WORLDRECON: GENERATIVE MULTI-VIEW MESH RECONSTRUCTION FOR LARGE-SCALE 3D WORLD

Anonymous authors
Paper under double-blind review

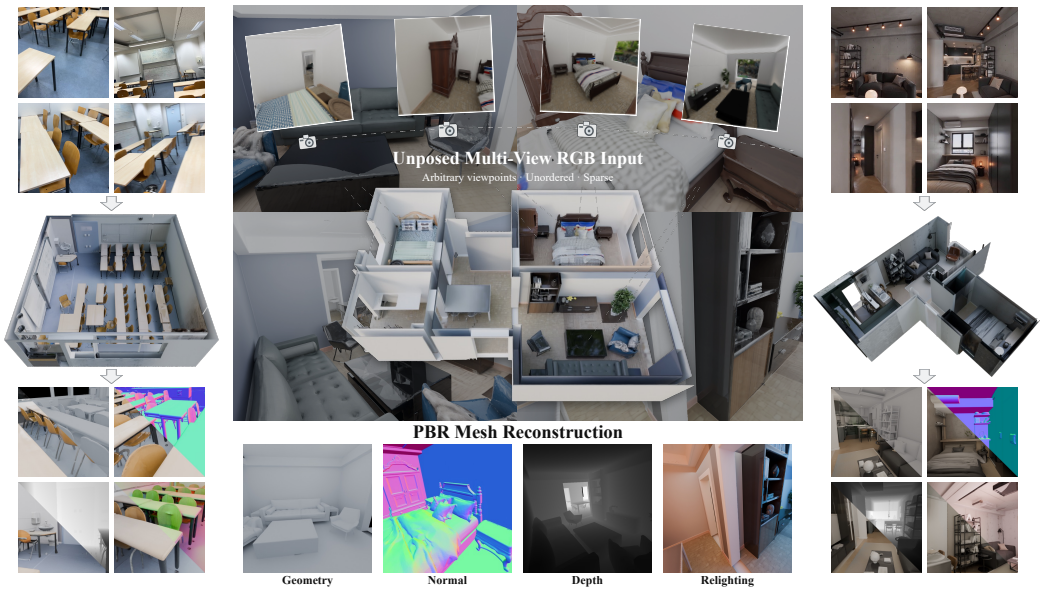

Figure 1: **WorldRecon Overview.** Given unposed multi-view RGB images of an indoor scene, WorldRecon reconstructs a complete scene-scale PBR mesh block-by-block via a native 3D generative prior at arbitrary scene scales, yielding high-fidelity geometry, depth, normal, and texture. From left to right, we demonstrate WorldRecon on real-world captures, synthetic images, and frames from generated videos, highlighting generalization across diverse input domains.

ABSTRACT

Reconstructing complete, high-quality meshes of large-scale scenes from unposed multi-view images requires faithful geometric grounding while plausibly completing surfaces that are unobserved or only sparsely captured. We present WorldRecon, a generative multi-view reconstruction framework for recovering complete, large-scale 3D scene meshes from unposed RGB images. WorldRecon uses a coarse, noisy global point cloud and camera poses estimated from the input images by a reconstruction model as a global scaffold for whole-scene generation and partitions it into overlapping spatial blocks. To robustly leverage these imperfect geometric priors, we equip the native 3D generative model TRELLIS.2 with a dedicated reconstruction-based conditioning architecture that generates each block from its local point-cloud crop and block-relevant image observations, producing sparse structure, detailed geometry, and PBR materials. To maintain scene-level coherence, we introduce neighbour-aware latent inpainting and a causal world-generation procedure that propagates structural, geometric, and material context across overlapping blocks. Together, these designs scale multi-view generative reconstruction to large-scale scenes, yielding coherent, unified meshes. Extensive experiments demonstrate that WorldRecon achieves state-of-the-art reconstruction performance, reducing Chamfer distance by 14.9% and 43.1% while improving F-score by 17.3% and 12.0% over the strongest baselines on synthetic and real-world indoor scenes, respectively, thereby demonstrating faithful and complete generative reconstruction as well as strong sim-to-real generalization.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

1 INTRODUCTION

Recovering a large-scale scene from unposed RGB images is a fundamental problem in computer vision, with broad applications in virtual and augmented reality, embodied intelligence, and digital content production. Reconstruction-based methods Wang et al. (2024); Leroy et al. (2024); Wang et al. (2026); Barron et al. (2022); Yu et al. (2022); Kerbl et al. (2023) recover scene geometry from visual observations, but often leave occluded regions incomplete and produce noisy geometry that is difficult to convert into clean meshes with detailed surfaces and materials.

Generative priors improve reconstruction completeness by inferring plausible geometry and appearance beyond the observations. Building on native 3D generative models Xiang et al. (2025; 2026); Zhang et al. (2024), recent multi-view extensions Chang et al. (2025); Huang et al. (2026) directly reconstruct structured 3D assets from unposed RGB images. Extending these methods to scene scale, however, remains challenging as the spatial extent and geometric complexity of an environment exceed the capacity of a single high-resolution generative representation.

Recent scene generation methods Chen et al. (2025); Wu et al. (2024b); Meng et al. (2025); Li et al. (2026); Engstler et al. (2025) scale to substantially larger environments by decomposing a scene into overlapping spatial blocks and generating them jointly or progressively. However, they are typically conditioned on text or layouts and thus lack visual guidance and 3D spatial grounding. More closely related to our task, scene-level generative reconstruction methods represent large scenes using hierarchical voxel latents Ren et al. (2024) or overlapping scene chunks Schmid et al. (2026). However, they require accurate camera poses to project image features into a 3D feature volume, making reconstruction quality sensitive to pose errors.

In this work, we introduce WorldRecon, a generative multi-view framework for reconstructing large-scale scene meshes from unposed RGB images. We employ VGGT-Ω Wang et al. (2026) to estimate coarse camera poses and recover a global point-cloud scaffold, which we partition into overlapping blocks. Our reconstruction-based conditioning leverages local point-cloud crops and block-relevant image observations, enabling the generation of complete, high-fidelity mesh blocks with PBR materials from imperfect geometric and camera estimates. To promote consistency across blocks, we train neighbour-aware latent inpainting models that condition each new block on overlap latents from previously generated neighbours. At inference, we progressively expand the scene following a causal block order, propagating sparse structure, geometry, and material context through shared world representations. This enables neighbouring blocks to transition seamlessly in both geometry and appearance before being decoded and assembled into a unified scene mesh.

Extensive experiments on 3D-FRONT Fu et al. (2021) and ScanNet++ Yeshwanth et al. (2023) demonstrate that WorldRecon reconstructs globally accurate and complete indoor scenes with fine geometric and material details from unposed RGB images on both synthetic and real-world settings.

In summary, our main contributions are as follows:

- We present WorldRecon, a framework that integrates 3D reconstruction and generation to recover large-scale, high-fidelity scene meshes with PBR materials from unposed multi-view RGB images.
- We develop reconstruction-based block generation with neighbour-aware inpainting, producing faithful mesh blocks from local observations and imperfect geometric and camera estimates.
- We propose a causal world-generation pipeline that propagates sparse structure, geometry, and material context across overlapping blocks for scalable and seamless scene generation.
- Extensive experiments demonstrate state-of-the-art performance in large-scale mesh reconstruction from unposed multi-view images across synthetic and real-world indoor scenes.

2 RELATED WORK

Multi-View 3D Reconstruction. Traditional 3D reconstruction relies on Structure-from-Motion Schonberger & Frahm (2016); Pan et al. (2024) and Multi-View Stereo Furukawa & Hernández (2015); Schönberger et al. (2016) to recover camera poses and scene geometry through triangulation. Learning-based MVS methods Yao et al. (2018; 2019); Wang et al. (2021a) further improve quality and efficiency via differentiable cost volumes. More recently, pose-free models Wang

et al. (2024); Leroy et al. (2024); Wang et al. (2025; 2026) cast the pairwise reconstruction problem as a regression of pointmaps, jointly regressing cameras and dense geometry. Beyond discrete point clouds, Neural Radiance Fields Mildenhall et al. (2021); Wang et al. (2021c); Barron et al. (2022) reframe reconstruction as inverse volumetric rendering for photorealistic view synthesis. Neural implicit surface methods Wang et al. (2021b); Yariv et al. (2021); Yu et al. (2022) represent scenes as continuous signed-distance functions. 3D Gaussian Splatting Kerbl et al. (2023); Huang et al. (2024); Charatan et al. (2024) optimize scenes via differentiable rasterization for fast rendering. However, these methods suffer from incomplete reconstruction and tend to produce over-smooth, blurred details in occluded regions.

Generative Multi-View 3D Reconstruction. Generative priors are introduced in multi-view 3D reconstruction to mitigate reconstructive incompleteness. Early efforts Liu et al. (2024); Gao et al. (2024); Wu et al. (2024a); Guo et al. (2025); Xu et al. (2025) leverage 2D or video diffusion priors to synthesize novel views or depth maps, but still require per-scene optimization and suffer from cross-view inconsistency. Other works Jin et al. (2024); Tang et al. (2024b); Ziwen et al. (2025) regress target-view pixels or 3D Gaussians in a single feed-forward pass, but output unstructured primitives rather than meshes. Recently, native 3D generative models Xiang et al. (2025; 2026); Zhang et al. (2024) have enabled direct mesh generation from image conditioning. ReconViaGen Chang et al. (2025) and UniRecGen Huang et al. (2026) extend this paradigm to multi-view conditioning, but remain confined to the object level. SCube Ren et al. (2024) and GenRecon Schmid et al. (2026) further extend multi-view 3D reconstruction to the scene level, yet both rely on accurate camera poses to project image features into a 3D feature volume, which induces cross-view conflicts and degrades reconstruction quality under imperfect poses. In contrast, our method conditions on pose-free RGB images, recovering coherent scene mesh block-by-block via a native 3D generative prior.

Unbounded 3D Scene Generation. Extending 3D generation from objects to unbounded scenes has recently drawn increasing attention. Early efforts focus on bounded environments such as a single room, either retrieving assets from a fixed database Tang et al. (2024a); Wang et al. (2021d), or hallucinating RGB-D frames with 2D diffusion models under text or layout conditions Höllein et al. (2023); Schult et al. (2024). To scale beyond bounded rooms, recent works pursue unbounded scene generation via block-wise generation, where neighbouring blocks are either jointly denoised and seamlessly blended Chen et al. (2025), or progressively synthesized by an extrapolation prior Wu et al. (2024b); Meng et al. (2025); Li et al. (2026); Engstler et al. (2025). However, these methods are either unconditional or conditioned only on text or layout, lacking both visual guidance and 3D spatial grounding. Our method instead grounds the scene in a coarse predicted point cloud and conditions each block on a few multi-view images, scaling multi-view generative reconstruction to arbitrarily large scenes.

3 WorldRecon Method

Given an arbitrary number of uncalibrated RGB images $\{I_n\}_{n=1}^{N}$ of a large-scale indoor scene, our goal is to recover a complete, high-fidelity, scene-level 3D mesh endowed with PBR materials. Our generative multi-view reconstruction framework couples a reconstruction prior for global layout consistency with a generative prior for surface fidelity and completeness. To this end, we curate a block-level training dataset (Section 3.1). We finetune the object-level generative prior of TRELLIS.2 Xiang et al. (2026) into a block-level mesh generator conditioned on each block's point-cloud slice and multi-view RGBA images, together with a neighbour-aware 3D inpainting variant (Section 3.2). At inference, we first extract a global point cloud with VGGT-Ω Wang et al. (2026) as a coarse geometric layout cue and partition it into overlapping point cloud blocks. Starting from a seed block, we sequentially generate the remaining blocks across three generation stages, conditioning each block on its overlaps with already-generated neighbours (Section 3.3). The overall pipeline is shown in Figure 2. In this way, the point-cloud scaffold pins the scene to the observed geometry, while the block-wise generative prior fills in complete, high-fidelity surfaces.

3.1 Data Curation

Scene Slicing. To enable block-level scene generation, we first construct a structured, scalable 3D block dataset. We build on two synthetic indoor scene datasets, 3D-FRONT Fu et al. (2021) and SAGE-10k Xia et al. (2026), which together cover diverse layouts, furniture styles, and material

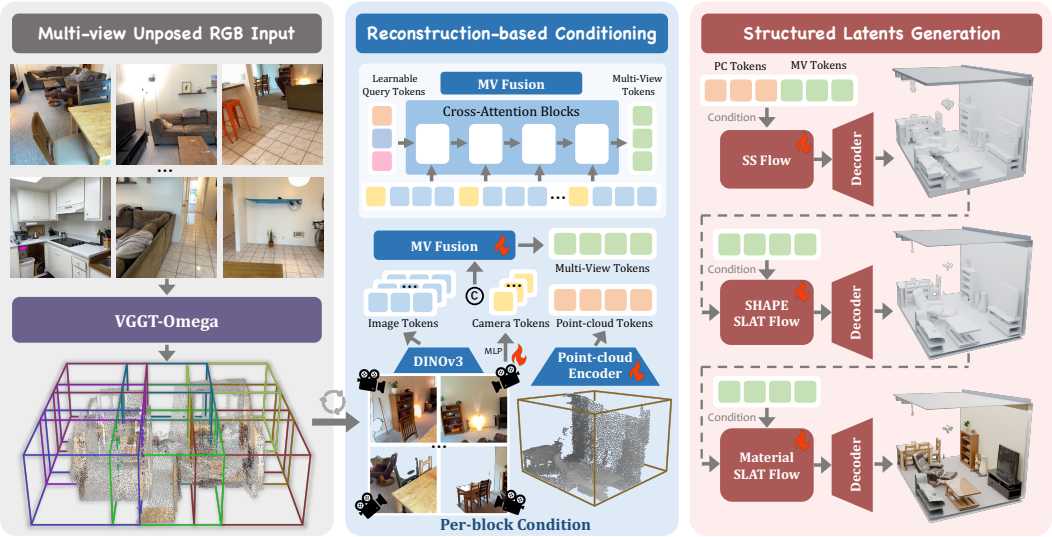

Figure 2: **WorldRecon Pipeline. Left:** Given unposed multi-view RGB images, VGGT-Ω reconstructs a global point cloud. The point cloud is divided into overlapping blocks, each paired with local RGBA images and predicted camera poses. **Middle:** For each block, DINOv3 image tokens and Plücker-ray camera tokens are fused into fixed-length multi-view tokens, while a point-cloud encoder extracts geometric tokens. **Right:** Three cascaded flow models progressively generate the sparse structure, detailed geometry, and PBR materials.

distributions. For each scene, we render RGBD views from randomised eye-level cameras under varied lighting. We then propose a scene-slicing strategy. The scene is partitioned into a regular grid of canonically oriented cubic blocks with side length ℓ, each covering the full vertical extent of the scene. This yields roughly 56k blocks from 6k 3D-FRONT scenes and 61k blocks from 10k SAGE-10k scenes, totalling around 117k training blocks. Each mesh block is normalised to canonical $[-0.5, 0.5]^3$ space.

Condition Extraction. Each block then yields a training tuple $(\mathcal{M}_b,\ \mathcal{P}_b,\ \{(I_v, K_v, R_v, t_v)\}_{v=1}^{16})$, denoting the block-local mesh, point cloud, RGBA renders, and camera poses, respectively. Every rendered RGBD view is back-projected into the block's canonical space, yielding a block-local camera pose (K_v, R_v, t_v). The pixels falling inside the block cube form the RGBA render I_v, with the in-cube mask as alpha. We retain the top-16 views per block by in-cube pixel coverage. For $\mathcal{P}_b$, we randomly sample from two complementary noisy sources during training: unprojected GT depth from a random subset of views with augmented noise, and a VGGT-Ω Wang et al. (2026) reconstruction from $\{I_v\}_{v=1}^{16}$, both downsampled to M points. This exposes the model to both controllable augmentation and the target inference distribution, keeping generation robust when only VGGT-Ω point clouds are available at test time.

3.2 Reconstruction-based Block Generation

Generative Prior. We build on TRELLIS.2 Xiang et al. (2026), an image-to-3D generation model that represents 3D asset as a sparse voxel structure termed *O-Voxel*, defined as a set of feature tuples $\mathbf{f} = \{(\mathbf{f}_i^{\text{geo}}, \mathbf{f}_i^{\text{mat}}, \mathbf{o}_i)\}_{i=1}^{L}$ over L active voxels on a regular G^3 grid, where $\mathbf{o}_i$ denotes the voxel position, $\mathbf{f}_i^{\text{geo}}$ encodes local geometric information, and $\mathbf{f}_i^{\text{mat}}$ encodes material properties. Generation cascades three DiT flow-matching stages: *Sparse Structure* predicts the active-voxel positions $\{\mathbf{o}_i\}$. *Geometry* predicts geometry latents $\mathbf{z}^{\text{geo}}$ within active voxels, which are decoded into $\{\mathbf{f}_i^{\text{geo}}\}$. *Material* synthesizes material latents $\mathbf{z}^{\text{mat}}$ aligned to $\mathbf{z}^{\text{geo}}$, decoded to $\{\mathbf{f}_i^{\text{mat}}\}$. We finetune all three stages from the pretrained object-level TRELLIS.2 for block-level scene generation, replacing its original single-image conditioning with our reconstruction-based conditioning as detailed below.

Reconstruction-Based Conditioning. We leverage the block-level conditions curated in Section 3.1 for controllable 3D generation. Multi-view images and their camera poses are jointly encoded into

a unified representation $\mathbf{c}^{\text{mv}}$ shared by all three stages. *Sparse Structure* stage additionally receives point-cloud tokens $\mathbf{c}^{\text{pc}}$ as a coarse geometric prior, giving $\mathbf{c} = [\mathbf{c}^{\text{mv}}, \mathbf{c}^{\text{pc}}]$. The resulting conditioning is injected into every cross-attention layer of the DiT.

Multi-View Image Condition. For each block, we randomly sample V views $\{I_v\}_{v=1}^{V}$ with their camera poses $\{(R_v, t_v, K_v)\}_{v=1}^{V}$. Each view is encoded by a frozen DINOv3 Siméoni et al. (2025) patch encoder into image tokens $\mathbf{c}_v^{\text{img}}$, while its camera is represented by five Plücker rays which are projected through an MLP into camera tokens $\mathbf{c}_v^{\text{cam}}$. Per-view camera tokens are prepended to the corresponding image patches, forming the multi-view context $\mathcal{S} = [\mathbf{c}_1^{\text{cam}}, \mathbf{c}_1^{\text{img}}, \ldots, \mathbf{c}_V^{\text{cam}}, \mathbf{c}_V^{\text{img}}]$. Starting from a set of learnable query tokens $\mathbf{q}_0$, we run N_t transformer cross-attention blocks that progressively fuse $\mathcal{S}$ into the fixed-length tokens:

$$\mathbf{q}_{l+1} = \text{CrossAttn}\big(Q(\mathbf{q}_l),\ K(\mathcal{S}),\ V(\mathcal{S})\big), \quad l \in \{0, 1, \ldots, N_t - 1\}, \tag{1}$$

yielding the multi-view condition $\mathbf{c}^{\text{mv}} = \mathbf{q}_{N_t}$.

Point Cloud Condition. Recovering geometry from RGB alone is inherently ambiguous, especially at scene scale where sparse views cover only partial structure. We therefore condition the Sparse Structure stage on a point cloud that supplies coarse spatial structure. Following the Hunyuan3D-Omni Team (2025) design, each point of $\mathcal{P}_b = \{\mathbf{p}_j\}_{j=1}^{M}$ is encoded by a Fourier positional embedder followed by a linear projection, and K learnable modality tokens are appended to form the point-cloud condition $\mathbf{c}^{\text{pc}}$. Point-cloud tokens are randomly dropped during training, allowing classifier-free guidance to support both point-free and point-conditioned inference.

Neighbour-Aware Latent Inpainting. At inference, the scene is extended block-by-block. Each new block only observes partial 3D context from its already-generated neighbours, yet it must produce a complete 3D representation that continues the surrounding geometry and appearance seamlessly. We formulate this as a 3D block inpainting task and train dedicated inpaint variants of all three denoisers. We augment each denoiser's native input with $C{+}1$ inpainting-specific channels, comprising a one-channel binary inpainting mask $\mathbf{m}$ and a C-channel masked known latent $\mathbf{x}_0 \odot (\mathbf{1} - \mathbf{m})$, where C denotes the latent feature dimension. The resulting hidden feature is:

$$\mathbf{h}_{\text{in}} = f_{\text{in}}\Big(\big[\,\mathbf{x}_t\,[, \mathbf{z}^{\text{geo}}],\ \mathbf{m},\ \mathbf{x}_0 \odot (\mathbf{1} - \mathbf{m})\,\big]\Big). \tag{2}$$

Here, $\mathbf{m}_i = 0$ marks known latent tokens supplied, whereas $\mathbf{m}_i = 1$ marks tokens to be freshly generated. $\mathbf{z}^{\text{geo}}$ denotes an optional *Geometry* latent included for the *Material* stage.

To retain the priors of our block-level generation models, we initialise each inpaint variant from its corresponding generation model. The original input projection is directly inherited, while the projection weights associated with the additional $C{+}1$ inpainting channels are zero-initialised. At the start of finetuning, the inpaint variant is thus functionally equivalent to the base generator, providing a stable initialisation for optimisation.

Training masks are sampled to emulate the overlap layouts arising at inference. Since all scenes fit within a single block along the vertical axis, we restrict masking to the XY plane. On each iteration, we sample the number of neighbour-supplied edges $E \in \{1, 2\}$ of the block, enforcing an *L-shape* constraint (one X-edge plus one Y-edge) when $E = 2$. The mask configurations cover the 1- and 2-neighbour cases encountered under our voxel-priority generation ordering (Section 3.3). The known margin along each edge is sampled between $3\ell/16$ and $\ell/4$. At inference, we shrink the raw overlap of width $\ell/4$ by one quarter, leaving a known margin of $3\ell/16$ and an editable transition band of width $\ell/16$. This design preserves sufficient neighbour context for cross-block consistency while allowing the model to adjust geometric and material transitions near block boundaries.

3.3 World Generation Pipeline

Given an arbitrary number of uncalibrated RGB images $\{I_n\}_{n=1}^{N}$, our inference pipeline generates a large-scale 3D mesh through block-based, neighbour-aware latent inpainting and overlap blending.

Block Preparation. We run VGGT-Ω Wang et al. (2026) to recover coarse camera parameters and a global point cloud from the input RGB images. After confidence filtering and outlier removal, we align and rescale the reconstruction to the canonical scene coordinates. We partition its XY bounding box into overlapping cubic blocks of side length ℓ, with adjacent blocks overlapping by

$\ell/4$. For each block, we retain only the in-block pixels of each input RGB image and consider images with at least 15% in-block pixel coverage as candidate views. We voxelise the corresponding visible points and select up to V views whose union maximises the block's voxel coverage. This produces the training-compatible tuple $(\hat{\mathcal{P}}_b, \{(I_v, \hat{K}_v, \hat{R}_v, \hat{t}_v)\}_{v=1}^{V})$, without requiring ground-truth geometry or calibrated cameras.

We determine a voxel-priority generation order based on block-wise voxel occupancy. Starting from the block with the highest occupancy, we greedily expand a connected frontier while ensuring that each block has at most two previously generated four-connected neighbours. This ordering matches the 1- and 2-neighbour masks used to train the block inpainters.

Sparse Structure World Generation. We maintain shared world-space *Sparse Structure* latent and occupancy. The seed latent is sampled by the generator, while subsequent blocks are inpainted from overlaps with generated neighbours. To ensure *Sparse Structure* continuity across block boundaries, we introduce a *trajectory-clamping* mechanism. At every denoising step t, the known overlap is restored to its forward-flow trajectory:

$$\mathbf{x}^{\text{ss}}_{t,\text{known}} = (1-t)\mathbf{x}^{\text{ss}}_{0,\text{known}} + t\boldsymbol{\epsilon}. \tag{3}$$

Here, $\mathbf{x}^{\text{ss}}_{0,\text{known}}$ is the clean latent extracted from the world-space latent. We then clamp the flow state $\mathbf{x}^{\text{ss}}_t$ predicted by the inpainter by replacing its known overlap:

$$\tilde{\mathbf{x}}^{\text{ss}}_t = \mathbf{m}^{\text{ss}} \odot \mathbf{x}^{\text{ss}}_t + (\mathbf{1} - \mathbf{m}^{\text{ss}}) \odot \mathbf{x}^{\text{ss}}_{t,\text{known}}. \tag{4}$$

This mechanism preserves the previously generated overlap while allowing the unknown region to evolve along the denoising trajectory. After denoising, the resulting clean latent $\mathbf{z}^{\text{ss}} = \tilde{\mathbf{x}}^{\text{ss}}_0$ is decoded into an occupancy grid $\mathbf{o}^{\text{ss}} = \mathcal{D}_{\text{ss}}(\mathbf{z}^{\text{ss}})$. Both are written only to unoccupied locations in the shared world-space tensors, which then provide context for subsequent blocks.

Geometry and Material World Generation. Given the completed world *Sparse Structure* latent, we process blocks in the same causal order. For each block, we generate or inpaint its *Geometry* latent, followed by its *Material* latent. To reduce block-boundary seams, each block's clean latent $\mathbf{z}^s_b$, where $s \in \{\text{geo}, \text{mat}\}$, is smoothly blended into the corresponding world latent $\mathbf{Z}^s_b$ immediately after generation. The blending weights are determined by the distance to the block boundaries. Since blocks overlap only in the XY plane, let $\mathcal{A} \subseteq \{x, y\}$ denote the overlapping axes, $\mathbf{u} = (x, y, z)$ the world-grid coordinate, and $\boldsymbol{\xi}_b(\mathbf{u}) \in \{0, \dots, G-1\}^3$ the corresponding block-local latent-voxel coordinate, where G is the latent-grid resolution along each axis. As adjacent blocks overlap by one quarter of the block, the overlap spans $G/4$ latent voxels. We define the linear feather weight as

$$\alpha_b(\mathbf{u}) = \prod_{a \in \mathcal{A}} \text{clip}\left(\frac{\min\{\xi_{b,a}(\mathbf{u}) + \frac{1}{2},\ G - \xi_{b,a}(\mathbf{u}) - \frac{1}{2}\}}{G/4}, 0, 1 \right). \tag{5}$$

This produces smooth linear transitions across seams and bilinear blending at block corners. Let $\mathbf{Z}^s_b(\mathbf{u})$ denote the world latent after integrating block b, and let $A_b(\mathbf{u})$ denote the accumulated weight, with $A_0(\mathbf{u}) = 0$. For each active world-grid location $\mathbf{u}$, we update them online as

$$\mathbf{Z}^s_b(\mathbf{u}) = \frac{A_{b-1}(\mathbf{u})\mathbf{Z}^s_{b-1}(\mathbf{u}) + \alpha_b(\mathbf{u})\mathbf{z}^s_b(\mathbf{u})}{A_{b-1}(\mathbf{u}) + \alpha_b(\mathbf{u})}, \qquad A_b(\mathbf{u}) = A_{b-1}(\mathbf{u}) + \alpha_b(\mathbf{u}). \tag{6}$$

Once the world latents are complete, we extract the aligned *Geometry* and *Material* latents for each block and decode them into block-local meshes with PBR materials. The block meshes are then transformed into the shared world frame and merged sequentially.

4 Experiments

4.1 Experimental Setup

Datasets. We evaluate on unseen scenes from two datasets. From 3D-FRONT Fu et al. (2021), we hold out 30 scenes to assess performance on synthetic data. From ScanNet++ Yeshwanth et al. (2023), we evaluate 30 scenes to assess out-of-domain real-world performance. Each scene provides 32 unposed RGB images as input and 50 held-out RGB, depth, and normal views for evaluation.

Table 1: **Comparisons on Synthetic Scenes.** We evaluate 3D geometry and 2D depth and normal metrics on 30 scenes from 3D-FRONT, which is synthetic and large-scale.

Method	3D Geometry			Coverage	Depth				Normal		
	CD↓	F-score↑	NC↑	Complete.↑	RMSE↓	AbsRel↓	LPIPS↓	CLIP↑	Angular↓	LPIPS↓	CLIP↑
VGGT-Ω+TSDF	0.2934	0.4596	0.5433	0.7363	0.5906	0.1548	0.4873	0.8679	20.9899	0.6188	0.8568
Murre	0.5295	0.2426	0.4729	0.4572	0.8847	0.2397	0.5273	0.8332	22.6896	0.6671	0.8542
MonoSDF	0.2558	0.4936	0.5730	0.9056	0.6052	0.1616	0.2566	0.9161	18.2223	0.3560	0.9062
FineRecon	0.2448	0.5301	0.6252	0.8126	0.5578	0.1490	0.3398	0.8906	17.6099	0.4419	0.8845
GenRecon	0.1861	0.6196	0.6128	0.9587	0.5348	0.1666	0.2548	0.9490	17.9822	0.3139	0.9543
WorldRecon (Ours)	**0.1583**	**0.7271**	**0.6839**	**0.9589**	**0.3910**	**0.1126**	**0.1966**	**0.9582**	**16.4044**	**0.2951**	**0.9601**

Table 2: **Comparisons on Real-world Scenes.** We evaluate 3D geometry and 2D depth and normal metrics on 30 scenes from Scannet++. Best results are **bold**, and second-best results are underlined.

Method	3D Geometry			Coverage	Depth				Normal		
	CD↓	F-score↑	NC↑	Complete.↑	RMSE↓	AbsRel↓	LPIPS↓	CLIP↑	Angular↓	LPIPS↓	CLIP↑
VGGT-Ω+TSDF	0.1121	0.7467	0.7036	0.8737	0.3984	0.1433	0.3907	0.8991	25.4852	0.5075	0.8888
Murre	0.1756	0.5640	0.6222	0.5567	0.6619	0.1863	0.5587	0.8690	28.4193	0.5642	0.8761
MonoSDF	0.1665	0.6224	0.6913	0.9670	0.6293	0.3043	0.2705	0.9273	23.7527	0.3861	0.9016
FineRecon	0.1259	0.6861	0.7226	0.9388	0.4405	0.1673	0.2707	0.9183	22.6830	0.4043	0.8886
GenRecon	0.1227	0.5901	0.5969	0.9722	0.6837	0.3048	0.3862	0.9291	31.5722	0.4711	0.9172
WorldRecon (Ours)	**0.0638**	**0.8363**	**0.7840**	**0.9918**	**0.3331**	**0.0992**	**0.2139**	**0.9522**	**20.0429**	**0.3631**	**0.9320**

Metrics. We evaluate reconstructed meshes in both 3D and 2D. In 3D, we measure geometric accuracy and coverage using Chamfer distance (CD), F-score (10 cm), and normal consistency (NC, thresholded at 20 cm). In 2D, we report geometric errors (RMSE, AbsRel, and angular normal error), computed over mutually valid pixels; perceptual and semantic metrics (LPIPS and CLIP) on depth and normal renderings; and completeness over valid reference pixels.

Baselines. We compare against five representative baselines spanning different reconstruction paradigms. ***VGGT-Ω*** Wang et al. (2026) predicts per-view depth maps and camera poses, then fuses them into a mesh via TSDF integration Newcombe et al. (2011) and Marching Cubes. All of the following methods consume its predictions as input. ***Murre*** Guo et al. (2025) conditions a large monocular depth estimator on the predicted point cloud and fuses the resulting depth into a mesh. ***MonoSDF*** Yu et al. (2022) optimizes a per-scene signed distance field guided by monocular depth and normal priors. ***FineRecon*** Stier et al. (2023) performs feed-forward TSDF regression conditioned on multi-view depth guidance. ***GenRecon*** Schmid et al. (2026) tiles the scene into overlapping chunks and projects multi-view image features into per-chunk 3D feature volumes.

Implementation details. We partition scenes into $4\,\mathrm{m}$ cubic blocks, each conditioned on up to 4 local views and an 8192-point cloud crop. Starting from the 512-resolution TRELLIS.2 pretrain model, we fully finetune the *Sparse Structure* and *Geometry* stages and apply LoRA Hu et al. (2021) to *Material* stage. At inference, we use a 12-step Heun sampler with guidance scales of 7.5, 7.5, and 3.0 and time-rescaling factors of 5.0, 3.0, and 3.0. All models are trained on 32 NVIDIA GPUs. Details are provided in Appendix.

4.2 Quantitative Results

Synthetic scene reconstruction. We evaluate 2D and 3D reconstruction performance on 3D-FRONT Fu et al. (2021) using 32 unposed RGB images. The evaluation set primarily comprises large-scale multi-room scenes, with an average of 7.1 rooms and $62.1\,\mathrm{m}^2$ of floor area. WorldRecon achieves the best overall performance, completing regions missing from the initial VGGT-Ω reconstruction. It substantially outperforms all baselines across the 3D geometry metrics while achieving the best results on all evaluated 2D depth and normal metrics, demonstrating accurate, consistent, and complete scene reconstruction.

Real-world scene reconstruction. We further evaluate generalization on ScanNet++ Yeshwanth et al. (2023). Despite being trained solely on synthetic scenes, WorldRecon reduces CD by 43.1% and improves F-score by 12.0% over the strongest baselines. Its rendered depth and normal maps also exhibit better geometric accuracy, perceptual similarity, and semantic alignment with the laser-scanned ground truth, demonstrating strong sim-to-real generalization.

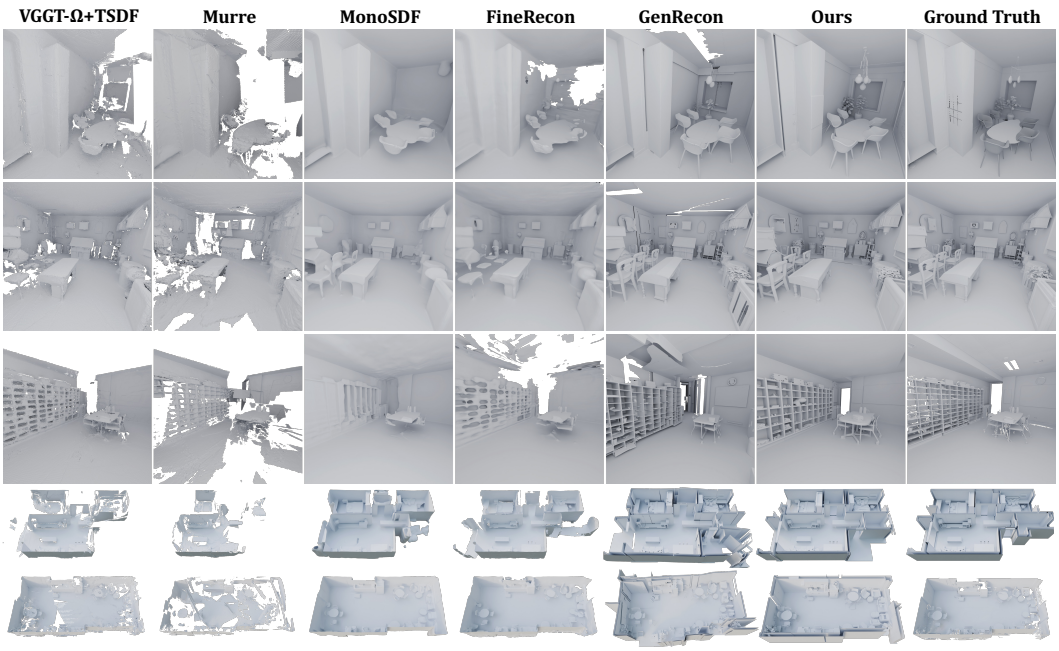

Figure 3: **Qualitative Comparison.** Mesh reconstruction results on evaluation scenes, including both synthetic and real-world examples. We show close-up views of interior details (top) and complete scene reconstructions (bottom). Compared with the baselines, WorldRecon produces more complete and geometrically consistent reconstructions with substantially fewer artifacts.

4.3 Qualitative Results

We present qualitative comparisons on both synthetic and real-world scenes in Figure 3. As a reconstruction-based approach, VGGT-Ω+TSDF recovers visible surfaces but leaves unobserved regions incomplete. Deviations in its predicted depths and camera poses can further propagate through TSDF fusion, resulting in noisy artifacts and distorted walls. Methods that use geometric predictions from VGGT-Ω, such as Murre and FineRecon, also exhibit artifacts and broken surfaces. GenRecon completes some unobserved regions; however, its feature-projection conditioning mechanism injects the distorted point cloud and inaccurate camera poses into the generative model, resulting in fragmented artifacts and distortions of the global scene layout. The optimisation-based MonoSDF completes missing regions and reduces geometric noise, but its over-smoothed surfaces lose fine geometric details. In comparison, WorldRecon produces more complete and geometrically coherent reconstructions, preserving sharper object boundaries, thin structures, and local details despite noise in the initial geometry and camera estimates.

Beyond geometric reconstruction, WorldRecon produces scene-level meshes with high-fidelity PBR materials. As illustrated in Figure 1, our method consistently reconstructs detailed geometry and texture from diverse input sources, including real-world captures, synthetic images, and frames from generated videos Seedance et al. (2026), demonstrating robust generalization across input domains. Additional qualitative results are provided in the Appendix. In particular, Figure S2 compares WorldRecon with an agentic baseline.

4.4 Ablation Study

We ablate the key components of WorldRecon. The ablation experiments are conducted on complete scenes generated from 15 3D-FRONT scenes and 15 ScanNet++ scenes. Starting from the vanilla pretrained TRELLIS.2 Xiang et al. (2026), we progressively add four components: reconstruction-based sparse-structure generation (*SS*), reconstruction-based geometry and material generation (*Geo+Mat*), neighbour-aware latent inpainting (*Inpaint*), and stage-wise world latent clamping and blending (*Stitch*). Quantitative results are reported in Table 3, while Figure 4 visualizes the corresponding effects on representative scene structures and local details.

Table 3: **Ablation Study.** We conduct quantitative ablation studies of the key components of WorldRecon on 30 scenes from 3D-FRONT and ScanNet++.

Components				3D Metrics			2D Metrics				
SS	Geo&Mat	Inpaint	Stitch	CD↓	F-score↑	NC↑	Complete.↑	MAE↓	AbsRel↓	Angular↓	RGB CLIP↑
×	×	×	×	0.4003	0.2013	0.1682	0.7377	1.1847	0.6160	48.6471	0.7255
✓	×	×	×	0.0996	0.7438	0.7173	0.9722	0.1659	0.0980	19.6434	0.8244
✓	✓	×	×	0.0954	0.7624	0.7349	0.9858	0.1589	0.0890	17.7610	0.8609
✓	✓	✓	×	0.1023	0.7482	0.7202	0.9868	0.1439	0.0830	17.2462	0.8609
✓	✓	✓	✓	**0.0830**	**0.8226**	**0.7640**	**0.9894**	**0.1394**	**0.0815**	**16.5789**	**0.8671**

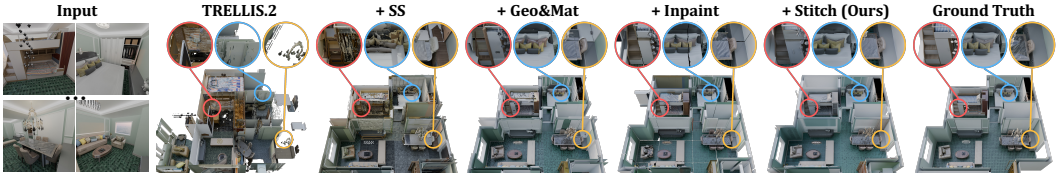

Figure 4: **Qualitative Ablation Study.** We qualitatively compare the same ablation settings as in Table 3 on a 3D-FRONT scene unseen during training. We show the complete scene reconstruction for each variant and the ground truth, while colored circles highlight corresponding local details.

Although vanilla pretrained TRELLIS.2 has been trained on 3D-FRONT scene data, it performs poorly in our task because its single-image conditioning provides only partial observations and does not enforce a consistent world orientation. Our reconstruction-based *SS* generator instead uses the reconstructed point cloud as a spatial prior while leveraging the generative prior to plausibly complete regions missing from the initial reconstruction. Our reconstruction-based *Geo&Mat* generators further exploit multi-view observations to recover finer geometry and more faithful materials. Nevertheless, because the blocks are generated independently, inconsistencies can remain in their overlapping regions. As shown in Figure 4, adjacent blocks may exhibit misaligned surface heights at their interfaces, accompanied by visible discontinuities in geometry and texture. Adding *Inpaint* propagates context between neighbouring blocks, encouraging their surfaces to connect coherently and improving the depth and normal metrics. However, small inconsistencies at block boundaries may remain visible as seams, explaining the slight degradation in the 3D metrics reported in Table 3. The complete WorldRecon pipeline employs stage-wise world latent clamping and blending (*Stitch*) to maintain cross-block continuity and global scene coherence, thereby eliminating visible boundary discontinuities. It consequently achieves the best performance across all reported metrics and produces the reconstruction most consistent with the ground truth.

5 Conclusion

We presented WorldRecon, a generative reconstruction framework for recovering complete, large-scale scene PBR meshes from unposed RGB images. WorldRecon leverages coarse geometry and camera estimates predicted by a reconstruction model as effective global grounding despite their imperfections. Based on this global scaffold, reconstruction-based block generation produces faithful mesh blocks with detailed geometry and PBR materials. Neighbour-aware latent inpainting and the causal world-generation pipeline stitch these blocks together, ensuring consistent transitions across block boundaries. WorldRecon achieves state-of-the-art performance in large-scale indoor scene reconstruction, with strong sim-to-real generalization despite training exclusively on synthetic data. More broadly, our work provides a practical path toward reconstructing editable 3D scene assets from casually captured or generated videos, supporting downstream applications such as simulation, embodied AI, and content creation.

Limitations and Future Work. As our training data comprise only indoor scenes, the current applicability of WorldRecon is limited to indoor environments, and its generalization to outdoor scenes remains to be established. In addition, our current scene representation comprises a single layer of $4\,\mathrm{m}$-high blocks, restricting reconstruction to single-story environments and precluding the modeling of multi-story buildings. Extending the training distribution to more diverse indoor, outdoor, and multi-story environments, together with supporting block generation along the vertical dimension, constitutes an important direction for future work.

AI USE STATEMENT

We used generative AI tools to aid and polish the writing of this paper to improve readability. We have not used generative AI tools for idea generation, hypothesis formulation, research methodology design, experiment design, data generation, method implementation, code development, result interpretation, or mathematical formulation. All AI-assisted text was reviewed and revised by the authors. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

ETHICS STATEMENT

Our method is trained exclusively on publicly available synthetic datasets (3D-FRONT and SAGE-10k) and does not involve human subjects or personal data. We note that 3D scene reconstruction technology could potentially be misused for unauthorized digital reproduction of real environments; we encourage responsible use of such technology.

REPRODUCIBILITY STATEMENT

We provide comprehensive details to support reproducibility. Section 4.1 describes the experimental setup, including datasets, metrics, and baselines. Section A details the block-level training data construction pipeline, covering scene rendering, slicing, multi-view condition extraction, and point-cloud preparation. Section B provides further implementation details of the WorldRecon framework, including reconstruction-based conditioning, neighbour-aware latent inpainting, and training configurations. Section C specifies the evaluation protocol, including baseline settings, alignment procedures, and metric definitions. Code and pretrained models will be released upon acceptance.

REFERENCES

Jonathan T Barron, Ben Mildenhall, Dor Verbin, Pratul P Srinivasan, and Peter Hedman. Mip-nerf 360: Unbounded anti-aliased neural radiance fields. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 5470–5479, 2022.

Jiahao Chang, Chongjie Ye, Yushuang Wu, Yuantao Chen, Yidan Zhang, Zhongjin Luo, Chenghong Li, Yihao Zhi, and Xiaoguang Han. Reconviagen: Towards accurate multi-view 3d object reconstruction via generation. *arXiv preprint arXiv:2510.23306*, 2025.

David Charatan, Sizhe Lester Li, Andrea Tagliasacchi, and Vincent Sitzmann. pixelsplat: 3d gaussian splats from image pairs for scalable generalizable 3d reconstruction. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 19457–19467, 2024.

Hanke Chen, Yuan Liu, and Minchen Li. Trellisworld: Training-free world generation from object generators. *arXiv preprint arXiv:2510.23880*, 2025.

Maximilian Denninger, Dominik Winkelbauer, Martin Sundermeyer, Wout Boerdijk, Markus Knauer, Klaus H. Strobl, Matthias Humt, and Rudolph Triebel. Blenderproc2: A procedural pipeline for photorealistic rendering. *Journal of Open Source Software*, 8(82):4901, 2023. doi: 10.21105/joss.04901. URL https://doi.org/10.21105/joss.04901.

Paul Engstler, Aleksandar Shtedritski, Iro Laina, Christian Rupprecht, and Andrea Vedaldi. Syncity: Training-free generation of 3d worlds. In *Proceedings of the International Conference on Computer Vision (ICCV)*, 2025.

Huan Fu, Bowen Cai, Lin Gao, Ling-Xiao Zhang, Jiaming Wang, Cao Li, Qixun Zeng, Chengyue Sun, Rongfei Jia, Binqiang Zhao, et al. 3d-front: 3d furnished rooms with layouts and semantics. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 10933–10942, 2021.

Yasutaka Furukawa and Carlos Hernández. Multi-view stereo: A tutorial. *Found. Trends Comput. Graph. Vis.*, 9(1-2):1–148, 2015.

Ruiqi Gao, Aleksander Holynski, Philipp Henzler, Arthur Brussee, Ricardo Martin-Brualla, Pratul Srinivasan, Jonathan T Barron, and Ben Poole. Cat3d: Create anything in 3d with multi-view diffusion models. *arXiv preprint arXiv:2405.10314*, 2024.

Haoyu Guo, He Zhu, Sida Peng, Haotong Lin, Yunzhi Yan, Tao Xie, Wenguan Wang, Xiaowei Zhou, and Hujun Bao. Multi-view reconstruction via sfm-guided monocular depth estimation. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 5272–5282, 2025.

Lukas Höllein, 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*, pp. 7909–7920, 2023.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models, 2021. URL `https://arxiv.org/abs/2106.09685`.

Binbin Huang, Zehao Yu, Anpei Chen, Andreas Geiger, and Shenghua Gao. 2d gaussian splatting for geometrically accurate radiance fields. In *ACM SIGGRAPH 2024 conference papers*, pp. 1–11, 2024.

Zhisheng Huang, Jiahao Chen, Cheng Lin, Chenyu Hu, Hanzhuo Huang, Zhengming Yu, Mengfei Li, Yuheng Liu, Zekai Gu, Zibo Zhao, et al. Unirecgen: unifying multi-view 3d reconstruction and generation. *arXiv preprint arXiv:2604.01479*, 2026.

Haian Jin, Hanwen Jiang, Hao Tan, Kai Zhang, Sai Bi, Tianyuan Zhang, Fujun Luan, Noah Snavely, and Zexiang Xu. Lvsm: A large view synthesis model with minimal 3d inductive bias. *arXiv preprint arXiv:2410.17242*, 2024.

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

Vincent Leroy, Yohann Cabon, and Jérôme Revaud. Grounding image matching in 3d with mast3r. In *European conference on computer vision*, pp. 71–91. Springer, 2024.

Sikuang Li, Chen Yang, Jiemin Fang, Taoran Yi, Jia Lu, Jiazhong Cen, Lingxi Xie, Wei Shen, and Qi Tian. Worldgrow: Generating infinite 3d world. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 6433–6441, 2026.

Yuan Liu, Cheng Lin, Zijiao Zeng, Xiaoxiao Long, Lingjie Liu, Taku Komura, and Wenping Wang. Syncdreamer: Generating multiview-consistent images from a single-view image. In *International conference on learning representations*, volume 2024, pp. 27676–27697, 2024.

Quan Meng, Lei Li, Matthias Nießner, and Angela Dai. Lt3sd: Latent trees for 3d scene diffusion. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 650–660, 2025.

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

Richard A Newcombe, Shahram Izadi, Otmar Hilliges, David Molyneaux, David Kim, Andrew J Davison, Pushmeet Kohi, Jamie Shotton, Steve Hodges, and Andrew Fitzgibbon. Kinectfusion: Real-time dense surface mapping and tracking. In *2011 10th IEEE international symposium on mixed and augmented reality*, pp. 127–136. Ieee, 2011.

Linfei Pan, Daniel Barath, Marc Pollefeys, and Johannes Lutz Schönberger. Global Structure-from-Motion Revisited. In *European Conference on Computer Vision (ECCV)*, 2024.

Xuanchi Ren, Yifan Lu, Hanxue Liang, Zhangjie Wu, Huan Ling, Mike Chen, Sanja Fidler, Francis Williams, and Jiahui Huang. Scube: Instant large-scale scene reconstruction using voxsplats. *Advances in Neural Information Processing Systems*, 37:97670–97698, 2024.

Katharina Schmid, Nicolas von Lützow, Jozef Hladkỳ, Angela Dai, and Matthias Nießner. Genrecon: Bridging generative priors for multi-view 3d scene reconstruction. *arXiv preprint arXiv:2605.23888*, 2026.

Johannes L Schonberger and Jan-Michael Frahm. Structure-from-motion revisited. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4104–4113, 2016.

Johannes L Schönberger, Enliang Zheng, Jan-Michael Frahm, and Marc Pollefeys. Pixelwise view selection for unstructured multi-view stereo. In *European conference on computer vision*, pp. 501–518. Springer, 2016.

Jonas Schult, Sam Tsai, Lukas Höllein, Bichen Wu, Jialiang Wang, Chih-Yao Ma, Kunpeng Li, Xiaofang Wang, Felix Wimbauer, Zijian He, et al. Controlroom3d: Room generation using semantic proxy rooms. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6201–6210, 2024.

Team Seedance, De Chen, Liyang Chen, Xin Chen, Ying Chen, Zhuo Chen, Zhuowei Chen, Feng Cheng, Tianheng Cheng, Yufeng Cheng, Mojie Chi, Xuyan Chi, Jian Cong, Qinpeng Cui, Fei Ding, Qide Dong, Yujiao Du, Haojie Duanmu, Junliang Fan, Jiarui Fang, Jing Fang, Zetao Fang, Chengjian Feng, Yu Gao, Diandian Gu, Dong Guo, Hanzhong Guo, Qiushan Guo, Boyang Hao, Hongxiang Hao, Haoxun He, Jiaao He, Qian He, Tuyen Hoang, Heng Hu, Ruoqing Hu, Yuxiang Hu, Jiancheng Huang, Weilin Huang, Zhaoyang Huang, Zhongyi Huang, Jishuo Jin, Ming Jing, Ashley Kim, Shanshan Lao, Yichong Leng, Bingchuan Li, Gen Li, Haifeng Li, Huixia Li, Jiashi Li, Ming Li, Xiaojie Li, Xingxing Li, Yameng Li, Yiying Li, Yu Li, Yueyan Li, Chao Liang, Han Liang, Jianzhong Liang, Ying Liang, Wang Liao, J. H. Lien, Shanchuan Lin, Xi Lin, Feng Ling, Yue Ling, Fangfang Liu, Jiawei Liu, Jihao Liu, Jingtuo Liu, Shu Liu, Sichao Liu, Wei Liu, Xue Liu, Zuxi Liu, Ruijie Lu, Lecheng Lyu, Jingting Ma, Tianxiang Ma, Xiaonan Nie, Jingzhe Ning, Junjie Pan, Xitong Pan, Ronggui Peng, Xueqiong Qu, Yuxi Ren, Yuchen Shen, Guang Shi, Lei Shi, Yinglong Song, Fan Sun, Li Sun, Renfei Sun, Wenjing Tang, Boyang Tao, Zirui Tao, Dongliang Wang, Feng Wang, Hulin Wang, Ke Wang, Qingyi Wang, Rui Wang, Shuai Wang, Shulei Wang, Weichen Wang, Xuanda Wang, Yanhui Wang, Yue Wang, Yuping Wang, Yuxuan Wang, Zijie Wang, Ziyu Wang, Guoqiang Wei, Meng Wei, Di Wu, Guohong Wu, Hanjie Wu, Huachao Wu, Jian Wu, Jie Wu, Ruolan Wu, Shaojin Wu, Xiaohu Wu, Xinglong Wu, Yonghui Wu, Ruiqi Xia, Xin Xia, Xuefeng Xiao, Shuang Xu, Bangbang Yang, Jiaqi Yang, Runkai Yang, Tao Yang, Yihang Yang, Zhixian Yang, Ziyan Yang, Fulong Ye, Bingqian Yi, Xing Yin, Yongbin You, Linxiao Yuan, Weihong Zeng, Xuejiao Zeng, Yan Zeng, Siyu Zhai, Zhonghua Zhai, Bowen Zhang, Chenlin Zhang, Heng Zhang, Jun Zhang, Manlin Zhang, Peiyuan Zhang, Shuo Zhang, Xiaohe Zhang, Xiaoying Zhang, Xinyan Zhang, Xinyi Zhang, Yichi Zhang, Zixiang Zhang, Haiyu Zhao, Huating Zhao, Liming Zhao, Yian Zhao, Guangcong Zheng, Jianbin Zheng, Xiaozheng Zheng, Zerong Zheng, Kuan Zhu, and Feilong Zuo. Seedance 2.0: Advancing video generation for world complexity, 2026. URL https://arxiv.org/abs/2604.14148.

Oriane Siméoni, Huy V. Vo, Maximilian Seitzer, Federico Baldassarre, Maxime Oquab, Cijo Jose, Vasil Khalidov, Marc Szafraniec, Seungeun Yi, Michaël Ramamonjisoa, Francisco Massa, Daniel Haziza, Luca Wehrstedt, Jianyuan Wang, Timothée Darcet, Théo Moutakanni, Leonel Sentana, Claire Roberts, Andrea Vedaldi, Jamie Tolan, John Brandt, Camille Couprie, Julien Mairal, Hervé Jégou, Patrick Labatut, and Piotr Bojanowski. DINOv3, 2025. URL https://arxiv.org/abs/2508.10104.

Noah Stier, Anurag Ranjan, Alex Colburn, Yajie Yan, Liang Yang, Fangchang Ma, and Baptiste Angles. Finerecon: Depth-aware feed-forward network for detailed 3d reconstruction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 18423–18432, 2023.

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*, pp. 20507–20518, 2024a.

Jiaxiang Tang, Zhaoxi Chen, Xiaokang Chen, Tengfei Wang, Gang Zeng, and Ziwei Liu. Lgm: Large multi-view gaussian model for high-resolution 3d content creation. In *European Conference on Computer Vision*, pp. 1–18. Springer, 2024b.

Yolo Y. Tang, Daiki Shimada, Jiayue Meng, Jing Bi, Pinxin Liu, Yicheng Wang, Yunzhong Xiao, Zhangyun Tan, Zeliang Zhang, Chao Huang, Susan Liang, Qianxiang Shen, Luchuan Song, Ali Vosoughi, Mingqian Feng, Melika Filvantorkaman, and Chenliang Xu. Bvb: Benchmarking agentic video understanding via programmatic reconstruction in blender, 2026. URL https://arxiv.org/abs/2609.15478.

Tencent Hunyuan3D Team. Hunyuan3d-omni: A unified framework for controllable generation of 3d assets, 2025. URL https://arxiv.org/abs/2509.21245.

Fangjinhua Wang, Silvano Galliani, Christoph Vogel, Pablo Speciale, and Marc Pollefeys. Patchmatchnet: Learned multi-view patchmatch stereo. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 14194–14203, 2021a.

Jianyuan Wang, Minghao Chen, Nikita Karaev, Andrea Vedaldi, Christian Rupprecht, and David Novotny. Vggt: Visual geometry grounded transformer. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 5294–5306, 2025.

Jianyuan Wang, Minghao Chen, Shangzhan Zhang, Nikita Karaev, Johannes Schönberger, Patrick Labatut, Piotr Bojanowski, David Novotny, Andrea Vedaldi, and Christian Rupprecht. Vggt-omega. *arXiv preprint arXiv:2605.15195*, 2026.

Peng Wang, Lingjie Liu, Yuan Liu, Christian Theobalt, Taku Komura, and Wenping Wang. Neus: Learning neural implicit surfaces by volume rendering for multi-view reconstruction. *arXiv preprint arXiv:2106.10689*, 2021b.

Qianqian Wang, Zhicheng Wang, Kyle Genova, Pratul P Srinivasan, Howard Zhou, Jonathan T Barron, Ricardo Martin-Brualla, Noah Snavely, and Thomas Funkhouser. Ibrnet: Learning multi-view image-based rendering. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 4690–4699, 2021c.

Shuzhe Wang, Vincent Leroy, Yohann Cabon, Boris Chidlovskii, and Jerome Revaud. Dust3r: Geometric 3d vision made easy. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 20697–20709, 2024.

Xinpeng Wang, Chandan Yeshwanth, and Matthias Nießner. Sceneformer: Indoor scene generation with transformers. In *2021 International conference on 3D vision (3DV)*, pp. 106–115. IEEE, 2021d.

Rundi Wu, Ben Mildenhall, Philipp Henzler, Keunhong Park, Ruiqi Gao, Daniel Watson, Pratul P Srinivasan, Dor Verbin, Jonathan T Barron, Ben Poole, et al. Reconfusion: 3d reconstruction with diffusion priors. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 21551–21561, 2024a.

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

Hongchi Xia, Xuan Li, Zhaoshuo Li, Qianli Ma, Jiashu Xu, Ming-Yu Liu, Yin Cui, Tsung-Yi Lin, Wei-Chiu Ma, Shenlong Wang, et al. Sage: Scalable agentic 3d scene generation for embodied ai. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 22358–22368, 2026.

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*, pp. 21469–21480, 2025.

Jianfeng Xiang, Xiaoxue Chen, Sicheng Xu, Ruicheng Wang, Zelong Lv, Yu Deng, Hongyuan Zhu, Yue Dong, Hao Zhao, Nicholas Jing Yuan, et al. Native and compact structured latents for 3d generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 14419–14429, 2026.

Tian-Xing Xu, Xiangjun Gao, Wenbo Hu, Xiaoyu Li, Song-Hai Zhang, and Ying Shan. Geometrycrafter: Consistent geometry estimation for open-world videos with diffusion priors. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 6632–6644, 2025.

Yao Yao, Zixin Luo, Shiwei Li, Tian Fang, and Long Quan. Mvsnet: Depth inference for unstructured multi-view stereo. In *Proceedings of the European conference on computer vision (ECCV)*, pp. 767–783, 2018.

Yao Yao, Zixin Luo, Shiwei Li, Tianwei Shen, Tian Fang, and Long Quan. Recurrent mvsnet for high-resolution multi-view stereo depth inference. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 5525–5534, 2019.

Lior Yariv, Jiatao Gu, Yoni Kasten, and Yaron Lipman. Volume rendering of neural implicit surfaces. In *Thirty-Fifth Conference on Neural Information Processing Systems*, 2021.

Chandan Yeshwanth, Yueh-Cheng Liu, Matthias Nießner, and Angela Dai. Scannet++: A high-fidelity dataset of 3d indoor scenes. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 12–22, 2023.

Zehao Yu, Songyou Peng, Michael Niemeyer, Torsten Sattler, and Andreas Geiger. Monosdf: Exploring monocular geometric cues for neural implicit surface reconstruction. *Advances in neural information processing systems*, 35:25018–25032, 2022.

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

Chen Ziwen, Hao Tan, Kai Zhang, Sai Bi, Fujun Luan, Yicong Hong, Li Fuxin, and Zexiang Xu. Long-lrm: Long-sequence large reconstruction model for wide-coverage gaussian splats. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 4349–4359, 2025.

APPENDIX

The appendix presents supplementary material related to data curation, the WorldRecon method, experiments details, and additional qualitative cases. Section A details the block-level training data construction pipeline, including scene rendering, scene slicing, multi-view condition extraction, and point-cloud preparation. Section B provides further implementation details of the WorldRecon framework, covering reconstruction-based conditioning, neighbour-aware latent inpainting, training configurations, and the world generation pipeline. Section C describes the evaluation protocol, including baseline settings, the alignment procedure, and metric definitions. Section D presents additional qualitative results, including large-scale scene generation, a comparison with an agentic LLM baseline, and failure case analysis.

A DATA CURATION DETAILS

We construct the block-level training set from two synthetic indoor scene datasets, 3D-FRONT Fu et al. (2021) and SAGE-10k Xia et al. (2026), as described in the main paper. We detail its rendering, scene-slicing, and condition-extraction procedures below.

A.1 SCENE RENDERING

For each scene, we render a set of full-scene RGB observations to simulate the multi-view images used for room reconstruction. We additionally render the corresponding depth maps for point-cloud construction during data preparation.

To increase the appearance diversity of 3D-FRONT, we augment its architectural surfaces with PBR materials from ambientCG (formerly CC0 Textures)[1]. For each scene, we construct separate material pools for floors (wood, planks, carpet, tiles, marble, and terrazzo), walls (wallpaper, marble, and plaster), and structural elements such as doors and baseboards (wood and marble). We first sample a scene-level material-assignment strategy

$$s \sim \mathrm{Categorical}(0.3,\, 0.6,\, 0.1), \qquad s \in \{\textit{instance},\, \textit{category},\, \textit{original}\}. \tag{S1}$$

Under the *instance* strategy, each architectural component independently draws a material from its corresponding pool. Under the *category* strategy, components belonging to the same category share a sampled material, producing a more coherent appearance. The *original* strategy applies no additional material augmentation and preserves the scene as provided by 3D-FRONT. When either *instance* or *category* strategy is selected, floors are always assigned a new material, while walls and structural elements are augmented with probabilities 0.6 and 0.8, respectively. Furniture is excluded from material augmentation and always retains its original 3D-FUTURE materials.

For illumination, in addition to the original light sources provided by each 3D-FRONT scene, we assign a randomly selected HDRI environment map. SAGE-10k scenes additionally receive one ceiling area light because their enclosed layouts receive limited illumination from the environment map. We randomise the horizontal position, size, energy, and warm/cool colour of each area light.

To mimic natural human capture, we sample near-level camera poses inside individual rooms at heights between 1.4m and 1.8m. Subject to the availability of valid poses, views are distributed across rooms approximately in proportion to their floor areas. We reject poses closer than 1m to scene geometry to avoid overly close-up observations and encourage broader scene coverage.

We render RGB and depth maps with BlenderProc Denninger et al. (2023) at 1024×1024 resolution and a 90° field of view (FoV). For every accepted view, we also store the camera intrinsics and camera-to-world transformation.

A.2 SCENE SLICING

We divide each scene into overlapping axis-aligned cubic blocks of side length $\ell = 4$m. For each horizontal axis $a \in \{x, y\}$, let $[a_{\min}, a_{\max}]$ denote the scene extent and $d_a = a_{\max} - a_{\min}$. The

[1] All assets are released under the CC0 1.0 license: `https://ambientcg.com/`.

number of blocks n_a, stride s_a, and block starting positions p_i^a are defined as

$$n_a = \left\lceil \frac{d_a}{\ell} \right\rceil, \qquad s_a = \begin{cases} 0, & n_a = 1, \\ \dfrac{d_a - \ell}{n_a - 1}, & n_a > 1, \end{cases} \qquad p_i^a = \begin{cases} \dfrac{a_{\min} + a_{\max} - \ell}{2}, & n_a = 1, \\ a_{\min} + i\, s_a, & n_a > 1, \end{cases} \tag{S2}$$

where $i = 0, \ldots, n_a - 1$. This construction uses the minimum number of blocks required to cover the scene and distributes their overlap uniformly. We retain only scenes with a height below 4m, so a single block centred on the scene is sufficient along the vertical axis.

For block index (i, j), the starting positions from Equation (S2) determine its world-space bounds:

$$\mathbf{b}_{\min}^{i,j} = \left(p_i^x,\, p_j^y,\, \frac{z_{\min} + z_{\max} - \ell}{2} \right), \qquad \mathbf{b}_{\max}^{i,j} = \mathbf{b}_{\min}^{i,j} + \ell \mathbf{1}. \tag{S3}$$

We construct each block mesh by clipping the complete scene mesh against four axis-aligned planes of its bounding box. Objects intersecting a block boundary are geometrically cropped rather than assigned wholly to one block, while their original material attributes are preserved. The newly created boundary surfaces are left uncapped.

The block centre is $\mathbf{c}^{i,j} = (\mathbf{b}_{\min}^{i,j} + \mathbf{b}_{\max}^{i,j})/2$. A world-space point $\mathbf{x}$ is then mapped to the canonical block frame by

$$\bar{\mathbf{x}} = \frac{\mathbf{x} - \mathbf{c}^{i,j}}{\ell}, \tag{S4}$$

such that every block occupies $[-0.5, 0.5]^3$. The same scale and translation are applied to meshes, point clouds, and camera-to-world translations.

A.3 Multi-View Condition Extraction

For each rendered image I_v, let $\mathbf{u} = (u_x, u_y)^\top$, $\tilde{\mathbf{u}} = (u_x, u_y, 1)^\top$, and $d_v(\mathbf{u})$ denote a pixel, its homogeneous coordinate, and its metric depth, respectively. Given the camera intrinsics $\mathbf{K}_v$ and camera-to-world transformation $[\mathbf{R}_v \mid \mathbf{t}_v]$, we unproject the pixel into world space as

$$\mathbf{x}_v(\mathbf{u}) = \mathbf{R}_v \left(d_v(\mathbf{u})\, \mathbf{K}_v^{-1} \tilde{\mathbf{u}} \right) + \mathbf{t}_v. \tag{S5}$$

Given the bounds $[\mathbf{b}_{\min}^{i,j}, \mathbf{b}_{\max}^{i,j}]$ of block (i, j), we define its visibility mask as

$$M_{i,j,v}(\mathbf{u}) = \mathbf{1}\left[\mathbf{b}_{\min}^{i,j} \le \mathbf{x}_v(\mathbf{u}) \le \mathbf{b}_{\max}^{i,j} \right]. \tag{S6}$$

Views are ranked by $\sum_{\mathbf{u}} M_{i,j,v}(\mathbf{u})$, and only those for which at least 25% of the image pixels fall inside the block are considered valid. We retain up to 16 valid views and use $M_{i,j,v}$ as the alpha channel of the corresponding RGB image.

We transform the camera-to-world translation to the canonical block frame using Equation (S4), while preserving the camera rotation. Camera intrinsics are rescaled to the resolution used by the image encoder. Thus, the RGBA images, cameras, point cloud, and target latents are expressed in the same block-local coordinate frame.

A.4 Point-Cloud Preparation

For each block, we construct two point-cloud conditions. To obtain the augmented depth-derived cloud, we randomly select 6–10 retained views and apply depth and camera perturbations, visibility dropout, and sparse outlier injection before unprojection. We unproject the perturbed views into the canonical block frame, merge and crop the points to $[-0.5, 0.5]^3$, and sparsify the cloud to 8192 points. The VGGT-Ω cloud is reconstructed from the retained block-masked RGB views described above. We first apply confidence filtering and statistical outlier removal. We then estimate a Horn Sim(3) transformation from the corresponding VGGT-predicted and canonical camera centres, jointly recovering the relative scale, rotation, and translation between the two coordinate frames. The estimated transformation is applied to the reconstructed point cloud, which is subsequently cropped to the canonical block bounds and sparsified to 8192 points.

When both sources are available, they are registered as separate training instances, ensuring balanced exposure to both point-cloud distributions within each epoch. For blocks with only one valid source, the available point cloud is used.

Table S1: **Training configurations**. We report the image and latent resolutions, point-cloud size, number of GPUs, global batch size, learning rate, EMA decay, and adaptation strategy for each generation and inpainting stage.

Stage	Image	Latent	Points	GPUs	Global Batch	LR	EMA	Adaptation
SS generation	512	16^3	8192	32	128	1×10^{-4}	0.9999	full
SS inpainting	512	16^3	8192	32	128	3×10^{-5}	0.9999	full
Geometry generation	512	32^3	–	16	64	1×10^{-4}	0.9995	full
Geometry inpainting	512	32^3	–	16	64	3×10^{-5}	0.9995	full
Material generation	512	32^3	–	16	96	3×10^{-5}	0.9995	LoRA
Material inpainting	512	32^3	–	16	96	1×10^{-5}	0.9995	LoRA

B WorldRecon Method Details

B.1 Reconstruction-Based Conditioning

Multi-View Image Conditioning. For each conditioning image, we extract patch features using a frozen DINOv3 ViT-L/16 encoder Siméoni et al. (2025) at 512×512 resolution.

To encode the camera pose, we sample five image-plane locations comprising the image centre and four corners. Given a homogeneous pixel coordinate $\tilde{\mathbf{u}}_{v,k}$, camera intrinsics $\mathbf{K}_v$, camera-to-world rotation $\mathbf{R}_v$, and camera centre $\mathbf{o}_v$, the corresponding world-space ray and Plücker coordinate are

$$\mathbf{d}_{v,k} = \frac{\mathbf{R}_v \mathbf{K}_v^{-1} \tilde{\mathbf{u}}_{v,k}}{\left\| \mathbf{R}_v \mathbf{K}_v^{-1} \tilde{\mathbf{u}}_{v,k} \right\|_2}, \qquad \boldsymbol{\pi}_{v,k} = (\mathbf{d}_{v,k}, \mathbf{o}_v \times \mathbf{d}_{v,k}) . \tag{S7}$$

A two-layer MLP projects each Plücker coordinate into a camera token, which is prepended to the corresponding image-patch tokens. A four-layer cross-attention module then uses learnable queries to fuse all views into a fixed-length conditioning sequence.

Point Cloud Condition. The SS stage additionally receives the XYZ coordinates of an 8192-point block cloud $\mathcal{P}_b = \{\mathbf{p}_j\}_{j=1}^{8192}$. Following Hunyuan3D-Omni Team (2025), we encode the normalised coordinates using Fourier features and append learned modality tokens:

$$\mathbf{c}^{\mathrm{pc}} = [\psi(\gamma(\mathbf{p}_1)), \ldots, \psi(\gamma(\mathbf{p}_{8192})), \mathbf{e}_1, \ldots, \mathbf{e}_{10}] , \tag{S8}$$

where $\gamma(\cdot)$ denotes the Fourier positional encoding, $\psi(\cdot)$ is a learnable projection, and $\mathbf{e}_{1:10}$ are modality tokens. All tokens share the 1024-dimensional feature space of the image condition. We drop the complete point-cloud stream with probability 0.1 during training.

B.2 Neighbour-Aware Latent Inpainting

We use the concat-inpainting formulation and overlap masks defined in the main paper for all three stages. Let $\mathbf{x}^s_{0,\mathrm{known}} = \mathbf{x}^s_0 \odot (\mathbf{1} - \mathbf{m}^s)$ denote the known latent provided by previously generated neighbours for stage $s \in \{\mathrm{ss}, \mathrm{geo}, \mathrm{mat}\}$, where $\mathbf{m}^s = 0$ marks known locations and $\mathbf{m}^s = 1$ marks locations to be generated. The input features for the three inpainters are defined separately as

$$\mathbf{h}^{\mathrm{ss}}_{\mathrm{in}} = f^{\mathrm{ss}}_{\mathrm{in}}\big([\mathbf{x}^{\mathrm{ss}}_t, \mathbf{m}^{\mathrm{ss}}, \mathbf{x}^{\mathrm{ss}}_{0,\mathrm{known}}]\big) , \tag{S9}$$

$$\mathbf{h}^{\mathrm{geo}}_{\mathrm{in}} = f^{\mathrm{geo}}_{\mathrm{in}}\Big(\Big[\mathbf{x}^{\mathrm{geo}}_t, \mathbf{m}^{\mathrm{geo}}, \mathbf{x}^{\mathrm{geo}}_{0,\mathrm{known}}\Big]\Big) , \tag{S10}$$

$$\mathbf{h}^{\mathrm{mat}}_{\mathrm{in}} = f^{\mathrm{mat}}_{\mathrm{in}}\big([\mathbf{x}^{\mathrm{mat}}_t, \mathbf{z}^{\mathrm{geo}}, \mathbf{m}^{\mathrm{mat}}, \mathbf{x}^{\mathrm{mat}}_{0,\mathrm{known}}]\big) . \tag{S11}$$

Here, $\mathbf{z}^{\mathrm{geo}}$ is the native Geometry condition of the Material stage.

B.3 Training Details

Table S1 summarises the final 512-resolution configurations used for the reported reconstruction results. All models are trained with AdamW and BF16 automatic mixed precision. We use adaptive gradient clipping at the 95-th percentile with a maximum norm of 1.0, and unconditional condition dropout of 0.1. Geometry and Material training samples are capped at 16k active sparse tokens.

The Sparse Structure and Geometry stages are fully fine-tuned, whereas the Material denoisers are adapted using rank-8, scale-16 LoRA modules inserted into their attention projections. For the Material stages, the frozen base denoiser is accompanied by trainable multi-view fusion and camera-tokenization modules, whose learning rates are 10^{-4} for generation and 3×10^{-5} for inpainting. The table reports the corresponding LoRA learning rates.

B.4 World Generation Pipeline

Global Reconstruction and Scene Alignment. Given N unposed RGB images, we resize them to the 512-pixel input resolution of VGGT-Ω. A single forward pass predicts a depth map $\hat{\mathbf{D}}_v$, depth confidence $\hat{\mathbf{C}}_v$, camera intrinsics $\hat{\mathbf{K}}_v$, and world-to-camera extrinsics $\hat{\mathbf{E}}_v = [\hat{\mathbf{R}}_v \mid \hat{\mathbf{t}}_v]$ for each view. We unproject the predicted depths using the corresponding camera parameters to obtain world-space point maps while preserving pixel-to-3D correspondences for constructing block-specific image conditions. We merge points satisfying $\hat{\mathbf{C}}_v(\mathbf{u}) \geq 2.0$ across all views and apply statistical outlier removal to suppress isolated depth artifacts. Finally, we apply connected-component filtering to the aligned global point cloud and retain its largest component as the scene scaffold.

The VGGT-Ω reconstruction is not necessarily aligned with the canonical scene axes. We therefore rotate the predicted point cloud such that the gravity direction is aligned with the positive Z-axis and the dominant wall directions are aligned with the X- and Y-axes, and then recover its physical scale. We first select an initial up direction from the three PCA axes of the global point cloud:

$$\bar{\mathbf{y}} = \frac{\sum_v \hat{\mathbf{R}}_v^\top \mathbf{e}_y}{\left\|\sum_v \hat{\mathbf{R}}_v^\top \mathbf{e}_y\right\|_2}, \qquad k^\star = \underset{k\in\{1,2,3\}}{\arg\max} \left|\mathbf{v}_k^\top \bar{\mathbf{y}}\right|, \qquad \mathbf{g}_0 = -\operatorname{sign}\left(\mathbf{v}_{k^\star}^\top \bar{\mathbf{y}}\right) \mathbf{v}_{k^\star}. \tag{S12}$$

Here, $\bar{\mathbf{y}}$ is the average predicted camera Y-axis expressed in world coordinates, $\mathbf{v}_k$ is the k-th PCA axis, and $\mathbf{g}_0$ is the selected initial up direction. The negative sign accounts for the OpenCV convention in which the camera Y-axis points downward. After rotating $\mathbf{g}_0$ to $\mathbf{e}_z = (0,0,1)^\top$, we refine the up direction using approximately horizontal surfaces in the lower portion of the scene:

$$\mathcal{H} = \{i \in \mathcal{P} : |\mathbf{n}_i^\top \mathbf{e}_z| > \tau_h\}, \qquad \mathbf{g} = \frac{\sum_{i\in\mathcal{H}} \operatorname{sign}(\mathbf{n}_i^\top \mathbf{e}_z)|\mathbf{n}_i^\top \mathbf{e}_z|\mathbf{n}_i}{\left\|\sum_{i\in\mathcal{H}} \operatorname{sign}(\mathbf{n}_i^\top \mathbf{e}_z)|\mathbf{n}_i^\top \mathbf{e}_z|\mathbf{n}_i\right\|_2}. \tag{S13}$$

Here, $\mathbf{n}_i$ is the local surface normal, $\mathcal{H}$ retains normals satisfying the horizontal-surface threshold $\tau_h = 0.7$. Their consistently oriented, weighted average gives the refined up direction $\mathbf{g}$, which is aligned with $\mathbf{e}_z$. We then estimate the dominant direction of wall-like normals:

$$\begin{aligned} &\mathcal{W} = \left\{i : \left|\mathbf{n}_i^\top \mathbf{e}_z\right| < \tau_w\right\}, \qquad \phi_i = \operatorname{atan2}(n_{i,y}, n_{i,x}) \bmod \frac{\pi}{2}, \\ &\phi^\star = \underset{\phi\in[0,\pi/2)}{\arg\max} \sum_{i\in\mathcal{W}} \|\mathbf{n}_{i,xy}\|_2 \; K_\sigma^{\pi/2}(\phi - \phi_i), \qquad \mathbf{R}_{\text{yaw}} = \mathbf{R}_z(-\phi^\star). \end{aligned} \tag{S14}$$

Here, $\mathcal{W}$ contains wall-like normals selected with $\tau_w = 0.2$, ϕ_i is the horizontal orientation of normal i folded modulo $\pi/2$, and $K_\sigma^{\pi/2}$ is a periodic Gaussian kernel used to aggregate their votes. The dominant angle $\phi^\star$ defines the yaw rotation $\mathbf{R}_{\text{yaw}}$.

Block Construction and Local Conditioning. We rescale the aligned scene to a vertical extent of $2.9\,\text{m}$ and cover the XY bounding box of its connected scaffold with the smallest centred grid of $4\,\text{m}$ blocks. Adjacent blocks use a $3\,\text{m}$ stride and share the same vertical centre. We discard blocks containing fewer than 100 reconstructed points. Each remaining crop is uniformly subsampled to at most $8N_{\text{pc}}$ points before farthest-point sampling produces the final $N_{\text{pc}} = 8192$-point condition.

To select the conditioning views, we voxelise each block crop at G^3 resolution with $G = 32$, obtaining a target set $\mathcal{Q}_b$. For every candidate view v satisfying the 15% in-block coverage criterion, we voxelise its aligned in-block point map and retain the intersection $\mathcal{V}_{b,v}$ with $\mathcal{Q}_b$. We then choose up to four views that maximise their joint coverage:

$$\mathcal{S}_b^\star = \underset{\substack{\mathcal{S}\subseteq\mathcal{C}_b \\ |\mathcal{S}|=\min(4,|\mathcal{C}_b|)}}{\arg\max} \left|\bigcup_{v\in\mathcal{S}} \mathcal{V}_{b,v}\right|, \tag{S15}$$

where $\mathcal{C}_b$ denotes the candidate-view set.

C Experimental Details

C.1 Baselines

We use VGGT-Ω Wang et al. (2026) prediction as the common geometric prior for all methods. It provides camera intrinsics and extrinsics, per-view depth and confidence, and a point cloud obtained by unprojecting the predicted depths. Each baseline follows its official pipeline, with any required ground-truth geometric inputs replaced by the corresponding predictions from VGGT-Ω.

- **VGGT-Ω+TSDF** directly integrates the predicted RGB-D frames using the VGGT-Ω cameras through TSDF fusion Newcombe et al. (2011).
- **Murre** Guo et al. (2025) receives the RGB frames and sparse depth samples selected from the VGGT-Ω depth maps using confidence, spatial subsampling, and multi-view consistency. Its refined dense depths are fused with the VGGT-Ω intrinsics and aligned poses using TSDF.
- **FineRecon** Stier et al. (2023) receives the RGB frames, confidence-filtered VGGT-Ω depths, intrinsics, and aligned poses. The predicted depths are used for depth-guided feature back-projection and as an additional TSDF feature channel.
- **MonoSDF** Yu et al. (2022) uses the VGGT-Ω intrinsics and aligned poses together with the RGB frames. Following its original formulation, its monocular depth and normal supervision is predicted from the RGB images by Omnidata, rather than taken from the VGGT-Ω depth maps.
- **GenRecon** Schmid et al. (2026) uses the aligned VGGT-Ω point cloud to determine the scene extent and chunk layout, and uses the predicted cameras to project multi-view RGB features into each chunk's 3D feature volume.

C.2 Evaluation Protocol

As VGGT-Ω estimates camera poses from uncalibrated images, its reconstructed coordinate frame has no absolute scale or canonical horizontal orientation. Our scene-alignment procedure recovers gravity and dominant wall directions, but does not resolve the remaining global similarity ambiguity with respect to the ground-truth frame. For evaluation, we therefore estimate a Sim(3) transform from corresponding predicted and reference camera centres using Horn alignment and apply it to the complete predicted mesh. The same transform is used for all 2D and 3D metrics. We convert dataset-specific axis conventions before computing the metrics.

Following the evaluation protocol of GenRecon Schmid et al. (2026), for synthetic scenes, we restrict prediction triangles to the GT axis-aligned bounding box enlarged by 20%, preventing geometry far outside the scene extent from dominating the evaluation. For ScanNet++, we restrict the evaluation domain to the observed scanner surface, represented by a $1.1\,\mathrm{cm}$ voxel grid and dilated by $15\,\mathrm{cm}$.

C.3 Metrics

3D Metrics. We uniformly sample 200k points from each mesh surface and compute bidirectional nearest-neighbour correspondences. We report:

- **Chamfer Distance (CD):** the average of the mean predicted-to-reference and reference-to-predicted nearest-neighbour Euclidean distances, reported in metres;
- **F-score@10cm (F-score):** the harmonic mean of predicted-to-GT precision and GT-to-prediction recall, where a point is considered matched if its nearest-neighbour distance is below $0.1\,\mathrm{m}$;
- **Normal Consistency (NC):** the bidirectional mean absolute dot product between the normals of nearest-neighbour surface samples. Correspondences farther than $0.2\,\mathrm{m}$ contribute zero.

2D Metrics. We render the aligned predicted and reference geometry using the same held-out camera poses and intrinsics. For each scene, we evaluate 50 views sampled from the held-out test set. The synthetic test views have a resolution of 512×512, whereas the ScanNet++ test views have a resolution of 640×480. Frames used as input to VGGT-Ω are excluded from the evaluation views.

We evaluate pixel-wise depth and surface-normal errors, pooling RMSE, AbsRel, and Angular over all valid view–pixel pairs across the evaluation views. Let $\mathcal{V}_d$ denote the set of view–pixel pairs for which both the predicted and reference renderings have finite, positive depth. The depth metrics are

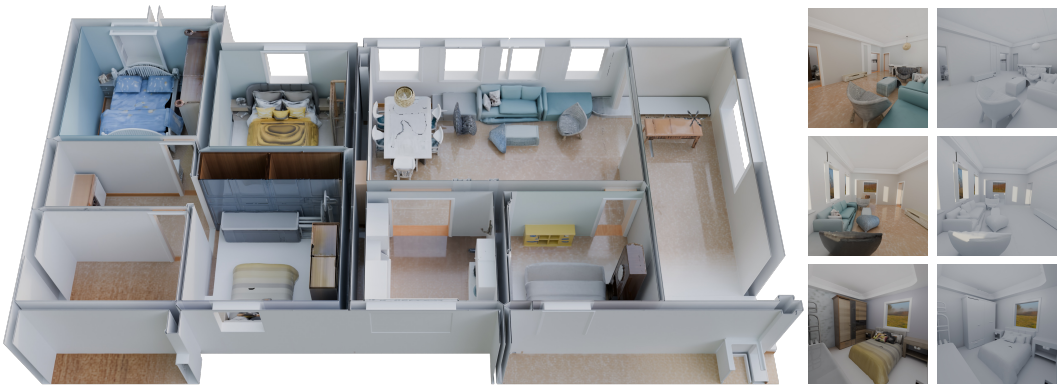

Figure S1: **Large-Scale Scene Generation.** Reconstruction results for large-scale, multi-room scenes, with top-down views (left) and detailed interior views (right).

defined as

$$\mathrm{RMSE} = \sqrt{\frac{1}{|\mathcal{V}_d|}\sum_{p\in\mathcal{V}_d}(\hat{d}_p - d_p)^2}, \qquad \mathrm{AbsRel} = \frac{1}{|\mathcal{V}_d|}\sum_{p\in\mathcal{V}_d}\frac{|\hat{d}_p - d_p|}{d_p}, \tag{S16}$$

where $\hat{d}_p$ and d_p denote the predicted and reference depths, respectively. For surface normals, let $\mathcal{V}_n$ denote the set of view–pixel pairs for which both the predicted and reference foreground normals are valid. Assuming unit-length normals, we report the mean angular error in degrees:

$$\mathrm{Angular} = \frac{180}{\pi|\mathcal{V}_n|}\sum_{p\in\mathcal{V}_n}\arccos\left(\left|\hat{\mathbf{n}}_p^\top \mathbf{n}_p\right|\right). \tag{S17}$$

The absolute dot product makes the metric invariant to normal sign ambiguity. Completeness measures the fraction of valid reference foreground pixels for which the predicted rendering contains valid geometry.

For perceptual evaluation, LPIPS and CLIP similarity are computed separately for depth and surface normals. LPIPS, CLIP similarity, and Completeness are computed independently for each view and then averaged across the evaluation views.

D Additional Results

D.1 Large Scene Generation

We further demonstrate our method on large-scale indoor scenes comprising multiple, largely isolated rooms. As shown in Fig. S1, our method reconstructs an expansive nine-room scene with high fidelity, demonstrating its scalability to large 3D indoor environments.

D.2 Comparison with Agentic Baseline.

We compare our method with a GPT-6 Astra agent configured with high reasoning effort and adapted from BVB's reconstruction protocol Tang et al. (2026). Given the same 32 RGB views, the agent inspects the images, writes and executes Blender scripts, and examines rendered previews to refine its reconstruction. It constructs scenes using geometric primitives and materials specified by numerical parameters, without image textures or imported assets. Our adaptation uses fixed RGB views instead of video input and does not enforce the original cost ceiling.

Figure S2 shows a densely furnished scene with complex object arrangements and mutual occlusions (top), and a multi-room interior with oblique partitions (bottom). The recorded reconstruction stages consume approximately 2.12M and 1.71M tokens, respectively, including repeated cached context, corresponding to estimated Standard API-equivalent costs of $2.93 and $2.26. In contrast, our method performs inference on a single GPU without external LLM calls, avoiding this additional API expense. Despite iterative refinement, the agent produces simplified object geometry and

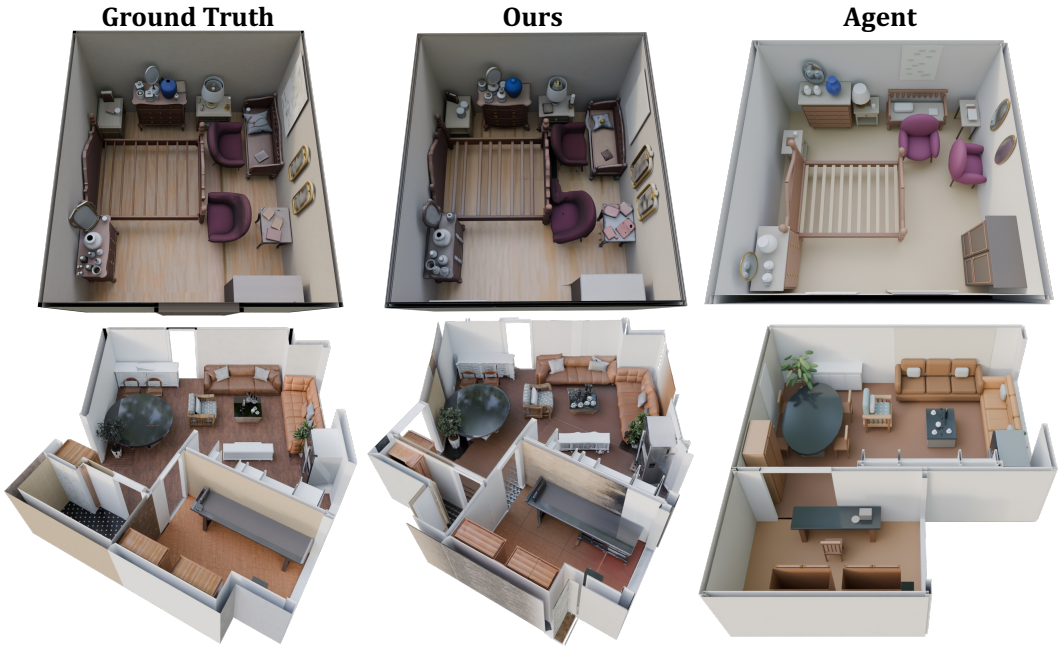

Figure S2: **Comparison with Agentic Baseline.** Given the same 32 views, we show GT, ours, and a GPT-6 Astra agent adapted from BVB Tang et al. (2026) (left to right). Our method yields more faithful reconstructions, while the agent simplifies or omits objects and distorts the scene layout.

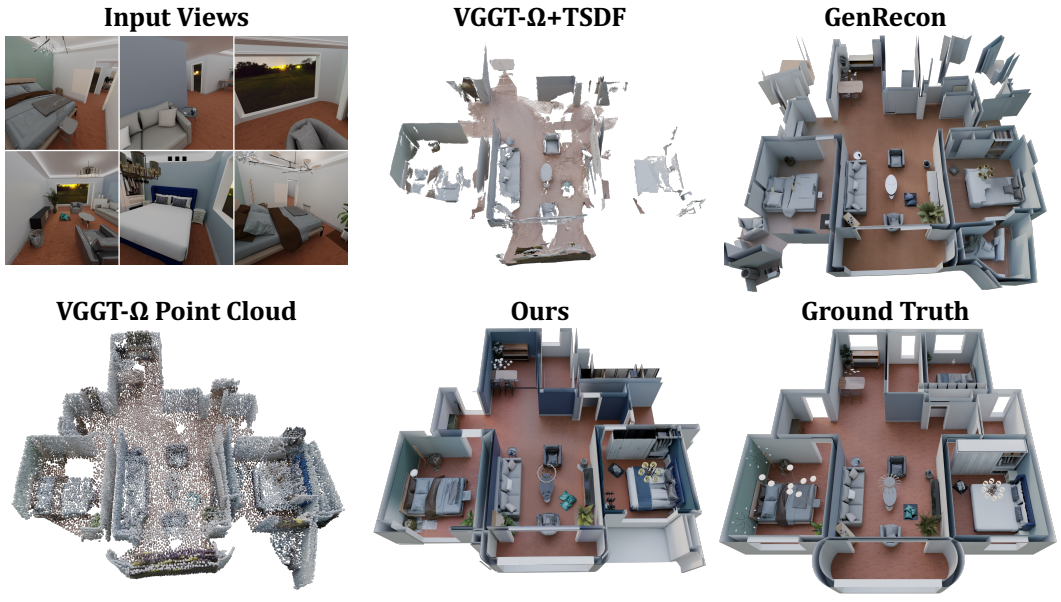

Figure S3: **Failure case.** Inaccurate reconstruction from VGGT-Ω lead to missing geometry in unobserved regions. Nevertheless, WorldRecon remains robust to substantial pose errors and still faithfully reconstructs the areas that are covered by the input views.

misplaced furniture. In the multi-room scene, it reconstructs an incorrect global layout. Our method more faithfully recovers furniture arrangement, architectural structure, and surface appearance.

D.3 Failure Cases and Limitations

Figure S3 illustrates a representative failure case on a multi-room scene. The 32 sampled input views (top-left) are visually rich, but parts of the scene are occluded or observed from only a handful of viewpoints. As a result, the VGGT-Ω point-cloud scaffold (bottom-left) incorrectly estimates the poses of several views: some regions are either left unreconstructed or filtered out due to low confidence, while other views are projected to entirely wrong locations.

Under this shared scaffold, the TSDF baseline can only reconstruct a small fraction of the scene, while the generative baseline GenRecon directly reprojects image features using the erroneous poses, scattering misaligned projections across the scene and producing large fragmented artifacts as well as distorted objects. WorldRecon likewise suffers from missing geometry in unobserved regions due to the incomplete input coverage. Nevertheless, it remains robust to substantial pose errors and still faithfully reconstructs the areas that are well covered by the input views, and is able to hallucinate plausible completions for the missing point-cloud regions based on sparse image cues.