WorldGrow: Generating Infinite 3D World

Sikuang Li[1*†], Chen Yang[2*], Jiemin Fang[2 ✉‡], Taoran Yi[3†], Jia Lu[3†], Jiazhong Cen[1†], Lingxi Xie[2], Wei Shen[1], Qi Tian[2 ✉]

[1]MoE Key Lab of Artificial Intelligence, School of Computer Science, Shanghai Jiao Tong University
[2]Huawei Inc.
[3]Huazhong University of Science and Technology
{uranusits, jiazhongcen, wei.shen}@sjtu.edu.cn, {chenyang.res, jaminfong, 198808xc}@gmail.com,
{taoranyi, jialu2023}@hust.edu.cn, tian.qi1@huawei.com

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, limiting 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 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.

Code — https://github.com/world-grow/WorldGrow
Project Page — https://world-grow.github.io/
Extended Version — https://arxiv.org/abs/2510.21682

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 (Xie et al. 2024; Li et al. 2024a), 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 (Yu et al. 2024; Chung et al. 2023; Engstler et al. 2025; World Labs 2025) relies on pre-trained 2D diffusion models (Rombach et al. 2022; Black Forest Labs 2024; Google DeepMind 2025) to generate images, which are then "lifted" to 3D scenes using camera poses, depth maps (Ranftl et al. 2022; Birkl, Wofk, and Müller 2023), or image-to-3D models (Xiang et al. 2025). 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 (Wu et al. 2024; Meng et al. 2025; Lee, Han, and Chang 2025) attempts to directly predict 3D representations (*e.g.*, triplanes (Chan et al. 2022; Wu et al. 2024), UDFs (Chibane, Mir, and Pons-Moll 2020; Yang, Chen, and Kumari 2023; Meng et al. 2025), global latents (Chou, Bahat, and Heide 2023)) 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 (Fu et al. 2021a; Lin et al. 2022). Recent powerful 3D generation models (Xiang et al. 2025; Zhang et al. 2024; Li et al. 2024b; Zeng et al. 2022), empowered by large-scale training data (Deitke et al. 2023), 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 textu-

*Equal contributions.
†Work done during internship at Huawei.
‡Project lead.
✉Corresponding authors.

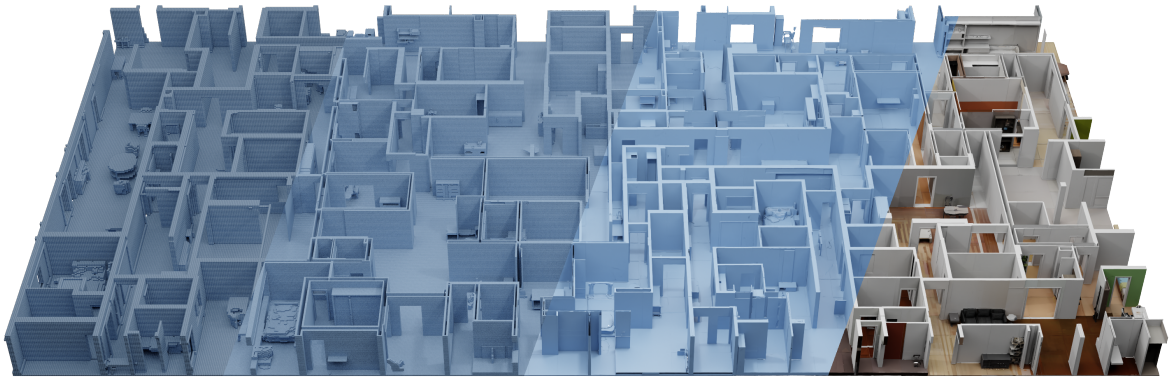

Figure 1: We introduce WorldGrow, a novel framework for infinite 3D scene generation via block-wise synthesis and growth, with coarse-to-fine refinement. Starting from a single seed block, WorldGrow progressively generates large-scale 3D scenes with coherent geometry and photorealistic appearance. The example showcases an indoor scene covering $1800\,\mathrm{m}^2$. From left to right: coarse voxel representation, fine voxel refinement, mesh reconstruction, and textured mesh rendering.

ral 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 (Chan et al. 2022) and 3D Gaussian Splatting (3DGS)(Kerbl et al. 2023), a number of feed-forward models(Hong et al. 2024b; Wei et al. 2024; Lan et al. 2024; Hong et al. 2024a; Zou et al. 2024; Tang et al. 2024; Xiang et al. 2025; Wu et al. 2025) have been developed to directly synthesize 3D content from a single image.

2.2 3D Unbounded Scene Generation

More recent efforts toward scene generation and even unbounded scene generation aim to produce 3D content that can be extended infinitely in all directions. BlockFusion (Wu et al. 2024) 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) (Chibane, Mir, and Pons-Moll 2020; Yang, Chen, and Kumari 2023; Meng et al. 2025) or vector-set latents (Lee, Han, and Chang 2025) 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 (Chen et al. 2024; Yang et al. 2024) to produce realistic surface appearances. SynCity (Engstler et al. 2025) proposes a training-free pipeline that divides a scene into grids, generates descriptive captions for each grid using Large Language Models (LLMs) (OpenAI et al. 2024), synthesizes images via text-to-image diffusion models (Black Forest Labs 2024), 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.

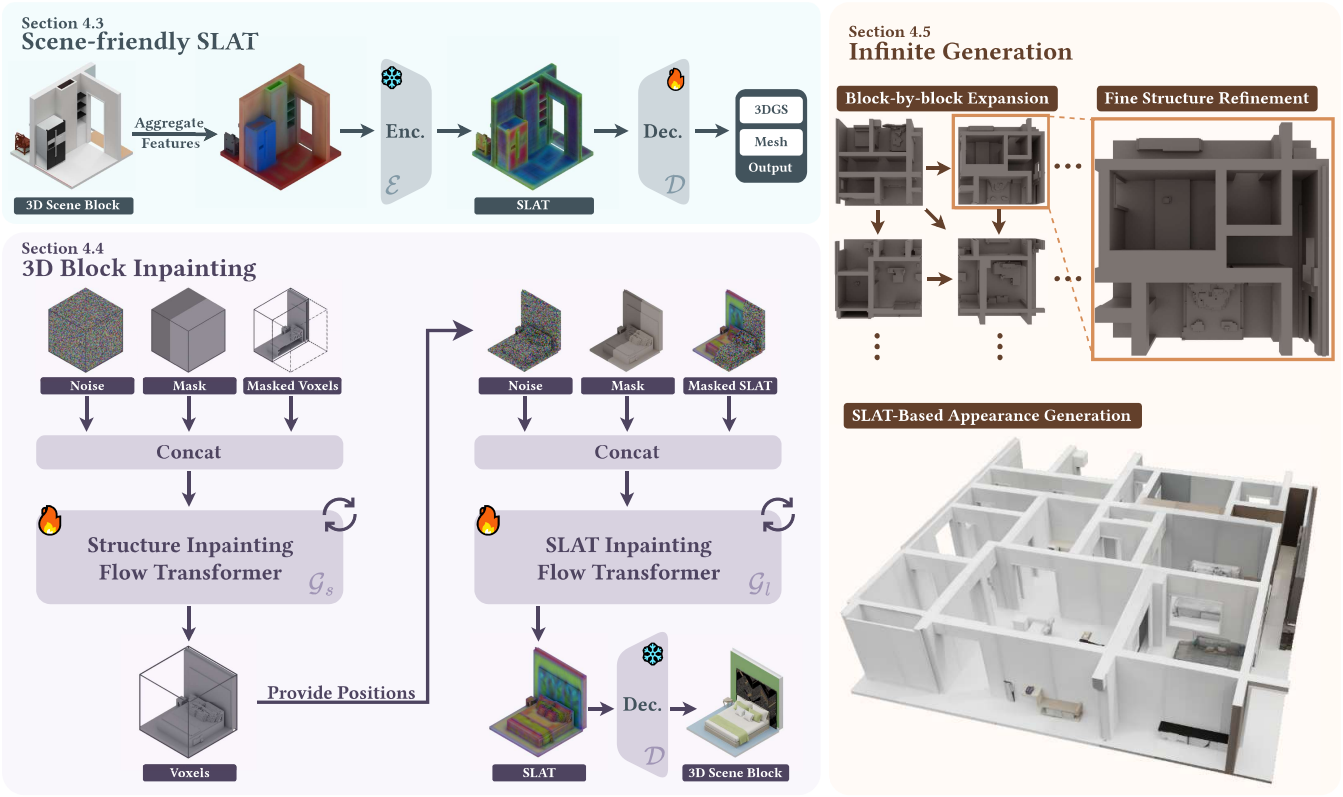

Figure 2: Overview of WorldGrow. Our goal is to generate infinite 3D scenes through modular, block-by-block synthesis. We begin by curating high-quality scene blocks and adapting SLAT to better model structured 3D context. A 3D block inpainting module enables spatially coherent extension, while a coarse-to-fine 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.

3 Preliminary: TRELLIS

TRELLIS (Xiang et al. 2025), foundational to our work, is a text/image-conditioned 3D generation model, denoising features in a sparse 3D latent space. We follow the mathematical symbol defination from TRELLIS. TRELLIS represents 3D objects via structured latents (SLATs): $\mathbf{z} = \{(\mathbf{z}_i, \mathbf{p}_i)\}_{i=1}^{L}$, where $\mathbf{z}_i \in \mathbb{R}^C$ is a latent feature at position $\mathbf{p}_i$ in a 3D grid, with L surface voxels. A Transformer-based VAE encodes sparse voxel features $\mathbf{f}$ into $\mathbf{z}$ and decodes $\mathbf{z}$ into 3D representations (*e.g.*, 3D Gaussians or meshes). SLAT generation is a two-stage process: Stage 1 predicts active voxel positions $\{\mathbf{p}_i\}_{i=1}^{L}$ using structure generation $\mathcal{G}_s$. Stage 2 recovers their latent features $\{\mathbf{z}_i\}_{i=1}^{L}$ using latent generation $\mathcal{G}_l$. Each stage employs a flow Transformer trained to reverse a noise addition process. Refer to our extended version and TRELLIS for more details.

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 (Deitke et al. 2023), are predominantly object-centric, consisting of isolated assets without spatial continuity. TRELLIS (Xiang et al. 2025) 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)

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

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, 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, and 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* blocks[1]. Coarse blocks are defined 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 miss-

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

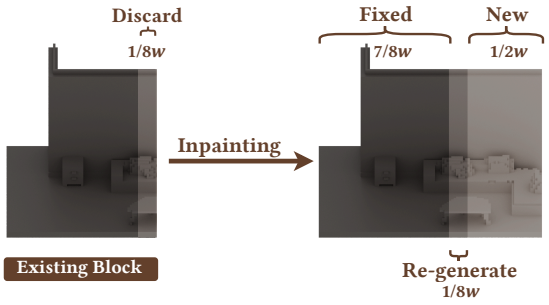

Figure 4: 1D illustration of our block-by-block expansion. Existing block's $[1/2w,\ 7/8w)$ area is used as context for inpainting the next block. Thus, the final region $[7/8w,\ w)$ is discarded and then re-generated during expansion.

ing 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:

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

where $\boldsymbol{\ell}_m^{(0)} = \boldsymbol{\ell}^{(0)} \otimes (1 - m)$ is the latent code masked, $\otimes$ denotes the Hadamard product, and $(\mathcal{G}, \boldsymbol{\ell}, m)$ 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.

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 (Meng et al. 2022). 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:

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

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, bridging global layout and fine-scale realism in a structure-aware manner.

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 (Fu et al. 2021a,b). 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 (Lin et al. 2022) in Fig. 6. Please refer to the extended version for details.

Implementation Details. We utilize a text-conditioned TRELLIS-XL model (Xiang et al. 2025) for 3D block inpainting, where the conditioning text consists of a fixed generic scene description generated by a large language model (OpenAI et al. 2024). During inference 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. Detailed training settings can be found in our extended version.

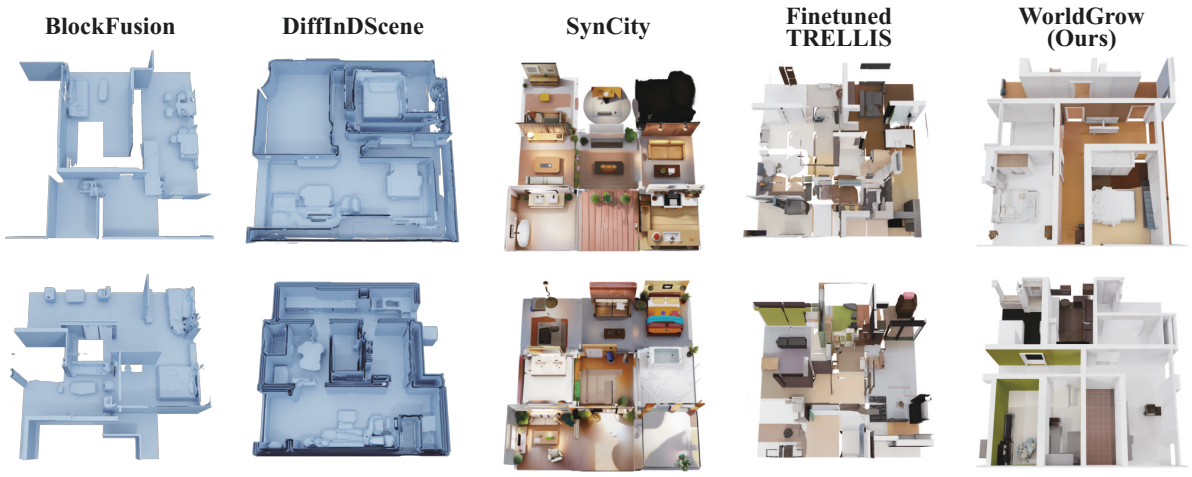

Figure 5: Qualitative comparison of indoor scene generation. We compare our method with state-of-the-art infinite scene generation approaches, indoor house generation methods and our baseline TRELLIS. WorldGrow produces high-resolution, continuous indoor scenes with realistic and coherent textures.

Metrics. We evaluate our method across two aspects: scene block generation and full scene synthesis. For block generation, we report three standard distribution-based metrics MMD, COV, and 1-NNA computed using both CD and EMD. We additionally adopt the perceptual FID (Heusel et al. 2017) with PointNet++ (Qi et al. 2017; Nichol et al. 2022). We provide extra visual quality metrics in the extended version. For full-scene synthesis, 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 generation methods, including BlockFusion (Wu et al. 2024) and SynCity (Engstler et al. 2025). Additionally, we evaluate against scene-scale generation baselines: Text2Room (Höllein et al. 2023) and DiffInDScene (Ju et al. 2024). The original and fine-tuned TRELLIS (Xiang et al. 2025) is included as a foundational baseline.

Method	MMD($\times 10^2$)↓		COV(%)↑		1-NNA(%)↓		FID↓
	CD	EMD	CD	EMD	CD	EMD	
DiffInDScene	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.

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 the observations. WorldGrow achieves SOTA performance across all geometry metrics, demonstrating superior connectivity and structural coherence.

Full Scene Generation. We conduct a human preference study following BlockFusion (Wu et al. 2024). We ask participants to evaluate structure plausibility (SP), geometry detail (GD), and appearance fidelity (AF) for indoor scenes, with an additional criterion of continuity (CO) for unbounded scenes. As shown in Table 2, 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.

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 and unfiltered 3D-FRONT data. As shown in Fig. 7, the unfiltered data results in object interpenetration and implausible arrangements, while

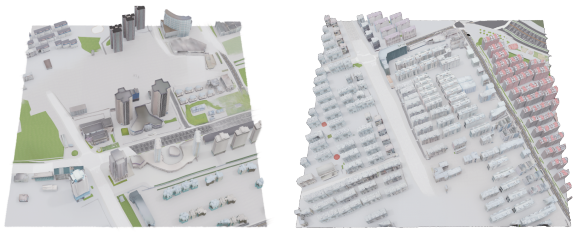

Figure 6: Infinite outdoor 3D scene generation by WorldGrow. Our method synthesizes diverse scenes such as urban streetscapes with plausible layouts, coherent suburban neighborhoods with consistent styles, showing WorldGrow's ability to be adapted to various domains.

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 2: Average of human preference scores (1–5).

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 3, 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.

Occ. Aware	Retrain $\mathcal{D}$	LPIPS ↓	PSNR ↑	SSIM ↑
✗	✗	0.0741	23.17	0.9273
✓	✗	0.0850	22.23	0.9046
✗	✓	0.0491	25.84	0.9531
✓	✓	**0.0311**	**31.32**	**0.9705**

Table 3: Ablation study about components of scene-friendly SLAT. Occ. Aware means occlusion aware feature aggregation and Retrain $\mathcal{D}$ is retraining VAE's decoder.

Coarse-to-Fine Generation. Here, we validate our coarse-to-fine generation strategy by comparing against direct fine-scale generation. As shown in Fig. 7, direct fine generation

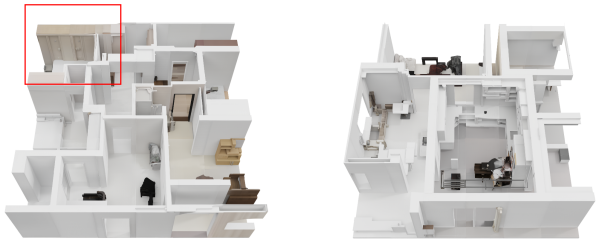

Figure 7: Ablation study on key components of WorldGrow. Left: Without Data Curation, the generated wardrobe intersects with multiple walls, indicating poor spatial alignment. Right: Without Coarse-to-Fine generation, the global furniture layout becomes cluttered and less coherent.

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.

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 limitations can be solved in future via multi-level generation strategies, larger-scale datasets, and enhanced LLM-generated captions .

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.

Acknowledgements

This work was supported by the NSFC under Grant 62322604 and 62576207.

References

Birkl, R.; Wofk, D.; and Müller, M. 2023. MiDaS v3.1 – A Model Zoo for Robust Monocular Relative Depth Estimation. *arXiv preprint arXiv:2307.14460*.

Black Forest Labs. 2024. FLUX. https://github.com/black-forest-labs/flux. Accessed: 2025-06-26.

Chan, E. R.; Lin, C. Z.; Chan, M. A.; Nagano, K.; Pan, B.; De Mello, S.; Gallo, O.; Guibas, L. J.; Tremblay, J.; Khamis, S.; Karras, T.; and Wetzstein, G. 2022. Efficient Geometry-Aware 3D Generative Adversarial Networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 16123–16133.

Chen, D. Z.; Li, H.; Lee, H.-Y.; Tulyakov, S.; and Nießner, M. 2024. 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)*, 21081–21091.

Chibane, J.; Mir, A.; and Pons-Moll, G. 2020. Neural unsigned distance fields for implicit function learning. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS '20. Red Hook, NY, USA: Curran Associates Inc. ISBN 9781713829546.

Chou, G.; Bahat, Y.; and Heide, F. 2023. Diffusion-SDF: Conditional Generative Modeling of Signed Distance Functions. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2262–2272.

Chung, J.; Lee, S.; Nam, H.; Lee, J.; and Lee, K. M. 2023. LucidDreamer: Domain-free Generation of 3D Gaussian Splatting Scenes. *arXiv preprint arXiv:2311.13384*.

Deitke, M.; Liu, R.; Wallingford, M.; Ngo, H.; Michel, O.; Kusupati, A.; Fan, A.; Laforte, C.; Voleti, V.; Gadre, S. Y.; VanderBilt, E.; Kembhavi, A.; Vondrick, C.; Gkioxari, G.; Ehsani, K.; Schmidt, L.; and Farhadi, A. 2023. Objaverse-XL: A Universe of 10M+ 3D Objects. In *Thirty-seventh Conference on Neural Information Processing Systems Datasets and Benchmarks Track*.

Engstler, P.; Shtedritski, A.; Laina, I.; Rupprecht, C.; and Vedaldi, A. 2025. SynCity: Training-Free Generation of 3D Worlds. arXiv:2503.16420.

Fu, H.; Cai, B.; Gao, L.; Zhang, L.-X.; Wang, J.; Li, C.; Zeng, Q.; Sun, C.; Jia, R.; Zhao, B.; and Zhang, H. 2021a. 3D-FRONT: 3D Furnished Rooms With layOuts and semaNTics. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 10933–10942.

Fu, H.; Jia, R.; Gao, L.; Gong, M.; Zhao, B.; Maybank, S.; and Tao, D. 2021b. 3D-FUTURE: 3D Furniture Shape with TextURE. *Int. J. Comput. Vision*, 129(12): 3313–3337.

Google DeepMind. 2025. Genie 3: A new frontier for world models. DeepMind Blog.

Heusel, M.; Ramsauer, H.; Unterthiner, T.; Nessler, B.; and Hochreiter, S. 2017. GANs Trained by a Two Time-Scale Update Rule Converge to a Local Nash Equilibrium. In Guyon, I.; Luxburg, U. V.; Bengio, S.; Wallach, H.; Fergus, R.; Vishwanathan, S.; and Garnett, R., eds., *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Höllein, L.; Cao, A.; Owens, A.; Johnson, J.; and Nießner, M. 2023. Text2Room: Extracting Textured 3D Meshes from 2D Text-to-Image Models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 7909–7920.

Hong, F.; Tang, J.; Cao, Z.; Shi, M.; Wu, T.; Chen, Z.; Wang, T.; Pan, L.; Lin, D.; and Liu, Z. 2024a. 3DTopia: Large Text-to-3D Generation Model with Hybrid Diffusion Priors. *arXiv preprint arXiv:2403.02234*.

Hong, Y.; Zhang, K.; Gu, J.; Bi, S.; Zhou, Y.; Liu, D.; Liu, F.; Sunkavalli, K.; Bui, T.; and Tan, H. 2024b. LRM: Large Reconstruction Model for Single Image to 3D. In *The Twelfth International Conference on Learning Representations*.

Ju, X.; Huang, Z.; Li, Y.; Zhang, G.; Qiao, Y.; and Li, H. 2024. DiffInDScene: Diffusion-based High-Quality 3D Indoor Scene Generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 4526–4535.

Kerbl, B.; Kopanas, G.; Leimkühler, T.; and Drettakis, G. 2023. 3D Gaussian Splatting for Real-Time Radiance Field Rendering. *ACM Transactions on Graphics*, 42(4).

Lan, Y.; Hong, F.; Yang, S.; Zhou, S.; Meng, X.; Dai, B.; Pan, X.; and Loy, C. C. 2024. LN3Diff: Scalable Latent Neural Fields Diffusion for Speedy 3D Generation. In *ECCV*.

Lee, H.-H.; Han, Q.; and Chang, A. X. 2025. NuiScene: Exploring Efficient Generation of Unbounded Outdoor Scenes. arXiv:2503.16375.

Li, B.; Guo, J.; Liu, H.; Zou, Y.; Ding, Y.; Chen, X.; Zhu, H.; Tan, F.; Zhang, C.; Wang, T.; et al. 2024a. UniScene: Unified Occupancy-centric Driving Scene Generation. *arXiv preprint arXiv:2412.05435*.

Li, W.; Liu, J.; Yan, H.; Chen, R.; Liang, Y.; Chen, X.; Tan, P.; and Long, X. 2024b. CraftsMan3D: High-fidelity Mesh Generation with 3D Native Generation and Interactive Geometry Refiner. *arXiv preprint arXiv:2405.14979*.

Lin, L.; Liu, Y.; Hu, Y.; Yan, X.; Xie, K.; and Huang, H. 2022. 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*, 93–109. Berlin, Heidelberg: Springer-Verlag. ISBN 978-3-031-20073-1.

Meng, C.; He, Y.; Song, Y.; Song, J.; Wu, J.; Zhu, J.-Y.; and Ermon, S. 2022. SDEdit: Guided Image Synthesis and Editing with Stochastic Differential Equations. In *International Conference on Learning Representations*.

Meng, Q.; Li, L.; Nießner, M.; and Dai, A. 2025. LT3SD: Latent Trees for 3D Scene Diffusion. arXiv:2409.08215.

Nichol, A.; Jun, H.; Dhariwal, P.; Mishkin, P.; and Chen, M. 2022. Point-E: A System for Generating 3D Point Clouds from Complex Prompts. arXiv:2212.08751.

OpenAI; et al. 2024. GPT-4 Technical Report. arXiv:2303.08774.

Qi, C. R.; Yi, L.; Su, H.; and Guibas, L. J. 2017. PointNet++: deep hierarchical feature learning on point sets in a metric space. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, 5105–5114. Red Hook, NY, USA: Curran Associates Inc. ISBN 9781510860964.

Ranftl, R.; Lasinger, K.; Hafner, D.; Schindler, K.; and Koltun, V. 2022. Towards Robust Monocular Depth Estimation: Mixing Datasets for Zero-Shot Cross-Dataset Transfer. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(3).

Rombach, R.; Blattmann, A.; Lorenz, D.; Esser, P.; and Ommer, B. 2022. High-Resolution Image Synthesis With Latent Diffusion Models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 10684–10695.

Tang, J.; Ren, J.; Zhou, H.; Liu, Z.; and Zeng, G. 2024. DreamGaussian: Generative Gaussian Splatting for Efficient 3D Content Creation. In *The Twelfth International Conference on Learning Representations*.

Wei, X.; Zhang, K.; Bi, S.; Tan, H.; Luan, F.; Deschaintre, V.; Sunkavalli, K.; Su, H.; and Xu, Z. 2024. MeshLRM: Large Reconstruction Model for High-Quality Mesh. *arXiv preprint arXiv:2404.12385*.

World Labs. 2025. Marble. Product site.

Wu, G.; Fang, J.; Yang, C.; Li, S.; Yi, T.; Lu, J.; Zhou, Z.; Cen, J.; Xie, L.; Zhang, X.; et al. 2025. UniLat3D: Geometry-Appearance Unified Latents for Single-Stage 3D Generation. *arXiv preprint arXiv:2509.25079*.

Wu, Z.; Li, Y.; Yan, H.; Shang, T.; Sun, W.; Wang, S.; Cui, R.; Liu, W.; Sato, H.; Li, H.; and Ji, P. 2024. BlockFusion: Expandable 3D Scene Generation using Latent Triplane Extrapolation. *ACM Trans. Graph.*, 43(4).

Xiang, J.; Lv, Z.; Xu, S.; Deng, Y.; Wang, R.; Zhang, B.; Chen, D.; Tong, X.; and Yang, J. 2025. Structured 3D Latents for Scalable and Versatile 3D Generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Xie, H.; Chen, Z.; Hong, F.; and Liu, Z. 2024. CityDreamer: Compositional Generative Model of Unbounded 3D Cities. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 9666–9675.

Yang, B.; Dong, W.; Ma, L.; Hu, W.; Liu, X.; Cui, Z.; and Ma, Y. 2024. DreamSpace: Dreaming Your Room Space with Text-Driven Panoramic Texture Propagation. In *2024 IEEE Conference Virtual Reality and 3D User Interfaces (VR)*, 650–660.

Yang, Z.; Chen, Y.; and Kumari, S. 2023. TUDF-NeRF: Generalizable Neural Radiance Field via Truncated Unsigned Distance Field. In *2023 Asia-Pacific Conference on Image Processing, Electronics and Computers (IPEC)*, 366–371.

Yu, H.-X.; Duan, H.; Hur, J.; Sargent, K.; Rubinstein, M.; Freeman, W. T.; Cole, F.; Sun, D.; Snavely, N.; Wu, J.; and Herrmann, C. 2024. WonderJourney: Going from Anywhere to Everywhere. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 6658–6667.

Zeng, X.; Vahdat, A.; Williams, F.; Gojcic, Z.; Litany, O.; Fidler, S.; and Kreis, K. 2022. LION: Latent Point Diffusion Models for 3D Shape Generation. In Oh, A. H.; Agarwal, A.; Belgrave, D.; and Cho, K., eds., *Advances in Neural Information Processing Systems*.

Zhang, L.; Wang, Z.; Zhang, Q.; Qiu, Q.; Pang, A.; Jiang, H.; Yang, W.; Xu, L.; and Yu, J. 2024. CLAY: A Controllable Large-scale Generative Model for Creating High-quality 3D Assets. *ACM Trans. Graph.*, 43(4).

Zou, Z.-X.; Yu, Z.; Guo, Y.-C.; Li, Y.; Liang, D.; Cao, Y.-P.; and Zhang, S.-H. 2024. 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)*, 10324–10335.