256 \times 256 \times 32$ voxels. In comparison, BEV-based NuScenes [9] contains 3D GT of $192 \times 192 \times 16$ voxels and lacks explicit geospatial structure as well as detailed 3D semantic distribution.

To evaluate the quality and diversity of the generated 3D semantic scenes, we adopt two widely used metrics: Fréchet Inception Distance (FID) [73] and Maximum Mean Discrepancy (MMD) [9]. Together, these metrics capture statistical similarity and feature-level realism, providing a comprehensive assessment of generative performance. Further details on the evaluation metrics are supplied in the Appendix.

4 Method

We propose a 3D semantic scene generation method that captures both geometric structure and semantic information, based on a given sketch and its corresponding pseudo-labeled satellite image annotations. Formally, let the sketch image be denoted as $I \in \mathbb{R}^{L \times W \times 1}$, and the associated pseudo-labeled satellite image annotations as $PSA \in \mathbb{R}^{L \times W \times 1}$. These two modalities are jointly projected into a structured 3D voxel grid $\mathbb{R}^{L \times W \times H \times 1}$, which encodes the spatial structure of the semantic scene, where L, W, H represent the length, width, and height of the 3D space, respectively. The goal is to generate a semantically complete 3D scene by predicting each voxel's occupancy state and semantic label. Each voxel in the generated grid is assigned a semantic class label $c \in 0, 1, 2, \ldots, C-1$, where C is the total number of semantic categories. By convention, $c = 0$ corresponds to empty or unoccupied space, while the remaining values represent distinct semantic classes.

4.1 Scene Structure Estimation Network

To facilitate efficient convergence of CymbaDiff, we introduce a scene structure estimation network (SSEN) that produces a coarse structural representation of the target 3D scene, as shown in Figure 2. This structural prior guides the diffusion model towards geometrically plausible outputs during early generation steps. Inspired by recent advances in structural scene modeling [64, 74], the SSEN architecture incorporates multi-scale feature extraction modules with Dimensional Decomposition Residual (DDR) blocks. Specifically, multi-scale feature extraction modules capture hierarchical contextual information by aggregating features across multiple receptive fields. It employs parallel branches of $3 \times 3 \times 3$ convolutions to replace $5 \times 5 \times 5$ and $7 \times 7 \times 7$ convolutions, which are progressively stacked and merged at multiple levels, as shown in Figure 3 (b). The DDR structure decomposes a standard $k \times k \times k$ 3D convolution into a sequence of three separable layers: $1 \times 1 \times k$, $1 \times k \times 1$, and $k \times 1 \times 1$, as illustrated in Figure 3 (d). The multi-scale modules capture spatial context and semantically-rich features across different receptive fields, while the DDR blocks enhance the network's representational capacity with limited computational cost. Through joint use of these components, SSEN generates a voxel-based structural representation that accelerates convergence during the diffusion-driven 3D generation while improving geometric fidelity.

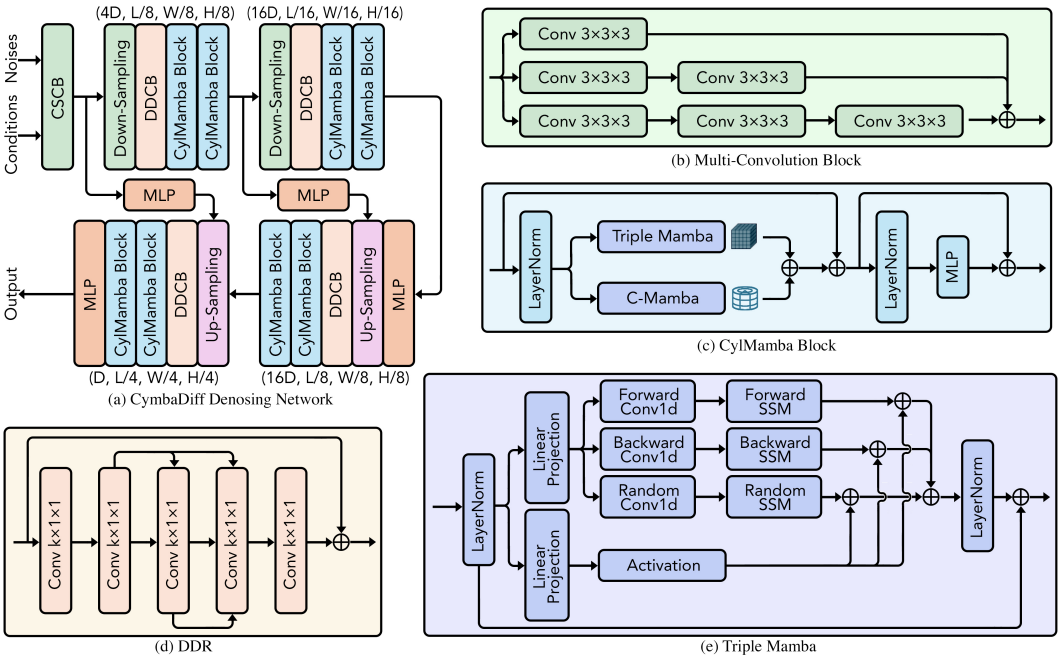

Figure 3: Architecture of the CymbaDiff denoising network. CylMamba denotes cylinder mamba block. Refer to the text for details.

4.2 Variational Autoencoder (VAE) / Latent Mapping Network

As illustrated in Figure 2, CymbaDiff operates in the latent space of a VAE, which provides a compact and informative representation for 3D semantic scenes. The VAE is trained with a combination of cross-entropy loss [75] and Lovász-Softmax loss [76]. This joint objective encourages alignment with the voxel grid manifold while mitigating the blurriness often introduced by conventional voxel-wise losses (like L_2 [77]). Given a voxelized input scene $V \in L \times W \times H$, the encoder $\mathbb{E}$ maps it to a latent representation $z = \mathbb{E}(V)$, the decoder $\mathbb{D}$ then reconstructs the scene as $\tilde{V} = \mathbb{D}(z) = \mathbb{D}(\mathbb{E}(V))$. In our implementation, $\mathbb{E}(\cdot)$ reduces the spatial resolution of the input voxel grid by a factor of $f = 4$, effectively compressing the scene while preserving key structural features. The VAE encoder consists of two down-sampling blocks, each comprising four consecutive convolutional layers. Every pair of convolutional layers is followed by a Batch Normalization layer and a ReLU activation function. Following these operations, a downsampling convolutional layer is applied, which is also followed by Batch Normalization and ReLU. To align with the VAE's latent distribution, the latent mapping network is designed to share the same architecture as the encoder.

4.3 Cross-Scale Contextual Block / Dilated Decomposed Convolution Block

We introduce the Cross-Scale Contextual Block (CSCB), inspired by hierarchical receptive fields in VGG [78] and multi-path processing in SCPNet [64]. CSCB efficiently captures local-to-global context from conditioning inputs with minimal memory overhead. Starting with a $3 \times 3 \times 3$ convolution, it has cascaded multi-covolution blocks (see Figure 3 (b)) with skip connections, and ends with another $3 \times 3 \times 3$ convolution before adding the residual output. Moreover, the Dilated Decomposed Convolution Block (DDCB) employs DDR blocks [74] with varying dilation rates of 1, 2 and 3 to capture diverse contextual features. The DDR structure is shown in Figure 3(d). The DDR block reduces computational cost of $C^{in} \times C^{out} \times k^3$ in traditional 3D convolutions to $C^{in} \times C^{out} \times 3k$ by breaking down the operations into $1 \times 1 \times k$, $1 \times k \times 1$, and $k \times 1 \times 1$ layers, which decreases the parameter count three times while maintaining detailed spatial layout. Therefore, this decomposition significantly reduces the number of parameters while preserving fine-grained spatial layout information.

4.4 CymbaDiff Denoising Network

As shown in Figure 2 (a), CymbaDiff generates scenes from conditional inputs and latent noise, drawing on the Mamba framework [79] to model sequences through a state-space formulation. A continuous input $x(t) \in \mathbb{R}$ is transformed into an output $y(t) \in \mathbb{R}$ via an intermediate hidden state

$h(t) \in \mathbb{R}^N$, before being discretized. The SSMs model [80] is typically formulated using linear ordinary differential equations (ODEs), defined as:

$$h^{'}(t) = Ah(t) + Bx(t), \quad y(t) = Ch(t), \tag{1}$$

where $A \in \mathbb{R}^{N\times N}$ and $B \in \mathbb{R}^{N\times 1}$, $C \in \mathbb{R}^{1\times N}$ denote the state matrix, input matrix, and output matrix, respectively. Since deriving the analytical solution for $h(t)$ is often intractable and real-world data is typically discrete, the system is discretized as follows:

$$h(t) = \overline{A}h(t-1) + \overline{B}x(t), \quad y(t) = \overline{C}h(t), \tag{2}$$

where $\overline{A} = \exp(\triangle A)$ and $\overline{B} = (\triangle A)^{-1}(\exp(\triangle A) - I) \cdot \triangle B$, $\overline{C} = C$ are the discretized state parameters and $\triangle$ is the discretization step size. The final output is obtained by applying a global convolution over a structured kernel. The downsampling and upsampling operations follow the design proposed in [25].

Cylinder Mamba Block. A core component of the CymbaDiff denoiser is the cylinder mamba block, illustrated in Figure 3 (c). This block integrates the Triple Mamba module [81] with our proposed cylinder mamba layer design to jointly leverage the advantages of both Cartesian and cylindrical coordinate representations. The Triple Mamba module, based on Cartesian grids, effectively preserves precise geometric distances, critical for modeling local physical neighborhoods. However, adjacent elements in Cartesian voxel sequences may misrepresent spatial relationships, limiting the effectiveness of sequential modeling. In contrast, the cylinder mamba layer (θ, r, z) imposes a structured spatial ordering that explicitly captures cylindrical continuity and vertical hierarchy. This ordering provides a vehicle-centric, geometrically coherent view, enabling angular-radial semantic tokenisation and supporting long-range context modelling with Mamba, for example, capturing structural information about sidewalks and buildings flanking the road.

The detailed structure of Triple Mamba layer is illustrated in Figure 3(e), and the cylinder mamba (C-Mamba) layer adopts the same architecture. Before entering the Mamba layers, input features undergo residual Layer Normalization (LN) on respective coordinate-based feature representation i.e, $z_{TMB}(t) = (LN(f_{TMB}(t))) + f_{TMB}(t)$ and $z_{CMB}(t) = (LN(f_{CMB}(t))) + f_{CMB}(t)$. $f_{TMB}(t)$ and $z_{TMB}(t)$ are the input and output features before the Triple Mamba layer, while $f_{CMB}(t)$ and $z_{CMB}(t)$ denote the corresponding features before cylinder mamba layer. The temporal dynamics of the Triple Mamba and C-Mamba layer input are thus governed by:

$$h(t) = \overline{A}h(t-1) + \overline{B}z_{TMB}(t), \quad y(t) = \overline{C}h(t), \tag{3}$$

$$h(t) = \overline{A}h(t-1) + \overline{B}z_{CMB}(t), \quad y(t) = \overline{C}h(t). \tag{4}$$

The Triple Mamba layer and C-mamba layer apply three separate Mamba modules, each operating on the same input $z_{TMB}(t)$ and $z_{CMB}(t)$ but with distinct ordering strategies: forward (ψ_i^f), backward (ψ_i^b), and random inter-slice (ψ_i^u) directions. The output of the i^{th} Triple Mamba layer and C-mamba layer are computed as:

$$\psi_i(z_{TMB}(t)) = \psi_i^f(z_{TMB}(t)) + \psi_i^b(z_{TMB}(t)) + \psi_i^u(z_{TMB}(t)), \tag{5}$$

$$\omega_i(z_{CMB}(t)) = \omega_i^f z_{CMB}(t)) + \psi_i^b(z_{CMB}(t)) + \psi_i^u(z_{CMB}(t)), \tag{6}$$

where $\psi_i(z_{TMB}(t))$ and $\omega_i(z_{CMB}(t))$ represent the outputs of the i^{th} triple Mamba and C-mamba layer. Fused 3D features from triple Mamba and C-mamba layers are formulated as $\psi_i^{all} = \phi_i^{all}(z_{TMB}(t)) + \omega_i^{all}(z_{CMB}(t))$, where $\phi_i^{all}(z_{TMB}(t)) = \text{MLP}(\text{LN}(\psi_i(z_{TMB}(t)))) + \psi_i(z_{TMB}(t))$ and $\omega_i^{all}(z_{CMB}(t)) = \text{MLP}(\text{LN}(\omega_i(z_{CMB}(t)))) + \omega_i(z_{CMB}(t))$. $\phi_i^{all}(z_{TMB}(t))$ and $\omega_i^{all}(z_{CMB}(t))$ denote the output feature from the triple Mamba and the C-mamba layer. MLP corresponds to stacked linear layers. Note that the input features in the C-Mamba layer are sorted by angular, radial, and vertical indices $((\theta, r, z))$, and the output features are mapped back to Cartesian spatial ordering (x, y, z) (the same ordering in the Triple Mamba layer) and fused with those from the Triple Mamba layer, allowing the model to jointly exploit radial and axis-aligned spatial cues. This joint representation enhances the model's ability to learn both local and global 3D spatial structures, capturing both Cartesian and cylindrical representations. Unlike the original Mamba [41, 82], which emphasizes directional context aggregation along scan lines with higher memory usage, our cylinder mamba block is specifically designed to efficiently capture spatially-structured 3D information.

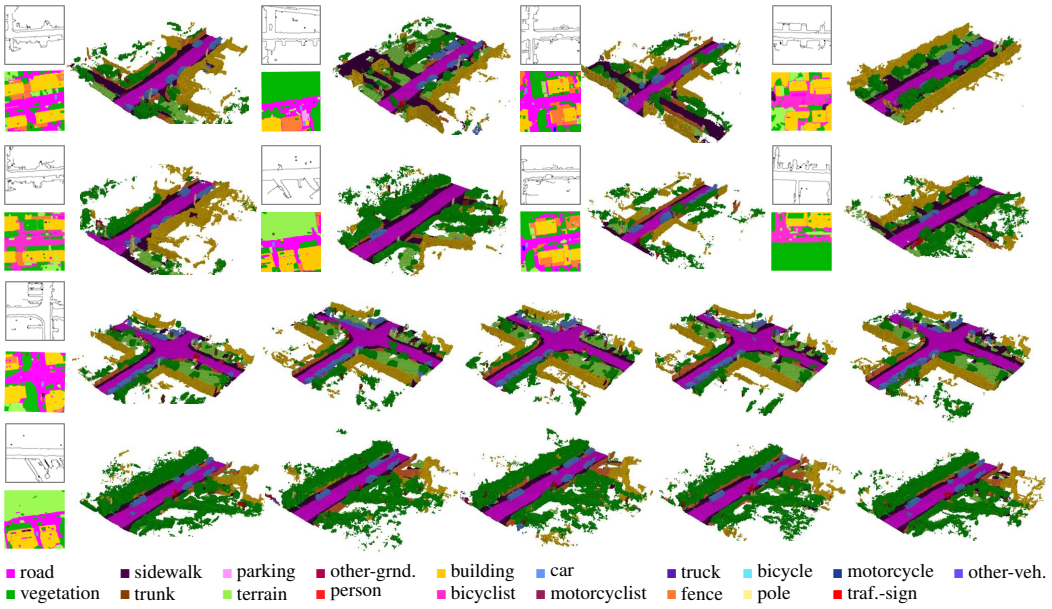

Figure 4: Qualitative results on the Sketch-based SemanticKITTI validation set. The 1st and 2nd rows show generated scenes conditioned on the corresponding freehand sketch and pseudo-labeled satellite images. The 3rd and 4th rows demonstrate the model's capability to generate moderately diverse 3D scenes (with different details) under identical input conditions.

5 Experiments

Implementation Details. Our model is trained on the Sketch-based SemanticKITTI training split from the SketchSem3D dataset. For evaluation, we use the validation splits of both the Sketch-based SemanticKITTI and Sketch-based KITTI-360 subsets, also from SketchSem3D. Following UrbanDiff [9], we train a dedicated network to extract latent features that encode both geometric and semantic information. These features are used to compute 3D FID and MMD, providing a joint assessment of generation quality and distributional similarity to ground-truth scenes. Additional implementation details are presented in the Appendix.

5.1 3D Semantic Scene Generation and Ablation Study

3D Semantic Scene Generation. As shown in Table 2, we compare our approach with two recent state-of-the-art baselines, SSD [57] and Semcity [11]. Across all evaluation metrics, our method consistently achieves superior performance. Notably, on the Sketch-based SemanticKITTI subset, it improves the FID score by approximately 16 points compared to Semcity [11], highlighting its effectiveness in interpreting sparse and abstract conditional inputs, such as freehand sketches and pseudo-labeled satellite image annotations.

SSD [57] and Semcity [11] both adopt 2D FID for evaluation. In contrast, we adopt more comprehensive 3D evaluation metrics, 3D FID and MMD, that more accurately assess geometric fidelity and semantic consistency in voxel space. To evaluate the effectiveness of our approach, we replace the CymbaDiff denoising network with two baselines: a 3D extension of the Latent Diffusion network [1] and the 3D DiT model [83], and conduct experiments on the SketchSem3D benchmark. Results in Table 2 show that our method consistently outperforms both baselines, demonstrating its superior performance in 3D semantic scene generation. For additional context, UrbanDiff [9] reports competitive performance, with a 3D FID of 291.4 and a 3D MMD of 0.11 on the NuScenes dataset. However, their experimental setting is less challenging, as the voxel resolution of NuScenes is $192 \times 192 \times 16$ and with only 17 semantic classes. In comparison, our benchmark dataset has a resolution of $256 \times 256 \times 256$ and with 20 classes for the Sketch-based SemanticKITTI subset and 16 classes for the Sketch-based KITTI-360 subset. The primary factor underlying this is that UrbanDiff operates solely within the Cartesian coordinate system, leading to the loss of important volumetric structural information. Furthermore, UrbanDiff does not release its source code or preprocessed data, which prevents direct comparison with our proposed task.

Table 2: Semantic scene generation results. SK: sketch, PSA: pseudo-labeled satellite image annotations. SSD and Semcity: 2D FID.

Datasets	Method	Condition	FID ↓	MMD ↓
SemanticKITTI	SSD [57]	-	112.82	-
	Semcity [11]	-	56.55	-
	3D Latent Diffusion [1]	SK+PSA	165.65	0.09
	3D DIT [83]	SK+PSA	138.86	0.08
	CymbaDiff (ours)	SK+PSA	**40.67**	**0.04**
KITTI-360	3D Latent Diffusion [1]	SK+PSA	330.86	0.12
	3D DIT [83]	SK+PSA	272.83	0.11
	CymbaDiff (ours)	SK+PSA	**107.53**	**0.08**

Table 3: Ablation study on Sketch-based SemanticKITTI test set. w/o: "without", C-Mamba: cylinder mamba.

Method	FID ↓	MMD ↓
w/o CSCB	90.53	0.06
w/o DDCB	76.57	0.06
w/o C-Mamba	74.09	0.05
CymbaDiff	40.67	0.04

In Table 2, to evaluate robustness and generalization, we directly applied our model, trained only on Sketch-based SemanticKITTI, to Sketch-based KITTI-360 without any fine-tuning. During this evaluation, only the overlapping class labels (16 classes) between the two subsets are used. Our model maintains top-tier performance, producing structurally coherent and semantically meaningful 3D scenes. This cross-dataset evaluation highlights the strong generalization capability of our approach.

We present qualitative results on the Sketch-based SemanticKITTI validation set in Figure 4. Rows 1 and 2 illustrate the generated semantic scenes conditioned on the input sketches and their corresponding PSAs. Rows 3 and 4 present additional generation results using the same input conditions to demonstrate both consistency and moderate diversity in scene synthesis. We see that our model effectively produces structurally accurate, and semantically meaningful 3D scenes that align well with inputs. These visualizations further demonstrate the model's ability to integrate abstract freehand sketches and pseudo-labeled satellite cues to generate high-quality semantic reconstructions. Some sketch-PSA pairs may have differences because the 3D ground truth annotations in SemanticKITTI were collected around 2013 and the satellite images used for PSA were captured around 2025. PSA generation, being automatic, is also prone to errors. In contrast, sketches originate directly from the 2013 ground-truth data, maintaining temporal consistency and serving as a stable spatial reference to mitigate the domain gap.

Observing the results of CymbaDiff on the proposed SketchSem3D dataset, it is apparent that it demonstrates strong performance, effectively handling challenges such as semantic misalignment caused by noisy pseudo-labels, e.g., due to confusion between vegetation and buildings. Nevertheless, this method does occasionally fail to accurately reconstruct small or occluded objects that are underrepresented in the training data or sparsely encoded in the sketch and PSA inputs. Although CymbaDiff mitigates this issue to some extent through the use of the Cross-Scale Contextual Block and Cylinder Mamba Block, which capture multi-scale contextual information, its performance could be further enhanced by increasing the representation of small objects in the dataset.

Ablation Study. We conducted systematic experiments to evaluate the impact of different components in our model and to quantify their individual contributions to the overall performance. As presented in Table 3, the ablation study offers valuable insights into the role and effectiveness of each component. These results allow us to isolate and identify the elements that most significantly enhance the model's performance in the 3D semantic scene generation task. Notably, the CSCB, DDCB, and cylinder mamba blocks play a critical role, as they enable the model to capture complex spatial and semantic relationships within 3D scenes more effectively. "w/o C-Mamba" refers to a variant that retains only the triple Mamba layers.

5.2 3D Semantic Scene Completion.

Since our work explores a new research direction and, currently, there are no directly comparable methods using the same input modalities, we compare CymbaDiff with existing state-of-the-art semantic scene completion methods that use monocular or stereo RGB inputs. However, we emphasize that our main contribution lies in 3D scene generation. Table 4 compares our method to 3D scene completion methods on the IoU and mIoU metrics reported in their respective publications. All methods are evaluated for 3D semantic scene completion on the SemanticKITTI validation set. The compared methods either use monocular or stereo (image) inputs. Remarkably, despite relying only on input SK and PSA, our method achieves highly competitive performance, matching or exceeding several leading methods that utilize richer input modalities. This demonstrates that SK and PSA offer

Table 4: Quantitative results on the SemanticKITTI validation set. The best results are indicated in **bold**. Mono and Stereo refer to methods using monocular and stereo inputs, respectively, while SK and PSA denote sketch and pseudo-labeled satellite annotations. Note that we demonstrate strong performance using SK+PSA, which are much easier to obtain than images.

Method	Input	IoU	mIoU	road	sidewalk	parking	other-grnd.	building	car	truck	bicycle	motorcycle	other-veh.	vegetation	trunk	terrain	person	bicyclist	motorcyclist	fence	pole	traf.-sign
MonoScene [84]	Mono	36.9	11.1	56.5	26.7	14.3	0.5	14.1	23.03	7.0	0.6	0.5	1.5	17.9	2.8	29.6	1.9	1.2	0.0	5.8	4.1	2.3
TPVFormer [85]	Mono	35.6	11.3	56.5	25.9	20.6	0.9	13.9	23.8	8.1	0.4	0.1	4.4	16.9	2.3	30.4	0.5	0.9	0.0	5.9	3.1	1.5
NDC-Scene [86]	Mono	37.2	12.7	59.2	28.2	**21.4**	1.7	14.9	26.3	14.8	**1.7**	**2.4**	7.7	19.1	3.5	31.0	3.6	2.7	0.0	6.7	4.5	2.7
OccFormer [87]	Mono	36.5	13.5	58.9	26.9	19.6	0.3	14.4	25.1	**25.5**	0.8	1.2	8.5	19.6	3.9	32.6	2.8	2.8	0.0	5.6	4.3	2.9
SparseOcc [88]	Mono	36.5	13.1	**59.6**	29.7	20.4	0.5	15.4	24.0	18.1	0.8	0.9	**8.9**	18.9	3.5	31.1	3.7	0.6	0.0	6.7	3.9	2.6
IAMSSC [60]	Mono	44.3	12.5	54.6	25.9	16.0	0.7	17.4	26.3	8.7	0.6	0.2	5.1	24.6	5.0	30.1	1.3	3.5	0.0	6.9	6.4	3.6
VoxFormer [89]	Stereo	44.2	13.4	53.6	26.5	19.7	0.4	19.5	26.5	7.3	1.3	0.6	7.8	26.1	6.1	33.1	1.9	2.0	0.0	7.3	9.2	4.9
DepthSSC [90]	Stereo	**45.8**	13.3	55.4	27.0	18.8	0.9	19.2	25.9	6.0	0.4	1.2	7.5	26.4	4.5	30.2	2.6	**6.3**	0.0	8.5	7.4	4.1
HASSC-S [91]	Stereo	44.8	13.5	57.1	28.3	15.9	1.1	19.1	27.2	9.9	0.9	0.9	5.6	25.5	6.2	32.9	2.8	4.7	0.0	6.6	7.7	4.1
H2GFormer-S [61]	Stereo	44.6	13.7	56.1	29.1	17.8	0.5	19.7	28.2	10.0	0.5	0.5	7.4	26.3	6.8	34.4	1.5	2.9	0.0	7.2	7.9	4.7
CymbaDiff	SK+PSA	43.2	**14.6**	52.4	**33.3**	13.1	**10.9**	**32.4**	32.1	0.8	1.0	0.0	3.2	**28.0**	**8.7**	22.2	**4.6**	4.9	0.0	**11.2**	**12.7**	**5.2**

a flexible alternative, especially when RGB data are unavailable or impractical, such as in remote sensing.

For semantic scene completion, our method achieves 43.2% IoU and 14.6% mIoU on the SemanticKITTI validation set, outperforming the leading monocular baseline by 1.1% mIoU and the best stereo-based method by 0.9%. This performance gain underscores the strong representational and generative capabilities of CymbaDiff, particularly in reconstructing large-scale structures such as sidewalks, buildings, vegetation, other-ground, and fences. In addition, our method maintains competitive accuracy for smaller objects like people, poles, traffic signs, and tree trunks, demonstrating robustness across a wide range of object sizes and semantic categories. These results collectively highlight the effectiveness and versatility of our approach in diverse urban scene contexts. We present further qualitative examples, including results on underrepresented classes in the Appendix.

6 Conclusion

We introduced a novel and scalable task: 3D outdoor semantic scene generation from sketches and pseudo-labeled satellite image annotations. This task offers a low-cost and flexible alternative to traditional annotation-intensive methods, particularly beneficial for applications such as autonomous driving, urban planning. To achieve this, we proposed SketchSem3D, the first publicly available dataset specifically designed for multi-conditioned scene generation in outdoor environments. We proposed CymbaDiff, a diffusion-based generative model designed to enforce structured spatial coherence by explicitly modeling angular continuity and vertical hierarchies, while preserving physical local and global spatial relationships within 3D scenes. CymbaDiff achieves top-tier performance for 3D scene generation and completion using only sparse and abstract input modalities, establishing a solid baseline for future advancements in this field. We hope our new task, dataset, and approach (including code) would foster advancements in related areas.

Broader Impacts. CymbaDiff model inherently neutral and designed for positive human-centric applications such as urban simulation and autonomous driving, may pose potential societal risks if misused, particularly in scenarios involving unauthorized mass surveillance.

Limitations. While CymbaDiff generates high-quality 3D semantic scenes from freehand (like) sketches and pseudo-labeled satellite image annotations (PSA), obtaining authentic human-drawn sketches could further improve its generalizability and effectiveness in practical human–AI interaction tasks. Future work could focus on using authentic human-drawn sketches for 3D semantic scene generation.

7 Acknowledgments

This research was supported by the Australian Government through the Australian Research Council's Discovery Projects funding scheme (project # DP240101926). Professor Ajmal Mian is the recipient of an ARC Future Fellowship Award (project # FT210100268) funded by the Australian Government.

Dr. Naveed Akhtar is a recipient of the ARC Discovery Early Career Researcher Award (project # DE230101058), funded by the Australian Government.

References

[1] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695, 2022.

[2] Florinel-Alin Croitoru, Vlad Hondru, Radu Tudor Ionescu, and Mubarak Shah. Diffusion models in vision: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(9):10850–10869, 2023.

[3] Ling Yang, Zhilong Zhang, Yang Song, Shenda Hong, Runsheng Xu, Yue Zhao, Wentao Zhang, Bin Cui, and Ming-Hsuan Yang. Diffusion models: A comprehensive survey of methods and applications. *ACM Computing Surveys*, 56(4):1–39, 2023.

[4] Hanqun Cao, Cheng Tan, Zhangyang Gao, Yilun Xu, Guangyong Chen, Pheng-Ann Heng, and Stan Z Li. A survey on generative diffusion models. *IEEE Transactions on Knowledge and Data Engineering*, 2024.

[5] Yizhi Song, Zhifei Zhang, Zhe Lin, Scott Cohen, Brian Price, Jianming Zhang, Soo Ye Kim, and Daniel Aliaga. Objectstitch: Object compositing with diffusion model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18310–18319, 2023.

[6] Chenfeng Xu, Huan Ling, Sanja Fidler, and Or Litany. 3difftection: 3d object detection with geometry-aware diffusion features. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10617–10627, 2024.

[7] Siyuan Huang, Zan Wang, Puhao Li, Baoxiong Jia, Tengyu Liu, Yixin Zhu, Wei Liang, and Song-Chun Zhu. Diffusion-based generation, optimization, and planning in 3d scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16750–16761, 2023.

[8] Xiaoliang Ju, Zhaoyang Huang, Yijin Li, Guofeng Zhang, Yu Qiao, and Hongsheng Li. Diffindscene: Diffusion-based high-quality 3d indoor scene generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4526–4535, 2024.

[9] Junge Zhang, Qihang Zhang, Li Zhang, Ramana Rao Kompella, Gaowen Liu, and Bolei Zhou. Urban scene diffusion through semantic occupancy map. *arXiv preprint arXiv:2403.11697*, 2024.

[10] Yuheng Liu, Xinke Li, Xueting Li, Lu Qi, Chongshou Li, and Ming-Hsuan Yang. Pyramid diffusion for fine 3d large scene generation. In *European Conference on Computer Vision*, pages 71–87. Springer, 2024.

[11] Jumin Lee, Sebin Lee, Changho Jo, Woobin Im, Juhyeong Seon, and Sung-Eui Yoon. Semcity: Semantic scene generation with triplane diffusion. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 28337–28347, 2024.

[12] Alaia Sola, Cristina Corchero, Jaume Salom, and Manel Sanmarti. Simulation tools to build urban-scale energy models: A review. *Energies*, 11(12):3269, 2018.

[13] Alaia Sola, Cristina Corchero, Jaume Salom, and Manel Sanmarti. Multi-domain urban-scale energy modelling tools: A review. *Sustainable Cities and Society*, 54:101872, 2020.

[14] Xuran Pan, Zhuofan Xia, Shiji Song, Li Erran Li, and Gao Huang. 3d object detection with pointformer. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 7463–7472, 2021.

[15] Yingwei Li, Adams Wei Yu, Tianjian Meng, Ben Caine, Jiquan Ngiam, Daiyi Peng, Junyang Shen, Yifeng Lu, Denny Zhou, Quoc V Le, et al. Deepfusion: Lidar-camera deep fusion for multi-modal 3d object detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 17182–17191, 2022.

[16] Junbo Yin, Jianbing Shen, Runnan Chen, Wei Li, Ruigang Yang, Pascal Frossard, and Wenguan Wang. Is-fusion: Instance-scene collaborative fusion for multimodal 3d object detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 14905–14915, 2024.

[17] Zhiwei Lin, Zhe Liu, Zhongyu Xia, Xinhao Wang, Yongtao Wang, Shengxiang Qi, Yang Dong, Nan Dong, Le Zhang, and Ce Zhu. Rcbevdet: radar-camera fusion in bird's eye view for 3d object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14928–14937, 2024.

[18] Hengwei Bian, Lingdong Kong, Haozhe Xie, Liang Pan, Yu Qiao, and Ziwei Liu. Dynamiccity: Large-scale 4d occupancy generation from dynamic scenes. *The Eleventh International Conference on Learning Representations*, 2024.

[19] Lucas Nunes, Rodrigo Marcuzzi, Jens Behley, and Cyrill Stachniss. Towards generating realistic 3d semantic training data for autonomous driving. *arXiv preprint arXiv:2503.21449*, 2025.

[20] Aryan Mikaeili, Or Perel, Mehdi Safaee, Daniel Cohen-Or, and Ali Mahdavi-Amiri. Sked: Sketch-guided text-based 3d editing. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14607–14619, 2023.

[21] Aditya Sanghi, Pradeep Kumar Jayaraman, Arianna Rampini, Joseph Lambourne, Hooman Shayani, Evan Atherton, and Saeid Asgari Taghanaki. Sketch-a-shape: Zero-shot sketch-to-3d shape generation. *arXiv preprint arXiv:2307.03869*, 2023.

[22] Zijie Wu, Yaonan Wang, Mingtao Feng, He Xie, and Ajmal Mian. Sketch and text guided diffusion model for colored point cloud generation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8929–8939, 2023.

[23] Zijie Wu, Mingtao Feng, Yaonan Wang, He Xie, Weisheng Dong, Bo Miao, and Ajmal Mian. External knowledge enhanced 3d scene generation from sketch. In *European Conference on Computer Vision*, pages 286–304. Springer, 2024.

[24] Lianghui Zhu, Bencheng Liao, Qian Zhang, Xinlong Wang, Wenyu Liu, and Xinggang Wang. Vision mamba: Efficient visual representation learning with bidirectional state space model. In *Forty-first International Conference on Machine Learning*.

[25] Zhaohu Xing, Tian Ye, Yijun Yang, Guang Liu, and Lei Zhu. Segmamba: Long-range sequential modeling mamba for 3d medical image segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 578–588. Springer, 2024.

[26] Jun Ma, Feifei Li, and Bo Wang. U-mamba: Enhancing long-range dependency for biomedical image segmentation. *arXiv preprint arXiv:2401.04722*, 2024.

[27] Jiuming Liu, Ruiji Yu, Yian Wang, Yu Zheng, Tianchen Deng, Weicai Ye, and Hesheng Wang. Point mamba: A novel point cloud backbone based on state space model with octree-based ordering strategy. *arXiv preprint arXiv:2403.06467*, 2024.

[28] Tao Zhang, Haobo Yuan, Lu Qi, Jiangning Zhang, Qianyu Zhou, Shunping Ji, Shuicheng Yan, and Xiangtai Li. Point cloud mamba: Point cloud learning via state space model. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 10121–10130, 2025.

[29] Mehdi Cherti, Romain Beaumont, Ross Wightman, Mitchell Wortsman, Gabriel Ilharco, Cade Gordon, Christoph Schuhmann, Ludwig Schmidt, and Jenia Jitsev. Reproducible scaling laws for contrastive language-image learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2818–2829, 2023.

[30] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4015–4026, 2023.

[31] Quentin Anthony, Yury Tokpanov, Paolo Glorioso, and Beren Millidge. Blackmamba: Mixture of experts for state-space models. *arXiv preprint arXiv:2402.01771*, 2024.

[32] Wenrui Li, Xiaopeng Hong, and Xiaopeng Fan. Spikemba: Multi-modal spiking saliency mamba for temporal video grounding. *arXiv preprint arXiv:2404.01174*, 2024.

[33] Ziyang Wang, Jian-Qing Zheng, Yichi Zhang, Ge Cui, and Lei Li. Mamba-unet: Unet-like pure visual mamba for medical image segmentation. *arXiv preprint arXiv:2402.05079*, 2024.

[34] Rui Deng and Tianpei Gu. Cu-mamba: Selective state space models with channel learning for image restoration. In *2024 IEEE 7th International Conference on Multimedia Information Processing and Retrieval (MIPR)*, pages 328–334. IEEE, 2024.

[35] Hang Guo, Jinmin Li, Tao Dai, Zhihao Ouyang, Xudong Ren, and Shu-Tao Xia. Mambair: A simple baseline for image restoration with state-space model. In *European conference on computer vision*, pages 222–241. Springer, 2024.

[36] Shida Wang and Qianxiao Li. Stablessm: Alleviating the curse of memory in state-space models through stable reparameterization. *arXiv preprint arXiv:2311.14495*, 2023.

[37] Roger Waleffe, Wonmin Byeon, Duncan Riach, Brandon Norick, Vijay Korthikanti, Tri Dao, Albert Gu, Ali Hatamizadeh, Sudhakar Singh, Deepak Narayanan, et al. An empirical study of mamba-based language models. *arXiv preprint arXiv:2406.07887*, 2024.

[38] Dingkang Liang, Xin Zhou, Wei Xu, Xingkui Zhu, Zhikang Zou, Xiaoqing Ye, Xiao Tan, and Xiang Bai. Pointmamba: A simple state space model for point cloud analysis. *arXiv preprint arXiv:2402.10739*, 2024.

[39] Yue Liu, Yunjie Tian, Yuzhong Zhao, Hongtian Yu, Lingxi Xie, Yaowei Wang, Qixiang Ye, Jianbin Jiao, and Yunfan Liu. Vmamba: Visual state space model. *Advances in neural information processing systems*, 37:103031–103063, 2024.

[40] Eric Nguyen, Karan Goel, Albert Gu, Gordon Downs, Preey Shah, Tri Dao, Stephen Baccus, and Christopher Ré. S4nd: Modeling images and videos as multidimensional signals with state spaces. *Advances in neural information processing systems*, 35:2846–2861, 2022.

[41] Shufan Li, Harkanwar Singh, and Aditya Grover. Mamba-nd: Selective state space modeling for multi-dimensional data. *arXiv preprint arXiv:2402.05892*, 2024.

[42] Ian J Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. *Advances in neural information processing systems*, 27, 2014.

[43] Diederik P Kingma, Max Welling, et al. Auto-encoding variational bayes, 2013.

[44] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

[45] Ruoshi Liu, Rundi Wu, Basile Van Hoorick, Pavel Tokmakov, Sergey Zakharov, and Carl Vondrick. Zero-1-to-3: Zero-shot one image to 3d object. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9298–9309, 2023.

[46] Guocheng Qian, Jinjie Mai, Abdullah Hamdi, Jian Ren, Aliaksandr Siarohin, Bing Li, Hsin-Ying Lee, Ivan Skorokhodov, Peter Wonka, Sergey Tulyakov, et al. Magic123: One image to high-quality 3d object generation using both 2d and 3d diffusion priors. *arXiv preprint arXiv:2306.17843*, 2023.

[47] Xuanchi Ren, Jiahui Huang, Xiaohui Zeng, Ken Museth, Sanja Fidler, and Francis Williams. Xcube: Large-scale 3d generative modeling using sparse voxel hierarchies. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4209–4219, 2024.

[48] Jianfeng Xiang, Zelong Lv, Sicheng Xu, Yu Deng, Ruicheng Wang, Bowen Zhang, Dong Chen, Xin Tong, and Jiaolong Yang. Structured 3d latents for scalable and versatile 3d generation. *arXiv preprint arXiv:2412.01506*, 2024.

[49] Chao Xu, Ang Li, Linghao Chen, Yulin Liu, Ruoxi Shi, Hao Su, and Minghua Liu. Sparp: Fast 3d object reconstruction and pose estimation from sparse views. In *European Conference on Computer Vision*, pages 143–163. Springer, 2024.

[50] Guandao Yang, Xun Huang, Zekun Hao, Ming-Yu Liu, Serge Belongie, and Bharath Hariharan. Pointflow: 3d point cloud generation with continuous normalizing flows. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4541–4550, 2019.

[51] Bo Miao, Mohammed Bennamoun, Yongsheng Gao, and Ajmal Mian. Spectrum-guided multi-granularity referring video object segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 920–930, October 2023.

[52] Bo Miao, Mingtao Feng, Zijie Wu, Mohammed Bennamoun, Yongsheng Gao, and Ajmal Mian. Referring human pose and mask estimation in the wild. *Advances in Neural Information Processing Systems*, 37:44791–44813, 2024.

[53] Alexey Bokhovkin, Quan Meng, Shubham Tulsiani, and Angela Dai. Scenefactor: Factored latent 3d diffusion for controllable 3d scene generation. *arXiv preprint arXiv:2412.01801*, 2024.

[54] Chuan Fang, Yuan Dong, Kunming Luo, Xiaotao Hu, Rakesh Shrestha, and Ping Tan. Ctrl-room: controllable text-to-3d room meshes generation with layout constraints. *arXiv preprint arXiv:2310.03602*, 2023.

[55] Jiapeng Tang, Yinyu Nie, Lev Markhasin, Angela Dai, Justus Thies, and Matthias Nießner. Diffuscene: Denoising diffusion models for generative indoor scene synthesis. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 20507–20518, 2024.

[56] Xiuyu Yang, Yunze Man, Junkun Chen, and Yu-Xiong Wang. Scenecraft: Layout-guided 3d scene generation. *Advances in Neural Information Processing Systems*, 37:82060–82084, 2024.

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

[58] Quan Meng, Lei Li, Matthias Nießner, and Angela Dai. Lt3sd: Latent trees for 3d scene diffusion. *arXiv preprint arXiv:2409.08215*, 2024.

[59] Zhu Yu, Runming Zhang, Jiacheng Ying, Junchen Yu, Xiaohai Hu, Lun Luo, Siyuan Cao, and Huiliang Shen. Context and geometry aware voxel transformer for semantic scene completion. *arXiv preprint arXiv:2405.13675*, 2024.

[60] Haihong Xiao, Hongbin Xu, Wenxiong Kang, and Yuqiong Li. Instance-aware monocular 3d semantic scene completion. *IEEE T-ITS*, 2024.

[61] Yu Wang and Chao Tong. H2gformer: Horizontal-to-global voxel transformer for 3d semantic scene completion. In *AAAI*, volume 38, pages 5722–5730, 2024.

[62] Yuwen Xiong, Wei-Chiu Ma, Jingkang Wang, and Raquel Urtasun. Learning compact representations for lidar completion and generation. In *CVPR*, pages 1074–1083, 2023.

[63] Ran Cheng, Christopher Agia, Yuan Ren, Xinhai Li, and Liu Bingbing. S3cnet: A sparse semantic scene completion network for lidar point clouds. In *CoRL*, pages 2148–2161, 2021.

[64] Zhaoyang Xia, Youquan Liu, Xin Li, Xinge Zhu, Yuexin Ma, Yikang Li, Yuenan Hou, and Yu Qiao. Scpnet: Semantic scene completion on point cloud. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 17642–17651, 2023.

[65] Yingjie Cai, Xuesong Chen, Chao Zhang, Kwan-Yee Lin, Xiaogang Wang, and Hongsheng Li. Semantic scene completion via integrating instances and scene in-the-loop. In *CVPR*, pages 324–333, 2021.

[66] Xuzhi Wang, Di Lin, and Liang Wan. Ffnet: Frequency fusion network for semantic scene completion. In *AAAI*, volume 36, pages 2550–2557, 2022.

[67] Jens Behley, Martin Garbade, Andres Milioto, Jan Quenzel, Sven Behnke, Cyrill Stachniss, and Jurgen Gall. Semantickitti: A dataset for semantic scene understanding of lidar sequences. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9297–9307, 2019.

[68] Yiming Li, Sihang Li, Xinhao Liu, Moonjun Gong, Kenan Li, Nuo Chen, Zijun Wang, Zhiheng Li, Tao Jiang, Fisher Yu, et al. Sscbench: Monocular 3d semantic scene completion benchmark in street views. 2023.

[69] Yibo Li and Bailun Liu. Improved edge detection algorithm for canny operator. In *2022 IEEE 10th Joint International Information Technology and Artificial Intelligence Conference (ITAIC)*, volume 10, pages 1–5. IEEE, 2022.

[70] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

[71] Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *2012 IEEE conference on computer vision and pattern recognition*, pages 3354–3361. IEEE, 2012.

[72] Yiyi Liao, Jun Xie, and Andreas Geiger. Kitti-360: A novel dataset and benchmarks for urban scene understanding in 2d and 3d. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(3):3292–3310, 2022.

[73] Despoina Paschalidou, Amlan Kar, Maria Shugrina, Karsten Kreis, Andreas Geiger, and Sanja Fidler. Atiss: Autoregressive transformers for indoor scene synthesis. *Advances in Neural Information Processing Systems*, 34:12013–12026, 2021.

[74] Jie Li, Yu Liu, Dong Gong, Qinfeng Shi, Xia Yuan, Chunxia Zhao, and Ian Reid. Rgbd based dimensional decomposition residual network for 3d semantic scene completion. In *CVPR*, pages 7693–7702, 2019.

[75] Zhilu Zhang and Mert Sabuncu. Generalized cross entropy loss for training deep neural networks with noisy labels. *Advances in neural information processing systems*, 31, 2018.

[76] Maxim Berman, Amal Rannen Triki, and Matthew B Blaschko. The lovász-softmax loss: A tractable surrogate for the optimization of the intersection-over-union measure in neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4413–4421, 2018.

[77] Batia Laufer and Stig Eliasson. What causes avoidance in l2 learning: L1-l2 difference, l1-l2 similarity, or l2 complexity? *Studies in second language acquisition*, 15(1):35–48, 1993.

[78] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[79] Albert Gu and Tri Dao. Mamba: Linear-time sequence modeling with selective state spaces. *arXiv preprint arXiv:2312.00752*, 2023.

[80] Albert Gu, Karan Goel, and Christopher Ré. Efficiently modeling long sequences with structured state spaces. *arXiv preprint arXiv:2111.00396*, 2021.

[81] Yijun Yang, Zhaohu Xing, and Lei Zhu. Vivim: a video vision mamba for medical video object segmentation. *arXiv preprint arXiv:2401.14168*, 2024.

[82] Lianghui Zhu, Bencheng Liao, Qian Zhang, Xinlong Wang, Wenyu Liu, and Xinggang Wang. Vision mamba: Efficient visual representation learning with bidirectional state space model. *arXiv preprint arXiv:2401.09417*, 2024.

[83] Shentong Mo, Enze Xie, Ruihang Chu, Lanqing Hong, Matthias Niessner, and Zhenguo Li. Dit-3d: Exploring plain diffusion transformers for 3d shape generation. *Advances in neural information processing systems*, 36:67960–67971, 2023.

[84] Anh-Quan Cao and Raoul De Charette. Monoscene: Monocular 3d semantic scene completion. In *The IEEE/CVF Conference: CVPR*, pages 3991–4001, 2022.

[85] Yuanhui Huang, Wenzhao Zheng, Yunpeng Zhang, Jie Zhou, and Jiwen Lu. Tri-perspective view for vision-based 3d semantic occupancy prediction. In *The IEEE/CVF Conference: CVPR*, pages 9223–9232, 2023.

[86] Jiawei Yao, Chuming Li, Keqiang Sun, Yingjie Cai, Hao Li, Wanli Ouyang, and Hongsheng Li. Ndc-scene: Boost monocular 3d semantic scene completion in normalized device coordinates space. In *The IEEE/CVF Conference: ICCV*, pages 9421–9431, 2023.

[87] Yunpeng Zhang, Zheng Zhu, and Dalong Du. Occformer: Dual-path transformer for vision-based 3d semantic occupancy prediction. In *The IEEE/CVF Conference: ICCV*, pages 9433–9443, 2023.

[88] Pin Tang, Zhongdao Wang, Guoqing Wang, Jilai Zheng, Xiangxuan Ren, Bailan Feng, and Chao Ma. Sparseocc: Rethinking sparse latent representation for vision-based semantic occupancy prediction. In *The IEEE/CVF Conference: CVPR*, pages 15035–15044, 2024.

[89] Yiming Li, Zhiding Yu, Christopher Choy, Chaowei Xiao, Jose M Alvarez, Sanja Fidler, Chen Feng, and Anima Anandkumar. Voxformer: Sparse voxel transformer for camera-based 3d semantic scene completion. In *The IEEE/CVF Conference: CVPR*, pages 9087–9098, 2023.

[90] Jiawei Yao and Jusheng Zhang. Depthssc: Depth-spatial alignment and dynamic voxel resolution for monocular 3d semantic scene completion. *arXiv preprint arXiv:2311.17084*, 2023.

[91] Song Wang, Jiawei Yu, Wentong Li, Wenyu Liu, Xiaolu Liu, Junbo Chen, and Jianke Zhu. Not all voxels are equal: Hardness-aware semantic scene completion with self-distillation. In *The IEEE/CVF Conference: CVPR*, pages 14792–14801, 2024.

[92] Haozhe Xie, Zhaoxi Chen, Fangzhou Hong, and Ziwei Liu. Citydreamer: Compositional generative model of unbounded 3d cities. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9666–9675, 2024.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction clearly state the claims, which are supported by our experimental results. See Abstract and Sec. 1.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: See Sec. 6.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [NA]

Justification: The paper does not contain any theoretical assumptions or proofs.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: We report the dataset, model, and training details in Sec. 3, Sec. 4, Sec. 5, and the Appendix. The introduced dataset and code are publicly available.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: The introduced dataset and code are publicly available.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (https://nips.cc/public/guides/CodeSubmissionPolicy) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (https://nips.cc/public/guides/CodeSubmissionPolicy) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: See Sec. 5 and the Appendix.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

   Justification: We follow the usual format used in previous related works to report and compare the results.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).

- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: See the Appendix.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics https://neurips.cc/public/EthicsGuidelines?

   Answer: [Yes]

   Justification: The research conducted in the paper conforms with NeurIPS Code of Ethics.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: See Sec. 6.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.

- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer:[NA]

    Justification: The paper poses no such risks.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: We cite the creators or original papers of all the related code, data, and models used in this paper. All the assets are free for research studies and widely used in previous related works. We also provide the licenses of the used datasets in the Appendix.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.

- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: We report the dataset, model, and training details in Sec. 3, Sec. 4, Sec. 5, and the Appendix. The introduced dataset and code are publicly available.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: We did not use crowdsourcing or conduct research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: We did not use crowdsourcing or conduct research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

A. Appendix

A.1 Training Objective

Due to the complexity of the task, our training objective combines multiple loss terms. For the 3D VAE, the objective is:

$$\mathcal{L} = \mathcal{L}_{CE} + \gamma \mathcal{L}_{Lovasz} - \beta D_{KL}\left(q_\phi\left(z|x\right)||\,p\left(z\right)\right), \tag{7}$$

where $\gamma = 1.0$ and $\beta = 0.001$ balance the contributions of each loss component, $\mathcal{L}_{CE}$ and $\mathcal{L}_{Lovasz}$ denote the standard cross-entropy and Lovasz-Softmax losses, respectively, following SCPNet [63]. D_{KL} denotes the Kullback–Leibler Divergence between the approximate posterior $q_\phi\left(z|x\right)$ and the prior $p\left(z\right)$, similar to the Latent Diffusion Model (LDM) [1]. The objective function for the CymbaDiff denoising network follows the LDM [1], minimizing the expected squared error between the predicted noise and true noise:

$$\mathcal{L}_{\text{LDM}} = \mathbb{E}_{x,\epsilon\sim N(0,1),t}\left[\|\epsilon - \epsilon_\theta\left(x_t, t\right)\|_2^2\right], \tag{8}$$

where $\epsilon_\theta(x_t, t)$ denotes a uniformly-weighted denoising autoencoder applied across time steps $t = 1, \ldots, T$. At each step t, the model predicts a denoised estimate of the input x_t, which is a noise-corrupted version of the original input x.

A.2 Evaluation Metrics

To evaluate the quality and diversity of the generated 3D semantic scenes, we use two widely used metrics: Fréchet Inception Distance (FID)[73] and Maximum Mean Discrepancy (MMD)[9]. Together, these metrics capture both the statistical similarity and feature-level fidelity between the generated and real data, providing a comprehensive assessment of generative performance. Specifically, FID measures the similarity between the distributions of generated and real samples in a latent feature space. Formally, FID is defined as:

$$\text{FID} = \|M_t - M_g\|_2^2 + Tr\left(C_t + C_g - 2(C_t C_g)^{\frac{1}{2}}\right), \tag{9}$$

where (M_t, M_g) and (C_t, C_g) are the mean and covariance of the real and generated feature distributions. MMD is a non-parametric, kernel-based metric that quantifies the distance between two probability distributions. Unlike FID, MMD does not rely on the assumption that features follow a Gaussian distribution, making it suitable for evaluating generative models under more flexible conditions. In our case, MMD is computed using a Gaussian kernel applied to features extracted from the same latent space as used for FID. The formal definition of MMD is:

$$\text{MMD}^2\left(X, Y\right) = \mathbb{E}_{x,x'}\left[k\left(x, x^{'}\right)\right] + \mathbb{E}_{y,y'}\left[k\left(y, y^{'}\right)\right] - 2\mathbb{E}_{x,y}\left[k\left(x, y\right)\right] \tag{10}$$

where $X = \{x_1, x_2, ..., x_m\}$ and $Y = \{y_1, y_2, ..., y_m\}$ denote the sets of latent features extracted from real and generated 3D scenes, respectively.

A.3 Additional Implementation Details

All experiments were conducted on a single NVIDIA GeForce RTX 4090 GPU with 24 GB of RAM. The Variational Autoencoder (VAE) was trained for 22 epochs using the AdamW optimizer with an initial learning rate of 3e-4. The VAE and the CymbaDiff denoising network were trained with a batch size of 2 and 4, each occupying approximately 20 GB of GPU memory. The CymbaDiff denoiser was trained for 31 epochs using the AdamW optimizer with a learning rate of 1e-3 and a weight decay of 1e-4. The number of denoising steps in CymbaDiff was set to 100. A WarmupCosineLR scheduler was used in all training stages to gradually decrease the learning rate, which helped ensure stable convergence.

A.4 VAE Results

Our CymbaDiff denosing network operates in the latent space of a VAE. To ensure high-quality semantic scene generation, this VAE needs to be accurate. We report the performance of the proposed VAE on the SemanticKITTI validation set in Table 5.

Table 5: VAE reconstruction performance on SemanticKITTI validation set. IoU and mIoU denote Intersection over Union and mean Intersection over Union, respectively.

Model	Original Spatial Size	Latent Spatial Size	Latent Channel	training epoch	batch size	IoU	mIoU
VAE	$256 \times 256 \times 32$	$64 \times 64 \times 8$	8	22	2	92.1	92.0

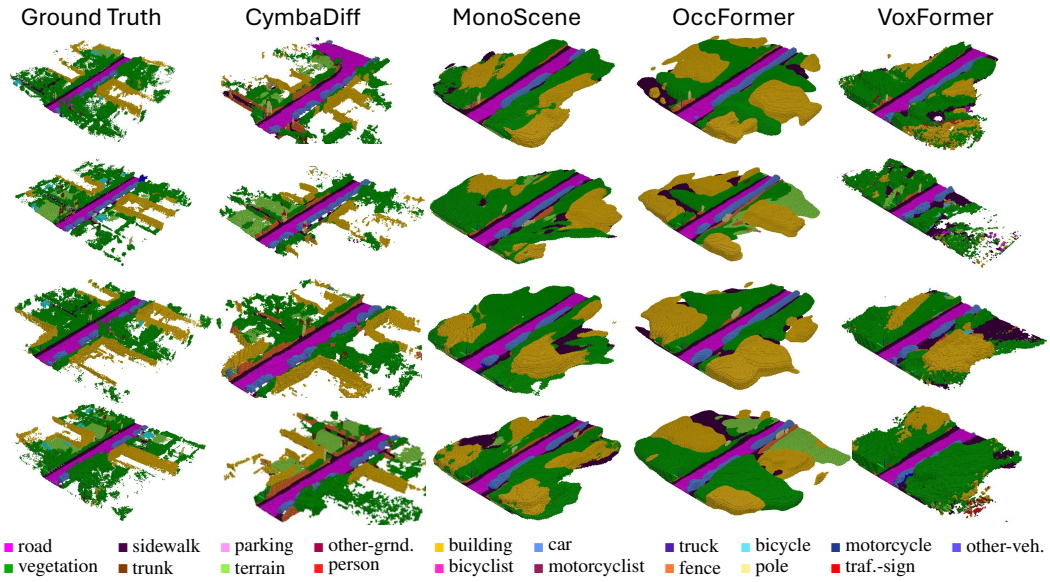

Figure 5: Qualitative results on the SemanticKITTI validation set. Columns from the left represent ground truth, and outputs of CymbaDiff (our method), MonoScene, OccFormer, and VoxFormer.

A.5 Effieicency Comparison

we provide quantitative comparisons in the Table 6 across methods in terms of parameter count and runtime performance. These results demonstrate that CymbaDiff achieves a favorable trade-off between model efficiency and computational cost, offering competitive performance with significantly fewer parameters compared to these two generative models.

A.6 Cross-domain Test

We have now trained SemCity[11] and CityDreamer[92]on the SketchSem3D dataset to compare with our CymbaDiff. To ensure compatibility with our 3D voxel-based setup, we integrated their denoisers into our framework. We also attempted to train the full SemCity pipeline directly, but it resulted in unstable training, with the VAE loss diverging to NaN, an issue also reported by other users on SemCity's official GitHub page. Please note, CityDreamer is designed for 2D generation and cannot be directly applied to 3D voxel scenes. As shown in the Table 7, CymbaDiff consistently outperforms both baselines across all evaluation metrics strongly.

The reason why Semcity and CityDreamer do not perform well in our experiments is their denoisers (provided in their official GitHub repositories). The denoiser in SemCity only has convolutional and linear layers, whereas that in CityDreamer relies on a simple stacking of transformer layers. Although transformer layers can model long-range dependencies, such simplified designs may be suboptimal for large-scale 3D voxel scene generation, where sparse and irregular data demand specialized mechanisms to effectively capture both local geometry and relevant global context.

A.7 Qualitative results on 3D Semantic Scene Completion

To demonstrate the effectiveness of our proposed framework for 3D semantic scene completion, we present additional qualitative results in Figures 5. The figure displays representative examples randomly selected from the SemanticKITTI validation set [67]. CymbaDiff accurately delineates fine-grained boundaries of 3D scenes and objects by incorporating the cylinder Mamba blocks, which

Table 6: Efficiency comparison. M: Million, and S: seconds.

Input Modality	Parameters (M)	Inference Times (S)
3D DIT	195	4.5
3D Latent Diffusion	1265	11.4
CymbaDiff	23	7.2

Table 7: Cross-domain Comparison

Method	Sketch-based SemanticKITTI FID ↓	Sketch-based SemanticKITTI MMD ↓	Sketch-based KITTI-360 FID ↓	Sketch-based KITTI-360 MMD ↓
3D SemCity[11]	987.91	0.26	740.09	0.25
3D CityDreamer[92]	950.16	0.26	754.47	0.25
CymbaDiff	40.67	0.04	107.53	0.08

promotes structured spatial coherence through explicit modeling of angular continuity and vertical hierarchies.

A.8 Licenses

Licenses of SemanticKITTI and SSCBench KITTI-360. The SemanticKITTI dataset is licensed under the CC BY-NC-SA 4.0, while the SSCBench KITTI-360 dataset is released under CC BY-NC-SA 3.0 license.

Terms of Use and License of SketchSem3D. The SketchSem3D dataset is licensed under CC BY-NC-SA 4.0.