ANCHORCOMPLETE: RECOVERING COMPLETE SCENE MESHES FROM PARTIAL POINT CLOUDS

Anonymous authors
Paper under double-blind review

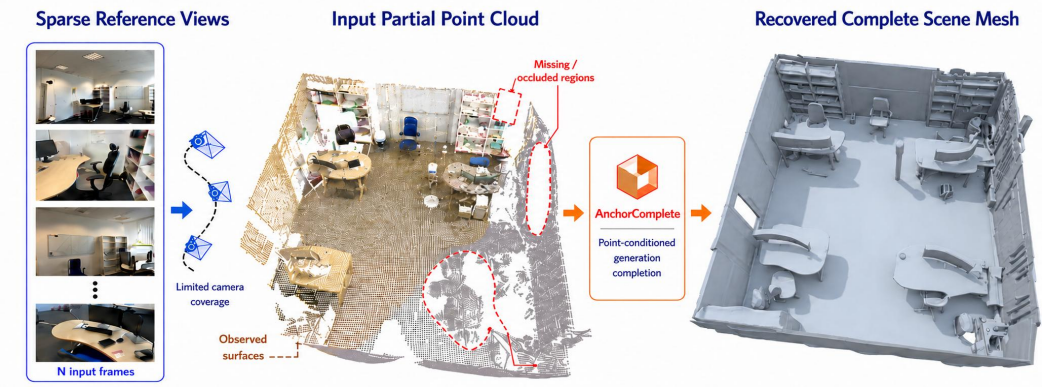


Figure 1: Given a partial point cloud with missing or occluded regions, **ANCHORCOMPLETE** recovers a complete scene mesh using a point-conditioned generative completion approach, including geometry beyond the observed surfaces. Sparse reference views are shown only for visualization.

ABSTRACT

Most image-based 3D reconstructors prioritize reconstructing visible surfaces, producing partial point clouds with large holes and missing areas due to limited field of view and occlusions. To address this gap, we introduce ANCHORCOMPLETE, a framework that recovers complete scene meshes from partial point reconstructions. Our design addresses two key challenges: anchoring generation to the observed geometry and scaling completion to diverse scene layouts. We introduce a *partial-point conditioning* module that guides both sparse-structure and sparse-latent generation using the input point cloud, together with *Swin-DiT backbones* that replace full self-attention with 3D windowed attention for efficient high-resolution generation. We instantiate these designs on top of the sparse latent representation of TRELLIS. The resulting model produces complete scene meshes that remain faithful to the observed geometry, providing a more complete digital representation of the physical environment. Experiments on Replica and SCRREAM show that ANCHORCOMPLETE consistently outperforms strong baselines in complete scene reconstruction, particularly under sparse-view settings.

1 INTRODUCTION

We study the problem of recovering a complete 3D mesh of a scene given only a partial point observation, such as the output of a feed-forward 3D reconstructor or a 3D scanning system. A robust solution to this problem would benefit downstream applications such as any-view AR/VR rendering (Xie et al., 2026), large-scale physical simulation (Savva et al., 2019; Szot et al., 2021), and robotics (Liu et al., 2023a), where complete scene meshes are often required for rendering, collision detection, and physical interaction.

Addressing this problem requires tackling two coupled challenges: (i) preserving fidelity to the observed geometry and (ii) scaling completion to *scene*-level inputs. Recently, feed-forward 3D reconstructors like DUSt3R (Wang et al., 2024) and its variants (Leroy et al., 2024; Wang et al., 2025b;a; Yang et al., 2025a; Sun et al., 2024; Keetha et al., 2025), as well as online SLAM systems like DROID-SLAM (Teed & Deng, 2021) and VGGT-SLAM (Maggio et al., 2025), have made it

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.

possible to obtain dense point clouds of scenes in a single forward pass by scaling the model and training data to directly regress 3D points from image observations. However, completion is not part of their formulation, so unobserved regions remain large holes, limiting their applicability in many downstream applications.

Early completion methods (Han et al., 2017; Lyu et al., 2021; Yu et al., 2021; Zhou et al., 2021) focus on completing point clouds of single objects in normalized coordinate spaces and thus do not scale to scene-level inputs. While Amodal3R (Wu et al., 2025), SAM 3D (Team et al., 2025), and ShapeR (Siddiqui et al., 2026) extend foundational 3D generators (Zhang et al., 2023; Vahdat et al., 2022; Zhang et al., 2024; Hunyuan3D et al., 2025a; Xiang et al., 2024; 2025; Li et al., 2025a) to occluded geometry, they mainly address separate *object* completion in normalized coordinate spaces and do not directly condition on partial point observations, so they may not remain faithful to the geometry actually observed by the cameras. Recent *scene* completion methods like PaSCo (Cao et al., 2024), NOVA3R (Chen et al., 2026b), and Seen2Scene (Meng et al., 2026) aim to generate complete scene geometry. While promising, these models typically complete within the given field of view, without considering geometry outside the camera view. They also do not output a complete mesh, but rather a point cloud or a voxel grid, which may not be directly usable for downstream applications.

Seeking to close this gap, we introduce ANCHORCOMPLETE, a generative framework that takes as input a partial point cloud, such as one recovered from sparse RGB observations, and outputs a complete scene mesh (Figure 1). We build on the sparse latent mesh representation of TRELLIS (Xiang et al., 2024; 2025), but adapting it to scene completion poses two key challenges: faithfully preserving observed geometry and scaling to higher latent resolutions, where global self-attention becomes increasingly memory-intensive.

We address these challenges with two main components. **Partial-point conditioning** anchors generation to the input through a hybrid scheme that combines global cross-attention to transformer-encoded anchor tokens with local spatial injection of observed voxels into the noisy latent. **Swin-DiT** enables high-resolution scene generation by replacing global self-attention over latent tokens with memory-efficient windowed self-attention, while retaining global cross-attention to the point observations. Together, they enable faithful and scalable scene completion from sparse observations.

We evaluate our method on the Replica (Straub et al., 2019) and SCRREAM (Jung et al., 2024) datasets, where we reconstruct complete indoor scenes under standard and sparse-view settings. Our experiments demonstrate that ANCHORCOMPLETE recovers more complete scene geometry than the baselines while remaining faithful to the visible input, and degrades more slowly as camera coverage drops. Our contributions are:

- We take a step towards recovering complete meshes of scenes from partial point observations, without requiring posed dense images, oriented normals, or per-scene optimization. For this new problem setting, we formulate strong baselines and evaluation protocols.
- We introduce ANCHORCOMPLETE, a generative framework that contributes two architectural components: (1) **partial-point conditioning** and (2) **Swin-DiT backbones**, which together enable efficient generative mesh completion at scene scale while remaining anchored to the observed reconstruction.
- ANCHORCOMPLETE achieves the best $\mathrm{CD}_{\mathrm{all}}$ and F-scores on Replica and SCRREAM, and degrades more slowly than the visible-surface baselines we evaluate as input views decrease.

2 RELATED WORK

Visible-surface reconstruction. DUSt3R (Wang et al., 2024) and MASt3R (Leroy et al., 2024) regress dense pixel-aligned point maps from unposed image collections, marking a shift toward generalizable, pose-free 3D reconstruction. Building on this paradigm, several recent works (Wang et al., 2025a;b; Lin et al., 2025; Yang et al., 2025a; Sun et al., 2024; Keetha et al., 2025; Wang et al., 2025c; Chen et al., 2026a) extend it to multi-view settings, while online dense SLAM systems (Teed & Deng, 2021; Murai et al., 2025; Liu et al., 2025; Maggio et al., 2025) integrate such reconstruction into camera-trajectory estimation. Surface fitters such as Poisson and Screened Poisson reconstruction (Kazhdan et al., 2006; Kazhdan & Hoppe, 2013), and learned methods such as NKSR and NoKSR (Huang et al., 2023; Li et al., 2025b), convert oriented points into smooth implicit surfaces. All remain tied to camera-visible geometry: occluded regions go unmodeled, and surfaces fitted

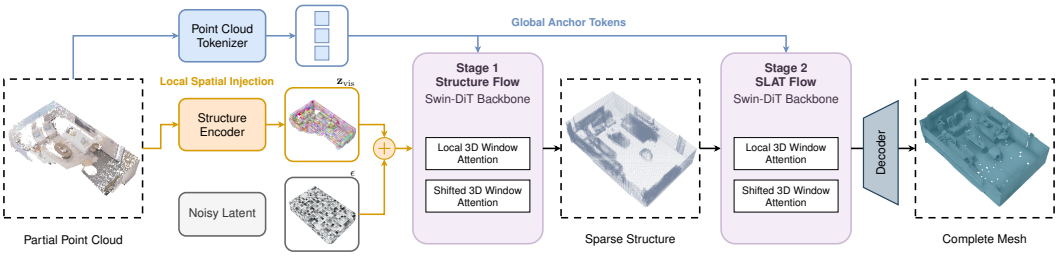


Figure 2: **Overview of AnchorComplete.** Given a partial point cloud, AnchorComplete completes a scene mesh with a two-stage generative pipeline. It conditions on the input via global cross-attention over anchor tokens and local spatial injection of encoded visible geometry into the noisy latent. Stage 1 predicts a sparse scene structure using a Swin-DiT backbone, while Stage 2 generates structured latent features on this structure and decodes them into the final complete mesh.

across holes follow smoothness rather than any prior over the unobserved scene. We instead complete those regions with a learned generative prior.

Generative 3D priors. Generative 3D methods synthesize complete geometry from images, text, layouts, or latent representations. Image-space methods such as Text2Room (Höllein et al., 2023), Invisible Stitch (Engstler et al., 2025), and GenRC (Li et al., 2024) iteratively inpaint and lift RGB-D views, while foundation 3D generators (Zhang et al., 2023; 2024; Xiang et al., 2024; Hunyuan3D et al., 2025a; Li et al., 2025a; Xiang et al., 2025) and amodal methods (Wu et al., 2025; Team et al., 2025) recover complete object geometry from image or text prompts. Scene-scale approaches (Shriram et al., 2025; Wu et al., 2024; Ju et al., 2024; Roessle et al., 2024; Bahmani et al., 2023; Yang et al., 2024a;b; Feng et al., 2023) generate rooms from learned scene representations or layouts. However, these priors are not directly constrained by the metric geometry of a specific partial reconstruction, motivating our point-conditioned formulation.

Completion from partial 3D geometry. Completion methods condition on partial 3D input to infer missing geometry. Object-centric approaches (Yuan et al., 2018; Yu et al., 2021; Zhou et al., 2022; Chu et al., 2023; Han et al., 2017; Lyu et al., 2021; Zhou et al., 2021) complete point clouds or shapes, while scene methods (Dai et al., 2018; 2020) predict occupancy or SDFs from partial voxel grids, typically at limited resolution. More recent generative models such as Hunyuan3D-Omni (Hunyuan3D et al., 2025b) and ShapeR (Siddiqui et al., 2026) condition foundation 3D generators on partial geometry but remain object-scale. At scene scale, PaSCo (Cao et al., 2024), NOVA3R (Chen et al., 2026b), Points-to-3D (Xia et al., 2026), Seen2Scene (Meng et al., 2026), and GenRecon (Schmid et al., 2026) explore occupancy, amodal reconstruction, latent generation, or chunk-based completion. Seen2Scene is the closest baseline to our setting and is evaluated in Section 4.2. Maintaining fidelity to observed geometry while plausibly completing missing regions remains a central challenge in scene completion. We condition a pretrained 3D generative prior on a raw scene-scale partial reconstruction and output a completed mesh in the original scene coordinate frame (Section 3).

3 Method

We develop AnchorComplete, a generative model that recovers a complete scene mesh $\mathcal{M}$ from a partial point cloud $\mathcal{P}$ obtained from feed-forward 3D reconstruction (Wang et al., 2025a; Lin et al., 2025), SLAM (Teed & Deng, 2021; Maggio et al., 2025), or 3D scanning (Dai et al., 2017). The key challenge is to hallucinate plausible geometry in unobserved regions while preserving the observed structure. Our model, illustrated in Figure 2, builds on TRELLIS (Xiang et al., 2024) and adapts it to scene-level completion through partial-point conditioning and Swin-DiT backbones. We first review TRELLIS in Section 3.1, then present these two components in Sections 3.2 and 3.3, followed by training details in Section 3.4.

3.1 PRELIMINARIES: TRELLIS

TRELLIS (Xiang et al., 2024) is a foundational 3D asset generator that takes a text prompt or image as the condition $\mathbf{c}$ and outputs various 3D representations, including meshes $\mathcal{M}$, Gaussian splats $\mathcal{G}$, and radiance fields $\mathcal{R}$. For flexibility, TRELLIS represents the given 3D asset $\mathcal{O}$ with a set of *Structured Latents* (SLATs) $\mathcal{Z} = \{\boldsymbol{z}_i, \boldsymbol{p}_i\}_{i=1}^{L}$ on a predefined 3D spatial grid $\Omega \in \mathbb{N}^{H\times W\times D}$, where $L \ll HWD$ is the number of active voxels, $\boldsymbol{p}_i \in \mathbb{R}^3$ is the 3D coordinate of the i-th active voxel, and $\boldsymbol{z}_i \in \mathbb{R}^C$ is the corresponding latent feature encoding local geometry and appearance information.

The TRELLIS pipeline consists of two VAE modules followed by two corresponding rectified-flow generation stages (Albergo & Vanden-Eijnden, 2022; Lipman et al., 2022; Liu et al., 2023b). The first *structure* VAE $(\mathcal{E}_{\text{s}}, \mathcal{D}_{\text{s}})$ compresses a dense binary voxel grid $\boldsymbol{O} \in \{0,1\}^{H\times W\times D}$ into a latent representation $\boldsymbol{z}_{\text{s}} \in \mathbb{R}^{\frac{H}{s}\times\frac{W}{s}\times\frac{D}{s}\times C}$, which is decoded back into a dense voxel grid to model coarse geometry. The second *SLAT* VAE $(\mathcal{E}_{\text{sl}}, \mathcal{D}_{\text{sl}})$ encodes the *Structured Latents* (SLATs) $\mathcal{Z}$, augmented with voxelized multi-view image features of the 3D asset, into a latent representation $\boldsymbol{z}_{\text{sl}} \in \mathbb{R}^{L\times C}$, which is then decoded into various 3D representations. During generation, two rectified-flow stages are applied to the structure and SLAT latent spaces, respectively. In the first sparse structure generation stage, a flow transformer $\mathcal{T}_{\text{s}}$ generates the structure latent $\boldsymbol{z}_{\text{s}}$ from Gaussian noise conditioned on the text or image prompts $\mathbf{c}$. The sampled $\boldsymbol{z}_{\text{s}}$ is then decoded into the active voxel support $\{\boldsymbol{p}_i\}_{i=1}^{L}$, which determines the coarse spatial layout of the asset. In the second SLAT generation stage, a sparse flow transformer $\mathcal{T}_{\text{sl}}$ generates the local SLAT features $\{\boldsymbol{z}_i\}_{i=1}^{L}$ on this support $\{\boldsymbol{p}_i\}_{i=1}^{L}$, again conditioned on $\mathbf{c}$. Finally, the generated SLAT $\mathcal{Z}$ is decoded into the desired 3D output. In our work, we adopt the TRELLIS SLAT representation and extend it with partial-point conditioning and scalable scene-level generation.

3.2 PARTIAL POINT CONDITIONING

Unlike image-conditioned 3D generators (Zhang et al., 2024; Hunyuan3D et al., 2025a; Xiang et al., 2024; 2025), we condition scene completion directly on partial point observations. This is advantageous for several reasons: (1) *Completeness*: the observed points delineate the visible scene boundary, providing direct geometric evidence of where completion is needed; (2) *Accuracy*: the condition is directly grounded in 3D geometry, maintaining better fidelity to the visible geometry; (3) *Efficiency*: the observed points $\mathcal{P} = \{\boldsymbol{p}_i\}_{i=1}^{N}$ can be uniformly subsampled to a fixed size N, avoiding conditioning costs that grow with the number of input images; and (4) *Composability*: point-based inputs can be chunked for scalable local processing, and the outputs can be directly merged in the same scene coordinate frame. In particular, we take a hybrid conditioning approach via (1) global cross-attention to *anchor tokens* encoded from the point cloud and (2) *local spatial injection* of visible-voxel evidence into the structure latent; both paths are shown in Figure 2. The two pathways are ablated in Section 4.

Global Anchor Tokens. We design a simple architecture to extract anchor tokens from the input point observations $\mathcal{P} = \{\boldsymbol{p}_i\}_{i=1}^{N}$. In particular, we implement the point-cloud encoder $\mathcal{E}_{\text{p}}$ using a transformer architecture inspired by the point encoder of NOVA3R (Chen et al., 2026b). To this end, we first map the 3D coordinates of all N input points to high-dimensional features $\{\boldsymbol{f}_i\}_{i=1}^{N}$. The encoder then selects $\mathcal{F}^{q} = \{\boldsymbol{f}_i\}_{i=1}^{m}$ point queries via farthest point sampling and uses the full set of point features $\mathcal{F}^{(k,v)} = \{\boldsymbol{f}_i\}_{i=1}^{N}$ as the keys and values for the encoder's cross-attention layers, where $m \ll N$ is a fixed number of point tokens. The anchor tokens are further refined by several interleaved self-attention and cross-attention blocks. The resulting anchor tokens $\boldsymbol{c}_{\text{p}} \in \mathbb{R}^{m\times C_{\text{xa}}}$, where C_{xa} is the cross-attention feature dimension, are then fed to the TRELLIS flow model as global conditioning tokens through cross-attention, replacing the original image/text conditioning tokens in both stages.

Local Spatial Injection. To further encourage spatial consistency between the generated geometry and the input point cloud, we introduce a local spatial conditioning path that injects visible-point evidence directly into the first-stage structure latent space. Inspired by condition injection strategies in image editing (Avrahami et al., 2023), we first voxelize the point observations $\mathcal{P}$ into a binary 3D grid $\boldsymbol{O}_{\text{vis}} \in \{0,1\}^{H\times W\times D}$, setting a voxel to 1 if it contains at least one observed point and to 0 otherwise. It is then encoded with the pretrained structure encoder $\mathcal{E}_{\text{s}}$ to obtain partial-geometry features $\boldsymbol{z}_{\text{vis}} \in \mathbb{R}^{\frac{H}{s}\times\frac{W}{s}\times\frac{D}{s}\times C}$. In parallel, we downsample $\boldsymbol{O}_{\text{vis}}$ to obtain a 3D binary mask

$\boldsymbol{O}_{\text{vis},m} \in \{0,1\}^{\frac{H}{s}\times\frac{W}{s}\times\frac{D}{s}}$, indicating which latent cells correspond to visible regions. The visible evidence enters Stage 1 in two ways. First, it initializes the flow: rather than starting from Gaussian noise everywhere, we replace the initial sample at visible cells by $\boldsymbol{z}_{\text{vis}}$,

$$\boldsymbol{z}_s^{(1)} \;=\; \boldsymbol{O}_{\text{vis},m} \odot \boldsymbol{z}_{\text{vis}} \;+\; (\mathbf{1} - \boldsymbol{O}_{\text{vis},m}) \odot \boldsymbol{\epsilon}, \qquad \boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I}), \tag{1}$$

where $\boldsymbol{z}_s^{(1)}$ is the latent the sampler starts from. Second, the mask is available at every denoising step: we concatenate $\boldsymbol{O}_{\text{vis},m}$ to the noisy latent and feed the result back to the denoiser, so it can separate observed from unobserved cells throughout sampling. Both act only in Stage 1; we do not apply them to Stage 2, since Stage 1 already predicts the complete active voxel support on which SLAT features are generated.

3.3 SWIN-DIT BACKBONES

To let ANCHORCOMPLETE handle large-scale 3D scenes with diverse spatial layouts and aspect ratios, we introduce local shifted-window attention into the DiT backbone.

Hierarchical Swin-DiT. Following Swin Transformer (Liu et al., 2021) and its 3D variants (Yang et al., 2025b), we introduce a hierarchical 3D Swin-DiT backbone with shifted-window attention, in which *self-attention* is computed within *local non-overlapping 3D windows*. Specifically, let $\mathbf{h}^{(k)} \in \mathbb{R}^{H_k \times W_k \times D_k \times C_k}$ denote the latent feature grid at resolution level k. We partition the feature grid into non-overlapping 3D windows of size w^3, where $w \leq \min(H_k, W_k, D_k)$ is a fixed window size. Within each window, each token attends only to its local spatial neighbors via self-attention. To enable communication across windows, we organize the Swin-DiT blocks in regular-window/shifted-window pairs. The first block applies self-attention within regular windows, while the second block shifts the window partition by an offset $\mathbf{s} = (w/2, w/2, w/2)$ before attention and reverses the shift afterward. To increase the effective receptive field, we also downsample the feature grid after every few blocks, following a hierarchical UNet design. We refer readers to (Liu et al., 2021; Yang et al., 2025b) for more detailed descriptions.

To leverage the pretrained TRELLIS DiT weights, we apply window partitioning and shifting only on the attention masks while initializing attention weights from *global self-attention*. For feature scaling between resolution levels, we use residual scaling blocks with zero-initialized convolutional branches. Given the feature grid $\mathbf{h}^{(k)}$, the scaling operation is defined as

$$\mathbf{h}^{(k\pm1)} = \mathcal{S}_{\downarrow/\uparrow}\Big(\mathbf{h}^{(k)}\Big) + \text{Conv}^{\text{zero}}_{3\times3\times3}\Big(\mathcal{S}_{\downarrow/\uparrow}\Big(\mathbf{h}^{(k)}\Big)\Big), \tag{2}$$

where $\mathcal{S}_{\downarrow}$ is average-pooling downsampling, $\mathcal{S}_{\uparrow}$ denotes reversed upsampling, and $\text{Conv}^{\text{zero}}_{3\times3\times3}$ is a zero-initialized 3D CNN. During upsampling, the scaled feature is further fused with the corresponding encoder feature through a skip connection before being refined by Swin-DiT blocks. The shifted-window attention design is applied to both generation stages. Note that we apply it only to the *self-attention* layer within each block, while keeping cross-attention to the global anchor tokens unchanged. More details on this design are provided in the appendix.

3.4 TRAINING

We fine-tune two flow models in TRELLIS with our new conditioning and backbone designs: one predicts the active voxels $\{\boldsymbol{p}_i\}_{i=1}^{L}$, and the other predicts the SLAT features $\{\boldsymbol{z}_i\}_{i=1}^{L}$ on this support. Each model $v_{\boldsymbol{\theta}}$ operates in a latent space $\boldsymbol{z}$ and is trained to remove Gaussian noise $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ added to the target latent $\boldsymbol{z}_0$, *i.e.* $\boldsymbol{z}_t = (1-t)\boldsymbol{z}_0 + t\boldsymbol{\epsilon}$, where $t \sim \mathcal{U}(0,1)$ is a random interpolation timestep. Formally, each model $v_{\boldsymbol{\theta}}$ is trained to predict the rectified-flow velocity:

$$\mathcal{L}(\boldsymbol{\theta}) = \mathbb{E}_{(\boldsymbol{z}_0, \mathbf{c}_p), t, \boldsymbol{\epsilon}}\left[\left\| v_{\boldsymbol{\theta}}(\boldsymbol{z}_t, \mathbf{c}_p, t) - (\boldsymbol{\epsilon} - \boldsymbol{z}_0)\right\|_2^2\right], \tag{3}$$

where $\mathbf{c}_p$ denotes the encoded partial point observations. The loss is applied independently to both stages, and we fine-tune the newly introduced conditioning modules and Swin-DiT backbones from the pretrained generative prior. The warm start of Equation (1) and the mask concatenation are active during both training and inference.

Table 1: **Quantitative reconstruction evaluation on Replica (Straub et al., 2019) and SCRREAM (Jung et al., 2024) with $N_{\text{views}} = 4$.** The best result in each metric column is shown in **bold**, and the second best is underlined. * denotes our scene finetune of TRELLIS, with its output aligned to the input cloud; the unstarred row is the released object-pretrained model. Runtime is per-scene reconstruction on an RTX PRO 6000 Max-Q.

Method	Replica				SCRREAM				Runtime ↓
	CD_{vis} ↓	CD_{all} ↓	$F_{0.1}$ ↑	$F_{0.05}$ ↑	CD_{vis} ↓	CD_{all} ↓	$F_{0.1}$ ↑	$F_{0.05}$ ↑	
Feed-forward dense reconstruction									
VGGT (Wang et al., 2025a)	0.248	0.380	0.381	0.247	0.074	0.225	0.533	0.391	<1s
DA3 (Lin et al., 2025)	**0.048**	0.220	0.611	0.508	**0.052**	0.227	0.607	0.453	<1s
CUT3R (Wang et al., 2025b)	0.244	0.360	0.355	0.214	0.189	0.359	0.455	0.311	<1s
PI-3 (Wang et al., 2025c)	0.053	0.224	0.573	0.426	0.096	0.253	0.485	0.315	<1s
Image-based Generation									
TRELLIS (Xiang et al., 2024)	0.537	0.584	0.239	0.127	0.346	0.416	0.262	0.160	∼4 s
TRELLIS* (Xiang et al., 2024)	0.244	0.288	0.308	0.150	0.221	0.264	0.323	0.170	∼4 s
GenRC (Li et al., 2024)	0.355	0.366	0.161	0.076	0.295	0.332	0.191	0.091	∼6.6 min
Scene completion									
SG-NN (Dai et al., 2020)	0.085	0.214	0.600	0.415	0.111	0.225	0.518	0.308	18 s
NKSR (Huang et al., 2023)	0.057	0.211	0.633	0.527	0.066	0.209	0.614	0.457	1 s
Seen2Scene (Meng et al., 2026)	0.070	0.132	0.777	0.647	0.075	0.167	0.692	0.525	∼10.4 min
AnchorComplete (ours)	0.053	**0.071**	**0.814**	**0.682**	0.056	**0.075**	**0.787**	**0.581**	∼20 s

4 Experiments

4.1 Experimental Setup

Evaluation datasets. We evaluate on two indoor datasets with ground-truth geometry for both visible and occluded regions: **Replica** (Straub et al., 2019) with 8 photorealistic indoor reconstructions with high-quality meshes and densely sampled trajectories, and **SCRREAM** (Jung et al., 2024) with 12 real-world captures with accurate per-scene ground truth, which we treat as our primary real-world benchmark.

Training data construction. We train AnchorComplete on **ScanNet++** (Yeshwanth et al., 2023) and **InternScenes** (Zhong et al., 2025). ScanNet++ supplies real-world RGB-D videos and scanned meshes, but the meshes are incomplete and not watertight. In contrast, we use around 16k scenes from InternScenes whose watertight glTF meshes serve as complete supervision $\mathcal{M}$, covering both observed surfaces and occluded regions. Per scene we render multi-view RGB-D with BlenderProc (Denninger et al., 2023) at 640×480 and an 85° horizontal field of view, placing cameras with an area-adaptive multi-anchor sampler plus corner viewpoints for coverage. Details are in the appendix.

Metrics. Following prior work (Wang et al., 2024; 2025a; Chen et al., 2026b), we report Chamfer Distance (CD) and F-score, using ground-truth poses to crop the **visible** mesh within the input view frustums. We decompose CD into *accuracy* (prediction-to-GT) and *completeness* (GT-to-prediction). CD_{vis} is symmetric over the visible region, while CD_{all} combines visible-region accuracy with full-scene completeness. F-scores at 0.1 and $0.05\,\text{m}$ use the same regions as CD_{all}; lower CD and higher F are better. Accuracy is computed only for predictions inside the input-view frustums, since completion outside the observed region is inherently multi-modal and the ground truth represents only one valid realization. See Section A.2 for details.

Baselines. We compare against three groups, grouped as in Table 1. *Feed-forward dense reconstruction*: VGGT (Wang et al., 2025a), DA3 (Lin et al., 2025), CUT3R (Wang et al., 2025b), and PI-3 (Wang et al., 2025c) predict dense geometry from images but target visible surfaces; none outputs a mesh, so we convert all of them with the same pipeline. *Image-based generation*: TRELLIS (Xiang et al., 2024) and GenRC (Li et al., 2024) test whether generative priors alone recover missing geometry. *Scene completion*: SG-NN (Dai et al., 2020), NKSR (Huang et al., 2023), and Seen2Scene (Meng et al., 2026) take partial point clouds and predict complete surfaces, matching our input-output setting most closely. We run Seen2Scene with the authors' released code and checkpoints; its input is a

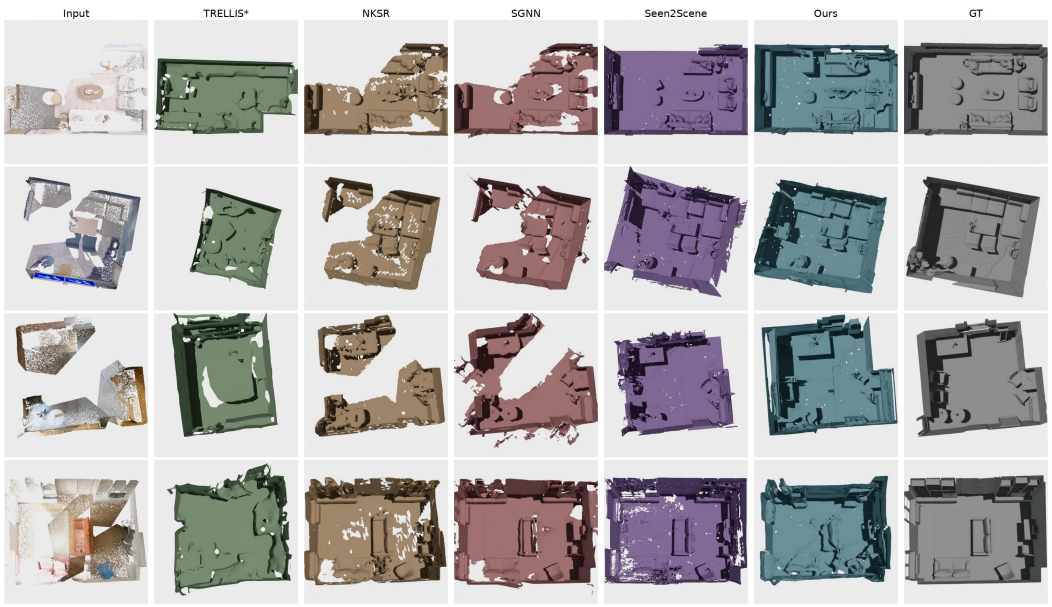


Figure 3: **Qualitative reconstruction results on Replica (Straub et al., 2019) and SCR-REAM (Jung et al., 2024).** Given the same DA3 point cloud as input, our method achieves better scene completion performance.

TSDF fused (1.1 cm voxels, VDBFusion) from the same DA3 depth maps our model conditions on, and both are scored with the same code after the same Sim(3) alignment.

Implementation details. We initialize AnchorComplete from the object-level TRELLIS checkpoints. The point-cloud encoder $\mathcal{E}_{\text{p}}$ adopts weights from NOVA3R (Chen et al., 2026b) and is kept frozen during training. Only the linear projection head that maps its features to the cross-attention dimension is fine-tuned. The Stage-1 denoiser is a Swin-DiT with 24 blocks, 1024 channels, 16 attention heads, and a window size of 8. It operates on an 8-channel 32^3 sparse-structure latent, which is decoded into a 128^3 occupancy grid. The Stage-2 denoiser uses a sparse Swin-DiT with the same window configuration and operates on the 128^3 structured-latent grid. Both stages are trained independently using AdamW with learning rate 5×10^{-5}. Stage 1 and Stage 2 are each trained for 50k steps on four NVIDIA L40 GPUs. Additional details are provided in the appendix.

4.2 Complete Scene Reconstruction

Table 1 evaluates complete-scene reconstruction at N_{views}=4. AnchorComplete leads the complete-region metrics on both datasets against every baseline. On the visible region DA3 (Lin et al., 2025) obtains the best CD_{vis} with AnchorComplete close behind, so the completion gains do not cost visible-surface fidelity; VGGT (Wang et al., 2025a) and CUT3R (Wang et al., 2025b) struggle even there, likely because four views give little coverage of a full room. Surface fitting such as NKSR (Huang et al., 2023) smooths visible points into surfaces but still lags, showing smoothing alone cannot recover unseen geometry, while TRELLIS (Xiang et al., 2024) without partial-point conditioning drifts off the captured scene. TRELLIS*, our scene-level fine-tuned version of the same prior, improves substantially over the released object model but still trails AnchorComplete by a wide margin on both datasets, indicating that our advantage comes from conditioning on the observed geometry rather than from scene-level training.

Figure 3 shows the same split visually. AnchorComplete keeps the visible surface and closes back walls and out-of-view floor where the feed-forward and surface-fitting baselines leave holes. TRELLIS* returns a plausible room unrelated to the input cloud. Seen2Scene (Meng et al., 2026), the completion baseline closest to our setting, returns a far flatter result, without the furniture relief and wall structure AnchorComplete preserves. It is also the slowest method in Table 1, at $\sim$10.4 min per scene against our $\sim$20 s.

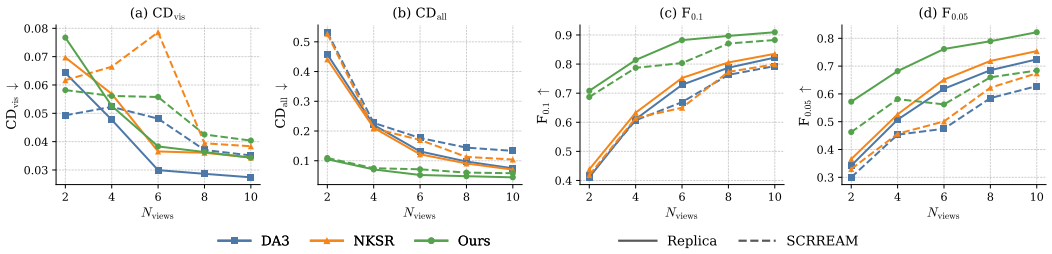


Figure 4: **Sparse-view robustness on Replica (Straub et al., 2019) and SCRREAM (Jung et al., 2024)** (mean across scenes).

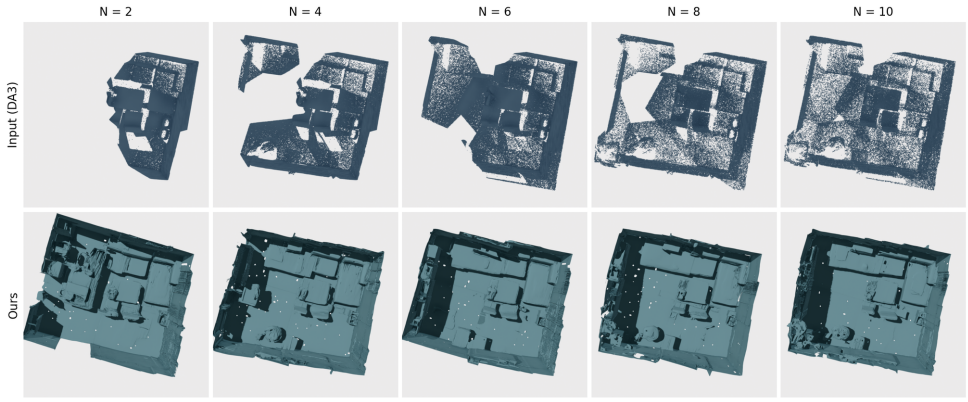


Figure 5: **Sparse-view conditional scene completion.** The DA3 input (top) loses coverage as views are removed, while our completion (bottom) stays closed across the range

.

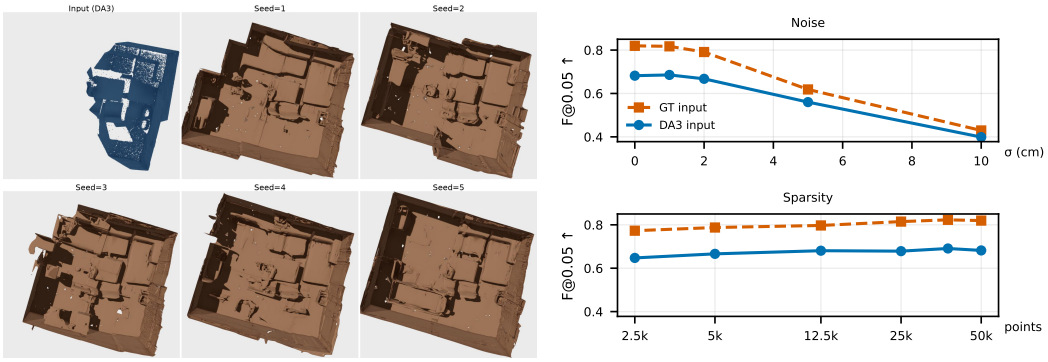
4.3 Robustness to Sparse Input Views

To evaluate robustness to sparse input views, we uniformly subsample each scene's saved views to $N_{\text{views}} \in \{2, 4, 6, 8, 10\}$ and report the same metrics as in Table 1, comparing against DA3 (Lin et al., 2025) and NKSR (Huang et al., 2023) as representative baselines. AnchorComplete leads CD_{all} and the F-scores at every view count we test, against DA3 and NKSR, with the largest gains in the sparsest setting: the gap to NKSR in CD_{all} is widest at $N{=}2$ on both datasets. It also degrades less as views decrease, maintaining higher F-scores from $N{=}10$ to $N{=}2$ than visible-surface baselines. CD_{vis} remains slightly better for DA3, especially at very sparse views, since it reconstructs only observed surfaces while our model generates complete geometry. Figure 5 shows that, as views are removed, the DA3 cloud becomes increasingly incomplete whereas our reconstruction remains complete. In addition, Figure 6a shows that AnchorComplete can generate diverse plausible completions from the same partial observation, suggesting that it does not simply memorize a single dataset-specific shape prior.

4.4 Ablation Study

We ablate on Replica (Straub et al., 2019) to analyze the contributions of key design choices in AnchorComplete.

Importance of Partial-point Conditioning. Partial-point conditioning anchors the generative prior through two complementary paths (Section 3.2): *global anchor tokens*, encoded from the input cloud and read via cross-attention, and *local spatial injection*, which injects the visible cloud into the structure latent. Removing local injection causes the largest drop, especially in visible-region fidelity (CD_{vis}), since per-voxel evidence localizes the observed geometry. Removing global anchor tokens causes a smaller drop, as the tokens supply scene-level context rather than localization. Replacing Swin-DiT with full-attention DiT, limited to a 16^3 latent, improves some coarse metrics but underperforms on finer-grained ones.



(a) (b)

Figure 6: **Generation diversity and input robustness. (a)** From one partial point cloud, different seeds give different plausible scenes; the observed structure is preserved and the variation sits in unobserved regions. **(b)** Robustness to input noise and sparsity, the noise sweep at a fixed 50k points, each run on the DA3 cloud and on ground-truth points.

Table 2: **Ablation study on Replica (Straub et al., 2019)**. CD in metres (↓), F-scores (↑) at 0.1 / 0.05 / 0.02 m. (a) N_{views}=4, 8 scenes; Global is global anchor tokens and Local is local spatial injection. (b) peak memory (GiB) and per-step wall clock (ms) for training the Stage-1 denoiser, measured on a single GPU at batch size 1; bold marks the cheaper backbone at each resolution.

(a) **Component Ablation**

Method	Global	Local	Backbone	CD_{vis}	CD_{all}	$F_{0.1}$	$F_{0.05}$	$F_{0.02}$
TRELLIS*	✗	✗	DiT (16^3)	0.244	0.288	0.308	0.150	0.030
w/o local injection	✓	✗	Swin-DiT (32^3)	0.072	0.105	0.749	0.532	0.131
w/o global tokens	✗	✓	Swin-DiT (32^3)	0.060	0.086	0.787	0.652	0.219
w/o Swin-DiT	✓	✓	DiT (16^3)	0.060	**0.065**	**0.841**	0.653	0.188
Ours	✓	✓	Swin-DiT (32^3)	**0.053**	0.071	0.814	**0.682**	**0.245**

(b) **Training Cost**

Backbone	GiB	ms
DiT 16^3	9.3	111
DiT 32^3	58.6	1886
Swin 16^3	**5.7**	**96**
Swin 32^3	**29.4**	**238**

The Cost of Full Attention. Table 2a shows full-attention DiT ahead on CD_{all} and $F_{0.1}$ at a fixed latent shape, but only at the loosest tolerance: the ordering reverses as the threshold tightens, and at 0.02 m Swin-DiT leads, 0.245 against 0.188. A 16^3 latent buys a coarse approximation that a 10 cm tolerance forgives and a 2 cm one does not. Table 2b prices running DiT at 32^3 instead. At 32^3, one training step requires 58.6 GiB and 1886 ms with full attention, compared with 29.4 GiB and 238 ms for Swin-DiT. The former exceeds the 48 GiB of the L40 GPUs we train on, which is why the DiT variant in Table 2a is evaluated at 16^3. At inference Swin-DiT is faster (91 vs. 593 ms at 32^3), with slightly higher peak memory (2.5 vs. 1.9 GiB) from the extra window-partitioning intermediates.

Robustness to Imperfect Inputs. All results above condition on raw DA3 clouds with no confidence filtering. We additionally perturb that input, with no retraining (Figure 6b). Severe sparsification costs little: dropping 95% of the input, 50k points down to 2.5k, costs 0.034 $F_{0.05}$, indicating limited sensitivity to point density. Geometric perturbation behaves differently: the curve is flat out to $\sigma = 2$ cm, the voxel scale, then falls steeply, losing ≈ 0.28 $F_{0.05}$ by $\sigma = 10$ cm, consistent with a reliance on spatially accurate anchors. Both sweeps keep their shape on ground-truth points as well as DA3, so the curves show the model's response to the perturbation, not an interaction with DA3's own error.

5 Conclusion

We presented AnchorComplete, a generative framework for recovering complete scene meshes from partial point clouds. Our approach combines partial-point conditioning with a scalable Swin-DiT backbone, enabling direct generation of complete scene geometry in the original scene coordinate frame. AnchorComplete improves complete-scene reconstruction over the baselines while maintaining high fidelity to the observed geometry. We further demonstrate robustness to sparse-view inputs and strong performance on real-world scenes, highlighting its potential for applications that require complete digital representations of physical environments.

AI USE STATEMENT

We did not use generative AI tools for research ideation, methodology design, or as an important, original, or non-standard component of the core methods in this work. We used AI-assisted writing tools for grammar and phrasing suggestions during manuscript preparation; all such suggestions were reviewed and edited by the authors, who take responsibility for the final content of this work.

ETHICS STATEMENT

This work addresses complete scene-mesh reconstruction from partial point clouds, evaluated on publicly available research datasets (ScanNet++, InternScenes, Replica, SCRREAM). We are not aware of ethical concerns specific to this research, including risks to human subjects, dataset consent issues, or foreseeable dual-use harms beyond those already associated with general-purpose 3D reconstruction and generative modeling research.

REPRODUCIBILITY STATEMENT

We describe the model architecture, training data construction, and evaluation protocol in Sections 3 and 4 of the main text, with the camera sampler in Section A.1 and the remaining implementation details, metric definitions, and evaluated scene subsets in Section A.2. We plan to release code and pretrained checkpoints upon publication to support independent verification of the reported results.

REFERENCES

Michael S Albergo and Eric Vanden-Eijnden. Building normalizing flows with stochastic interpolants. *arXiv preprint arXiv:2209.15571*, 2022.

Omri Avrahami, Ohad Fried, and Dani Lischinski. Blended latent diffusion. *ACM transactions on graphics (TOG)*, 42(4):1–11, 2023.

Sherwin Bahmani, Jeong Joon Park, Despoina Paschalidou, Xingguang Yan, Gordon Wetzstein, Leonidas Guibas, and Andrea Tagliasacchi. Cc3d: Layout-conditioned generation of compositional 3d scenes. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 7171–7181, 2023.

Anh-Quan Cao, Angela Dai, and Raoul De Charette. Pasco: Urban 3d panoptic scene completion with uncertainty awareness. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 14554–14564, 2024.

Lin-Zhuo Chen, Jian Gao, Yihang Chen, Ka Leong Cheng, Yipengjing Sun, Liangxiao Hu, Nan Xue, Xing Zhu, Yujun Shen, Yao Yao, et al. Geometric context transformer for streaming 3d reconstruction. *arXiv preprint arXiv:2604.14141*, 2026a.

Weirong Chen, Chuanxia Zheng, Ganlin Zhang, Andrea Vedaldi, and Daniel Cremers. NOVA3R: Non-pixel-aligned visual transformer for amodal 3D reconstruction. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2026b.

Ruihang Chu, Enze Xie, Shentong Mo, Zhenguo Li, Matthias Nießner, Chi-Wing Fu, and Jiaya Jia. Diffcomplete: Diffusion-based generative 3d shape completion. *Advances in neural information processing systems*, 36:75951–75966, 2023.

Angela Dai, Angel X Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. ScanNet: Richly-annotated 3D reconstructions of indoor scenes. In *Proc. CVPR*, 2017.

Angela Dai, Daniel Ritchie, Martin Bokeloh, Scott Reed, Jürgen Sturm, and Matthias Nießner. Scancomplete: Large-scale scene completion and semantic segmentation for 3d scans. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4578–4587, 2018.

Angela Dai, Christian Diller, and Matthias Nießner. Sg-nn: Sparse generative neural networks for self-supervised scene completion of rgb-d scans. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 849–858, 2020.

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.

Paul Engstler, Andrea Vedaldi, Iro Laina, and Christian Rupprecht. Invisible stitch: Generating smooth 3D scenes with depth inpainting. In *Proceedings of the International Conference on 3D Vision (3DV)*, 2025.

Weixi Feng, Wanrong Zhu, Tsu-jui Fu, Varun Jampani, Arjun Akula, Xuehai He, Sugato Basu, Xin Eric Wang, and William Yang Wang. Layoutgpt: Compositional visual planning and generation with large language models. *Advances in Neural Information Processing Systems*, 36:18225–18250, 2023.

Xiaoguang Han, Zhen Li, Haibin Huang, Evangelos Kalogerakis, and Yizhou Yu. High-resolution shape completion using deep neural networks for global structure and local geometry inference. In *Proc. ICCV*, 2017.

Lukas Höllein, Ang Cao, Andrew Owens, Justin Johnson, and Matthias Nießner. Text2Room: Extracting textured 3D meshes from 2D text-to-image models. In *Proc. ICCV*, 2023.

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

Team Hunyuan3D, Shuhui Yang, Mingxin Yang, Yifei Feng, Xin Huang, Sheng Zhang, Zebin He, Di Luo, Haolin Liu, Yunfei Zhao, Qingxiang Lin, Zeqiang Lai, Xianghui Yang, Huiwen Shi, Zibo Zhao, Bowen Zhang, Hongyu Yan, Lifu Wang, Sicong Liu, Jihong Zhang, Meng Chen, Liang Dong, Yiwen Jia, Yulin Cai, Jiaao Yu, Yixuan Tang, Dongyuan Guo, Junlin Yu, Hao Zhang, Zheng Ye, Peng He, Runzhou Wu, Shida Wei, Chao Zhang, Yonghao Tan, Yifu Sun, Lin Niu, Shirui Huang, Bojian Zheng, Shu Liu, Shilin Chen, Xiang Yuan, Xiaofeng Yang, Kai Liu, Jianchen Zhu, Peng Chen, Tian Liu, Di Wang, Yuhong Liu, Linus, Jie Jiang, Jingwei Huang, and Chunchao Guo. Hunyuan3D 2.1: From images to high-fidelity 3D assets with production-ready PBR material. *arXiv*, 2506.15442, 2025a.

Team Hunyuan3D, Bowen Zhang, Chunchao Guo, Haolin Liu, Hongyu Yan, Huiwen Shi, Jingwei Huang, Junlin Yu, Kunhong Li, Penghao Wang, et al. Hunyuan3d-omni: A unified framework for controllable generation of 3d assets. *arXiv preprint arXiv:2509.21245*, 2025b.

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

HyunJun Jung, Weihang Li, Shun-Cheng Wu, William Bittner, Nikolas Brasch, Jifei Song, Eduardo Pérez-Pellitero, Zhensong Zhang, Arthur Moreau, Nassir Navab, and Benjamin Busam. Scrream : Scan, register, render and map:a framework for annotating accurate and dense 3d indoor scenes with a benchmark, 2024. URL `https://arxiv.org/abs/2410.22715`.

Michael Kazhdan and Hugues Hoppe. Screened poisson surface reconstruction. *ACM Transactions on Graphics (ToG)*, 32(3):1–13, 2013.

Michael Kazhdan, Matthew Bolitho, and Hugues Hoppe. Poisson surface reconstruction. In *Proceedings of the fourth Eurographics symposium on Geometry processing*, volume 7, 2006.

Nikhil Keetha, Norman Müller, Johannes Schönberger, Lorenzo Porzi, Yuchen Zhang, Tobias Fischer, Arno Knapitsch, Duncan Zauss, Ethan Weber, Nelson Antunes, Jonathon Luiten, Manuel Lopez-Antequera, Samuel Rota Bulò, Christian Richardt, Deva Ramanan, Sebastian Scherer, and Peter Kontschieder. MapAnything: universal feed-forward metric 3D reconstruction. *arXiv*, 2509.13414, 2025.

Vincent Leroy, Yohann Cabon, and Jerome Revaud. Grounding image matching in 3d with MASt3R. In *Proc. ECCV*, 2024.

Ming-Feng Li, Yueh-Feng Ku, Hong-Xuan Yen, Chi Liu, Yu-Lun Liu, Albert Y Chen, Cheng-Hao Kuo, and Min Sun. GenRC: generative 3D room completion from sparse image collections. In *Proc. ECCV*, 2024.

Yangguang Li, Zi-Xin Zou, Zexiang Liu, Dehu Wang, Yuan Liang, Zhipeng Yu, Xingchao Liu, Yuan-Chen Guo, Ding Liang, Wanli Ouyang, et al. Triposg: High-fidelity 3d shape synthesis using large-scale rectified flow models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2025a.

Zhen Li, Weiwei Sun, Shrisudhan Govindarajan, Shaobo Xia, Daniel Rebain, Kwang Moo Yi, and Andrea Tagliasacchi. NoKSR: kernel-free neural surface reconstruction via point cloud serialization. *arXiv*, 2502.12534, 2025b.

Haotong Lin, Sili Chen, Junhao Liew, Donny Y. Chen, Zhenyu Li, Guang Shi, Jiashi Feng, and Bingyi Kang. Depth anything 3: Recovering the visual space from any views. *arXiv*, 2511.10647, 2025.

Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. *arXiv.cs*, abs/2210.02747, 2022.

Bo Liu, Yifeng Zhu, Chongkai Gao, Yihao Feng, Qiang Liu, Yuke Zhu, and Peter Stone. LIBERO: benchmarking knowledge transfer for lifelong robot learning. In *Proc. NeurIPS*, 2023a.

Xingchao Liu, Chengyue Gong, and qiang liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. In *Proc. ICLR*, 2023b.

Yuzheng Liu, Siyan Dong, Shuzhe Wang, Yingda Yin, Yanchao Yang, Qingnan Fan, and Baoquan Chen. Slam3r: Real-time dense scene reconstruction from monocular rgb videos. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 16651–16662, 2025.

Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *IEEE International Conference on Computer Vision (ICCV)*, 2021.

Zhaoyang Lyu, Zhifeng Kong, Xudong Xu, Liang Pan, and Dahua Lin. A conditional point diffusion-refinement paradigm for 3D point cloud completion. In *Proc. ICLR*, 2021.

Dominic Maggio, Hyungtae Lim, and Luca Carlone. VGGT-SLAM: Dense RGB SLAM optimized on the $sl(4)$ manifold. In *Proc. NeurIPS*, 2025.

Quan Meng, Yujin Chen, Lei Li, Matthias Nießner, and Angela Dai. Seen2scene: Completing realistic 3d scenes with visibility-guided flow. *arXiv preprint arXiv:2603.28548*, 2026.

Riku Murai, Eric Dexheimer, and Andrew J Davison. Mast3r-slam: Real-time dense slam with 3d reconstruction priors. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 16695–16705, 2025.

Junfeng Ni, Yu Liu, Ruijie Lu, Zirui Zhou, Song-Chun Zhu, Yixin Chen, and Siyuan Huang. Decompositional neural scene reconstruction with generative diffusion prior. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 6022–6033, 2025.

Barbara Roessle, Norman Müller, Lorenzo Porzi, Samuel Rota Bulò, Peter Kontschieder, Angela Dai, and Matthias Nießner. L3DG: Latent 3D Gaussian diffusion. *arXiv*, 2410.13530, 2024.

Manolis Savva, Abhishek Kadian, Oleksandr Maksymets, Yili Zhao, Erik Wijmans, Bhavana Jain, Julian Straub, Jia Liu, Vladlen Koltun, Jitendra Malik, Devi Parikh, and Dhruv Batra. Habitat: A Platform for Embodied AI Research. In *Proc. ICCV*, 2019.

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 Lutz Schönberger and Jan-Michael Frahm. Structure-from-motion revisited. In *Proc. CVPR*, 2016.

Jaidev Shriram, Alex Trevithick, Lingjie Liu, and Ravi Ramamoorthi. RealmDreamer: text-driven 3d scene generation with inpainting and depth diffusion. In *Proc. 3DV*, 2025.

Yawar Siddiqui, Duncan Frost, Samir Aroudj, Armen Avetisyan, Henry Howard-Jenkins, Daniel DeTone, Pierre Moulon, Qirui Wu, Zhengqin Li, Julian Straub, et al. Shaper: Robust conditional 3d shape generation from casual captures. *arXiv preprint arXiv:2601.11514*, 2026.

Julian Straub, Thomas Whelan, Lingni Ma, Yufan Chen, Erik Wijmans, Simon Green, Jakob J Engel, Raul Mur-Artal, Carl Ren, Shobhit Verma, et al. The Replica dataset: A digital replica of indoor spaces. *arXiv*, 1906.05797, 2019.

Libo Sun, Jia-Wang Bian, Huangying Zhan, Wei Yin, Ian D. Reid, and Chunhua Shen. Sc-depthv3: Robust self-supervised monocular depth estimation for dynamic scenes. *PAMI*, 46(1), 2024.

Andrew Szot, Alex Clegg, Eric Undersander, Erik Wijmans, Yili Zhao, John Turner, Noah Maestre, Mustafa Mukadam, Devendra Chaplot, Oleksandr Maksymets, Aaron Gokaslan, Vladimir Vondrus, Sameer Dharur, Franziska Meier, Wojciech Galuba, Angel Chang, Zsolt Kira, Vladlen Koltun, Jitendra Malik, Manolis Savva, and Dhruv Batra. Habitat 2.0: Training home assistants to rearrange their habitat. In *Proc. NeurIPS*, 2021.

SAM 3D Team, Xingyu Chen, Fu-Jen Chu, Pierre Gleize, Kevin J Liang, Alexander Sax, Hao Tang, Weiyao Wang, Michelle Guo, Thibaut Hardin, Xiang Li, Aohan Lin, Jiawei Liu, Ziqi Ma, Anushka Sagar, Bowen Song, Xiaodong Wang, Jianing Yang, Bowen Zhang, Piotr Dollár, Georgia Gkioxari, Matt Feiszli, and Jitendra Malik. SAM 3D: 3Dfy anything in images. *arXiv*, 2511.16624, 2025.

Zachary Teed and Jia Deng. DROID-SLAM: deep visual SLAM for monocular, stereo, and RGB-D cameras. In *Proc. NeurIPS*, 2021.

Zachary Teed, Lahav Lipson, and Jia Deng. Deep patch visual odometry. *arXiv.cs*, abs/2208.04726, 2022.

Arash Vahdat, Francis Williams, Zan Gojcic, Or Litany, Sanja Fidler, Karsten Kreis, et al. Lion: Latent point diffusion models for 3d shape generation. *Advances in neural information processing systems*, 35:10021–10039, 2022.

Jianyuan Wang, Minghao Chen, Nikita Karaev, Andrea Vedaldi, Christian Rupprecht, and David Novotny. VGGT: Visual geometry grounded transformer. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025a.

Jianyuan Wang, Minghao Chen, Shangzhan Zhang, Nikita Karaev, Johannes Schönberger, Patrick Labatut, Piotr Bojanowski, David Novotny, Andrea Vedaldi, and Christian Rupprecht. Vggt-ω. In *Proc. CVPR*, 2026.

Qianqian Wang, Yifei Zhang, Aleksander Holynski, Alexei A Efros, and Angjoo Kanazawa. Continuous 3d perception model with persistent state. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 10510–10522, 2025b.

Shuzhe Wang, Vincent Leroy, Yohann Cabon, Boris Chidlovskii, and Jerome Revaud. DUSt3R: Geometric 3D vision made easy. In *Proc. CVPR*, 2024.

Yifan Wang, Jianjun Zhou, Haoyi Zhu, Wenzheng Chang, Yang Zhou, Zizun Li, Junyi Chen, Jiangmiao Pang, Chunhua Shen, and Tong He. π^3: Permutation-equivariant visual geometry learning. *arXiv preprint arXiv:2507.13347*, 2025c.

Tianhao Wu, Chuanxia Zheng, Frank Guan, Andrea Vedaldi, and Tat-Jen Cham. Amodal3R: Amodal 3D reconstruction from occluded 2D images. In *Proceedings of the International Conference on Computer Vision (ICCV)*, 2025.

Zhennan Wu, Yang Li, Han Yan, Taizhang Shang, Weixuan Sun, Senbo Wang, Ruikai Cui, Weizhe Liu, Hiroyuki Sato, Hongdong Li, and Pan Ji. BlockFusion: Expandable 3D scene generation using latent tri-plane extrapolation. *arXiv.cs*, 2024.

Jiatong Xia, Zicheng Duan, Anton van den Hengel, and Lingqiao Liu. Points-to-3d: Structure-aware 3d generation with point cloud priors. *arXiv preprint arXiv:2603.18782*, 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. *arXiv*, 2412.01506, 2024.

Jianfeng Xiang, Xiaoxue Chen, Sicheng Xu, Ruicheng Wang, Zelong Lv, Yu Deng, Hongyuan Zhu, Yue Dong, Hao Zhao, Nicholas Jing Yuan, and Jiaolong Yang. Native and compact structured latents for 3D generation. *arXiv*, 2512.14692, 2025.

Mingyang Xie, Numair Khan, Tianfu Wang, Naina Dhingra, Seonghyeon Nam, Haitao Yang, Zhuo Hui, Christopher Metzler, Andrea Vedaldi, Hamed Pirsiavash, and Lei Luo. LaVR: Latent space conditioned video re-rendering using large 4D reconstruction models. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2026.

Jianing Yang, Alexander Sax, Kevin J Liang, Mikael Henaff, Hao Tang, Ang Cao, Joyce Chai, Franziska Meier, and Matt Feiszli. Fast3R: towards 3D reconstruction of 1000+ images in one forward pass. *Proc. CVPR*, 2025a.

Yandan Yang, Baoxiong Jia, Peiyuan Zhi, and Siyuan Huang. PhyScene: physically interactable 3d scene synthesis for embodied AI. In *Proc. CVPR*, 2024a.

Yu-Qi Yang, Yu-Xiao Guo, Jian-Yu Xiong, Yang Liu, Hao Pan, Peng-Shuai Wang, Xin Tong, and Baining Guo. Swin3d: A pretrained transformer backbone for 3d indoor scene understanding. *Computational Visual Media*, 11(1):83–101, 2025b.

Yue Yang, Fan-Yun Sun, Luca Weihs, Eli VanderBilt, Alvaro Herrasti, Winson Han, Jiajun Wu, Nick Haber, Ranjay Krishna, Lingjie Liu, Chris Callison-Burch, Mark Yatskar, Aniruddha Kembhavi, and Christopher Clark. Holodeck: Language guided generation of 3d embodied AI environments. In *Proc. CVPR*, 2024b.

Chandan Yeshwanth, Yueh-Cheng Liu, Matthias Nießner, and Angela Dai. ScanNet++: a high-fidelity dataset of 3d indoor scenes. In *IEEE International Conference on Computer Vision (ICCV)*, 2023.

Xumin Yu, Yongming Rao, Ziyi Wang, Zuyan Liu, Jiwen Lu, and Jie Zhou. Pointr: Diverse point cloud completion with geometry-aware transformers. *arXiv.cs*, abs/2108.08839, 2021.

Wentao Yuan, Tejas Khot, David Held, Christoph Mertz, and Martial Hebert. Pcn: Point completion network. In *2018 international conference on 3D vision (3DV)*, pp. 728–737. IEEE, 2018.

Biao Zhang, Jiapeng Tang, Matthias Niessner, and Peter Wonka. 3DShape2VecSet: A 3D shape representation for neural fields and generative diffusion models. In *ACM Transactions on Graphics*, 2023.

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. In *Proc. SIGGRAPH*, 2024.

Weipeng Zhong, Peizhou Cao, Yichen Jin, Li Luo, Wenzhe Cai, Jingli Lin, Hanqing Wang, Zhaoyang Lyu, Tai Wang, Bo Dai, et al. Internscenes: A large-scale simulatable indoor scene dataset with realistic layouts. *arXiv preprint arXiv:2509.10813*, 2025.

Haoran Zhou, Yun Cao, Wenqing Chu, Junwei Zhu, Tong Lu, Ying Tai, and Chengjie Wang. Seedformer: Patch seeds based point cloud completion with upsample transformer. In *European conference on computer vision*, pp. 416–432. Springer, 2022.

Linqi Zhou, Yilun Du, and Jiajun Wu. 3D shape generation and completion through point-voxel diffusion. In *Proc. CVPR*, 2021.

A Appendix

A.1 Training Data Construction Details

For each `InternScenes` room we place 1–9 camera anchors by floor area, clear of furniture, and render three pitch rings at $-10°$, $-45°$, and $+50°$ from each, with 10 uniformly spaced yaw angles per ring, giving 30 frames per anchor. Zenith and nadir views and corner cameras facing the room interior cover the boundaries that centre-facing views miss. The sampler yields 40 to roughly 420 frames per scene depending on room size, 106 on average; training draws random subsets of these to form the partial point cloud.

A.2 More Implementation Details

Stage-1. The Stage-1 denoiser operates on the $32^3 \times 8$ TRELLIS sparse-structure latent, with windowed self-attention across the two levels of a UNet and adaptive layer-norm time conditioning. Cross-attention attends to 768 tokens produced by the frozen NOVA3R encoder from the 50k-point input cloud. Two modifications make it visibility-aware: (i) a *warm-started prior* that voxelizes the input cloud at 128^3, runs the frozen TRELLIS SS-encoder to obtain $z_{\text{vis}} \in \mathbb{R}^{8\times 32^3}$, and replaces the Gaussian initial noise at visible voxels by z_{vis}; and (ii) a VisibleMaskConcat adapter, which concatenates the binary visibility mask with the noisy latent at each denoising step and maps the result back to 8 channels with an identity-initialized 3D convolution. After 50 flow-sampling steps with classifier-free guidance, the latent is decoded by the unmodified TRELLIS sparse-structure decoder into a 128^3 occupancy field, whose active voxel coordinates are exported to Stage 2.

Stage-2. The Stage-2 denoiser is *sparse*, operating directly on the active-coord tensor produced by Stage 1. Tokens are formed by patchifying the 128^3 sparse grid with patch size 2, giving an effective 64^3 token grid sparsified to the active set. Positional encoding is local-RoPE inside each Swin window rather than a global APE, so the same operator applies to scenes of differing spatial extent. Cross-attention reuses the Stage-1 tokens. After 50 flow steps with CFG 3.0, the predicted SLAT is fed to the unmodified TRELLIS mesh decoder.

Metric definitions. Let $\mathcal{P}$ be points sampled from the predicted mesh, $\mathcal{G}$ points sampled from the complete ground-truth mesh, and $\mathcal{G}_{\text{vis}}$ the ground-truth points returned by raycasting the input cameras, so $\mathcal{G}_{\text{vis}}$ is occlusion-aware. Let $\mathcal{P}_{\text{vis}}$ be the subset of $\mathcal{P}$ inside the union of the input view frustums; this is a geometric frustum test, not an occlusion test, since occlusion of the prediction cannot be resolved without raycasting it. With $d(x, \mathcal{S}) = \min_{y\in\mathcal{S}} \|x - y\|$,

$$a = \frac{1}{|\mathcal{P}_{\text{vis}}|} \sum_{x\in\mathcal{P}_{\text{vis}}} d(x, \mathcal{G}), \qquad c_{\text{vis}} = \frac{1}{|\mathcal{G}_{\text{vis}}|} \sum_{y\in\mathcal{G}_{\text{vis}}} d(y, \mathcal{P}), \qquad c_{\text{all}} = \frac{1}{|\mathcal{G}|} \sum_{y\in\mathcal{G}} d(y, \mathcal{P}),$$

$$\text{CD}_{\text{vis}} = \tfrac{1}{2}(a + c_{\text{vis}}), \qquad \text{CD}_{\text{all}} = \tfrac{1}{2}(a + c_{\text{all}}), \qquad \text{F}_\tau = \tfrac{2P_\tau R_\tau}{P_\tau + R_\tau},$$

with $P_\tau = |\{x \in \mathcal{P}_{\text{vis}} : d(x, \mathcal{G}) < \tau\}|/|\mathcal{P}_{\text{vis}}|$ and $R_\tau = |\{y \in \mathcal{G} : d(y, \mathcal{P}) < \tau\}|/|\mathcal{G}|$. Precision is therefore restricted to the observed region while recall covers the scene, so geometry generated outside the frustums is not charged. We sample 10^5 points per mesh.

Evaluation Datasets. For Replica (Straub et al., 2019), we use 8 of the 18 scenes, following Ni et al. (2025). For SCRREAM (Jung et al., 2024), we exclude scenes whose videos focus only on a single corner of the scene.

A.3 Additional Qualitative Results

Real-world scans. Figure 7 takes the model outside the benchmarks, onto two real-world rooms captured with a 3D scanner (Yeshwanth et al., 2023). Input sits beside output: top-down, then two viewpoints inside the room. The inside views show what the top-down hides, that the input thins into scattered points wherever the scanner did not reach, while the output is a closed surface. Being real scans, they have no complete ground-truth mesh, which is why they are shown qualitatively rather than scored: the meshes this dataset provides are themselves incomplete, so complete geometry for these rooms was never available as a training target either (Section 4.1).

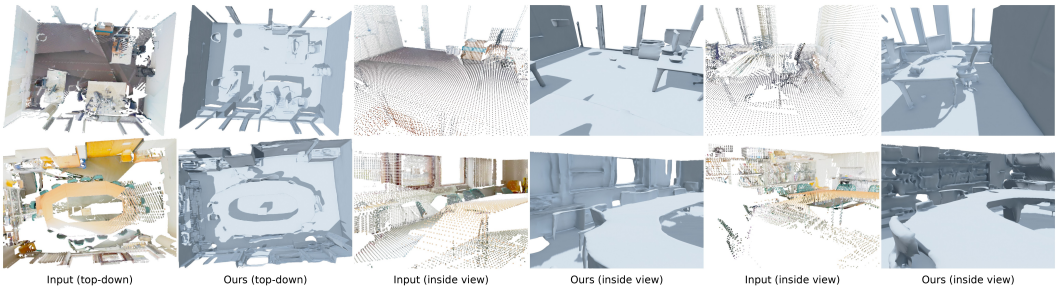


Figure 7: **Real-world scans.** Two real-world rooms captured with a 3D scanner (Yeshwanth et al., 2023), one per row. Columns alternate input and output: top-down, then two viewpoints inside the room.

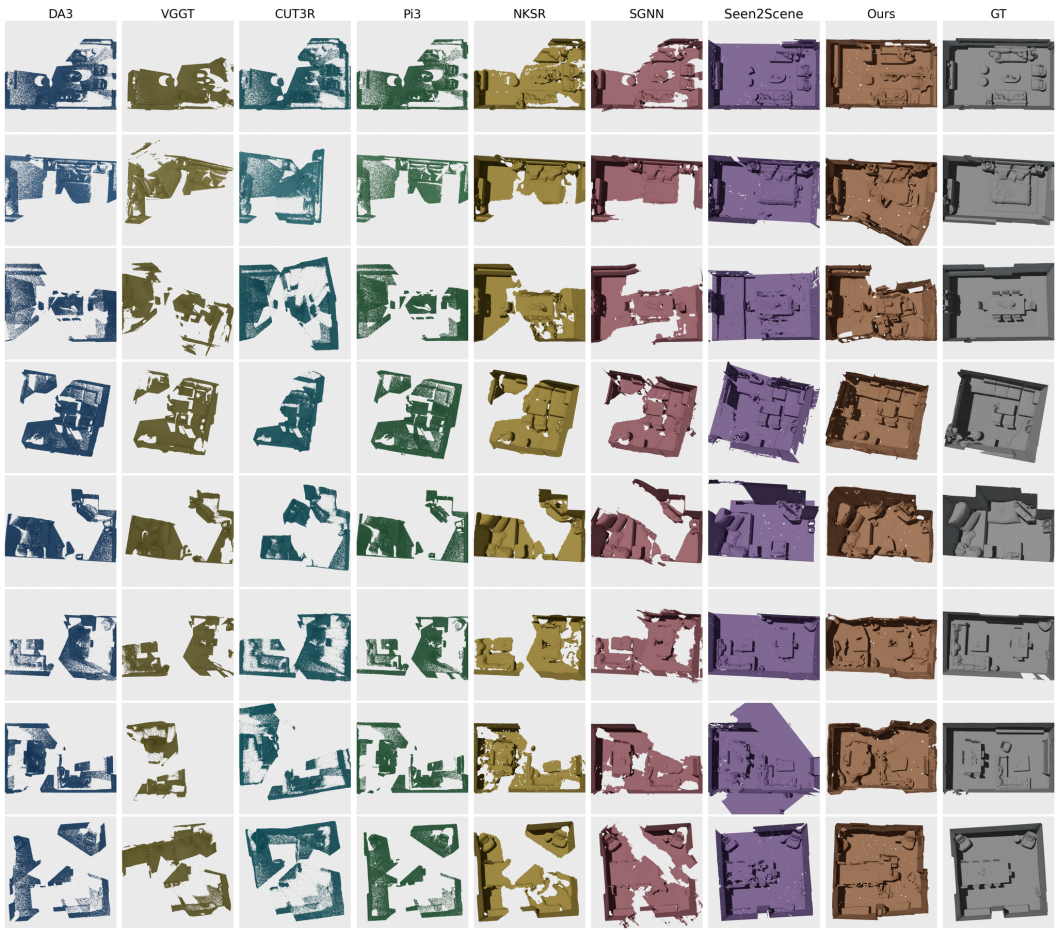


Figure 8: **Additional qualitative results on Replica (Straub et al., 2019) at N=4.** Top-down views of the meshes each method reconstructs from the same partial DA3 point cloud, with the ground truth in the last column.

More reconstructions on Replica and SCRREAM. We provide additional top-down qualitative reconstructions on Replica (Straub et al., 2019) and SCRREAM (Jung et al., 2024) at N_{views}=4 in Figures 8 and 9, respectively, comparing the input partial point cloud against representative baselines and AnchorComplete. Across both datasets, our method recovers a complete room mesh, including floors, walls, and large objects in regions where the partial input has no coverage, while baselines leave visible holes wherever the cameras did not directly observe.

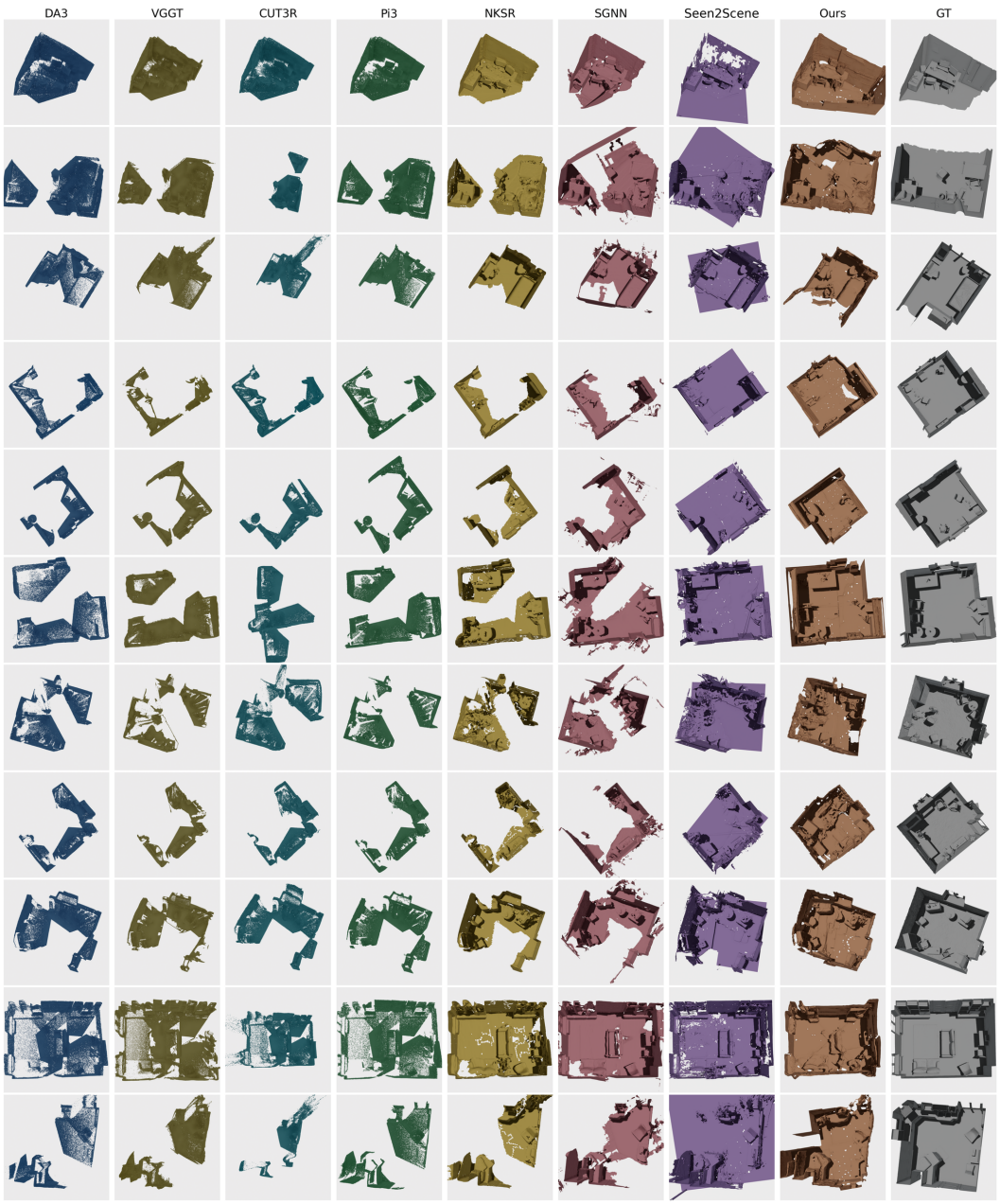


Figure 9: **Additional qualitative results on SCRREAM (Jung et al., 2024) at $N{=}4$.** Top-down views of the meshes each method reconstructs from the same partial DA3 point cloud, with the ground truth in the last column.

A.4 Generalization Across Input Sources

The experiments above condition on point clouds produced by DA3 (Lin et al., 2025), a per-view point-map estimator. Because AnchorComplete conditions directly on points rather than on images or estimator-specific features, it is not architecturally tied to this source. To test this, we apply the same pretrained checkpoint, with no retraining or source-specific tuning, to partial point clouds from four acquisition paradigms: learned per-view point maps (DA3, Pi3 (Wang et al., 2025c), VGGT-Ω (Wang et al., 2026)), classical structure-from-motion and multi-view stereo (COLMAP (Schönberger & Frahm, 2016)), monocular visual SLAM (DROID-SLAM (Teed & Deng, 2021), DPVO (Teed et al.,

Table 3: **Generalization across input point-cloud sources on Replica (Straub et al., 2019).** The same pretrained ANCHORCOMPLETE checkpoint is evaluated without retraining on partial point clouds from multiple acquisition paradigms, using the default DA3 evaluation protocol (Table 1). Visual-SLAM and LiDAR use full trajectories/scans and therefore demonstrate source generality rather than budget-matched performance. † LiDAR uses conservative $5\,\mathrm{cm}$ range noise.

Category	**Input source**	$\mathrm{CD}_{\mathrm{vis}} \downarrow$	$\mathrm{CD}_{\mathrm{all}} \downarrow$	$\mathrm{F}_{0.1} \uparrow$	$\mathrm{F}_{0.05} \uparrow$
Per-view point map	DA3 (default)	0.053	0.071	0.814	0.682
	Pi3	0.047	0.067	0.822	0.641
	VGGT-Ω	0.066	0.082	0.748	0.555
SfM/MVS	COLMAP	0.067	0.085	0.760	0.618
Visual SLAM	DROID-SLAM	0.043	0.055	0.872	0.761
	DPVO	0.048	0.060	0.863	0.682
LiDAR	32-beam, σ=5 cm†	0.040	0.044	0.942	0.744

2022)), and a simulated 32-beam LiDAR scan. All inputs are registered into the same evaluation frame and scored with the protocol used for the default DA3 setting.

Table 3 shows that ANCHORCOMPLETE remains effective across every source we test, with $\mathrm{F}_{0.05}$ ranging from 0.555 to 0.761. VGGT-Ω is the weakest of them and DROID-SLAM the strongest; the same checkpoint handles both without tuning. In particular, the SfM/MVS, visual-SLAM, and LiDAR rows confirm that the method is not restricted to the per-view point-map setting used elsewhere in the paper.

A.5 SYMMETRIC FULL-SCENE CHAMFER

The accuracy term of Table 1 counts only predicted points inside the input frustums, so geometry generated in unobserved space is never charged. Table 4 repeats $\mathrm{CD}_{\mathrm{all}}$ with accuracy taken over all predicted points instead. The baselines barely move, since they place almost no geometry outside the frustums; ours rises, which is what generating into unobserved space costs. The ordering is unchanged on both datasets. We keep the asymmetric form as the headline because a completion that differs from the recorded ground truth in unobserved space is not thereby wrong.

Table 4: **Charging for generated geometry** (N_{views}=4, metres ↓). $\mathrm{CD}_{\mathrm{all}}$ is the value reported in Table 1, whose accuracy term counts only predicted points inside the input frustums; $\mathrm{CD}_{\mathrm{all}}^{\mathrm{sym}}$ takes accuracy over all predicted points.

Method	**Replica**		**SCRREAM**	
	$\mathrm{CD}_{\mathrm{all}}$	$\mathrm{CD}_{\mathrm{all}}^{\mathrm{sym}}$	$\mathrm{CD}_{\mathrm{all}}$	$\mathrm{CD}_{\mathrm{all}}^{\mathrm{sym}}$
DA3 (Lin et al., 2025)	0.220	0.220	0.227	0.228
NKSR (Huang et al., 2023)	0.211	0.211	0.209	0.208
Seen2Scene (Meng et al., 2026)	<u>0.132</u>	<u>0.134</u>	<u>0.167</u>	<u>0.175</u>
ANCHORCOMPLETE (ours)	**0.071**	**0.098**	**0.075**	**0.110**

A.6 FAILURE CASES

Figure 6b perturbs the point cloud synthetically, which is not how a depth estimator actually fails. To probe that, we degrade the *images* DA3 consumes and re-run the full pipeline on the resulting clouds. Figure 10 shows a SCRREAM scene under low light and a ScanNet++ scene under motion blur and sensor noise, both simulated on real captures. The SCRREAM scene is held out; the ScanNet++ one is in our Stage-1 training set, so the low-light row carries the weight of the claim.

The degraded imagery costs DA3 large parts of the room, and the reconstruction inherits exactly those gaps: ANCHORCOMPLETE still closes the surface it is given and produces a coherent room, but it completes around the missing observations rather than recovering them. The model is trained only

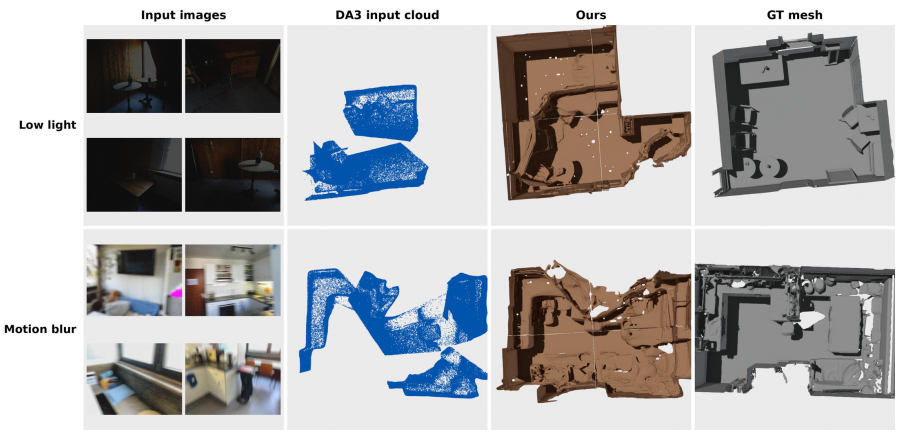


Figure 10: **Failure cases under degraded imagery.** Real captures with simulated low light (top) and motion blur/noise (bottom). The degraded inputs cause DA3 to miss large scene regions, and our model completes around these gaps rather than recovering them.

on clean geometry, so it treats the input as a reliable anchor and preserves systematic upstream error rather than repairing it. Training with degraded inputs is a natural remedy we leave for future work.

A.7 LIMITATIONS

While ANCHORCOMPLETE is robust to sparse input views (Section 4.3), it still has several limitations. First, the current model predicts geometry only: we use the inherited mesh decoder and do not model appearance. Extending the framework with RGB supervision and an appearance-aware decoder could enable textured mesh generation. Second, performance depends on the quality of the upstream geometry, so drift, scale errors, or systematic depth bias may propagate to the generated mesh. Section A.6 shows the sharper form of this: severe upstream failures are inherited rather than corrected. Third, our training and evaluation focus on indoor scenes, and generalization to outdoor, unbounded, or substantially different scene distributions remains open.