

WorldGrow: Generating Infinite 3D World

Sikuang Li^{1*†} Chen Yang^{2*} Jiemin Fang^{2✉} Taoran Yi^{3†} Jia Lu^{3†}
 Jiazhong Cen^{1†} Lingxi Xie² Wei Shen¹ Qi Tian^{2✉}

¹MoE Key Lab of Artificial Intelligence, School of Computer Science, SJTU

²Huawei Inc. ³Huazhong University of Science and Technology

World-Grow.github.io

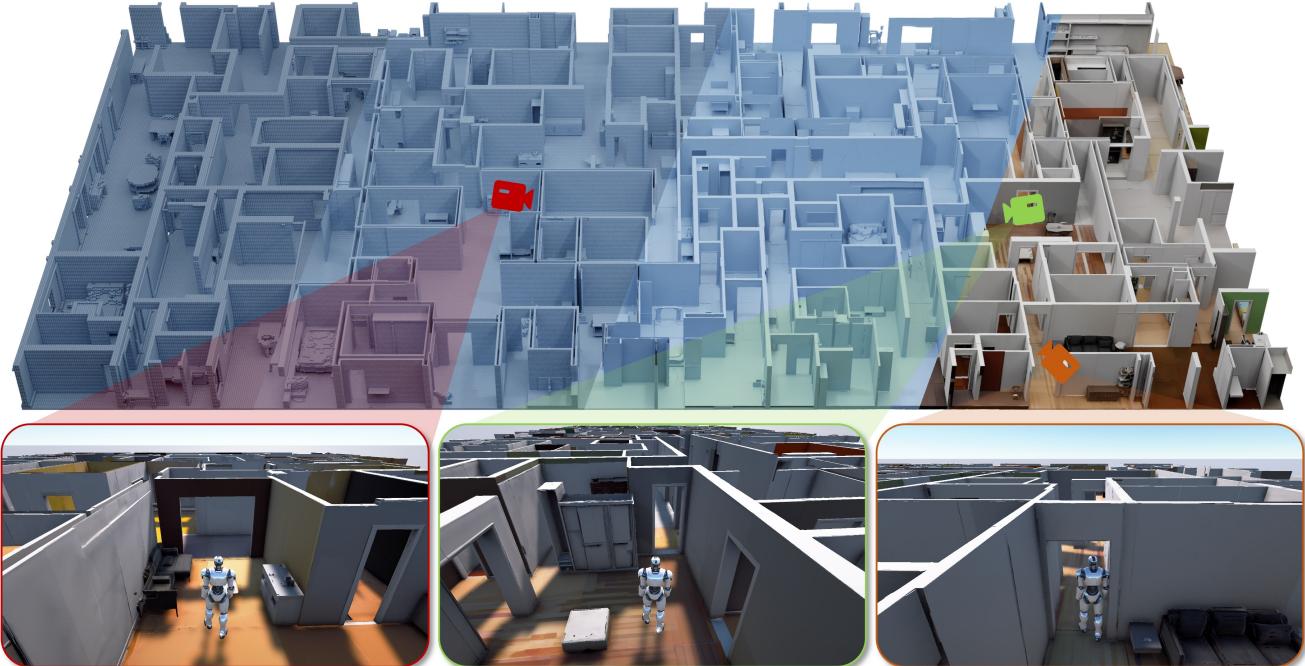


Figure 1. We introduce **WorldGrow**, a novel framework for infinite 3D world generation via block-wise synthesis and growth with coarse-to-fine refinement. Starting from a single seed block, WorldGrow progressively generates large-scale 3D scenes with coherent geometry and photorealistic appearance. *Top*: An indoor scene of 19×39 blocks (covering $\sim 1,800 \text{ m}^2$); left→right: coarse voxel layout, refined voxels, mesh reconstruction, and textured rendering. *Bottom*: An embodied agent navigates the generated world, demonstrating diverse room layouts and traversable spaces suitable for interactive AI tasks (e.g., navigation and planning).

Abstract

We tackle the challenge of generating the infinitely extendable 3D world – large, continuous environments with coherent geometry and realistic appearance. Existing methods face key challenges: 2D-lifting approaches suffer from geometric and appearance inconsistencies across views, 3D implicit representations are hard to scale up, and current 3D foundation models are mostly object-centric, limit-

ing their applicability to scene-level generation. Our key insight is leveraging strong generation priors from pre-trained 3D models for structured scene block generation. To this end, we propose **WorldGrow**, a hierarchical framework for unbounded 3D scene synthesis. Our method features three core components: (1) a data curation pipeline that extracts high-quality scene blocks for training, making the 3D structured latent representations suitable for scene generation; (2) a 3D block inpainting mechanism that enables context-aware scene extension; and (3) a coarse-to-fine generation strategy that ensures both global layout plausibility and local geometric/textural fidelity. Evaluated on the

*Equal contributions.

†Work done during internship at Huawei.

✉Corresponding authors.

large-scale 3D-FRONT dataset, WorldGrow achieves SOTA performance in geometry reconstruction, while uniquely supporting infinite scene generation with photorealistic and structurally consistent outputs. These results highlight its capability for constructing large-scale virtual environments and potential for building future world models.

1. Introduction

This paper addresses the critical challenge of generating the infinitely extendable 3D world, aiming to automatically create vast, continuous, and content-rich virtual environments. Such technology holds significant potential for industries including video games, virtual/augmented reality (VR/AR), computer-aided design, and film production. More importantly, infinite 3D world generation is foundational for developing *World Models* and embodied AI systems [31, 66], as it provides continuously expandable environments essential for open-ended learning, where agents can navigate, plan, and interact without the constraints of fixed-size worlds.

To achieve infinite 3D world generation, existing efforts have primarily explored two main approaches. One line of works [10, 14, 60, 70] relies on pre-trained 2D diffusion models [2, 18, 49] to generate images, which are then “lifted” to 3D scenes using camera poses, depth maps [1, 47], or image-to-3D models [65]. These methods optimize based on local viewpoints and lack a holistic understanding of the full 3D structure. As a result, they often suffer from geometric inaccuracies and appearance inconsistencies (*e.g.*, aliasing or distortion) across different views or extended regions, which further limits their ability to generate large-scale scenes. Another line of works [30, 39, 64] attempts to directly predict 3D representations (*e.g.*, triplanes [4, 64], UDFs [8, 39, 69], global latents [9]) by learning from 3D data for scene generation. However, their performance and generalization are often constrained by the limited scale and diversity of available scene-level datasets [16, 35]. Recent powerful 3D generation models [33, 65, 72, 73], empowered by large-scale training data [12], have demonstrated impressive capabilities in producing high-quality 3D assets. Though powerful, they are predominantly designed for single object generation, not applicable for infinite scene generation.

We propose to leverage the powerful generative capabilities of 3D generation models for block-based infinite scene generation – a promising yet challenging direction. The key challenges are threefold: 1) transferring rich geometric and textural priors from object-level models to generate scene blocks that are contextually coherent, rather than isolated assets; 2) ensuring seamless geometric, stylistic, and textural coherence between adjacent 3D blocks during iterative scene growth; 3) achieving global structural plausibility and

semantic diversity in large-scale compositions, avoiding incoherent arrangements.

To address these challenges, we introduce WorldGrow, a novel framework that, for the first time, enables the generation of infinite continuous 3D Worlds with plausible layouts and high-fidelity appearances in a region-growing manner. First, we design a data preparation pipeline to extract sufficient high-quality ground-truth scene blocks from existing datasets. In addition, we adapt object-level 3D representation to be scene-friendly, enabling the migration of learned object priors for generating scene blocks with fine-grained geometry and appearance. Second, we develop a 3D block inpainting pipeline to ensure robust and context-aware completion of missing blocks during iterative extension. Finally, to ensure both global coherence and local detail, we curate coarse and fine datasets focused on layouts and appearances, respectively. During generation, a coarse-trained model builds the scene structure first, then a fine-trained model refines detailed geometry and textures. As shown in Fig. 1, WorldGrow generates detail-rich, photorealistic, and infinitely extendable 3D scenes, highlighting its strong potential for large-scale virtual world construction.

In summary, our main contributions are as follows:

- 1) A systematic data construction pipeline and the created scene block datasets, enabling scalable training and evaluation for block-based infinite scene generation.
- 2) An infinite 3D scene generation framework, WorldGrow, which synthesizes continuous and unbounded 3D worlds with coherent layouts and photorealistic appearances.
- 3) A set of novel techniques enabling high-quality world generation, including scene-friendly SLATs for adapting object-level priors, a 3D inpainting method for seamless block completion, and a coarse-to-fine generation strategy that balances the global structure and local details.

2. Related Work

2.1. 3D Generation Pretrained Models

Recent advances in 3D pretraining have shown great promise in single-image 3D object generation. Leveraging representations such as triplanes [4] and 3D Gaussian Splatting (3DGS)[26], a number of feed-forward models[23, 24, 29, 55, 57, 59, 76] have been developed to directly synthesize 3D content from a single image.

To better capture 3D structural priors, several works [7, 33, 48, 62, 72–75] focus on sampling voxels or point clouds from 3D shapes to coarsely define geometry, which are then embedded into latent spaces [28] via generative models [21, 51]. TRELLIS [65] introduces a novel 3D representation named Structured LATents (SLAT), which encodes object shapes into sparse voxel grids with DI-NOv2 features [43]. Later extensions [6, 19, 56, 63] im-

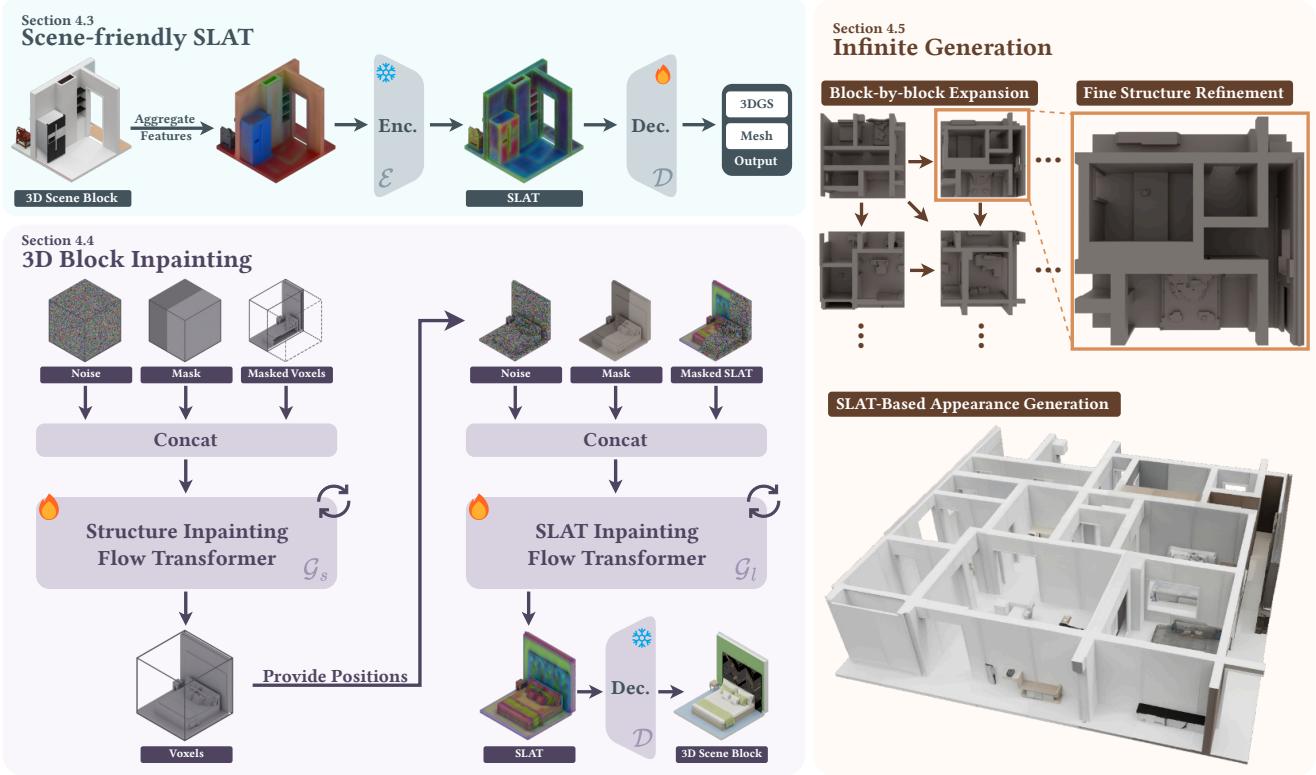


Figure 2. Overview of WorldGrow. Our goal is to generate infinite 3D scenes through modular, block-by-block synthesis. We begin by curating high-quality scene blocks and adapting SLAT to better model structured 3D context. A 3D block inpainting module enables spatially coherent extension, while a coarse-to-fine generation strategy ensures global layout plausibility and local detail fidelity. Together, these components allow WorldGrow to progressively construct photorealistic and structurally consistent 3D worlds with theoretically unbounded spatial extent.

prove SLAT by revising voxel sampling strategies. UniLat3D [61] unifies geometry and appearance into a single compact latent space for direct single-stage generation. These methods are typically trained on large-scale 3D object datasets [11, 12, 17, 27, 52], achieving high-quality object reconstruction. However, their applicability to scene-level generation is still underexplored, and we perform some preliminary attempts in this paper.

2.2. 3D Unbounded Scene Generation

To extend 3D generation to scene-level tasks, recent works explore the use of 2D inpainting diffusion models [10, 70] or video diffusion models with camera control [3, 34, 58, 67, 71] to hallucinate plausible multi-view images from single-view or textual inputs. Other approaches [13, 22, 25, 32, 54] aim to generate more complete geometry, such as 360-degree scenes within indoor environments. However, these methods typically generate only small-scale scenes, such as a single room.

More recent efforts toward unbounded scene generation aim to produce 3D content that can be extended in-

initely in all directions. BlockFusion [64] partitions 3D scenes into local blocks, encodes them as triplanes, and employs triplane extrapolation to synthesize neighboring blocks. Other methods utilize Truncated Unsigned Distance Field (TUDF) [8, 39, 69] or vector-set latents [30] to reconstruct the 3D scene blocks.

While these methods achieve compelling unbounded geometry generation, they typically lack explicit texture modeling. Instead, they rely on external texture synthesis and mapping pipelines [5, 68] to produce realistic surface appearances. SynCity [14] proposes a training-free pipeline that divides a scene into grids, generates descriptive captions for each grid using Large Language Models (LLMs) [42], synthesizes images via text-to-image diffusion models [2], and finally reconstructs textured 3D scenes using pretrained 3D generation models. Despite its scalability, this method suffers from limited view consistency: high-quality rendering is restricted to camera poses seen during diffusion generation, with fidelity degrading as the viewpoint diverges.

3. Preliminary: TRELLIS

TRELLIS [65], serving as a foundational model to our work, is a text/image-conditioned 3D generation model operating by denoising features in a sparse 3D latent space.

Structured Latent Representation. TRELLIS represents 3D objects via structured latents (SLATs): $\mathbf{z} = \{\mathbf{z}_i, \mathbf{p}_i\}_{i=1}^L$. Here, $\mathbf{z}_i \in \mathbb{R}^C$ is a latent feature at position $\mathbf{p}_i \in \{0, \dots, N-1\}^3$ (N : grid resolution), and $L \ll N^3$ is the count of active surface voxels. SLATs encode coarse geometry structure and fine appearance of 3D objects by linking latent features to active voxel locations using a Transformer-based variational autoencoder (VAE) [28], including an encoder \mathcal{E} and a decoder \mathcal{D} . \mathcal{E} maps sparse voxel features $\mathbf{f} = \{(\mathbf{f}_i, \mathbf{p}_i)\}_{i=1}^L$ of a 3D object to structured latents \mathbf{z} , where \mathbf{f}_i is a local visual feature created by projecting DINOv2 [43] feature maps (derived from multiview renders of the object) on to the voxel \mathbf{p}_i and averaging the retrieved features. \mathcal{D} then decodes \mathbf{z} into 3D representations, *e.g.*, 3D Gaussians [26], radiance fields [40], and meshes.

Structured Latent Generation. SLAT generation is a two-stage pipeline: Stage 1 predicts active voxel centers $\{\mathbf{p}_i\}_{i=1}^L$, and stage 2 recovers their latent features $\{\mathbf{z}_i\}_{i=1}^L$. Each stage uses a flow Transformer [36] v_θ on a latent code ℓ . It learns to reverse noise addition ($\ell^{(t)} = (1-t)\ell^{(0)} + t\epsilon$, where $\epsilon \sim \mathcal{N}(0, \mathbf{I})$ and $t \in [0, 1]$) by minimizing the flow loss:

$$\min_{\theta} \mathbb{E}_{(\ell^{(0)}, x), t, \epsilon} \|v_\theta(\ell^{(t)}, x, t) - (\epsilon - \ell^{(0)})\|_2^2,$$

where x is a conditioning prompt which can be either an image or a text prompt. In Stage 1 (for $\{\mathbf{p}_i\}$), $\ell \in \mathbb{R}^{L' \times C'}$ comprises L' tokens from a compressed N^3 occupancy volume. In Stage 2 (for $\{\mathbf{z}_i\}$), $\ell = \{\mathbf{z}_i\}_{i=1}^L \in \mathbb{R}^{L \times C}$ is a matrix of L C -dimensional tokens (L : active voxels from Stage 1). For better understanding, we term the flow Transformer in stage 1 as structure generation \mathcal{G}_s and flow Transformer in stage 2 as latent generation \mathcal{G}_l .

4. Method

4.1. Task Definition and Overall Framework

We define the task of synthesizing an infinite 3D world \mathcal{W} exhibiting plausible layouts and high-fidelity appearances as follows. The world \mathcal{W} is conceptualized as an unbounded composition of interconnected 3D blocks. Each block \mathcal{B} within this world is generated iteratively, conditioned on previously synthesized blocks. For simplicity, we define \mathcal{B} as a rectangular block aligned with the horizontal axes (*i.e.*, XY), with equal widths in the X and Y directions.

To enable infinite scene generation, WorldGrow first curates high-quality scene blocks for training (Sec. 4.2). We adapt the SLAT representation for structured 3D block modeling (Sec. 4.3), implement a 3D block inpainting module for context-aware completion (Sec. 4.4), and describe our coarse-to-fine generation strategy that achieves global layout plausibility with local detail fidelity (Sec. 4.5). The pipeline is shown in Fig. 2.

4.2. Data Curation

To enable infinite scene generation, we begin by constructing a dataset of structured, extendable 3D blocks. Existing 3D datasets, such as Objaverse-XL [12], are predominantly object-centric, consisting of isolated assets without spatial continuity. TRELLIS [65] performs well on such object-level data, but is not applicable to scene-level generation, which requires modular units that are spatially aligned and context-aware.

Scene Slicing. To bridge this gap, we propose a scene slicing strategy that partitions full 3D scenes (*e.g.*, a house or city) into coherent and reusable blocks. Given a full scene mesh, we extract training-ready blocks through the following process: we import the mesh into Blender [15], place a cuboid within its bounding box, and extract content via Boolean Intersection with the scene geometry. To ensure spatial density and avoid sparse regions, we render a top-down view and compute the occupancy of each extracted cuboid—if less than 95% of the surface contains visible content, the cuboid is repositioned and re-evaluated. This iterative sampling process yields multiple valid placements per scene, constructing a diverse set of spatially dense scene blocks. Our curated, topologically consistent data significantly reduces unrealistic geometry compared to naive partitioning approaches.

Coarse-to-Fine Data Strategy. Our method aims to synthesize unbounded, high-fidelity virtual worlds composed of 3D scene blocks with plausible global layouts. However, each block must be encoded into a SLAT, whose limited representational capacity constrains the amount of geometry and appearance detail it can effectively preserve. This introduces a fundamental trade-off in block design: larger 3D blocks capture broader scene context, benefiting global layout learning, but may suffer in rendering fidelity; conversely, smaller blocks support finer visual quality but lack sufficient spatial context to learn coherent scene structures.

To address this, we adopt a coarse-to-fine data strategy that balances context and detail. We prepare two distinct datasets: *coarse* and *fine* blocks¹. Coarse blocks are defined

¹Throughout this paper, superscripts c and f on symbols denote their association (typically via training or definition) with the coarse and fine datasets, respectively.



Figure 3. Scene-friendly SLAT better models 3D scene blocks, particularly in areas with occlusions and near block boundaries.

with four times the area in the XY plane while maintaining the same height, thereby capturing larger spatial volumes and richer contextual information. Both types of blocks are extracted using the random spatial partitioning method described previously. These dual-resolution datasets form the foundation for training our generative pipeline across global layout generation and local detail refinement.

4.3. Scene-friendly SLAT

While SLAT has demonstrated strong performance in object-level generation, its direct application to 3D scene block synthesis faces critical limitations. We identify two primary challenges: 1) Direct feature aggregation. SLAT’s VAE training projects multiview DINOv2 features onto each voxel \mathbf{p}_i and aggregates them to form its visual feature \mathbf{f}_i . While effective for objects with minimal self-occlusion, this projection-based aggregation degrades in cluttered scenes where self-occlusions are prevalent. As a result, vanilla SLAT often fails to capture accurate spatial relationships, leading to artifacts such as color bleeding between adjacent surfaces. 2) Inadequate decoder for scene blocks. SLAT’s decoder \mathcal{D} is pre-trained on object-level data, which typically lacks detailed 3D content near object boundaries. As a result, when applied to scene blocks, \mathcal{D} often produces floaters and artifacts near the block edges. These decoding failures lead to visual discontinuities, such as floating geometry or broken transitions, when multiple blocks are composed into large-scale scenes.

To address these limitations, we introduce two key modifications to make SLAT more scene-friendly. First, we incorporate an occlusion-aware strategy during feature aggregation. While conceptually simple, this adjustment significantly improves the representation of occluded regions and yields more consistent voxel features in cluttered scenes. Second, we retrain the decoder \mathcal{D} on scene block data, shifting its focus from isolated objects to structured scene content. This adaptation enables the decoder to better handle boundary regions, resulting in cleaner geometry and more coherent textures, especially at block edges. Together, these adaptations substantially reduce structural artifacts and enable more reliable scene block synthesis, as shown in Fig. 3.

4.4. 3D Block Inpainting

While scene-friendly SLAT improves the quality and consistency of individual block synthesis, extending a scene block-by-block requires reasoning over partial context and ensuring continuity with surrounding geometry and appearance. To address this challenge, we formulate scene expansion as a 3D block inpainting task, where a missing target block is synthesized based on its surrounding spatial neighbors.

Inherent from TRELLIS, we use a two-stage inpainting framework that operates on structure and latent space. Given a partially observed block with missing regions, our model first predicts the 3D structure (\mathbf{p}_b) and then reconstructs the corresponding latent features (\mathbf{z}_b) for high-fidelity appearance synthesis. To enable the model to better localize and infer missing regions, we modify the input layer of the models. Specifically, instead of using noisy latents as input, we concatenate three components along the channel dimension: the noisy latents, a binary mask indicating the inpainting region, and the masked known region itself. This design allows the model to condition its prediction on both the known context and explicit spatial cues of the missing area. By learning to denoise this composite input, the network is able to infer the structure and appearance of missing regions while preserving the observed content, improving the spatial continuity and stability of 3D block inpainting.

To train the inpainting model, we randomly select two splitting positions along the X and Y axes to divide each scene block into four quadrants, keeping one as context and masking the remaining three. For *structure inpainting*, we define a voxel-level binary mask $m_s \in \{0, 1\}^{N \times N \times N}$, where $m_s = 1$ denotes voxels to be inpainted. The structure generator \mathcal{G}_s takes this mask as input to complete the missing geometry. For *latent inpainting*, we define a sparse mask $m_l = \{(m_i, \mathbf{p}_i)\}_{i=1}^L$, where \mathbf{p}_i is the spatial coordinate and $m_i \in \{0, 1\}$ indicates whether to inpaint. This guides the latent generator \mathcal{G}_l to reconstruct corresponding features.

Both generators are optimized using a flow-matching loss:

$$\min_{\theta} \mathbb{E}_{(\ell^{(0)}, m, x), t, \epsilon} \|\mathcal{G}(\ell^{(t)}, m, \ell_m^{(0)}, x, t) - (\epsilon - \ell^{(0)})\|_2^2,$$

where $\ell_m^{(0)} = \ell^{(0)} \otimes (1 - m)$ is the latent code masked, \otimes denotes the Hadamard product, and (\mathcal{G}, ℓ, m) corresponds to either $(\mathcal{G}_s, \mathbf{p}_b, m_s)$ or $(\mathcal{G}_l, \mathbf{z}_b, m_l)$, depending on the task.

To support coarse-to-fine generation, we train separate models: \mathcal{G}_s^c on coarse blocks for structure inpainting, and $\mathcal{G}_s^f, \mathcal{G}_l^f$ on fine blocks for structure and latent inpainting, respectively – balancing global coherence and local detail.

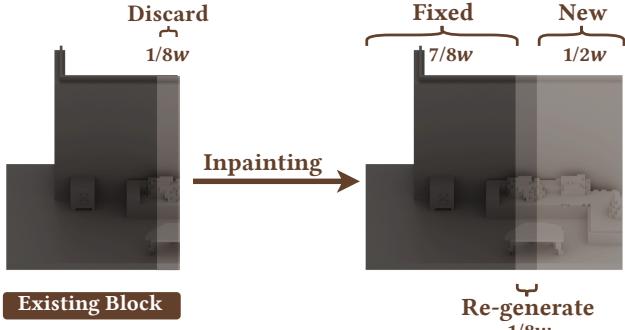


Figure 4. 1D illustration of our block-by-block expansion. Existing block’s $[1/2w, 7/8w]$ area is used as context for inpainting the next block. Thus, the final region $[7/8w, w)$ is discarded and then re-generated during expansion.

4.5. Infinite Scene Generation

With all components available, we now describe how WorldGrow constructs an infinite 3D world \mathcal{W} via a block-based, coarse-to-fine generation strategy. Starting from a seed block, the world is progressively extended in the XY plane through iterative 3D block inpainting. A coarse model first lays out the global structure across blocks, which is then refined by fine-level models to recover detailed geometry and generate the corresponding SLATs for each region.

Block-by-Block Expansion. We initiate scene generation from a seed block, which can either be synthesized by our inpainting model with a full 3D mask or initialized using a sample from vanilla TRELLIS. The scene is then expanded block by block, typically along the $+X$ and $+Y$ directions.

For each new block, the inpainting model takes as context the previously generated blocks to its left, top, and top-left (if available). See Fig. 4 for a 1D illustration of the expansion. To ensure continuity, we reuse a portion of these existing blocks: Specifically, we reuse a $3/8w$ -wide margin from each neighboring block along X and Y axes. This overlapping region corresponds to $[1/2w, 7/8w]$ on each axis. Based on this context, we inpaint the central $5/8w \times 5/8w$ region to complete a new $12/8w \times 12/8w$ block. This overlapping design ensures smooth transitions across block boundaries and provides a consistent context window for each expansion step.

Coarse Structure Generation. To establish the large-scale layout of the scene, we first apply the block-by-block generation process using the coarse structure model \mathcal{G}_s^c . This produces a low-resolution but spatially coherent structure \mathbf{p}_w^c that defines the overall geometry of the world.

Fine Structure Refinement. To enrich local geometry, we refine \mathbf{p}_w^c using the fine structure generator \mathcal{G}_s^f . We begin by upsampling \mathbf{p}_w^c via trilinear interpolation to match the voxel resolution of the fine stage, producing $\mathbf{p}_w^{c \uparrow f}$. This high-resolution structure is then partitioned into standard fine blocks.

Rather than generating each fine block from scratch, we adopt a structure-guided denoising approach inspired by SDEdit [38]. For each upsampled fine block $\mathbf{p}_{\text{fblock}}^{c \uparrow f}$, we encode it into an initial latent $\ell_{\text{fblock}}^{(0)}$. We then perturb this latent with controlled Gaussian noise:

$$\ell_{\text{fblock}}^{(t')} = (1 - t')\ell_{\text{fblock}}^{(0)} + t'\epsilon, \quad \text{where } 0 < t' < t.$$

The fine generator \mathcal{G}_s^f denoises $\ell_{\text{fblock}}^{(t')}$ to reconstruct the refined structure $\mathbf{p}_{\text{fblock}}^f$. This strategy enables preserving space distribution priors while enhancing details, effectively bridging global layout and fine-scale realism in a structure-aware generation process.

SLAT-Based Appearance Generation. Once the fine-level structure of the world \mathcal{W} , denoted as \mathbf{p}_w^f , is complete, we generate the corresponding SLATs \mathbf{z}_w . This stage follows the same block-by-block generation strategy as used for structure, but operates in the latent space. For each block, the latent generator \mathcal{G}_l^f synthesizes latents based on previously generated SLAT and current structure mask. Unlike structure inpainting, which uses dense voxel masks, latent inpainting is guided by sparse latent masks. After all latent blocks are generated, the full SLAT \mathbf{z}_w is decoded by our retrained \mathcal{D} into a renderable 3D world \mathcal{W} .

5. Experiments

5.1. Experiment Settings

Datasets. To align with previous infinite generation methods, We train WorldGrow on the dataset processed from 3D-FRONT [16, 17]. From the original 6,811 houses, we retain 3,072 after filtering, and include 353 additional houses that were manually corrected for higher quality. Consequently, our final dataset comprises 3,425 curated houses with reasonable layouts and detailed furnishings. From these, we generate 120k fine blocks and 38k coarse blocks. We also verify WorldGrow with city dataset UrbanScene3D [35] in Fig. 8. Please refer to the Appendix for details.

Implementation Details. We utilize a text-conditioned TRELLIS-XL model [65] for 3D block inpainting, where the conditioning text consists of a fixed generic scene description generated by a large language model [42]. This prompt provides minimal semantic guidance, allowing the model to focus on spatial and structural reasoning. For training, we optimize the inpainting model on our curated

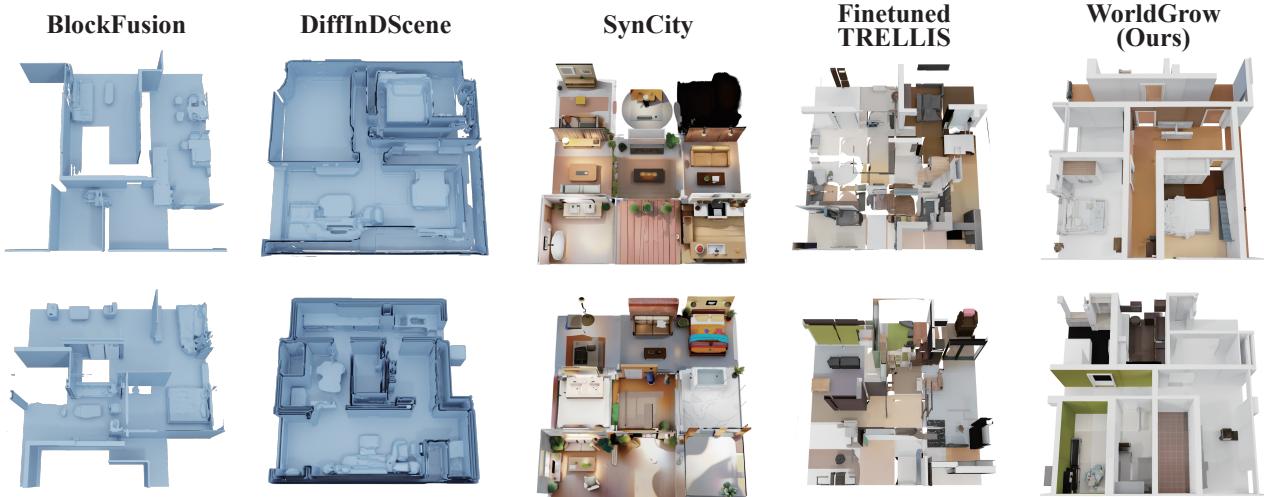


Figure 5. Qualitative comparison of indoor scene generation. We compare our method with state-of-the-art infinite scene generation approaches, indoor house generation methods and our baseline TRELLIS. WorldGrow produces high-resolution, continuous indoor scenes with realistic and coherent textures.

Method	MMD($\times 10^2$) \downarrow		COV(%) \uparrow		1-NNA(%) \downarrow		FID \downarrow
	CD	EMD	CD	EMD	CD	EMD	
DiffInDScape	6.57	27.70	2.83	5.26	99.30	97.69	84.41
BlockFusion	2.90	28.79	16.60	13.16	97.89	98.19	25.09
SynCity	1.37	19.54	19.03	11.94	90.04	93.56	34.69
TRELLIS	3.15	23.75	13.97	11.74	99.20	98.79	53.49
TRELLIS †	1.47	15.03	46.56	45.95	81.59	74.55	24.61
Ours	0.97	13.33	51.82	46.56	66.30	69.01	7.52
Ours w/o DC	1.00	13.84	46.76	40.49	69.01	74.65	9.09
Ours w/o CSG	1.08	13.62	43.93	40.28	73.24	72.33	17.04

Table 1. Quantitative results on scene block geometry evaluation. We report comparisons with state-of-the-art scene generation methods, along with results from our ablation study. TRELLIS † denotes TRELLIS fine-tuned on 3D-FRONT. “DC” refers to Data Curation, and “CSG” denotes Coarse Scene Generation.

dataset for 200k iterations using AdamW [37] with a learning rate of 0.0001, while the VAE backbone is trained separately for 100k iterations with the same configuration. During inference, we maintain the same noise scheduling as training with 50 sampling steps. On a single A100 GPU, each block generation takes 20 seconds (6 times faster than SynCity’s 2 minutes), and a complete 10×10 indoor scene (around 272 m^2) can be generated in 30 minutes using only 13GB of peak memory.

Metrics. We evaluate our method across two aspects: scene block generation and full scene synthesis. For block generation, we assess both geometric and visual quality. Following prior works [9, 39, 64], we report three stan-

Method	CLIP \uparrow	FID _{Incep} \downarrow	FID _{DINOv2} \downarrow	FID _{CLIP} \downarrow
DiffInDScape †	0.768	156.80	2066.13	42.43
BlockFusion †	0.758	138.34	1776.79	42.04
SynCity	0.804	101.83	655.60	16.22
TRELLIS †	0.813	101.94	674.65	13.17
Ours	0.843	29.87	313.54	3.95

Table 2. Visual fidelity evaluation of generated blocks. Methods with † generate geometry only; we apply uniform white texture for rendering and evaluation. TRELLIS † denotes TRELLIS fine-tuned on 3D-FRONT.

dard distribution-based metrics (MMD, COV, and 1-NNA) computed using both Chamfer Distance (CD) and Earth Mover’s Distance (EMD). We additionally adopt the perceptual Fréchet Inception Distance (FID) [20] with PointNet++ [44] following the protocol in [41] to assess 3D geometric quality. For visual quality, we render generated blocks from fixed multiple viewpoints and compute perceptual metrics including CLIP score [45] and FID variants with different feature extractors (Inception V3 [53], DINOv2 [43], and CLIP). For full-scene synthesis, where ground-truth meshes are unavailable, we conduct a human preference study with 91 participants who compare 5 methods across 10 scenes (4 house-level, 6 unbounded) presented in random order, evaluating structural plausibility, geometric detail, appearance fidelity, and scene continuity.

Compared Methods. We compare WorldGrow with SOTA infinite scene generation methods, including BlockFusion [64] and SynCity [14]. Additionally, we eval-



Figure 6. Gallery of scenes generated by WorldGrow. Top: 5×5 block layouts. Bottom: 9×9 blocks.



Figure 7. Large-scale scene generated by WorldGrow. The layout spans 19×39 blocks ($\sim 1,800 \text{ m}^2$). Left: reconstructed mesh. Right: textured rendering. This example is distinct from Figure 1 and illustrates scalability to large environments.

uate against scene-scale generation baselines such as Text2Room [22] and DiffInDScape [25]. The original text-conditioned TRELLIS [65] is included as a foundational baseline for comparison.

5.2. 3D Scene Generation

Scene Block Generation. To evaluate scene block connectivity, we modify the evaluation protocol from BlockFusion and LT3SD. Instead of generating individual blocks in isolation, we task each method with synthesizing larger 3×3 scenes and randomly sample 1×1 blocks for evaluation against the 3D-FRONT dataset distribution.

As shown in Fig. 5, SynCity exhibits poor continuity with visible discontinuities between segments, while other methods like fine-tuned TRELLIS produce locally valid blocks but lack outpainting capabilities. Quantitative results

in Table 1 confirm these observations, where WorldGrow achieves SOTA performance across all geometry metrics, demonstrating superior connectivity and structural coherence. To evaluate visual fidelity, we render the synthesized blocks from 10 fixed viewpoints and compare these multi-view images against renders from the 3D-FRONT dataset. As shown in Table 2, WorldGrow achieves significantly better perceptual quality than all baselines, demonstrating its ability to generate high-quality scene blocks with realistic appearance.

Full Scene Generation. We conduct a human preference study on textured indoor and unbounded scene generation. Following BlockFusion [64], we ask participants to evaluate structure plausibility (SP), geometry detail (GD), and appearance fidelity (AF) for indoor scenes, with an additional

Method	Textured Scenes			Unbounded Scenes			
	SP	GD	AF	SP	GD	AF	CO
Text2Room	2.07	1.56	2.07	/	/	/	/
Blockfusion	/	/	/	3.48	3.30	1.20	3.36
TRELLIS	2.82	2.26	2.89	2.15	2.96	3.33	2.38
SynCity	2.48	3.11	3.59	2.48	3.07	4.08	2.74
Ours	4.48	4.44	4.33	4.46	4.37	4.33	4.69

Table 3. Average of human preference scores (1–5).

Method	MMD($\times 10^2$) \downarrow		COV(%) \uparrow		1-NNA(%) \downarrow		FID \downarrow
	CD	EMD	CD	EMD	CD	EMD	
SynCity	1.68	19.39	15.38	13.97	94.27	93.76	51.97
Ours	0.96	12.83	48.99	48.18	59.66	64.79	5.43

Table 4. Expansion stability evaluation on outer regions. We evaluate 1×1 blocks sampled from regions beyond the initial 3×3 part among 7×7 generated scenes. Our method maintains consistent quality in distant expansions, while SynCity shows significant degradation.

criterion of continuity (CO) for unbounded scenes. As shown in Table 3, our method outperforms baseline methods across all criteria, particularly excelling in scene structure layout and continuity—demonstrating the effectiveness of our block-by-block expansion and coarse-to-fine generation strategy. Fig. 6 presents multiple distinct scenes produced by WorldGrow at increasing sizes, illustrating open-ended scalability across scene generation. We additionally present a 19×39 indoor scene to highlight WorldGrow’s scalability and consistency at large extents in Fig. 7, demonstrating that WorldGrow can sustain quality when expanding far beyond the initial region, with minimal seams or drift, and yielding navigable, walk-only spaces suitable for planning-oriented embodied evaluation.

We also propose an expansion stability experiment to quantitatively assess long-run generation quality and error accumulation. In this experiment, we synthesize large scenes with 7×7 blocks and randomly sample 1×1 blocks exclusively from the outer regions (beyond the initial 3×3 region) for evaluation using our block evaluation metrics. As shown in Table 4, WorldGrow maintains consistent generation quality even at distant expansions, achieving scores comparable to those in Table 1, while SynCity shows significant performance degradation (*e.g.*, FID increases from 34.69 to 51.97). Note that SynCity fails in 70% of expansion attempts, with only successful cases reported in the table. These results demonstrate WorldGrow’s robust stability in infinite scene generation without quality deterioration or seam accumulation over extended expansions.

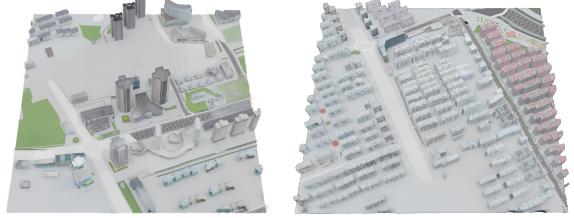


Figure 8. Infinite outdoor 3D scene generation by WorldGrow. Our method synthesizes diverse scenes such as urban streetscapes with plausible layouts, coherent suburban neighborhoods with consistent styles, showing WorldGrow’s ability to be adapted to various domains.

5.3. Ablation Study

We perform a series of experiments to validate the effectiveness of each component.

Data Curation. We first validate our data curation by comparing models trained on filtered versus unfiltered 3D-FRONT data. As shown in Fig. 9, training without data curation results in object interpenetration and implausible arrangements, while our curated dataset produces spatially coherent scenes.

Scene-Friendly SLAT. Our scene-friendly adaptation modifies TRELLIS’s VAE to better support scene-level generation, introducing two key components: an occlusion-aware feature aggregation mechanism and a decoder retrained on scene blocks. To assess their impact on SLAT’s ability to reconstruct realistic scene blocks, we conduct an ablation study against three variants: (i) the original object-centric VAE, (ii) a version with only occlusion-aware aggregation, and (iii) a version with only the retrained decoder.

As shown in Table 5, applying occlusion-aware aggregation alone, without retraining the decoder, results in performance degradation due to encoder-decoder mismatch. However, combining two components yields significant improvements, demonstrating their synergy in adapting SLAT for coherent scene-level reconstruction.

Coarse-to-Fine Generation. Here, we validate our coarse-to-fine generation strategy by comparing against direct fine-scale generation. As shown in Fig. 9, direct fine generation struggles with global layout consistency, producing implausible furniture arrangements. Our coarse-to-fine approach establishes coherent structure via \mathcal{G}_s^c , then enriches details through \mathcal{G}_s^f , achieving superior balance between global coherence and local realism.



Figure 9. Ablation study on key components of WorldGrow. Left: Without Data Curation, the generated wardrobe intersects with multiple walls, indicating poor spatial alignment. Right: Without Coarse-to-Fine generation, the global furniture layout becomes cluttered and less coherent.

Occ. Aware	Retrain \mathcal{D}	LPIPS \downarrow	PSNR \uparrow	SSIM \uparrow
\times	\times	0.0741	23.17	0.9273
\checkmark	\times	0.0850	22.23	0.9046
\times	\checkmark	0.0491	25.84	0.9531
\checkmark	\checkmark	0.0311	31.32	0.9705

Table 5. Ablation study about components of scene-friendly SLAT. Occ. Aware means occlusion aware feature aggregation and Retrain \mathcal{D} is retraining VAE’s decoder.

6. Discussion and Future Work

While WorldGrow demonstrates strong results, several limitations remain. Currently, our method extends scenes only in the XY plane, leaving vertical expansion along the Z -axis—essential for multi-story buildings—as an important direction for future work. Generation quality and diversity are also bounded by current 3D dataset limitations in scale, variety, and semantic annotations. Our block-wise design trades off fine geometric details for computational feasibility, prioritizing infinite generation capability over local detail resolution. Additionally, while WorldGrow naturally supports conditional control, the current implementation focuses on unconditional generation without semantic conditioning.

These points present clear opportunities for future research. Multi-level generation strategies could enable vertical expansion for complete buildings. Larger-scale dataset curation – particularly for outdoor environments where our preliminary experiments on city scenes show promising results – would enhance both diversity and quality. Introducing LLM-generated captions could enable fine-grained semantic control over room types and layouts. Moreover, it could be promising to integrate WorldGrow into geometry-appearance unified generation models [61] for more efficient pipelines.

7. Conclusion

We presented WorldGrow, a novel framework for infinite 3D world generation that constructs unbounded environments with coherent layout and photorealistic appearance. Through our block-based context-aware inpainting mechanism and coarse-to-fine refinement strategy, we leverage pre-trained 3D priors to overcome the fundamental scalability and coherence limitations that have constrained prior methods. Our comprehensive evaluation demonstrates SOTA performance in geometry reconstruction and visual fidelity, while uniquely enabling the generation of large-scale scenes that maintain both local detail and global consistency. As virtual worlds become increasingly important for embodied AI training and simulation, WorldGrow provides a practical path toward scalable, high-quality 3D content generation for future world models.

References

- [1] Reiner Birkl, Diana Wofk, and Matthias Müller. Midas v3.1 – a model zoo for robust monocular relative depth estimation. *arXiv preprint arXiv:2307.14460*, 2023. [2](#)
- [2] Black Forest Labs. Flux. <https://github.com/black-forest-labs/flux>, 2024. [2, 3](#)
- [3] Andreas Blattmann, Tim Dockhorn, Sumith Kulal, Daniel Mendelevitch, Maciej Kilian, Dominik Lorenz, Yam Levi, Zion English, Vikram Voleti, Adam Letts, Varun Jampani, and Robin Rombach. Stable video diffusion: Scaling latent video diffusion models to large datasets, 2023. [3](#)
- [4] Eric R. Chan, Connor Z. Lin, Matthew A. Chan, Koki Nagano, Boxiao Pan, Shalini De Mello, Orazio Gallo, Leonidas J. Guibas, Jonathan Tremblay, Sameh Khamis, Tero Karras, and Gordon Wetzstein. Efficient geometry-aware 3d generative adversarial networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 16123–16133, 2022. [2](#)
- [5] Dave Zhenyu Chen, Haoxuan Li, Hsin-Ying Lee, Sergey Tulyakov, and Matthias Nießner. Scenetex: High-quality texture synthesis for indoor scenes via diffusion priors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 21081–21091, 2024. [3](#)
- [6] Yiwen Chen, Zhihao Li, Yikai Wang, Hu Zhang, Qin Li, Chi Zhang, and Guosheng Lin. Ultra3d: Efficient and high-fidelity 3d generation with part attention, 2025. [2](#)
- [7] Zhaoxi Chen, Jiaxiang Tang, Yuhao Dong, Ziang Cao, Fangzhou Hong, Yushi Lan, Tengfei Wang, Haozhe Xie, Tong Wu, Shunsuke Saito, Liang Pan, Dahu Lin, and Ziwei Liu. 3dtopia-xl: High-quality 3d pbr asset generation via primitive diffusion. *arXiv preprint arXiv:2409.12957*, 2024. [2](#)
- [8] Julian Chibane, Aymen Mir, and Gerard Pons-Moll. Neural unsigned distance fields for implicit function learning. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, Red Hook, NY, USA, 2020. Curran Associates Inc. [2, 3](#)

- [9] Gene Chou, Yuval Bahat, and Felix Heide. Diffusion-sdf: Conditional generative modeling of signed distance functions. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 2262–2272, 2023. 2, 7
- [10] Jaeyoung Chung, Suyoung Lee, Hyeongjin Nam, Jaerin Lee, and Kyoung Mu Lee. Luciddreamer: Domain-free generation of 3d gaussian splatting scenes. *arXiv preprint arXiv:2311.13384*, 2023. 2, 3
- [11] Jasmine Collins, Shubham Goel, Kenan Deng, Achleshwar Luthra, Leon Xu, Erhan Gundogdu, Xi Zhang, Tomas F. Yago Vicente, Thomas Dideriksen, Himanshu Arora, Matthieu Guillaumin, and Jitendra Malik. Abo: Dataset and benchmarks for real-world 3d object understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 21126–21136, 2022. 3
- [12] Matt Deitke, Ruoshi Liu, Matthew Wallingford, Huong Ngo, Oscar Michel, Aditya Kusupati, Alan Fan, Christian Laforte, Vikram Voleti, Samir Yitzhak Gadre, Eli VanderBilt, Aniruddha Kembhavi, Carl Vondrick, Georgia Gkioxari, Kiana Ehsani, Ludwig Schmidt, and Ali Farhadi. Objaverse-XL: A universe of 10m+ 3d objects. In *Thirty-seventh Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2023. 2, 3, 4
- [13] Wenqi Dong, Bangbang Yang, Zesong Yang, Yuan Li, Tao Hu, Hujun Bao, Yuewen Ma, and Zhaopeng Cui. Hiscene: Creating hierarchical 3d scenes with isometric view generation, 2025. 3
- [14] Paul Engstler, Aleksandar Shtedritski, Iro Laina, Christian Rupprecht, and Andrea Vedaldi. Syncity: Training-free generation of 3d worlds, 2025. 2, 3, 7, 15
- [15] Blender Foundation. Blender, 2025. 4
- [16] Huan Fu, Bowen Cai, Lin Gao, Ling-Xiao Zhang, Jiaming Wang, Cao Li, Qixun Zeng, Chengyue Sun, Rongfei Jia, Binqiang Zhao, and Hao Zhang. 3d-front: 3d furnished rooms with layouts and semantics. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 10933–10942, 2021. 2, 6, 14
- [17] Huan Fu, Rongfei Jia, Lin Gao, Mingming Gong, Binqiang Zhao, Steve Maybank, and Dacheng Tao. 3d-future: 3d furniture shape with texture. *Int. J. Comput. Vision*, 129(12): 3313–3337, 2021. 3, 6
- [18] Google DeepMind. Genie 3: A new frontier for world models, 2025. DeepMind Blog. 2
- [19] Xianglong He, Zi-Xin Zou, Chia-Hao Chen, Yuan-Chen Guo, Ding Liang, Chun Yuan, Wanli Ouyang, Yan-Pei Cao, and Yangguang Li. Sparseflex: High-resolution and arbitrary-topology 3d shape modeling. *arXiv preprint arXiv:2503.21732*, 2025. 2
- [20] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. In *Advances in Neural Information Processing Systems*. Curran Associates, Inc., 2017. 7, 15
- [21] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *Advances in Neural Information Processing Systems*, pages 6840–6851. Curran Associates, Inc., 2020. 2
- [22] Lukas Höller, Ang Cao, Andrew Owens, Justin Johnson, and Matthias Nießner. Text2room: Extracting textured 3d meshes from 2d text-to-image models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 7909–7920, 2023. 3, 8, 15
- [23] Fangzhou Hong, Jiaxiang Tang, Ziang Cao, Min Shi, Tong Wu, Zhaoxi Chen, Tengfei Wang, Liang Pan, Dahua Lin, and Ziwei Liu. 3dtopia: Large text-to-3d generation model with hybrid diffusion priors. *arXiv preprint arXiv:2403.02234*, 2024. 2
- [24] 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. In *The Twelfth International Conference on Learning Representations*, 2024. 2
- [25] 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 (CVPR)*, pages 4526–4535, 2024. 3, 8, 15
- [26] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering. *ACM Transactions on Graphics*, 42(4), 2023. 2, 4
- [27] Mukul Khanna, Yongsen Mao, Hanxiao Jiang, Sanjay Haresh, Brennan Shacklett, Dhruv Batra, Alexander Clegg, Eric Undersander, Angel X. Chang, and Manolis Savva. Habitat synthetic scenes dataset (hssd-200): An analysis of 3d scene scale and realism tradeoffs for objectgoal navigation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 16384–16393, 2024. 3
- [28] Diederik P. Kingma and Max Welling. Auto-encoding variational bayes. In *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*, 2014. 2, 4
- [29] Yushi Lan, Fangzhou Hong, Shuai Yang, Shangchen Zhou, Xuyi Meng, Bo Dai, Xingang Pan, and Chen Change Loy. Ln3diff: Scalable latent neural fields diffusion for speedy 3d generation. In *ECCV*, 2024. 2
- [30] Han-Hung Lee, Qinghong Han, and Angel X. Chang. Nuiscene: Exploring efficient generation of unbounded outdoor scenes, 2025. 2, 3
- [31] Bohan Li, Jiazhe Guo, Hongsi Liu, Yingshuang Zou, Yikang Ding, Xiwu Chen, Hu Zhu, Feiyang Tan, Chi Zhang, Tiancai Wang, et al. Uniscene: Unified occupancy-centric driving scene generation. *arXiv preprint arXiv:2412.05435*, 2024. 2
- [32] Haoran Li, Haolin Shi, Wenli Zhang, Wenjun Wu, Yong Liao, Lin Wang, Lik-hang Lee, and Peng Yuan Zhou. Dreamscene: 3d gaussian-based text-to-3d scene generation via formation pattern sampling. In *European Conference on Computer Vision*, pages 214–230. Springer, 2024. 3
- [33] Weiyu Li, Jiarui Liu, Hongyu Yan, Rui Chen, Yixun Liang, Xuelin Chen, Ping Tan, and Xiaoxiao Long. Craftsman3d: High-fidelity mesh generation with 3d native gen-

- eration and interactive geometry refiner. *arXiv preprint arXiv:2405.14979*, 2024. 2
- [34] Hanwen Liang, Junli Cao, Vudit Goel, Guocheng Qian, Sergei Korolev, Demetri Terzopoulos, Konstantinos Plataniotis, Sergey Tulyakov, and Jian Ren. Wonderland: Navigating 3d scenes from a single image. *arXiv preprint arXiv:2412.12091*, 2024. 3
- [35] Liqiang Lin, Yilin Liu, Yue Hu, Xinguang Yan, Ke Xie, and Hui Huang. Capturing, reconstructing, and simulating: The urbanscene3d dataset. In *Computer Vision - ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23-27, 2022, Proceedings, Part VIII*, pages 93–109, Berlin, Heidelberg, 2022. Springer-Verlag. 2, 6, 14, 15
- [36] Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matthew Le. Flow matching for generative modeling. In *The Eleventh International Conference on Learning Representations*, 2023. 4
- [37] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019. 7
- [38] Chenlin Meng, Yutong He, Yang Song, Jiaming Song, Jiajun Wu, Jun-Yan Zhu, and Stefano Ermon. SDEdit: Guided image synthesis and editing with stochastic differential equations. In *International Conference on Learning Representations*, 2022. 6
- [39] Quan Meng, Lei Li, Matthias Nießner, and Angela Dai. Lt3sd: Latent trees for 3d scene diffusion, 2025. 2, 3, 7, 14, 15
- [40] 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. In *ECCV*, 2020. 4
- [41] Alex Nichol, Heewoo Jun, Prafulla Dhariwal, Pamela Mishkin, and Mark Chen. Point-e: A system for generating 3d point clouds from complex prompts, 2022. 7
- [42] OpenAI et al. Gpt-4 technical report, 2024. 3, 6, 15
- [43] Maxime Oquab, Timothée Darcret, Théo Moutakanni, Huy V. Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel HAZIZA, Francisco Massa, Alaaeldin El-Nouby, Mido Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Herve Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. DINOv2: Learning robust visual features without supervision. *Transactions on Machine Learning Research*, 2024. Featured Certification. 2, 4, 7, 14
- [44] Charles R. Qi, Li Yi, Hao Su, and Leonidas J. Guibas. Pointnet++: deep hierarchical feature learning on point sets in a metric space. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, page 5105–5114, Red Hook, NY, USA, 2017. Curran Associates Inc. 7, 15
- [45] Alec Radford, Jong Wook Kim, Chris Hallacy, A. Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *ICML*, 2021. 7, 15
- [46] Alexander Raistrick, Lahav Lipson, Zeyu Ma, Lingjie Mei, Mingzhe Wang, Yiming Zuo, Karhan Kayan, Hongyu Wen, Beining Han, Yihan Wang, et al. Infinite photorealistic worlds using procedural generation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12630–12641, 2023. 15
- [47] René Ranftl, Katrin Lasinger, David Hafner, Konrad Schindler, and Vladlen Koltun. Towards robust monocular depth estimation: Mixing datasets for zero-shot cross-dataset transfer. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(3), 2022. 2
- [48] 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 (CVPR)*, pages 4209–4219, 2024. 2
- [49] 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 (CVPR)*, pages 10684–10695, 2022. 2
- [50] 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 (CVPR)*, pages 20875–20886, 2023. 15
- [51] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *International Conference on Learning Representations*, 2021. 2
- [52] Stefan Stojanov, Anh Thai, and James M Rehg. Using shape to categorize: Low-shot learning with an explicit shape bias. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1798–1808, 2021. 3
- [53] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2818–2826, 2016. 7
- [54] 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 (CVPR)*, pages 20507–20518, 2024. 3
- [55] Jiaxiang Tang, Jiawei Ren, Hang Zhou, Ziwei Liu, and Gang Zeng. Dreamgaussian: Generative gaussian splatting for efficient 3d content creation. In *The Twelfth International Conference on Learning Representations*, 2024. 2
- [56] Tencent Hunyuan3D Team. Hunyuan3d 2.0: Scaling diffusion models for high resolution textured 3d assets generation, 2025. 2
- [57] Peng Wang, Hao Tan, Sai Bi, Yinghao Xu, Fujun Luan, Kalyan Sunkavalli, Wenping Wang, Zexiang Xu, and Kai Zhang. PF-LRM: Pose-free large reconstruction model for joint pose and shape prediction. In *The Twelfth International Conference on Learning Representations*, 2024. 2
- [58] Zhouxia Wang, Ziyang Yuan, Xintao Wang, Yaowei Li, Tianshui Chen, Menghan Xia, Ping Luo, and Ying Shan.

- Motionctrl: A unified and flexible motion controller for video generation. In *ACM SIGGRAPH 2024 Conference Papers*, New York, NY, USA, 2024. Association for Computing Machinery. 3
- [59] Xinyue Wei, Kai Zhang, Sai Bi, Hao Tan, Fujun Luan, Valentin Deschaintre, Kalyan Sunkavalli, Hao Su, and Zexiang Xu. Meshlrm: Large reconstruction model for high-quality mesh. *arXiv preprint arXiv:2404.12385*, 2024. 2
- [60] World Labs. Marble, 2025. Product site. 2
- [61] Guanjun Wu, Jiemin Fang, Chen Yang, Sikuang Li, Taoran Yi, Jia Lu, Zanwei Zhou, Jiazhong Cen, Lingxi Xie, Xiaopeng Zhang, et al. Unilat3d: Geometry-appearance unified latents for single-stage 3d generation. *arXiv preprint arXiv:2509.25079*, 2025. 3, 10
- [62] Shuang Wu, Youtian Lin, Yifei Zeng, Feihu Zhang, Jingxi Xu, Philip Torr, Xun Cao, and Yao Yao. Direct3d: Scalable image-to-3d generation via 3d latent diffusion transformer. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. 2
- [63] Shuang Wu, Youtian Lin, Feihu Zhang, Yifei Zeng, Yikang Yang, Yajie Bao, Jiachen Qian, Siyu Zhu, Xun Cao, Philip Torr, and Yao Yao. Direct3d-s2: Gigascale 3d generation made easy with spatial sparse attention, 2025. 2
- [64] Zhennan Wu, Yang Li, Han Yan, Taizhang Shang, Weixuan Sun, Senbo Wang, Ruikai Cui, Weizhe Liu, Hiroyuki Sato, Hongdong Li, and Pan Ji. Blockfusion: Expandable 3d scene generation using latent tri-plane extrapolation. *ACM Trans. Graph.*, 43(4), 2024. 2, 3, 7, 8, 15
- [65] 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. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025. 2, 4, 6, 8, 14, 15
- [66] 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 (CVPR)*, pages 9666–9675, 2024. 2
- [67] Jinbo Xing, Menghan Xia, Yong Zhang, Haoxin Chen, Wangbo Yu, Hanyuan Liu, Gongye Liu, Xintao Wang, Ying Shan, and Tien-Tsin Wong. Dynamicrafter: Animating open-domain images with video diffusion priors. In *Computer Vision - ECCV 2024: 18th European Conference, Milan, Italy, September 29–October 4, 2024, Proceedings, Part XLVI*, pages 399–417, Berlin, Heidelberg, 2024. Springer-Verlag. 3
- [68] Bangbang Yang, Wenqi Dong, Lin Ma, Wenbo Hu, Xiao Liu, Zhaopeng Cui, and Yuewen Ma. Dreamsphere: Dreaming your room space with text-driven panoramic texture propagation. In *2024 IEEE Conference Virtual Reality and 3D User Interfaces (VR)*, pages 650–660, 2024. 3
- [69] Zhongqi Yang, Yihua Chen, and Saru Kumari. Tudf-nerf: Generalizable neural radiance field via truncated unsigned distance field. In *2023 Asia-Pacific Conference on Image Processing, Electronics and Computers (IPEC)*, pages 366–371, 2023. 2, 3
- [70] Hong-Xing Yu, Haoyi Duan, Junhwa Hur, Kyle Sargent, Michael Rubinstein, William T. Freeman, Forrester Cole, Deqing Sun, Noah Snavely, Jiajun Wu, and Charles Hermann. Wonderjourney: Going from anywhere to everywhere. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6658–6667, 2024. 2, 3
- [71] Wangbo Yu, Jinbo Xing, Li Yuan, Wenbo Hu, Xiaoyu Li, Zhipeng Huang, Xiangjun Gao, Tien-Tsin Wong, Ying Shan, and Yonghong Tian. Viewcrafter: Taming video diffusion models for high-fidelity novel view synthesis. *arXiv preprint arXiv:2409.02048*, 2024. 3
- [72] Xiaohui Zeng, Arash Vahdat, Francis Williams, Zan Gojcic, Or Litany, Sanja Fidler, and Karsten Kreis. LION: Latent point diffusion models for 3d shape generation. In *Advances in Neural Information Processing Systems*, 2022. 2
- [73] 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 Trans. Graph.*, 43(4), 2024. 2
- [74] Zibo Zhao, Wen Liu, Xin Chen, Xianfang Zeng, Rui Wang, Pei Cheng, BIN FU, Tao Chen, Gang YU, and Shenghua Gao. Michelangelo: Conditional 3d shape generation based on shape-image-text aligned latent representation. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.
- [75] Xin-Yang Zheng, Hao Pan, Peng-Shuai Wang, Xin Tong, Yang Liu, and Heung-Yeung Shum. Locally attentional sdf diffusion for controllable 3d shape generation. *ACM Trans. Graph.*, 42(4), 2023. 2
- [76] Zi-Xin Zou, Zhipeng Yu, Yuan-Chen Guo, Yangguang Li, Ding Liang, Yan-Pei Cao, and Song-Hai Zhang. Triplane meets gaussian splatting: Fast and generalizable single-view 3d reconstruction with transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10324–10335, 2024. 2

In the appendix, we provide additional contents for:

1. Method Details (Sec. A)
2. Implementation Details (Sec. B)
3. Experiment Details (Sec. C)
4. More Experimental Results (Sec. D)

A. Method Details

A.1. Coarse-to-Fine Generation

WorldGrow introduces a hierarchical layout strategy where coarse and fine stages operate at different semantic levels rather than merely different resolutions. In our approach, each coarse block represents a 2×2 grid of fine blocks, capturing high-level spatial relationships and room-scale structure. The coarse model \mathcal{G}_s^c establishes the global scene layout—defining where rooms connect and how spaces flow—while the fine model \mathcal{G}_s^f fills in detailed geometry through structure-guided denoising. This hierarchical decomposition enables our model to reason about both global coherence (room-to-room relationships) and local detail (furniture, architectural elements) simultaneously. This design differs from prior coarse-to-fine approaches like LT3SD [39], which uses a latent tree representation to encode both lower-frequency geometry and higher-frequency details in a multi-resolution hierarchy. While LT3SD models the same scene content at different resolutions, our method explicitly separates layout reasoning from detail generation, with each stage operating at a different semantic level. This hierarchical decomposition better aligns with the natural structure of indoor environments—where room arrangements constrain local details—enabling more coherent generation of large-scale scenes through explicit modeling of spatial relationships at multiple scales.

A.2. Scene-Friendly SLAT

Our occlusion-aware SLAT is specifically designed to handle the complex visibility patterns in indoor scenes, where walls, furniture, and architectural elements create extensive self-occlusion. Vanilla TRELLIS [65] was designed for single objects: it samples 150 views around an object and extracts pixel-wise DINOv2 features [43] for each view. For every pixel, TRELLIS casts a ray and aggregates features along all voxels intersected by the ray, regardless of visibility. This approach works well for isolated objects but fails in scene contexts—features from a visible surface (*e.g.*, a wall) get incorrectly projected onto occluded voxels behind it (*e.g.*, furniture in another room), causing severe artifacts and inconsistent appearance generation.

In WorldGrow, we introduce an occlusion-aware visual feature aggregation strategy to address this issue. To compute the sparse voxel features $\mathbf{f} = \{(\mathbf{f}_i, \mathbf{p}_i)\}_{i=1}^L$, each voxel center \mathbf{p}_i is projected onto multiple camera views, where we compute binary visibility masks $\{M_v\}$ using depth test-

ing. The occlusion-aware feature \mathbf{f}_i is then computed by averaging DINOv2 features from only the views where \mathbf{p}_i is visible according to M_v . This ensures that each voxel only receives features from views where it is actually observable, preventing feature contamination across occluded surfaces.

This occlusion-aware lifting significantly improves representation quality when integrated into the TRELLIS encoder \mathcal{E} . Consider walls with distinct textures on opposite sides—the original method would incorrectly blend features from both sides, while our approach preserves these distinctions, resulting in sharper boundaries and more accurate appearance modeling.

To fully leverage these improved features, we retrain the SLAT decoder \mathcal{D} on scene-scale data with occlusion-aware features \mathbf{f} . This retraining adapts the decoder to the unique challenges of scene generation, improving photorealism, preserving geometric boundaries, and ensuring coherent appearance across block transitions—critical capabilities for high-quality unbounded scene synthesis.

B. Implementation Details

B.1. Datasets

We curate a high-quality subset from 3D-FRONT [16] by filtering out houses with mesh penetration (1,971), incorrect furniture placement (1,232), small layouts (324), sparse furnishing (456), and other anomalies (585), yielding 3,072 clean houses. We additionally select 353 houses with minor issues for manual refinement. From these datasets, we extract 100K fine and 30K coarse blocks from the filtered houses, plus 20K fine and 8K coarse blocks from the manually refined subset. Our models are first trained on the larger filtered set, then fine-tuned on the curated subset for improved quality.

For block extraction, we use a standard house height of $h \approx 3m$. Fine blocks are cubes with width $w^f = h$, while coarse blocks span $w^c = 2h$ (covering a 2×2 grid of fine blocks), both maintaining height h .

To evaluate generalization to outdoor scenes, we train WorldGrow on UrbanScene3D [35]. Among its seven cities, we select *Shanghai* for its architectural diversity and texture quality. We extract 10K fine and 3K coarse blocks, with each fine block covering 100m—suitable for long-range outdoor synthesis.

B.2. Text Prompts

We use the following prompt for indoor scene generation:

A photorealistic 3D house mesh with contemporary architectural style, featuring clean lines and a balanced mix of natural and industrial materials. Use large windows for natural lighting. Prioritize smooth mesh topology and modular components

for adaptability, maintaining a cohesive modern aesthetic.

For outdoor scenes:

A realistic 3D urban street scene in daylight, featuring modern buildings, parked cars, street lamps and sidewalks. The environment should be detailed and clean.

Both prompts were generated using GPT-4 [42] and encoded with a frozen CLIP [45] text encoder, following TRELLIS-text-xlarge’s pipeline.

C. Experiment Details

C.1. Metrics

We implement MMD, COV, and 1-NNA following NFD [50]. For FID [20], we diverge from LT3SD’s and TRELLIS’s image-based evaluation, which renders views and computes FID on 2D images. We also compute FID on 3D mesh data using PointNet++ [44] features, ensuring fair comparison across all methods.

C.2. Compared Methods

We compare WorldGrow with representative scene-scale generators spanning image lifting, block-wise extrapolation, diffusion-based volumetric synthesis, and TRELLIS-family baselines. Text2Room [22] reconstructs 3D scenes by lifting multi-view 2D images via depth estimation and iterative fusion. BlockFusion [64] grows scenes block-by-block through latent tri-plane extrapolation. DiffInD-Scene [25] employs a cascaded diffusion pipeline to produce room-level TSDFs that are fused locally. Vanilla TRELLIS [65] serves as our base model; we also report a fine-tuned variant trained on our curated 3D-FRONT subset, denoted TRELLIS[†]. SynCity [14] is a training-free TRELLIS variant that couples TRELLIS geometry with 2D generators for large-scale synthesis. We exclude LT3SD [39] due to the absence of released checkpoints and training details necessary for reproducible evaluation. We also omit WonderWorld, which optimizes from a small set of given 2D views and does not produce a globally consistent 3D scene.

D. Additional Results

D.1. Outdoor 3D Scene Generation

To assess outdoor performance, we train WorldGrow on *UrbanScene3D* [35]. Among its seven cities, we select *Shanghai* for its architectural diversity and texture quality. As shown in Fig. 8 and Table 6, WorldGrow attains comparable or better geometric statistics than SynCity (lower MMD and 1-NNA; higher COV) and substantially improves perceptual quality (FID: 23.49 vs. 93.45), indicating that our

Method	MMD($\times 10^2$) \downarrow		COV(%) \uparrow		1-NNA(%) \downarrow		FID \downarrow
	CD	EMD	CD	EMD	CD	EMD	
SynCity	0.42	6.78	29.00	34.80	95.30	90.00	93.45
Ours	0.41	6.35	41.80	44.80	81.30	84.40	23.49

Table 6. Outdoor/urban scene generation on *UrbanScene3D* [35]. We train WorldGrow on the *Shanghai* split using 10K fine and 3K coarse blocks (each fine block covers 100 m). Even with this small training set, WorldGrow achieves markedly better coverage and perceptual quality than SynCity.

coarse-to-fine refinement and masked conditioning remain effective in large-scale urban layouts.

Given the limited size of the training subset, these results should be viewed as preliminary but encouraging rather than a conclusive benchmark. We expect further gains from larger and more diverse outdoor datasets, *e.g.*, city-scale aerial (drone or satellite) and street-level captures—as well as highly diverse synthetic assets from procedural systems such as Infinigen [46].