Sat2City: 3D City Generation from A Single Satellite Image with Cascaded Latent Diffusion

Tongyan Hua[1] Lutao Jiang [1] Ying-Cong Chen [1,2] Wufan Zhao[1*]
[1]HKUST(GZ) [2]HKUST
{thua388, ljiang553}@connect.hkust-gz.edu.cn, yingcongchen@ust.hk, wufanzhao@hkust-gz.edu.cn
Project Page: https://ai4city-hkust.github.io/Sat2City/

Abstract

*Recent advancements in generative models have enabled 3D urban scene generation from satellite imagery, unlocking promising applications in gaming, digital twins, and beyond. However, most existing methods rely heavily on neural rendering techniques, which hinder their ability to produce detailed 3D structures on a broader scale, largely due to the inherent structural ambiguity derived from relatively limited 2D observations. To address this challenge, we propose **Sat2City**, a novel framework that synergizes the representational capacity of sparse voxel grids with latent diffusion models, tailored specifically for our novel 3D city dataset. Our approach is enabled by three key components: (1) A cascaded latent diffusion framework that progressively recovers 3D city structures from satellite imagery, (2) a Re-Hash operation at its Variational Autoencoder (VAE) bottleneck to compute multi-scale feature grids for stable appearance optimization, and (3) an inverse sampling strategy enabling implicit supervision for smooth appearance transitioning. To overcome the challenge of collecting real-world city-scale 3D models with high-quality geometry and appearance, we introduce a dataset of synthesized large-scale 3D cities paired with satellite-view height maps. Validated on this dataset, our framework generates detailed 3D structures from a single satellite image, achieving superior fidelity compared to existing city generation models.*

1. Introduction

Generating realistic 3D urban scenes [30, 31, 55, 56] has gained significant attention in recent years, driven by applications in gaming, urban planning, and digital twin systems. With advancements in neural rendering techniques, most prior works have focused on generating novel street-view images or videos [10, 27–29, 33, 34, 38, 44, 59, 64] within urban environments. These methods are restricted to rendering cities from a highly limited set of viewpoints and trajectories, and the generated images or videos often lead to poor results when explicitly reconstructing a 3D scene.

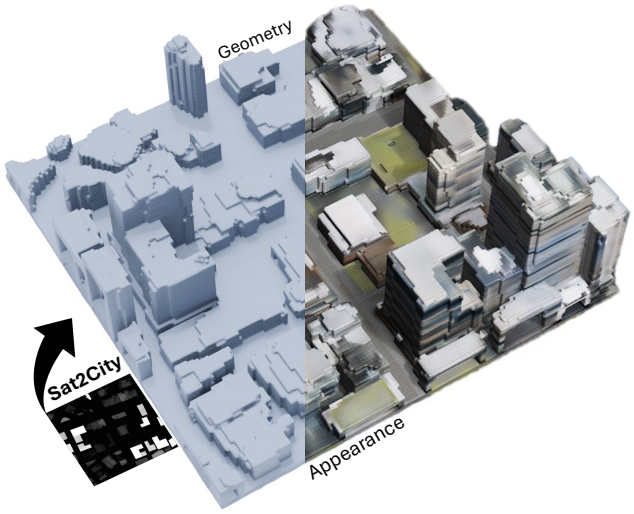


Figure 1. We present **Sat2City**, a novel framework for generating high-fidelity 3D city models with detailed geometry and appearance from a single satellite observation, in $\sim$ **1 minute**, using cascaded latent diffusion. This process is achieved even without the need for auxiliary inputs like a segmentation map.

Recent methods have evolved to integrate more reliable priors, such as segmentation and height maps from satellite imagery, enabling a more salient abstraction of urban environments and extending novel view synthesis to broader scales. Early on, InfiniCity [31] directly uses lifted voxels from satellite imagery as feature volumes of neural radiance fields (NeRFs), training a generative adversarial network (GAN) to transform ray-sampled features into rendered pixels. Building upon this framework, CityDreamer [55] and its Gaussian variants [56] achieve more realistic rendering effects by implementing feature-voxel partitioning through per-pixel semantic annotation. However, these 2D

*Corresponding author.

rendering-based generation approaches still suffer from significant 3D ambiguity due to the absence of direct supervision from 3D textured coordinates, leading to limited capability for reconstructing detailed 3D structures. Recently, Sat2Scene [30] pioneered the use of diffusion models for direct color generation on 3D point clouds. While this constitutes a conceptual advance, the method faces practical limitations: The color generation process relies on a densely allocated point cloud constructed around predefined geometry derived from satellite height maps. Despite achieving photorealistic appearance rendering, this process lacks the capability to refine the geometry and, more importantly, is computationally expensive to scale. Furthermore, its applicability is constrained by the scarcity of high-quality, textured point cloud data at urban scales, limiting its usage to single street-level scenarios.

To address these challenges, we introduce **Sat2City**, a novel generative framework that jointly synthesizes geometry and appearance. Our model learns directly from our high-quality 3D city dataset represented as colorized point clouds, ensuring a unified representation for structured 3D city generation. Recent advances in 3D generative modeling have demonstrated significant progress in scalability with XCube [39], an efficient framework that combines sparse voxel grids and latent diffusion models for large-scale outdoor scene synthesis. At its core, XCube leverages a compact VDB data structure, enabling fast querying and logarithmic memory scaling while preserving expressive power. Building on this foundation, we adopt XCube's sparse voxel grid as a unified neural representation for joint geometry and appearance encoding. In contrast to its closed-form successor, SCube [40], which employs Gaussian Splatting [24] with image-space texture priors, our framework directly encodes appearance as voxel color attributes. Rather than introducing excessive complexity, we leverage native sparse voxels to streamline the pipeline while ensuring consistent 3D geometry and appearance, free from multi-view reprojection artifacts.

We identify two crucial findings for capacitating the sparse latent grids to coherently encode the appearance. First, performing multi-level coarsening, dubbed Re-Hash, at the VAE bottleneck plays an important role not only in facilitating stable optimization, but also in providing global context for smooth appearance encoding. Second, supervising vertex color attribute learning implicitly through inverse sampling at the input point cloud level could enhance smooth visual transitions. In order to recover the geometric distribution of the original 3D data—onto which our appearance is assigned—from height maps that are heavily perturbed by noise, we additionally train a VAE with a densified bottleneck to encode geometry. By conditioning the sampling of this dense latent volume on elevated height maps, our cascaded latent diffusion model can progressively correct, refine, and colorize the geometry derived from noisy height maps through triple-level latent space conditioning, where each level is conditioned on the previous one.

	City-Scale	Appearance	Explicit 3D
XCube [39]	✗	✗	✓
SCube [40]	✗	✓	✓
CityDreamer [55]	✓	✓	✗
Sat2Scene [30]	✗	✓	✓
Ours	✓	✓	✓

Table 1. Comparison of 3D urban scene generation methods.

In summary, our primary contributions are as follows:

- We present the first 3D city generation framework to achieve city-scale appearance modeling with explicit geometry, producing high-fidelity and controllable urban environments from a single satellite image (Table 1).
- We propose three key innovations enabling joint appearance and geometry generation with sparse voxel grids and latent diffusion: (1) Re-Hash, (2) Inverse Sampling, and (3) a triplet cascaded latent diffusion framework.
- We introduce a new dataset of orthorectified satellite images paired with 3D colorized point clouds, facilitating training of our framework.

2. Related Works

3D Generation in Object-level. The recent 3D generation methods can be divided into two paradigms: per-scene optimization and direct generation. Per-scene optimization methods [1, 8, 9, 14, 22, 37, 66, 73] typically require several hours of training and are limited to specific scenes, making them less flexible. In contrast, we opt to follow the direct generation paradigm, which offers greater efficiency. The direct generation can be further categorized into three types, including feed-forward [18, 58], diffusion [39, 40, 68, 69], and auto-regressive [4–6, 46, 49, 52, 53]. Given that feed-forward methods are constrained by limited diversity and auto-regressive methods are not well-suited for large-scale scene generation, we focus on diffusion-based methods due to their merits of large-scale, diversity, and speed. In diffusion-based methods, two main representations are commonly used: latent sets [67–69] and sparse voxels [39, 40]. We choose the latter one as our scene representation for its inherent computational efficiency and proven superiority in handling large-scale scenes.

3D Generation in Scene-level. Two prominent strategies have emerged to specialize scene generation [16, 60, 65, 70] for city generation: asset retrieval and volumetric neural rendering. The first strategy focuses on agent-based procedural retrieval of pre-built 3D assets, such as those available in Blender, to enable the automated planning of 3D cities [11, 43, 63, 71] or more generalized scenes [32, 72]. However, this approach is inherently limited by the ex-

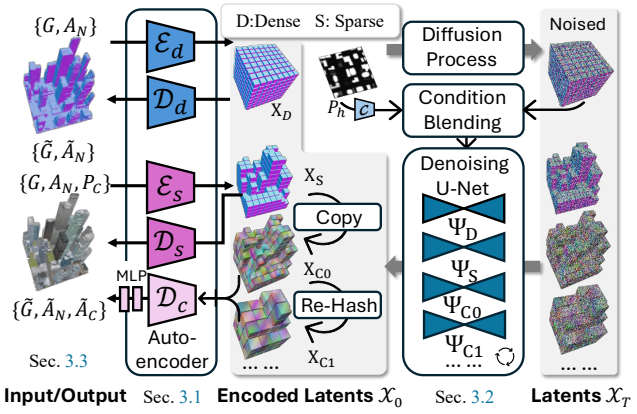


Figure 2. Sat2City Training Pipeline.

isting asset library, which makes it challenging to generate novel content. In contrast, neural rendering techniques focus on generating novel street-view images or videos [2, 7, 10, 16, 17, 27, 33, 60, 64, 65, 70]. Recent advancements have significantly improved cross-view consistency over long trajectories, particularly when leveraging large datasets [10]. More recent work has explored the use of salient urban scene representations, such as semantic volumes [27, 33, 64], as controllable priors for stable city-scale navigation. Yet, the inherent ambiguity of 3D structures inferred from 2D generative priors continues to constrain the robustness of these approaches when handling diverse viewpoints. Multi-view consistency has been significantly improved by separately modeling elements in urban environments with fine-grained categories [31, 55, 56], though 3D consistency remains suboptimal. Training directly on 3D data presents a promising opportunity; however, existing methods are still constrained to street-level generation and have not scaled to entire cities [30, 35, 40]. In contrast, our approach enables city-level generation by directly learning from large-scale 3D urban data, ensuring both geometric and appearance consistency.

3. Method

Our method is trained on 3D city data, represented as a colorized point cloud $P_C \in \mathbb{R}^{N\times 6}$, along with corresponding height maps, which are also elevated to a point cloud P_h (see Figure 3). For geometry encoding, the point clouds are voxelized into sparse voxel grids G, with each vertex assigned trilinearly splatted normal attributes A_N from the point cloud. For appearance encoding, the per-vertex color attribute $\tilde{A}_C$ is implicitly derived from the colorized point cloud during training via trilinear interpolation at P_C.

As depicted in Figure 2, our architecture integrates multiple VAEs with encoder-decoder pairs $(\mathcal{E}, \mathcal{D})$ and latent diffusion U-Nets Ψ through a two-stage training paradigm. The VAEs initially encode sparse voxel grids into compact latent representations $\mathcal{X}_0 = \{X_D, X_S, \{X_{Ck}\}_{k=0}^{n}\}$, where X_D and X_S respectively capture dense and sparse geometric features while $\{X_{Ck}\}_{k=0}^{n}$ encodes hierarchical appearance details through n progressively coarsen levels (Section 3.1). Following this encoding phase, the latent diffusion model undergoes optimization to iteratively denoise Gaussian-distributed features $\mathcal{X}_T$ across T timesteps, progressively recovering the target distribution $\mathcal{X}_0$ (Section 3.2). The framework is ultimately validated on our proposed dataset detailed in Section 3.3.

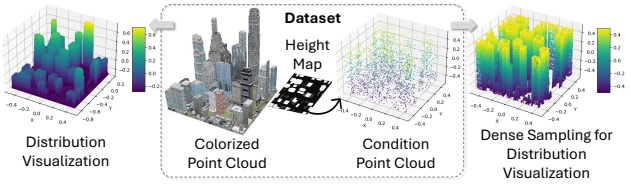


Figure 3. **Raw Input Data.** The colorized point cloud P_C and height map-derived conditional point cloud P_h are normalized for visualization, with P_h upsampled to enhance distribution clarity.

3.1. Triplet Bottleneck VAE

Our VAEs utilize sparse convolutional neural networks as encoders to downsample the input sparse voxel grid. It then employs a structure prediction backbone [20] as a decoder, which iteratively subdivides existing voxels and prunes excessive ones, ultimately upsampling back to the same resolution as the input [39]. As illustrated in Figure 4, instead of maintaining identical sparse latent grids for all VAE bottlenecks, we introduce two additional structures, namely the dense neck and the re-hash neck, to represent the overall occupancy and appearance field for downstream diffusion sampling, respectively. The primary motivation behind densification is to enable diffusion models to explicitly distinguish between occupied and unoccupied volumes, thereby eliminating falsely allocated volumes introduced by conditional inputs, which are highly perturbed by noise (Figure 3). Additionally, implementing densification in the bottleneck rather than in the input structure sustains an affordable computational cost for large-scale scene encoding.

Re-Hash Neck. Although dense volumetric representations are recommended for preserving detailed structures when modeling 3D shapes in various tasks [19, 21], multi-level encoding, which is essential for smooth appearance transitions [23, 51], imposes a substantial computational burden when applied to scene-level data encoding [74]. Therefore, we introduce Re-Hash, a hierarchical coarsening mechanism that progressively restructures the voxel representation through iterative resampling. Given an initial voxel grid X_{C0}, derived from sparse geometric feature grids via direct "copy" (Figure 2), we construct a multi-depth hierarchy in which each subsequent level is coarsened by a factor of two in voxel size (Figure 4). Formally, for each depth n, the

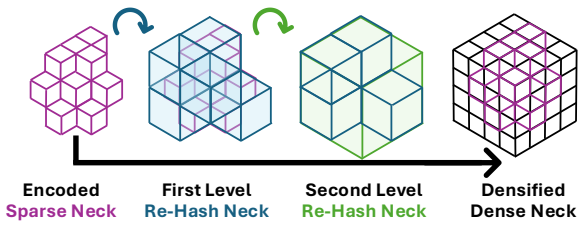


Figure 4. **Triplet Bottleneck Architecture for VAE.** All VAEs first downsample the input sparse voxel grid into an **Encoded Sparse Neck**, directly decoded by $\mathcal{D}_s$, while undergoing densification into a **Dense Neck** for $\mathcal{D}_d$ and re-hashing into a multi-level **Re-Hash Neck** for $\mathcal{D}_c$.

voxel resolution is updated as: $v_n = 2^n v_0, \quad o_n = \frac{v_n}{2}$, where v_n and o_n denote the voxel size and origin at depth n, respectively. At each level, the new sparse feature grid X_{Cn} is assigned with the features from X_{Cn-1} using trilinear interpolation, ensuring smooth transitions between resolutions. Specifically, given a set of 3D query positions from the restructured grid G_{Cn}, we sample latent features from the previous depth using trilinear interpolation (Tri):

$$X_{Cn} = \text{Tri}(G_{Cn}, X_{Cn-1}), \tag{1}$$

where X_{Cn} represents the encoded features at depth n. The final Re-Hash hierarchy preserves critical appearance details at multiple scales while facilitating a progressively refined latent space for downstream processing.

Dual-Stage Appearance Training. We train two independent VAEs for dense and sparse scenarios, respectively. For dense geometry, we employ a conventional VAE, consisting of an encoder and a decoder. Whereas in the sparse scenario, our architecture employs a unified encoder $\mathcal{E}_s$ to approximate the posterior distributions of the sparse geometric latent variable X_S and the multi-level appearance latent variable X_{Cn}. The model then utilizes two distinct decoders: the sparse decoder $\mathcal{D}_s$ reconstructs geometry and normal attributes via the conditional likelihood $p_{\mathcal{D}_s}(G, A_N|X_S)$, while the appearance decoder $\mathcal{D}_c$ processes each level of the re-hashed feature grids, concatenates them, and feeds the result into a small multilayer perceptron (MLP) to synthesize appearance, as demonstrated in Figure 2 and detailed in the following paragraph. Notably, the sparse latent X_S serves dual purposes: it provides structural guidance for progressive grid pruning in X_{Cn} during training, while also eliminating unnecessary voxel grids that were not suppressed during dense neck pruning. The training of the sparse VAE begins by exclusively optimizing $q_{\mathcal{E}_s}(X_S|G, A_N)$ and $p_{\mathcal{D}_s}(G, A_N|X_S)$ for geometric fidelity until epoch E. After this stage, X_S initializes the finest-level appearance latent volume X_{C0}, which is subsequently integrated with re-hashed coarser grids and processed through $\mathcal{D}_c$ to constrain the appearance latent X_{Cn}.

Inverse Sampling. Learning on 3D datasets provides generative models with significant advantages in terms of speed and spatial consistency. Moreover, for appearance learning, it offers direct supervision, reducing ambiguity. However, directly performing per-vertex color attribute learning is challenging. As illustrated in Figure 5, voxelizing colorized point clouds presents challenges with both naive strategies. (1) Direct assignment from the nearest point results in a lack of smoothness. (2) Trilinear splatting causes color blending conflicts due to overlapping contributions from multiple points. To address these issues, this work constrains A_N only at the vertex level while implicitly guiding the learning of per-vertex color attributes A_C through inverse sampling, a strategy inspired by the sampling process in NeRF's voxel grid variants [36, 48, 74]. During the training stage, for each level of the appearance latent hierarchy X_{Cn}, it is separately decoded through $\mathcal{D}_c$ into per-vertex color features, which are then trilinearly sampled on the colorized point cloud P_C. The interpolated features from all levels are concatenated and processed by an MLP, resulting in estimated point cloud color $\tilde{P}_C$:

$$\tilde{P}_C = \text{MLP}\{\oplus_{k=0}^{n} Tri\,(P_C, \mathcal{D}_c(X_{Cn}))\}, \tag{2}$$

where $\oplus$ denotes the concatenation operation applied to the decoder outputs. At inference time, given the predicted city geometry in the form of grid vertices $\tilde{G}$, we can infer its color in the same manner:

$$\tilde{A}_C = \text{MLP}\{\oplus_{k=0}^{n} Tri\left(\tilde{G}, \mathcal{D}_c(X_{Cn})\right)\}. \tag{3}$$

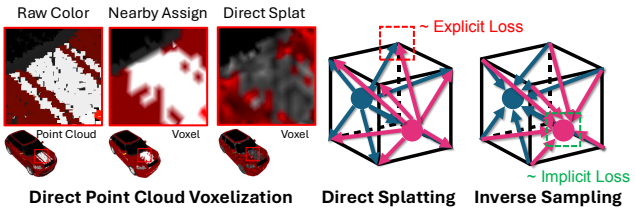


Figure 5. **Direct Point Cloud Voxelization and Inverse Sampling Process.** Examples of challenging scenarios in per-vertex color attribute learning, illustrated using a dense point cloud generated from the ShapeNet dataset [3] for the ease of demonstration.

3.2. Conditional Cascaded 3D Latent Diffusion

Diffusion models are probabilistic frameworks designed to model target data distributions, such as the latent space $\mathcal{X}_0 \sim q(\mathcal{X}_0)$ in VAEs (*i.e.* latent diffusion model), by progressively denoising a randomly sampled variable $\mathcal{X}_T \sim \mathcal{N}(0, I)$, initially drawn from a Gaussian distribution [41]. The process is based on learning the reverse denoising procedure of a fixed Markov Chain with T steps. Training starts with the diffusion process, which progressively adds noise to $\mathcal{X}_0$ over T steps to form $\mathcal{X}_T$. The model then iteratively denoises $\mathcal{X}_T$ using U-Nets, gradually recovering the original latent representation $\mathcal{X}_0$ (Figure 2). However, single-stage latent diffusion is proven to be inadequate when addressing expansive 3D landscapes [39], especially when the appearance is also required [35, 40]. Our

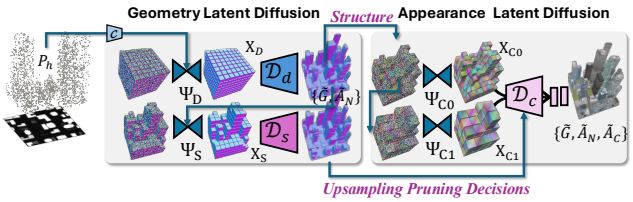


Figure 6. Inference pipeline of Sat2City.

model therefore adopts a sequentially conditioned diffusion pipeline, where later stages incorporate contents from earlier stages, as illustrated in Figure 6. Our generation process comprises three consecutive sampling stages for dense geometry latent X_D, sparse geometry latent X_S, and multi-level sparse appearance latent X_{Cn}, using DDIM [47]. The process begins by formulating the joint distribution factorization with respect to the dense grid and its corresponding latent representations as follows:

$$p(X_D, G, A_N) = p_{\mathcal{D}_d}(G, A_N|X_D)p_{\Psi_D}(X_D|c(P_h)). \tag{4}$$

Here, the initial dense geometry latent X_D encodes the overall spatial layout based on the encoded conditional height field inputs P_h. During inference, the denoised dense latent is decoded by $\mathcal{D}_d$ to generate the sparse grid output $\{G, A_N\}$, which facilitates sparse latent diffusion. This intermediate representation serves as a bridge between geometry and appearance, leveraging $\{G, A_N\}$ to fit a sparse latent volume for finer surface reconstruction. Crucially, the sparse latent grid decoder records voxel pruning decisions, *struct*, at each upsampling step, guiding structured pruning during appearance decoding, since its re-hashed latent grid does not explicitly encode geometric distributions.

$$p(X_S, \textit{struct}) = p_{\mathcal{D}_s}(G, A_N|X_S)p_{\Psi_S}(X_S|G, A_N). \tag{5}$$

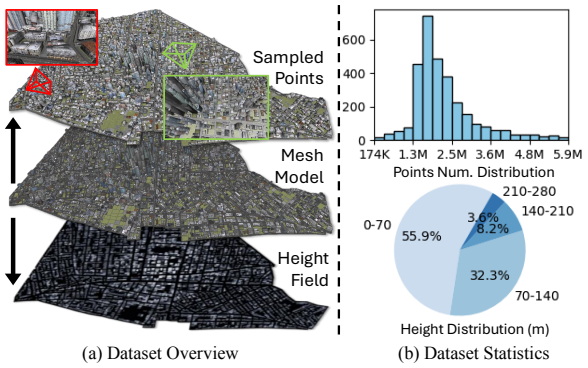
struct are subsequently leveraged to guide the appearance upsampling process, ensuring that the finest-level appearance feature grid maintains a consistent geometric structure with sparse latents at each upsampling step of the appearance decoder:

$$p(G, A_N, A_C) = p_{\mathcal{D}_c}(G, A_N, A_C|X_C)\cdot \prod_{n=0}^{N} p_{\Psi_{Cn}}(X_{Cn}|\textit{struct}). \tag{6}$$

Specifically, the finest-level appearance feature grid X_{C0} undergoes a *re-hashing* process, generating multiple coarser levels. During each upsampling step, these coarser levels (*e.g.*, X_{C1} in Figure 6) are iteratively re-fitted by adapting to the pruned voxel structure defined by X_{C0}. Refer to *Supplementary Material* for more details.

3.3. Sat2City Dataset

The 3D city dataset is created by artists in Blender [15] as mesh-based models. To generate the point cloud, we randomly sample 100 million points from the mesh surfaces using CloudCompare [50]. The height field is then simulated in Blender by mapping the texture coordinates of the elevation axis (in this case, the y-axis) into grayscale, with the lowest and highest elevations being linearly assigned to 0 and 255, respectively. We then position a vertically aligned orthographic camera to render a high-resolution synthetic height map of the entire city in a single shot. The output image, at a resolution of 2268×3423 pixels, covers a land area of $2090 \times 3449.4\ m^2$, achieving a ground sampling distance of $0.93\ m^2$ per pixel. To enhance realism and align our simulation with real-world height maps—commonly referred to as digital surface models (DSMs) in remote sensing and photogrammetry—the height field rendering incorporates ambient lighting variations. Additionally, noise is introduced by applying contrast scaling to the rendered height map (Figure 3).



(a) Dataset Overview (b) Dataset Statistics

Figure 7. Overview of the Sat2City dataset.

Figure 7 presents a layered visualization of the raw dataset alongside its statistical insights. In Figure 7 (a), the mesh model represents the original 3D city design in a mesh form, with the corresponding height field displayed in the bottom layer and the sampled point cloud in the top layer. The raw sampled point cloud is first spatially aligned with the height field images, ensuring a one-to-one correspondence. The aligned data is then uniformly cropped into 300×300 pixel segments, preserving the structural consistency between the point cloud and height field throughout the dataset. The entire dataset is randomly partitioned into 3110 instances, with 10% reserved for testing and validation, ensuring no data leakage during training. An example of a cropped height map and its corresponding point cloud is shown in Figure 3. These cropped point cloud instances contain between 174,000 and 5.9 million points, as visualized in Figure 7 (b). Please refer to *Supplementary Material* for more details.

4. Experiments

4.1. Evaluation Protocols

Geometric Metrics. To assess the quality of the generated 3D city models, we extract their mesh representa-

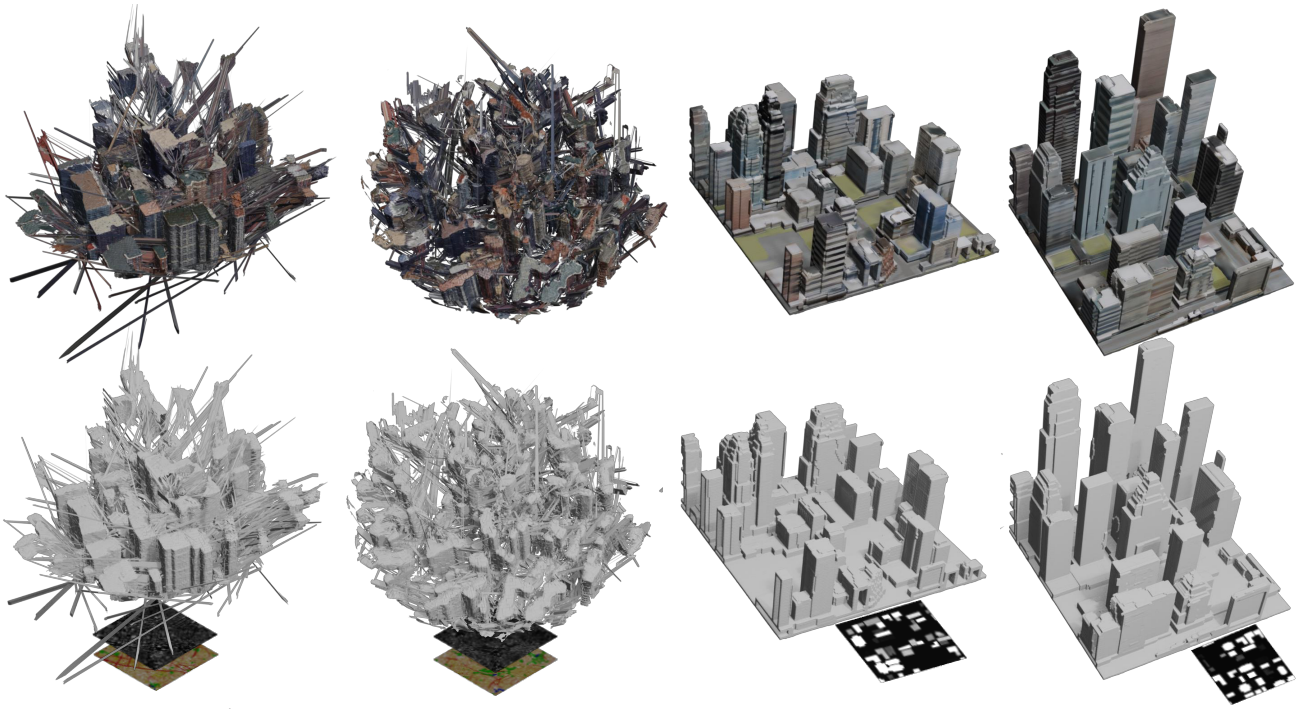


Figure 8. **Qualitative Comparison with CityDreamer [55]**. CityDreamer requires both segmentation and height maps, while Sat2City relies solely on height maps. CityDreamer generates 2D city renderings, reconstructed into meshes via MASt3R [26].

tions and compute three widely used metrics for 3D generation [21, 39, 46, 54, 61]: Coverage score (COV) and Minimum Matching Distance (MMD). COV quantifies the proportion of points in the reference point cloud that have at least one corresponding match in the generated point cloud, while MMD computes the average distance by identifying the nearest neighbor in the generated set for each point in the reference point cloud. These metrics are evaluated using Chamfer Distance (CD) and Earth Mover's Distance (EMD), which serve as metrics to quantify the geometric similarity between the generated and reference point clouds. 10,000 points are randomly sampled from both the generated mesh surface and the reference point cloud for metric computation. Please refer to [61] for more details.

Appearance Metric. A user study was conducted with 60 participants who evaluated the scene generation results based on two criteria: Perceptual Quality (PQ) and Structural Completeness (SC), using a 10-point scale. Evaluations were performed in both appearance-based mode (TPQ and TSC) and geometry-only mode (GPQ and GSC). In the appearance-based mode, participants assessed a textured mesh, while in the geometry-only mode, the appearance was replaced with a monochromatic material to emphasize the geometric structure [54]. Please refer to the *Supplementary Material* for the survey design.

	MMD↓		COV(%,↑)	
	CD	EMD	CD	EMD
NFD (unconditional) [45]	0.0445	0.2363	22.66	29.66
BlockFusion (unconditional) [54]	0.0326	0.1865	50.49	55.66
Ours (conditional)	**0.0165**	**0.1157**	**100.00**	**60.00**

Table 2. **Geometric Quality Comparison**. Results for NFD and BlockFusion are obtained through unconditional generation methods [54], while our Sat2City framework operates as a conditional generation pipeline.

4.2. Comparison

Baselines. To the best of our knowledge, this work presents the first 3D city generation framework trained directly with 3D supervision for both geometry and appearance encoding. In the absence of directly comparable methods, we adopt a two-fold evaluation strategy. For geometry generation quality assessment, we compare against established 3D scene generation pipelines, including BlockFusion [54] and its reproduction of NFD [45]. For visual quality evaluation, we benchmark against rendering-based urban generation baselines, CityDreamer [55] and Sat2Scene [30]. Notably, Sat2Scene serves as the most relevant baseline, as it is the only architecture compatible with our dataset configuration. Retraining all baselines is challenging, as BlockFusion (watertight meshes) and CityDreamer (aerial images) require inputs incompatible with Sat2City (point clouds). Therefore, to ensure a fair comparison, we faithfully re-

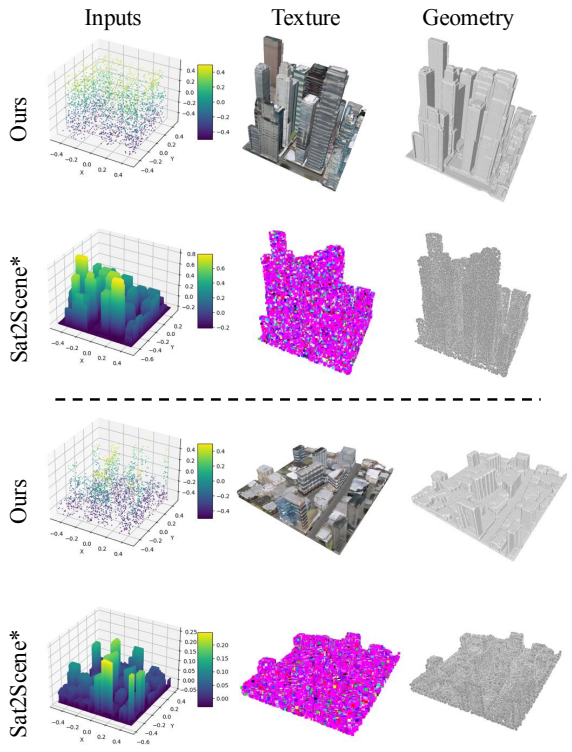


Figure 9. **Qualitative Comparison with Sat2Scene [30] Trained on Our Dataset**. Since Sat2Scene lacks geometry generation, we provide ground truth point clouds as input, whereas Sat2City generates geometry from noised height maps.

implement Sat2Scene on our dataset using its official code.

Qualitative Comparison. Figure 8 presents qualitative results comparing our method to CityDreamer [55], a state-of-the-art (SOTA) 3D city generation pipeline. To ensure a fair comparison, we reproduce CityDreamer by rendering 24 images per scene (its default setting in the official implementation) and applying a SOTA reconstruction model [26] to extract comparable meshes. CityDreamer exhibits geometric inconsistencies due to its reliance on image-based rendering, whereas our Sat2City framework achieves significantly improved spatial coherence without compromising appearance quality. Additionally, CityDreamer struggles with prompt adherence, often failing to generate cities that align with the conditional height field and segmentation map. Although CityDreamer can cover a larger area than our method, it remains restricted to limited viewpoints, constraining explicit 3D reconstruction. Our framework addresses these limitations by directly integrating control conditions into 3D generative priors, enabling precise regulation of building quantities and spatial arrangements that strictly adhere to textual specifications, even without requiring auxiliary inputs such as segmentation maps. Please refer to *Supplementary Material* for more visual results.

Initial efforts to adapt diffusion models for urban modeling with 3D sparse representations were pioneered by Sat2Scene. While effective at encoding street-level appearance, its representation fails to scale to entire cities. As shown in Figure 9, even when provided with ground truth point clouds, Sat2Scene struggles to reconstruct meaningful appearances when trained on our dataset. In contrast, our framework generates high-fidelity urban structures using only noised height-field prompts. We attribute the scalability limitations of Sat2Scene to its dependence on high point cloud densities: While successful generation of Sat2Scene typically requires approximately 400 points per square meter, our dataset provides only around 14 points per square meter. This fundamental constraint underscores the advantage of our 3D latent sparse representation.

	TPQ↑	TSC↑	GPQ↑	GSC↑
Sat2Scene (2D) [30]	6.17	5.90	-	-
Sat2Scene (3D) [30]	5.57	5.47	3.83	4.05
Sat2Scene*	3.18	3.30	3.03	3.02
CityDreamer (2D) [55]	6.40	6.63	-	-
CityDreamer (3D)*	4.48	4.48	3.60	3.38
Ours	**7.35**	**8.03**	**6.27**	**7.02**

Table 3. **Appearance Quality Comparison**. Asterisk (*) denotes methods adapted for fair comparison: Sat2Scene* is retrained on our proposed dataset, and CityDreamer (3D)* leverages MASt3R [26] for 3D mesh reconstruction of rendered video sequences. Noted that the Render Quality is directly retrieved from the original records.

Quantitative Comparison. Table 2 demonstrates that our method consistently outperforms well-established scene-level baselines across all geometric metrics. Sat2City achieves a 98.1% and 7.8% improvement in COV (CD) and COV (EMD) over previous SOTA, indicating more stable generation quality with a lower probability of mode collapse. Additionally, our method reduces MMD (CD) by 49.4% and MMD (EMD) by 37.9% compared to BlockFusion, suggesting superior geometric fidelity and a closer alignment with the reference distribution. It should be noted that the comparison here is between the complete BlockFusion solution and the full Sat2City pipeline, each encompassing both model architecture and associated dataset.

For user perception, Table 3 shows that Sat2City consistently achieves the highest scores in both textured and geometry-only evaluations. Notably, in metrics evaluating perceived textural quality (TPQ and TSC), our direct 3D generation (7.35 TPQ, 8.03 TSC) surpasses the 2D image rendering-based results of Sat2Scene(2D) (6.17 TPQ, 5.90 TSC) and CityDreamer(2D) (6.40 TPQ, 6.63 TSC). These findings highlight that in large-scale city generation, learning appearance directly from 3D data can also achieve higher user approval than 2D rendering-based methods.

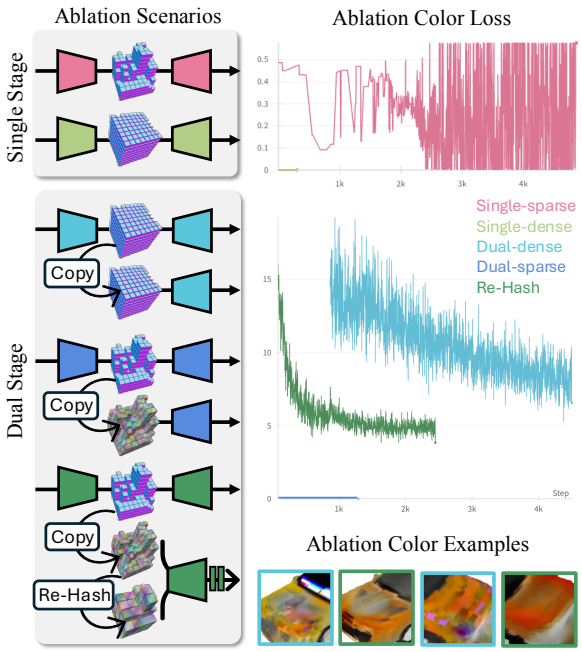


Figure 10. Ablation on Re-Hash operation. Note that the Single-dense and Dual-sparse variants produce zero values consistently throughout the training process.

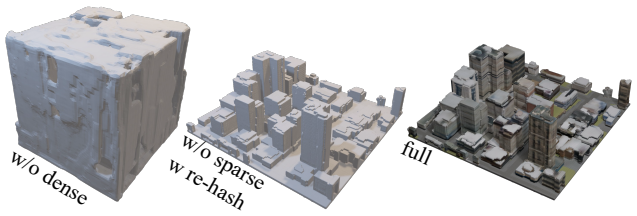


Figure 11. Ablation on cascaded latent diffusion.

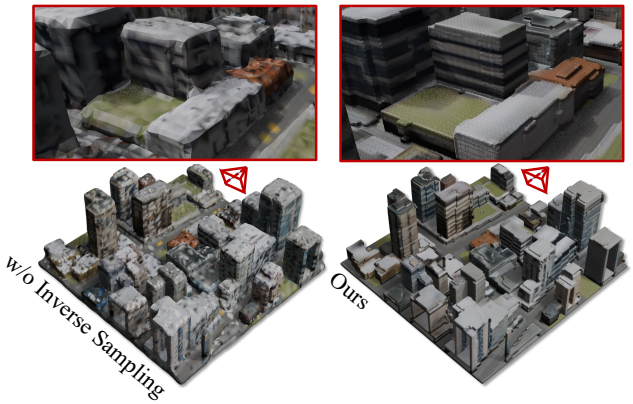


Figure 12. Ablation on inverse sampling.

4.3. Ablations

Bottleneck Design. The primary objective of our Re-Hash neck and its dual-stage training is to ensure stable appearance optimization. To validate its effectiveness, we present the color training loss patterns for various bottleneck and training designs in Figure 10 as evidence of our design's superiority. Our ablation reveals three key findings: First, encoding both appearance and geometric features into a unified continuous latent representation results in significant gradient direction conflicts, especially for sparse structural configurations. Second, when employing separate latent spaces for color representation through dual-stage training, these gradient conflicts are substantially alleviated for both sparse and dense structures, with the latter beginning to exhibit meaningful gradient descent patterns. Notably, the Re-Hash structure demonstrates faster and more stable convergence compared to the dual-dense design, which shows greater loss fluctuation and requires more iterations to stabilize. Furthermore, the color rendering results indicate that the dual-dense configuration is prone to generating artifacts in appearance. This deficiency stems from its lack of coarse-grained contextual information, which our Re-Hash structure explicitly incorporates to enable smooth spatial transitions in color representation.

Inverse-Sampling. Figure 12 presents qualitative ablation study results that underscore the importance of our inverse-sampling mechanism. In conventional direct color splatting approaches, where vertex color attributes lack implicit supervision from enclosed voxel points, we observe significant rendering artifacts. Our proposed inverse-sampling enables the color attributes assigned to mesh vertices to be continuously constrained by the spatial distribution of points within their associated voxels, which is crucial for maintaining appearance consistency.

Diffusion Model Hierarchy. We ablate each level of our cascaded latent diffusion framework in Figure 11. Results indicate that relying solely on sparse structures fails to capture unoccupied regions, leading to chaotic generations (w/o dense scenario) under direct sparse latent diffusion. Nevertheless, the sparse latent grid encoder is indispensable for conveying structured pruning decisions essential for appearance decoding (w/o sparse & w/ re-hash scenario).

5. Conclusion

We present Sat2City, a novel framework that integrates sparse voxel grids with latent diffusion models for large-scale 3D city generation. By introducing a cascaded latent diffusion framework with a VAE-based Re-Hash operation and inverse sampling for effectively and efficiently representing appearance. Sat2City, trained on our novel dataset of satellite-view simulations and hand-crafted 3D cities, generates detailed 3D urban structures from a single satellite image, surpassing existing models in fidelity. For further discussion of limitations and the potential applicability to real-world satellite–3D Tiles pairs, please refer to the *Supplementary Material*.

Acknowledgement. This paper is supported by the Young

Scientists Fund of the National Natural Science Foundation of China under Grant No. 42401567 and the Open Fund of the Technology Innovation Center for 3D Real Scene Construction and Urban Refined Governance, Ministry of Natural Resources under Grant No. 2024PF-1.

References

[1] Haotian Bai, Yuanhuiyi Lyu, Lutao Jiang, Sijia Li, Haonan Lu, Xiaodong Lin, and Lin Wang. Componerf: Text-guided multi-object compositional nerf with editable 3d scene layout. *arXiv preprint arXiv:2303.13843*, 2023. 2

[2] Lucy Chai, Richard Tucker, Zhengqi Li, Phillip Isola, and Noah Snavely. Persistent nature: A generative model of unbounded 3d worlds. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 20863–20874, 2023. 3

[3] Angel X Chang, Thomas Funkhouser, Leonidas Guibas, Pat Hanrahan, Qixing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, et al. Shapenet: An information-rich 3d model repository. *arXiv preprint arXiv:1512.03012*, 2015. 4

[4] Sijin Chen, Xin Chen, Anqi Pang, Xianfang Zeng, Wei Cheng, Yijun Fu, Fukun Yin, Yanru Wang, Zhibin Wang, Chi Zhang, et al. Meshxl: Neural coordinate field for generative 3d foundation models. *arXiv preprint arXiv:2405.20853*, 2024. 2

[5] Yiwen Chen, Tong He, Di Huang, Weicai Ye, Sijin Chen, Jiaxiang Tang, Xin Chen, Zhongang Cai, Lei Yang, Gang Yu, et al. Meshanything: Artist-created mesh generation with autoregressive transformers. *arXiv preprint arXiv:2406.10163*, 2024.

[6] Yiwen Chen, Yikai Wang, Yihao Luo, Zhengyi Wang, Zilong Chen, Jun Zhu, Chi Zhang, and Guosheng Lin. Meshanything v2: Artist-created mesh generation with adjacent mesh tokenization. *arXiv preprint arXiv:2408.02555*, 2024. 2

[7] Zhaoxi Chen, Guangcong Wang, and Ziwei Liu. Scenedreamer: Unbounded 3d scene generation from 2d image collections. *IEEE transactions on pattern analysis and machine intelligence*, 2023. 3

[8] Xinhua Cheng, Tianyu Yang, Jianan Wang, Yu Li, Lei Zhang, Jian Zhang, and Li Yuan. Progressive3d: Progressively local editing for text-to-3d content creation with complex semantic prompts. *arXiv preprint arXiv:2310.11784*, 2023. 2

[9] Dana Cohen-Bar, Elad Richardson, Gal Metzer, Raja Giryes, and Daniel Cohen-Or. Set-the-scene: Global-local training for generating controllable nerf scenes. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2920–2929, 2023. 2

[10] Boyang Deng, Richard Tucker, Zhengqi Li, Leonidas Guibas, Noah Snavely, and Gordon Wetzstein. Streetscapes: Large-scale consistent street view generation using autoregressive video diffusion. In *ACM SIGGRAPH 2024 Conference Papers*, pages 1–11, 2024. 1, 3

[11] Jie Deng, Wenhao Chai, Junsheng Huang, Zhonghan Zhao, Qixuan Huang, Mingyan Gao, Jianshu Guo, Shengyu Hao, Wenhao Hu, Jenq-Neng Hwang, et al. Citycraft: A real crafter for 3d city generation. *arXiv preprint arXiv:2406.04983*, 2024. 2

[12] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in neural information processing systems*, 34:8780–8794, 2021. 1, 2

[13] Paul Engstler, Aleksandar Shtedritski, Iro Laina, Christian Rupprecht, and Andrea Vedaldi. Syncity: Training-free generation of 3d worlds. *arXiv:2503.16420*, 2025. 3

[14] Dave Epstein, Ben Poole, Ben Mildenhall, Alexei A Efros, and Aleksander Holynski. Disentangled 3d scene generation with layout learning. *arXiv preprint arXiv:2402.16936*, 2024. 2

[15] Blender Foundation. Blender - a 3d modeling and animation software, 2025. Version 4.2, accessed: 2025-02-19. 5

[16] Rafail Fridman, Amit Abecasis, Yoni Kasten, and Tali Dekel. Scenescape: Text-driven consistent scene generation. *Advances in Neural Information Processing Systems*, 36, 2024. 2, 3

[17] Zekun Hao, Arun Mallya, Serge Belongie, and Ming-Yu Liu. Gancraft: Unsupervised 3d neural rendering of minecraft worlds. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14072–14082, 2021. 3

[18] Yicong Hong, Kai Zhang, Jiuxiang Gu, Sai Bi, Yang Zhou, Difan Liu, Feng Liu, Kalyan Sunkavalli, Trung Bui, and Hao Tan. Lrm: Large reconstruction model for single image to 3d. *arXiv preprint arXiv:2311.04400*, 2023. 2

[19] Tongyan Hua and Lin Wang. Benchmarking implicit neural representation and geometric rendering in real-time rgb-d slam. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21346–21356, 2024. 3

[20] Jiahui Huang, Zan Gojcic, Matan Atzmon, Or Litany, Sanja Fidler, and Francis Williams. Neural kernel surface reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4369–4379, 2023. 3, 1

[21] Ka-Hei Hui, Ruihui Li, Jingyu Hu, and Chi-Wing Fu. Neural wavelet-domain diffusion for 3d shape generation. In *SIGGRAPH Asia 2022 Conference Papers*, pages 1–9, 2022. 3, 6

[22] Lutao Jiang, Hangyu Li, and Lin Wang. A general framework to boost 3d gs initialization for text-to-3d generation by lexical richness. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pages 6803–6812, 2024. 2

[23] Mohammad Mahdi Johari, Camilla Carta, and François Fleuret. Eslam: Efficient dense slam system based on hybrid representation of signed distance fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17408–17419, 2023. 3

[24] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering. *ACM Trans. Graph.*, 42(4):139–1, 2023. 2

[25] Han-Hung Lee, Qinghong Han, and Angel X Chang. Nuiscene: Exploring efficient generation of unbounded outdoor scenes. *arXiv:2503.16375*, 2025. 3

[26] Vincent Leroy, Yohann Cabon, and Jérôme Revaud. Grounding image matching in 3d with mast3r. In *European Conference on Computer Vision*, pages 71–91. Springer, 2024. 6, 7

[27] Leheng Li, Weichao Qiu, Yingjie Cai, Xu Yan, Qing Lian, Bingbing Liu, and Ying-Cong Chen. Syntheocc: Synthesize geometric-controlled street view images through 3d semantic mpis. *arXiv preprint arXiv:2410.00337*, 2024. 1, 3

[28] Weijia Li, Jun He, Junyan Ye, Huaping Zhong, Zhimeng Zheng, Zilong Huang, Dahua Lin, and Conghui He. Crossviewdiff: A cross-view diffusion model for satellite-to-street view synthesis. *arXiv preprint arXiv:2408.14765*, 2024.

[29] Zuoyue Li, Zhenqiang Li, Zhaopeng Cui, Rongjun Qin, Marc Pollefeys, and Martin R Oswald. Sat2vid: Street-view panoramic video synthesis from a single satellite image. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 12436–12445, 2021. 1

[30] Zuoyue Li, Zhenqiang Li, Zhaopeng Cui, Marc Pollefeys, and Martin R Oswald. Sat2scene: 3d urban scene generation from satellite images with diffusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7141–7150, 2024. 1, 2, 3, 6, 7

[31] Chieh Hubert Lin, Hsin-Ying Lee, Willi Menapace, Menglei Chai, Aliaksandr Siarohin, Ming-Hsuan Yang, and Sergey Tulyakov. Infinicity: Infinite-scale city synthesis. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 22808–22818, 2023. 1, 3

[32] Xinhang Liu, Chi-Keung Tang, and Yu-Wing Tai. Worldcraft: Photo-realistic 3d world creation and customization via llm agents. *arXiv preprint arXiv:2502.15601*, 2025. 2

[33] Fan Lu, Kwan-Yee Lin, Yan Xu, Hongsheng Li, Guang Chen, and Changjun Jiang. Urban architect: Steerable 3d urban scene generation with layout prior. *arXiv preprint arXiv:2404.06780*, 2024. 1, 3

[34] Xiaohu Lu, Zuoyue Li, Zhaopeng Cui, Martin R Oswald, Marc Pollefeys, and Rongjun Qin. Geometry-aware satellite-to-ground image synthesis for urban areas. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 859–867, 2020. 1

[35] Yifan Lu, Xuanchi Ren, Jiawei Yang, Tianchang Shen, Zhangjie Wu, Jun Gao, Yue Wang, Siheng Chen, Mike Chen, Sanja Fidler, et al. Infinicube: Unbounded and controllable dynamic 3d driving scene generation with world-guided video models. *arXiv preprint arXiv:2412.03934*, 2024. 3, 4

[36] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. *Communications of the ACM*, 65(1):99–106, 2021. 4

[37] Ben Poole, Ajay Jain, Jonathan T Barron, and Ben Mildenhall. Dreamfusion: Text-to-3d using 2d diffusion. *arXiv preprint arXiv:2209.14988*, 2022. 2

[38] Ming Qian, Jincheng Xiong, Gui-Song Xia, and Nan Xue. Sat2density: Faithful density learning from satellite-ground image pairs. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3683–3692, 2023. 1

[39] 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. 2, 3, 4, 6, 1

[40] Xuanchi Ren, Yifan Lu, Hanxue Liang, Zhangjie Wu, Huan Ling, Mike Chen, Sanja Fidler, Francis Williams, and Jiahui Huang. Scube: Instant large-scale scene reconstruction using voxsplats. *arXiv preprint arXiv:2410.20030*, 2024. 2, 3, 4

[41] 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. 4

[42] Tim Salimans and Jonathan Ho. Progressive distillation for fast sampling of diffusion models. *arXiv preprint arXiv:2202.00512*, 2022. 1

[43] Yu Shang, Jiansheng Chen, Hangyu Fan, Jingtao Ding, Jie Feng, and Yong Li. Urbanworld: An urban world model for 3d city generation. *arXiv preprint arXiv:2407.11965*, 2024. 2

[44] Yujiao Shi, Dylan Campbell, Xin Yu, and Hongdong Li. Geometry-guided street-view panorama synthesis from satellite imagery. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(12):10009–10022, 2022. 1

[45] J Ryan Shue, Eric Ryan Chan, Ryan Po, Zachary Ankner, Jiajun Wu, and Gordon Wetzstein. 3d neural field generation using triplane diffusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20875–20886, 2023. 6

[46] Yawar Siddiqui, Antonio Alliegro, Alexey Artemov, Tatiana Tommasi, Daniele Sirigatti, Vladislav Rosov, Angela Dai, and Matthias Nießner. Meshgpt: Generating triangle meshes with decoder-only transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19615–19625, 2024. 2, 6

[47] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020. 5

[48] Cheng Sun, Min Sun, and Hwann-Tzong Chen. Direct voxel grid optimization: Super-fast convergence for radiance fields reconstruction. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5459–5469, 2022. 4

[49] Jiaxiang Tang, Zhaoshuo Li, Zekun Hao, Xian Liu, Gang Zeng, Ming-Yu Liu, and Qinsheng Zhang. Edgerunner: Auto-regressive auto-encoder for artistic mesh generation. *arXiv preprint arXiv:2409.18114*, 2024. 2

[50] CloudCompare Development Team. Cloudcompare (version 2.12.4) [gpl software], 2025. Retrieved on 2025-02-19. 5

[51] Hengyi Wang, Jingwen Wang, and Lourdes Agapito. Co-slam: Joint coordinate and sparse parametric encodings for neural real-time slam. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13293–13302, 2023. 3

[52] Zhengyi Wang, Jonathan Lorraine, Yikai Wang, Hang Su, Jun Zhu, Sanja Fidler, and Xiaohui Zeng. Llama-mesh:

Unifying 3d mesh generation with language models. *arXiv preprint arXiv:2411.09595*, 2024. 2

[53] Haohan Weng, Yikai Wang, Tong Zhang, CL Chen, and Jun Zhu. Pivotmesh: Generic 3d mesh generation via pivot vertices guidance. *arXiv preprint arXiv:2405.16890*, 2024. 2

[54] Zhennan Wu, Yang Li, Han Yan, Taizhang Shang, Weixuan Sun, Senbo Wang, Ruikai Cui, Weizhe Liu, Hiroyuki Sato, Hongdong Li, et al. Blockfusion: Expandable 3d scene generation using latent tri-plane extrapolation. *ACM Transactions on Graphics (TOG)*, 43(4):1–17, 2024. 6

[55] 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. 1, 2, 3, 6, 7

[56] Haozhe Xie, Zhaoxi Chen, Fangzhou Hong, and Ziwei Liu. Gaussiancity: Generative gaussian splatting for unbounded 3d city generation. *arXiv preprint arXiv:2406.06526*, 2024. 1, 3

[57] Zhitong Xiong, Sining Chen, Yi Wang, Lichao Mou, and Xiao Xiang Zhu. Gamus: A geometry-aware multi-modal semantic segmentation benchmark for remote sensing data. *arXiv preprint arXiv:2305.14914*, 2023. 2

[58] Jiale Xu, Weihao Cheng, Yiming Gao, Xintao Wang, Shenghua Gao, and Ying Shan. Instantmesh: Efficient 3d mesh generation from a single image with sparse-view large reconstruction models. *arXiv preprint arXiv:2404.07191*, 2024. 2

[59] Ningli Xu and Rongjun Qin. Geospecific view generation – geometry-context aware high-resolution ground view inference from satellite views, 2024. 1

[60] Yongzhi Xu, Yonhon Ng, Yifu Wang, Inkyu Sa, Yunfei Duan, Yang Li, Pan Ji, and Hongdong Li. Sketch2scene: Automatic generation of interactive 3d game scenes from user's casual sketches. *arXiv preprint arXiv:2408.04567*, 2024. 2, 3

[61] 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. 6

[62] Lihe Yang, Bingyi Kang, Zilong Huang, Xiaogang Xu, Jiashi Feng, and Hengshuang Zhao. Depth anything: Unleashing the power of large-scale unlabeled data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10371–10381, 2024. 2

[63] Shuqin Yang, Haopu Yuan, Tianqi Wang, Rui Zhong, Chenggang Song, Ying Fu, Wenyi Ge, and Xia Yuan. Procedural generation of 3d scenes for urban landscape based on remote sensing images. In *2024 IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, pages 1–7. IEEE, 2024. 2

[64] Yuanbo Yang, Yifei Yang, Hanlei Guo, Rong Xiong, Yue Wang, and Yiyi Liao. Urbangiraffe: Representing urban scenes as compositional generative neural feature fields. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9199–9210, 2023. 1, 3

[65] Yiying Yang, Fukun Yin, Jiayuan Fan, Xin Chen, Wanzhang Li, and Gang Yu. Scene123: One prompt to 3d scene generation via video-assisted and consistency-enhanced mae. *arXiv preprint arXiv:2408.05477*, 2024. 2, 3

[66] Xuening Yuan, Hongyu Yang, Yueming Zhao, and Di Huang. Dreamscape: 3d scene creation via gaussian splatting joint correlation modeling. *arXiv preprint arXiv:2404.09227*, 2024. 2

[67] Biao Zhang and Peter Wonka. Lagem: A large geometry model for 3d representation learning and diffusion. *arXiv preprint arXiv:2410.01295*, 2024. 2

[68] Biao Zhang, Jiapeng Tang, Matthias Niessner, and Peter Wonka. 3dshape2vecset: A 3d shape representation for neural fields and generative diffusion models. *ACM Transactions on Graphics (TOG)*, 42(4):1–16, 2023. 2

[69] Longwen Zhang, Ziyu Wang, Qixuan Zhang, Qiwei Qiu, Anqi Pang, Haoran Jiang, Wei Yang, Lan Xu, and Jingyi Yu. Clay: A controllable large-scale generative model for creating high-quality 3d assets. *ACM Transactions on Graphics (TOG)*, 43(4):1–20, 2024. 2

[70] Songchun Zhang, Yibo Zhang, Quan Zheng, Rui Ma, Wei Hua, Hujun Bao, Weiwei Xu, and Changqing Zou. 3d-scenedreamer: Text-driven 3d-consistent scene generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10170–10180, 2024. 2, 3

[71] Shougao Zhang, Mengqi Zhou, Yuxi Wang, Chuanchen Luo, Rongyu Wang, Yiwei Li, Xucheng Yin, Zhaoxiang Zhang, and Junran Peng. Cityx: Controllable procedural content generation for unbounded 3d cities. *arXiv preprint arXiv:2407.17572*, 2024. 2

[72] Mengqi Zhou, Jun Hou, Chuanchen Luo, Yuxi Wang, Zhaoxiang Zhang, and Junran Peng. Scenex: Procedural controllable large-scale scene generation via large-language models. *arXiv preprint arXiv:2403.15698*, 2024. 2

[73] Xiaoyu Zhou, Xingjian Ran, Yajiao Xiong, Jinlin He, Zhiwei Lin, Yongtao Wang, Deqing Sun, and Ming-Hsuan Yang. Gala3d: Towards text-to-3d complex scene generation via layout-guided generative gaussian splatting. *arXiv preprint arXiv:2402.07207*, 2024. 2

[74] Zihan Zhu, Songyou Peng, Viktor Larsson, Weiwei Xu, Hujun Bao, Zhaopeng Cui, Martin R Oswald, and Marc Pollefeys. Nice-slam: Neural implicit scalable encoding for slam. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12786–12796, 2022. 3, 4