TranSpace: Progressive Anchoring for Metric-Consistent Scene Synthesis

Hyeshim Kim
Korea Advanced Institute of Science and Technology
Daejeon, Republic of Korea
hyedeep@kaist.ac.kr

Taehei Kim*
Korea Advanced Institute of Science and Technology
Daejeon, Republic of Korea
hayleyy321@kaist.ac.kr

Jihun Shin*
Korea Advanced Institute of Science and Technology
Daejeon, Republic of Korea
jihun.shin@kaist.ac.kr

Hyeonjin Kim
Korea Advanced Institute of Science and Technology
Daejeon, Republic of Korea
hyeonjin@kaist.ac.kr

Sung-Hee Lee
Korea Advanced Institute of Science and Technology
Daejeon, Republic of Korea
sunghee.lee@kaist.ac.kr

Figure 1: TranSpace synthesizes transition geometry between separately captured scenes. Each row shows a different structure—wing wall, pony wall, and internal window (top to bottom)—viewed along an arc trajectory. Scenes from the Deep Blending dataset [14].

Abstract

Connecting two photo-realistic reconstructed spaces is largely unexplored yet opens new opportunities in spatial dataset creation and immersive telepresence. While existing works focus on single room-wise generation, we present TranSpace, a method that synthesizes transition geometry to seamlessly connect two distinct reconstructed spaces via video generation models. This setting introduces a fundamental challenge: the transition region requires plausible hallucination, while the pre-existing geometry must be strictly preserved. Because video diffusion operates in pixel space rather than metric space, this asymmetry exposes the 3D inconsistency of video diffusion—the tendency to gradually distort existing geometry or hallucinate artifacts that degrade downstream novel view synthesis. To address this, we introduce Progressive Anchoring, an iterative scheme that uses camera pose estimation as a quality gate for generation. Two mechanisms are central: volumetric blending fuses generated frames with pre-existing geometry to preserve trusted regions, and bidirectional PnP localization estimates camera poses against blended references to bound error accumulation from both endpoints. We evaluate against state-of-the-art camera-conditioned video generation and pose estimation baselines, demonstrating improvements across PSNR, SSIM, and LPIPS metrics. A user study further confirms superior perceptual quality over text-to-3D scene generation and 3D inpainting methods.

CCS Concepts

• **Computing methodologies** → **Reconstruction**; **Image-based rendering**; *Shape modeling*.

*Equal contribution.

MM '26, Rio de Janeiro, Brazil

ACM ISBN 979-8-4007-2213-4/2026/11
https://doi.org/10.1145/3767308.3836316

Keywords

Generative 3D Scene Synthesis, Video Generation Model, Camera Pose Estimation

ACM Reference Format:

Hyeshim Kim, Taehei Kim, Jihun Shin, Hyeonjin Kim, and Sung-Hee Lee. 2026. TranSpace: Progressive Anchoring for Metric-Consistent Scene Synthesis. In *Proceedings of the 34th ACM International Conference on Multimedia (MM '26), November 10–14, 2026, Rio de Janeiro, Brazil.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3767308.3836316

1 Introduction

Photo-realistic 3D reconstruction of indoor spaces has advanced rapidly, improving realism in immersive multimedia. As 3D scan data becomes widely available, new scene-level applications are emerging: VR-GS [17] makes reconstructed objects physically interactive, Voronoi Rooms [18] splits and merges users' scanned rooms for immersive telepresence, and GaussianNexus [16] enables room-scale telepresence through real-time Gaussian Splatting. Beyond using complete room scans, an appealing direction is to connect components from different rooms by synthesizing intermediate regions, creating more diverse indoor spaces for immersive multimedia applications—from architectural walkthroughs and autonomous agent training to mixed-reality telepresence. Yet bridging independently captured scenes remains largely unaddressed.

Recent progress in 3D scene generation enables creating large environments from scratch. Some methods iteratively expand scenes outward from a single view [8, 12, 15, 41, 42], while others aim for global semantic and geometric consistency [9, 22, 31, 33, 39]. In a complementary direction, 3D inpainting fills missing regions within a single scene and optimizes consistency in a shared reference frame [7, 11, 23, 25, 36].

Building upon previous works, we tackle a different problem: connecting two independently captured scenes that exhibit discontinuities in lighting, boundary geometry, and architectural style. Unlike room-wise generation or local inpainting within a single scene, the synthesized region must satisfy constraints from two fixed endpoints simultaneously. To guide our approach, we draw inspiration from two architectural concepts: transition space [4] and interior windows [1]. Transition space is a deliberate buffer zone that creates a psychological and experiential link between distinct spatial entities. Interior windows, often appearing as frames resembling traditional doors or windows, serve to visually articulate a boundary while simultaneously acting as a connector. Together, these two elements support a natural transition between distinct spaces. We translate these architectural principles into a computational framework for generating volumetric transition zones that are not merely visual overlays but geometrically plausible spatial connectors.

Transition synthesis poses two key challenges. First, the connector must match its surroundings in geometry, material appearance, and reconstruction density. Because we target novel view synthesis (NVS), the transition region must provide dense, multi-view-consistent observations comparable to real captures. Second, the two endpoint scenes must remain unchanged: generation should not distort or corrupt existing regions. To address these challenges, we leverage off-the-shelf video generation models, which produce highly realistic and temporally coherent sequences from large-scale visual priors. In particular, start-to-end frame interpolation naturally suits transition synthesis by preserving both endpoints and filling the bounded region. However, naively applying video generation leads to 3D inconsistency: producing plausible 2D sequences without explicit 3D reasoning, the model may distort existing geometry or hallucinate unwanted objects, degrading downstream reconstruction.

Recently, camera-conditioned generation approaches [26, 27, 43, 45] have shown promising results by prescribing camera trajectories during video generation. However, these methods require model fine-tuning and offer no mechanism for detecting generation failures. Inspired by InterPose [6], we explore an alternative: recovering poses after generation and using pose estimation as a quality gate to filter unreliable frames. However, state-of-the-art camera pose estimation methods [21, 34] estimate poses in their own coordinate frame and require post-hoc alignment to existing reconstructions, offering no guarantee of metric consistency. To address this, we propose Progressive Anchoring: we iteratively generate new frames and localize each against volumetrically blended anchors that composite generated content with rendered geometry from the fixed scenes, preventing error accumulation and maintaining metric consistency as the transition region grows.

In summary, we present TranSpace, a method for synthesizing metric-consistent transitions between separately captured indoor spaces. Our key contributions are:

- We formulate the problem of bounded transition synthesis: generating plausible 3D geometry that connects two fixed reconstructed scenes while satisfying constraints from both endpoints simultaneously.
- We propose Progressive Anchoring, an iterative scheme that uses pose estimation as a quality gate for video generation, mitigating 3D inconsistency through volumetric blending and bidirectional PnP localization without requiring camera-conditioned models or fine-tuning.
- We evaluate against camera-conditioned video generation and pose estimation baselines, achieving higher metric consistency in the reconstructed transition region. A user study further shows a perceptual preference over global scene generation and local inpainting baselines.

2 Related Work

2.1 Scene Generation and 3D Inpainting

Diffusion-based 3D scene generation has advanced rapidly. Early works such as SceneScape [12], Text2Room [15] and Text2NeRF [44] iteratively expanded scenes outward from a single viewpoint through warping-inpainting loops. WonderJourney [42] and WonderWorld [41] extended this paradigm to long-range exploration. To improve global coherence, MVDiffusion [33] introduced correspondence-aware multi-view generation, ControlRoom3D [31] and SceneCraft [39] incorporated bounding-box proxies for layout control, and WorldExplorer [29] enabled long-range camera exploration through memory-aware video generation. Despite their diversity, these methods generate entire scenes from scratch under a unified reference frame, either expanding outward from a single origin or optimizing global consistency within a single scene.

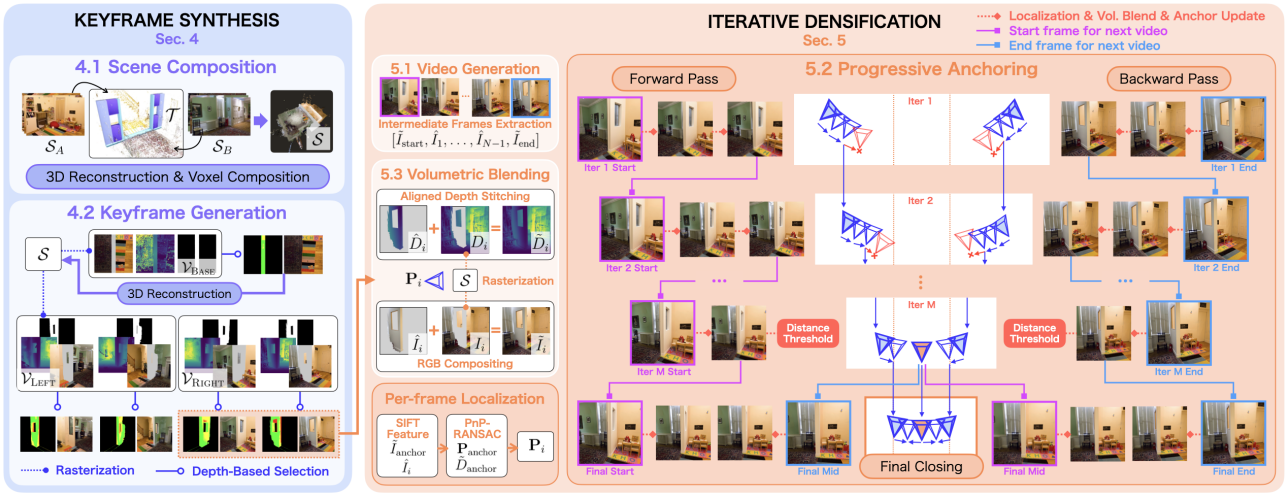

Figure 2: Overview of TranSpace pipeline. From two anchor spaces, we compose a unified scene with placeholder voxels defining the transition layout. Keyframes are inpainted, then densified into continuous sequences by video diffusion. Progressive Anchoring localizes each frame against blended anchors, yielding metrically consistent observations for novel view synthesis.

Complementarily, 3D inpainting completes missing regions within an existing reconstruction. SPIn-NeRF [25] pioneered multi-view-consistent object removal via perceptual optimization, and subsequent works improved robustness through diffusion-based depth completion [23], multi-view score distillation [7], and the Grid Prior for joint multi-view sampling in NeRFiller [36]. ObjFiller-3D [11] further demonstrated that video diffusion yields more coherent completions than frame-wise 2D inpainting. However, all inpainting methods assume a single underlying scene and optimize consistency within a unified coordinate frame.

In contrast, our work synthesizes transition regions that must simultaneously satisfy geometric constraints from two independently captured scenes—a bounded two-endpoint problem that neither outward expansion nor single-scene inpainting addresses.

2.2 Camera-Conditioned Generation and Pose Estimation

Recent work injects camera information into video generation to improve controllability and 3D consistency. Implicit approaches encode camera poses as auxiliary signals [2, 3, 13, 19], but struggle to generalize beyond their training distributions. Explicit geometry-conditioned methods [26, 43] construct intermediate 3D representations as conditioning inputs. Gen3C [27] maintains a progressively updated 3D cache concatenated with noisy latents. Stable Virtual Camera [45] achieves multi-view consistency through 3D self-attention without explicit geometry. While these methods improve self-consistency among generated views, unobserved or occluded regions remain prone to hallucinated geometry that conflicts with existing scene content. Furthermore, all camera-conditioned approaches require model fine-tuning, limiting their applicability and imposing substantial memory requirements. Because these methods prescribe camera poses and generate all frames in a single pass, they have no mechanism for detecting or recovering from generation failures.

An alternative strategy, inspired by InterPose [6] and pursued concurrently by [37], is to recover camera poses after generation rather than prescribing them. Classical Structure-from-Motion [30] suffers from large trajectory drift caused by 3D inconsistency in generated content, while transformer-based methods [21, 34, 35] improve robustness through learned dense 3D prediction and feature matching. However, these methods estimate poses in their own coordinate frame and require post-hoc alignment to existing reconstructions, introducing residual registration error. Camera pose estimation alone cannot guarantee metric consistency when generated frames lack grounded anchors.

Our method bridges both paradigms: we maintain metric fidelity to both endpoint reconstructions by iterative localization and volumetric blending while using pose estimation as a principled filter for video generation quality.

3 Overview

Our goal is to address an asymmetry challenge: anchor spaces (the two pre-existing reconstructions) must remain metrically authentic while the transition region requires plausible hallucination to fill in unobserved geometry. Unlike methods that treat all pixels uniformly, our pipeline must fundamentally distinguish between these regions. We first build a composite scene $\mathcal{S}$ with placeholder voxels defining the transition layout (Section 4.1), then render and inpaint strategic viewpoints with depth-based candidate selection (Section 4.2). Video diffusion densifies these sparse keyframes into continuous sequences (Section 5.1). To recover metric-consistent poses, we introduce Progressive Anchoring (Section 5.2): an iterative scheme that localizes each frame against blended anchors—frames that composite generated content with rendered geometry, preserving metric grounding. Volumetric blending (Section 5.3) produces both training data for reconstruction and anchors for subsequent localization. The result is a dense set of blended frames with metric-consistent camera poses spanning the transition region.

These observations are then combined with the composite scene $\mathcal{S}$ to yield a complete scene suitable for novel view synthesis.

External Components. Our pipeline leverages several off-the-shelf components: Google Vertex AI Imagen 2 [28] for keyframe inpainting, Depth-Anything-V2-Metric-Indoor [38] for monocular metric depth estimation, DeeVid AI Video Generation V2.1 [10] for intermediate frame densification, and SVRaster [32] for high-fidelity novel view synthesis via adaptive sparse voxel rasterization.

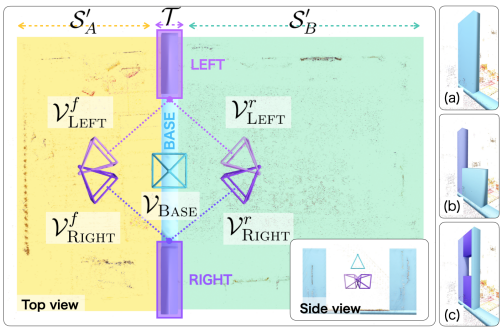

Figure 3: Scene composition and blueprinting. The composite scene combines pruned anchor spaces $\mathcal{S}'_A$, $\mathcal{S}'_B$, and transition region $\mathcal{T}$, subdivided into Base, Left, and Right with corresponding viewpoints. Right: (a) wing wall, (b) pony wall, (c) internal window.

4 Blueprint-Guided Keyframe Synthesis

4.1 Scene Composition

Synthesizing a transition region between two captured spaces requires careful spatial planning to ensure geometric consistency. Given two anchor spaces $\mathcal{S}_A$ and $\mathcal{S}_B$, each consisting of RGB images I, camera intrinsics $\mathbf{K}$, and camera-to-world poses $\mathbf{P}$, we obtain 3D reconstructions using SVRaster [32]. For the transition region between the two spaces, we use axis-aligned bounding boxes to define a coarse layout.

Since our goal is to synthesize connecting geometry that seamlessly bridges two existing reconstructions, we build a composite 3D scene that provides the generative model with visual context from both anchor spaces:

$$\mathcal{S} = \mathcal{S}'_A \oplus \mathcal{S}'_B \oplus \mathcal{T}, \tag{1}$$

where $\oplus$ denotes voxel composition and $\mathcal{T}$ represents the transition region as a set of empty voxels defined by bounding box primitives. Specifically, $\mathcal{S}'_A$ and $\mathcal{S}'_B$ are each pruned to retain only voxels on their respective sides of the transition center, preventing overlap while preserving context for generation as shown in Figure 3.

Our pipeline necessitates distinguishing between pixels that can be trusted metrically (from anchor spaces) versus pixels that are hallucinated (from the transition region). We achieve this by assigning each voxel a space identifier $s \in \{A, B, T\}$ by its source region $\mathcal{S}'_A$, $\mathcal{S}'_B$, or $\mathcal{T}$. The identifiers enable selective masking: transition pixels are masked for inpainting while anchor pixels serve as geometric references for localization.

4.2 Keyframe Generation and Depth-Based Selection

The transition region $\mathcal{T}$ is subdivided into Left, Right, and Base sub-regions (Figure 3). For each sub-region, we render views from the composite scene $\mathcal{S}$. Given camera pose $\mathbf{P} \in SE(3)$ and shared intrinsics $\mathbf{K}$, each view is

$$\mathcal{V} = (I, D, M) \tag{2}$$

where $I \in \mathbb{R}^{H\times W\times 3}$ is the RGB image, $D \in \mathbb{R}_+^{H\times W}$ is the depth map, and $M \in \{0,1\}^{H\times W}$ is a binary mask indicating transition pixels ($s = T$). For Left and Right, we place cameras at opposite ends of the transition space, producing front–rear view pairs $\{\mathcal{V}^f, \mathcal{V}^r\}$. For Base, we render a single downward-facing view $\mathcal{V}_{\text{Base}}$.

Using these rendered views, we inpaint masked regions using a pre-trained image generation model [28]:

$$\hat{I} = f_{\text{inpaint}}(I, M, y), \tag{3}$$

where y is a text prompt describing the user-defined transition structure (Figure 3). Because inpainting may violate geometric constraints, we generate multiple candidates and select those consistent with the existing scene.

To assess geometric compatibility, we estimate monocular depth for each generated image and align it to the rendered depth. Following Text2Room [15], we align the estimated depth $\hat{D}_{\text{est}}$ to the rendered depth D, but operate directly in metric space rather than disparity. We use Depth Anything V2 [38] fine-tuned for indoor metric depth:

$$\min_{\gamma,\beta} \left\| \neg M \odot \left(\gamma \hat{D}_{\text{est}} + \beta - D \right) \right\|^2, \tag{4}$$

where γ and β are a global scale and shift, $\odot$ denotes element-wise multiplication, and $\neg M$ is the complement of the transition mask M. This yields aligned depth $\hat{D} = \gamma^* \hat{D}_{\text{est}} + \beta^*$. We compute the mean absolute error between $\hat{D}$ and D over the masked region M and reject candidates exceeding a threshold of 0.25 m (Figure 4). From the remaining candidates, the user selects the final keyframes to ensure stylistic consistency between front and rear views:

$$\hat{\mathcal{V}} = (\hat{I}, \hat{D}). \tag{5}$$

We generate $\hat{\mathcal{V}}_{\text{Base}}$ first, as the floor plane provides a gravity-aligned reference visible in both lateral views, improving cross-view consistency. After reconstructing the base region with floor planarity regularization (see supplementary), we update the sparse voxel scene and render $\mathcal{V}_{\text{Left}}$ and $\mathcal{V}_{\text{Right}}$ (Figure 2). The resulting keyframes serve as anchors for progressive video generation (Section 5).

5 Iterative Intermediate Frames Densification

5.1 Video Generation

Given the keyframes $\hat{\mathcal{V}}_{\text{Left}} = \{\hat{\mathcal{V}}^f, \hat{\mathcal{V}}^r\}$ and $\hat{\mathcal{V}}_{\text{Right}} = \{\hat{\mathcal{V}}^f, \hat{\mathcal{V}}^r\}$ from Section 4, we first integrate each inpainted keyframe with the existing 3D scene through volumetric blending (Section 5.3), producing blended keyframes $\tilde{\mathcal{V}} = (\tilde{I}, \tilde{D})$.

For novel view synthesis, we generate intermediate frames from our initial keyframes using an off-the-shelf image-to-video model [10]:

$$f_{\text{vid}}(\tilde{I}^f, \tilde{I}^r, y) = [\hat{I}_0, \hat{I}_1, \ldots, \hat{I}_N], \tag{6}$$

where $\tilde{I}^f = \hat{I}_0$ and $\tilde{I}^r = \hat{I}_N$ are the blended keyframe images, and y is a text prompt describing an arc trajectory between viewpoints. Our keyframes share substantial visual context through the common anchor spaces $\mathcal{S}'_A \cup \mathcal{S}'_B$. This enables the video model to generate smooth transitions for the unobserved portions of the transition region, connecting the separately generated front and rear views. These temporally coherent video frames naturally resolve lighting discontinuities between anchors through gradual transitions. However, video diffusion introduces geometric inconsistencies that degrade the final reconstruction. To address this, we introduce Progressive Anchoring, which interleaves generation with localization using blended anchors that preserve metric grounding.

5.2 Progressive Anchoring

Given the generated video frames $[\hat{I}_0, \hat{I}_1, \ldots, \hat{I}_N]$ from Section 5.1, we now localize each frame to obtain metric-consistent camera poses. The key insight is that blended frames make better anchors than raw generated frames. A raw generated frame contains hallucinated content throughout the transition region—features extracted from these pixels have no metric grounding and will corrupt localization. A blended frame composites generated content with rendered scene geometry, preserving metrically consistent features in the anchor regions $\mathcal{S}'_A$ and $\mathcal{S}'_B$. By localizing against blended anchors, we extract correspondences primarily from trusted pixels while ignoring hallucinated regions. This is what distinguishes Progressive Anchoring from standard pose refinement: the anchor itself evolves through blended frames, accumulating metric constraints while tolerating hallucination locally.

Bidirectional Propagation. Processing frames in a single forward pass would accumulate localization error toward the sequence end, regardless of anchoring quality. By processing bidirectionally—forward from the front keyframe ($\hat{I}_0 \rightarrow \hat{I}_{N/2}$) and backward from the rear keyframe ($\hat{I}_N \rightarrow \hat{I}_{N/2}$)—we bound error accumulation to half the sequence length in each direction. When both passes converge within a distance threshold at the center, a final closing iteration fills the remaining gap (detailed below).

Per-Frame Localization. For each generated frame $\hat{I}_i$, we estimate its pose relative to the current anchor using Perspective-n-Point (PnP). We extract SIFT features [24] from the anchor image $\tilde{I}_{\text{anchor}}$ and target image $\hat{I}_i$, and establish correspondences via brute-force matching with Lowe's ratio test.

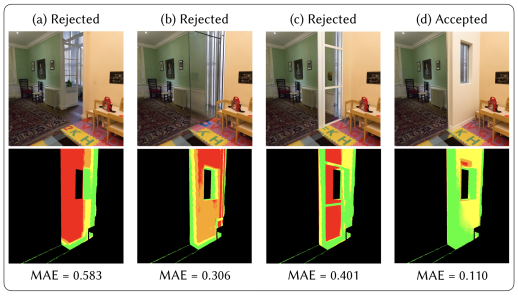

Figure 4: Inpainting candidate selection based on depth alignment error. Candidates (a–c) are rejected due to high MAE; candidate (d) is accepted.

A subtle failure mode occurs when correspondences concentrate in a single anchor space. If most matches originate from $\mathcal{S}'_A$, the pose estimate becomes biased toward that geometry, causing drift away from $\mathcal{S}'_B$. This is especially problematic near the center of the transition, where both anchors should contribute equally. We detect this imbalance using the rendering state map s, which labels matches by anchor origin ($s = A$ or $s = B$), and balance correspondences by capping contributions from the dominant space.

Using the anchor depth map $\tilde{D}_{\text{anchor}}$, we back-project matched keypoints to 3D:

$$\mathbf{X} = \tilde{D}_{\text{anchor}}(u, v) \cdot \mathbf{K}^{-1}[u, v, 1]^\top, \tag{7}$$

and recover the target pose with PnP-RANSAC [20] followed by Levenberg–Marquardt refinement.

Quality Control. Not all frames can be reliably localized. Frames where the video model hallucinated aggressively may have too few matchable features in the anchor regions, or matches may be spatially biased. We enforce three criteria before accepting a localized frame:

(1) **Sufficient inliers**: Above an adaptive threshold, ensuring enough geometric constraints for reliable pose estimation.
(2) **Low reprojection error**: Mean error below 1.0 pixel, indicating consistent 2D-3D correspondences.
(3) **Balanced distribution**: Minimum 0.2 inlier ratio between anchor spaces, preventing drift toward one side.

Frames failing these checks terminate the current pass—continuing would propagate unreliable poses to subsequent frames.

Progressive Anchor Update. After successful localization, we apply volumetric blending (Section 5.3) to composite the generated frame with the rendered scene at the estimated pose, producing $\tilde{I}_i$ and $\tilde{D}_i$. This blended result becomes the new anchor for localizing the next frame:

$$(\tilde{I}_{\text{anchor}}, \tilde{D}_{\text{anchor}}, \mathbf{P}_{\text{anchor}}) \leftarrow (\tilde{I}_i, \tilde{D}_i, \mathbf{P}_i) \tag{8}$$

where $\leftarrow$ denotes assignment: the anchor image, depth map, and pose are all replaced by those of frame i before proceeding to frame i+1. As we move through the sequence, the viewpoint changes, so features visible in the keyframe may become occluded or leave the field of view. By anchoring on the most recently localized blended frame, we maintain feature overlap with the next target frame while preserving metric grounding through the rendered scene component.

Iterative Densification and Final Closing. If the forward and backward passes do not converge after processing one video, we update the keyframe anchors with the last successfully localized frames and generate a new video spanning the reduced gap. Each iteration reduces the unlocalized region, and the new keyframes—being blended frames—provide richer context for the video model than the original sparse keyframes.

When both passes converge within a distance threshold, we perform a final closing iteration. We extract the middle frame $\hat{I}_{\text{mid}}$ from the current video, located midway between the last successfully localized frames of each pass. This frame is localized using *both*

anchors independently via dual-anchor PnP, selecting the result with higher quality under the same criteria. The blended middle frame then serves as the endpoint for two final video segments—one connecting to the forward anchor, one to the backward anchor—completing the transition region as shown in Figure 2.

5.3 Volumetric Blending

Blending plays two roles in our pipeline: it provides metrically consistent input for final reconstruction and, more importantly, produces anchors for Progressive Anchoring. Given a localized frame $\hat{I}_i$ with pose $\mathbf{P}_i$, we render the composite scene $\mathcal{S}$ at $\mathbf{P}_i$ to obtain (I, D, M). Blending is performed in depth and RGB space.

We first estimate monocular depth from $\hat{I}_i$ and align it to the rendered depth using Eq. (4), yielding $\hat{D}$. The stitched depth map is

$$\tilde{D} = M \odot \hat{D} + \neg M \odot D, \tag{9}$$

where M denotes pixels with state $s = T$. Naive 2D masking, however, cannot reject boundary artifacts, which can introduce spurious geometry that propagates across frames. We therefore apply 3D filtering: pixels are back-projected to world space and retained only if they fall within $\mathcal{T}$; others are replaced with rendered depth. Gaussian blending at boundaries ensures smooth transitions.

For RGB compositing, we use multi-band blending to smoothly merge low-frequency appearance while preserving high-frequency details. Specifically, we apply 5-level Laplacian pyramid blending [5] using a downsampled mask at each level, then reconstruct the blended image $\tilde{I}$. Pixels rejected during depth filtering are replaced with rendered RGB values.

6 Results

We evaluate TranSpace on two axes: (1) downstream NVS reconstruction quality against state-of-the-art camera-conditioned generation and pose estimation baselines, and (2) perceptual quality of synthesized transitions against global scene generation and 3D inpainting methods. We present quantitative metrics, qualitative comparisons, ablation studies, and a user evaluation assessing perceptual plausibility.

Datasets. We evaluate on 15 indoor scenes: 2 from Deep Blending [14] and 13 from ScanNet++ [40]. From these, we construct 12 scene pairs for transition synthesis, where each pair contains two transition regions (Left and Right), resulting in 24 total regions for evaluation. Scene pairs are categorized by semantic similarity—6 same-type (e.g., office–office) and 6 cross-type (e.g., apartment–office)—and by transition structure: 8 wing wall, 2 pony wall, and 2 internal window configurations. All scene imagery shown in this paper, and all results derived from it, originate from these two datasets: Deep Blending © Hedman et al., and ScanNet++ © Technical University of Munich, used under the ScanNet++ Terms of Use.

6.1 NVS Reconstruction Quality

Baselines. For camera-conditioned generation, we evaluate against SVC [45] and Gen3C [27]. For camera pose estimation, we compare against classical Structure-from-Motion [30] and transformer-based methods including DUSt3R [35], MASt3R [21], and VGGT [34]. All methods are evaluated using PSNR, SSIM, and LPIPS on the transition region.

Table 1: Quantitative evaluation on 12 scene pairs (24 regions).

Method	PSNR ↑	SSIM ↑	LPIPS ↓	End-Gap ↓	
				(m)	(°)
Camera-Conditioned Generation					
SVC	23.89	0.962	0.053	*prescribed*	
Gen3C	25.42	0.967	0.046	*prescribed*	
Camera Pose Estimation					
COLMAP	23.01	0.963	0.055	0.38	21.7
DUSt3R	22.37	0.951	0.069	0.11	2.72
MASt3R	25.08	0.963	0.052	0.07	2.42
VGGT	25.42	0.966	0.047	0.06	1.75
Ours	**26.18**	**0.971**	**0.039**	*anchored*	

Implementation Details. For fair comparison, all methods receive the same number of frames, and anchor regions are frozen so that errors only affect transitions. For camera-conditioned generation, we prescribe the arc camera trajectory from front view to rear view. For pose estimation methods, we align the scale and start pose to ground truth. Since no ground truth exists for the transition region, metrics measure how consistently each method reconstructs the generated content. Full implementation details are provided in the supplementary material.

Qualitative Results. Figures 5 and 6 present NVS reconstruction results for each method. Progressive Anchoring produces more coherent structures across the transition. The baselines more often exhibit blurriness in the transition region and near boundary areas.

Quantitative Results. Table 1 reports reconstruction quality metrics across 12 scene pairs (24 transition regions). Since anchor regions are frozen identically across all methods, we compute PSNR, SSIM, and LPIPS only within the transition region mask ($s = T$). Note that transition-region views partially capture anchor geometry in the background; differences in these areas reflect how faithfully each method preserves pre-existing content during generation, not differences in the frozen reconstruction itself.

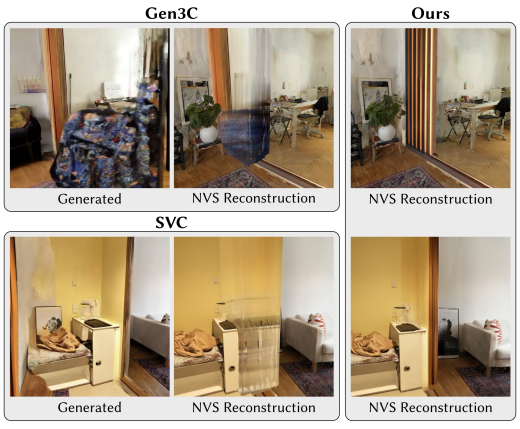

Figure 5: Camera-conditioned generation failures.

Among camera-conditioned generation methods, both fall short of Progressive Anchoring, which achieves a 0.76 dB improvement over Gen3C and 2.29 dB over SVC. This gap stems from two factors. First, without volumetric blending, hallucinated artifacts corrupt shared background regions. In particular, occluded regions in Gen3C's 3D cache tend to produce contradicting geometry against existing anchors (e.g., a chair occluding a desk causes different hallucinated objects across viewpoints), degrading the reconstructed results. Second, both methods exhibit catastrophic failures on several scenes as shown in Figure 5. Progressive Anchoring avoids these failures by using pose estimation as a quality gate, rejecting unreliable frames.

Among pose estimation methods, VGGT achieves the strongest results (25.42 dB), matching Gen3C's PSNR—suggesting that with sufficiently powerful pose estimation, the generate-then-recover paradigm becomes competitive with camera-conditioned generation. However, Progressive Anchoring still outperforms VGGT across all metrics. This gap suggests that the bottleneck is not pose estimation accuracy but the lack of metric anchoring in generated frames—a problem that Progressive Anchoring addresses by re-grounding each frame against the existing reconstruction.

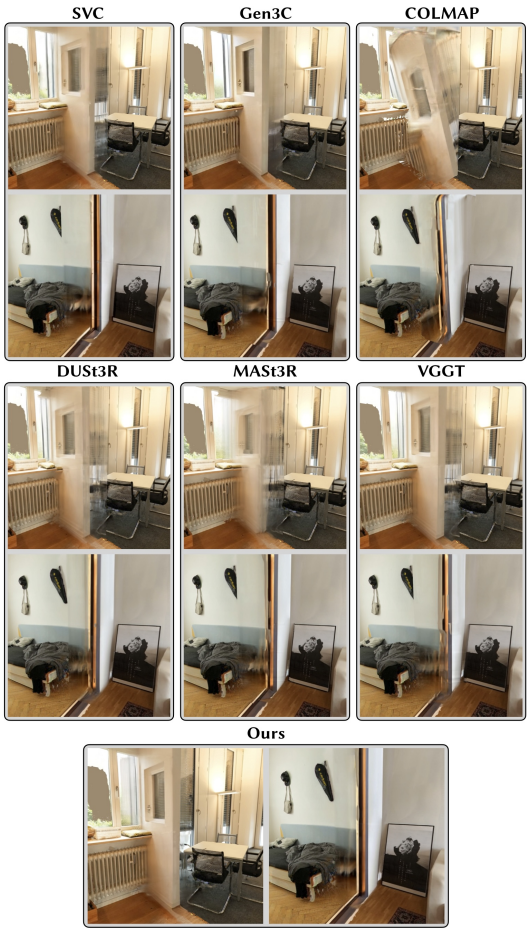

Figure 6: Qualitative comparison of NVS reconstruction across camera-conditioned generation and camera pose estimation baselines.

Table 2: Ablation study on 6 scene pairs (12 regions).

Variant	PSNR ↑	SSIM ↑	LPIPS ↓	Geometric Metrics	
				Inliers ↑	Reproj ↓
w/o Iterative Gen.	24.11	0.962	0.052	191	0.846px
w/o Vol. Blending	21.72	0.953	0.064	727	0.687px
Full Pipeline	**26.13**	**0.972**	**0.035**	555	0.721px

Ablation Study. We ablate two key components of Progressive Anchoring on 6 scene pairs (12 transition regions): iterative video generation and volumetric blending. Qualitative results are shown in Figure 7 and quantitative results in Table 2. To evaluate geometric consistency, we report the number of Inliers (the count of SIFT feature matches between generated frames and anchor regions that satisfy the epipolar constraint) and Reproj (the mean reprojection error in pixels of those matches). Removing iterative generation drops PSNR by 2.02 dB, indicating that 3D inconsistency compounds without intermediate anchor updates. Removing volumetric blending yields a counterintuitive result: stronger geometric metrics (more inliers, lower reprojection error) yet a 4.4 dB PSNR drop—without blending, anchor images lack grounding to the existing reconstruction, so the system locks onto the generated content's internal geometry rather than the metric-consistent scene, the failure mode Progressive Anchoring is designed to mitigate.

6.2 Transition Region Synthesis

Baselines. We compare our synthesized transitions against two methods that leverage off-the-shelf 2D inpainting models: Text2Room [15], which generates indoor scenes at a global level, and NeRFiller [36], which completes localized masked regions using coarse 3D proxies similar to our blueprinting approach. This combination covers both global scene generation and localized 3D inpainting.

Implementation Details. To the best of our knowledge, no existing work targets transition synthesis between pre-existing reconstructions. We adapt Text2Room and NeRFiller to our setting: Text2Room receives the rendered center view with empty regions masked, while NeRFiller receives 30 frames along the arc trajectory with voxel proxies as structural hints. Neither method was designed for bounded transition synthesis; these adaptations represent our best effort to enable comparison in an underexplored problem setting.

Qualitative Results. Figure 8 presents three representative views (Front, Center, Rear) from each method. Our method produces realistic architectural structures that seamlessly connect the two anchor spaces. In contrast, both Text2Room and NeRFiller fail to generate plausible spatial connections despite receiving the same prompts. Text2Room, designed for global scene generation, overwrites the anchor spaces with new content rather than preserving them. NeRFiller preserves the anchors but produces visible boundaries at masked regions—appearing as transparent walls between spaces rather than coherent architectural transitions.

User Evaluation. We recruited 23 participants to rate the synthesized scenes on a scale of 1 to 5 for four questions: Overall Realism (OR), 3D Structure Completeness (3DSC), Prompt Alignment (PA), and Transition Quality (TQ). There were a total of 9 scene pairs (9 transition regions) to evaluate, each evaluated using 3 different methods. We render 5-second-long videos for each generated scene. Users rate each scene individually. The Shapiro-Wilk test indicated that the data violated the normality assumption for OR, 3DSC, and PA, while TQ followed the normality assumption. As such, we used Friedman tests to compare the three conditions for OR, 3DSC, and PA, and used repeated-measures ANOVA for TQ. The results showed significant differences among methods for all questions (OR(Q1): $\chi^2(2) = 45.52$, $p < .001$, Kendall's $W = 0.989$; 3DSC(Q2): $\chi^2(2) = 44.09$, $p < .001$, $W = 0.958$; PA(Q3): $\chi^2(2) = 42.35$, $p < .001$, $W = 0.921$; TQ(Q4): $F(2, 42) = 139.50$, $p < .001$). In general, our method achieved the highest mean scores (3.99–4.32) compared to Text2Room (1.36–1.58) and NeRFiller (2.51–2.62) as illustrated in Figure 9. We conducted a short interview with participants to ask about their preferences and whether the space would feel natural in XR applications. 22 out of 23 participants preferred our method compared to others. Some participants recognized slight inconsistencies in our method, including different appearances between anchors. Despite these perceptual discrepancies, most participants preferred our method because of the naturalness and connectivity secured through the generated architectural structure.

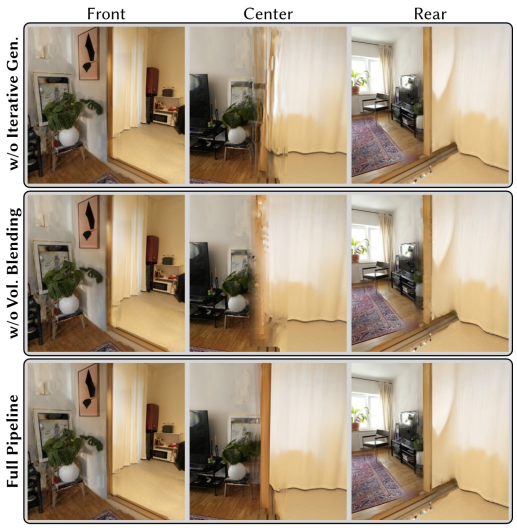

Figure 7: Ablation study results.

Figure 8: Qualitative comparison of transition synthesis methods.

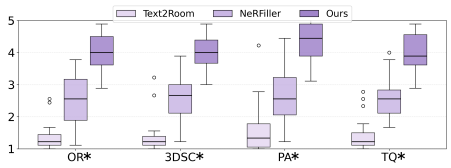

Figure 9: Statistical results of the user evaluation.

7 Limitations, Future Works, and Conclusion

Our method addresses 3D inconsistency but leaves room for improvement. Our framework targets indoor scenes with three user-specified connector types; automating the layout decision, supporting more complex connector geometries, and extending to outdoor scenes remain open directions. Progressive Anchoring requires multiple video generation iterations, resulting in longer processing times than single-pass baselines (e.g., 64 minutes per sub-region on an RTX 4500 GPU versus around 20 minutes for camera-conditioned generation on an A100 GPU). Future work could explore adaptive iteration strategies or learned pose estimators tailored for generated content to reduce this cost. When front and rear keyframes hallucinate inconsistent geometry, ghosting appears around transition structures. Outside the transition bounding box, foreground objects occluding the generation mask can leave holes that propagate through subsequent frames. Future work may address these through uncertainty-aware reconstruction with per-voxel reliability priors and occlusion-aware inpainting.

We presented TranSpace, a method for synthesizing consistent 3D transitions between two reconstructed indoor scenes. Video generation models lack geometric consistency across bounded transitions; Progressive Anchoring overcomes this by localizing generated frames against volumetrically blended anchors to preserve metric fidelity throughout synthesis. Experiments showed consistent gains in metric consistency over camera-conditioned generation and pose estimation baselines, and user studies indicated a perceptual preference over scene generation and 3D inpainting methods.

Acknowledgments

This research was supported by the Culture, Sports and Tourism R&D Program through the Korea Creative Content Agency (KOCCA) grant funded by the Ministry of Culture, Sports and Tourism (MCST) (RS-2026-25525207), and by Institute of Information & Communications Technology Planning & Evaluation (IITP) grants funded by the Korea government (MSIT) (RS-2026-25507551; RS-2022-00156435).

References

[1] Christopher Alexander. 2018. *A pattern language: towns, buildings, construction.* Oxford university press.

[2] Jianhong Bai, Menghan Xia, Xiao Fu, Xintao Wang, Lianrui Mu, Jinwen Cao, Zuozhu Liu, Haoji Hu, Xiang Bai, Pengfei Wan, et al. 2025. ReCamMaster: Camera-controlled generative rendering from a single video. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*.

[3] Jianhong Bai, Menghan Xia, Xintao Wang, Ziyang Yuan, Zuozhu Liu, Haoji Hu, Pengfei Wan, and Di Zhang. 2025. SynCamMaster: Synchronizing multi-camera video generation from diverse viewpoints. In *International Conference on Learning Representations*.

[4] Till Boettger. 2014. *Threshold spaces: Transitions in architecture. Analysis and design tools.* Birkhäuser.

[5] Peter J Burt and Edward H Adelson. 1983. The Laplacian pyramid as a compact image code. *IEEE Transactions on Communications* 31, 4 (1983), 532–540.

[6] Ruojin Cai, Jason Y Zhang, Philipp Henzler, Zhengqi Li, Noah Snavely, and Ricardo Martin-Brualla. 2025. Can Generative Video Models Help Pose Estimation?. In *CVPR*.

[7] Honghua Chen, Chen Change Loy, and Xingang Pan. 2024. Mvip-nerf: Multi-view 3d inpainting on nerf scenes via diffusion prior. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 5344–5353.

[8] Jaeyoung Chung, Suyoung Lee, Hyeongjin Nam, Jaerin Lee, and Kyoung Mu Lee. 2023. Luciddreamer: Domain-free generation of 3d gaussian splatting scenes. *arXiv preprint arXiv:2311.13384* (2023).

[9] Dana Cohen-Bar, Elad Richardson, Gal Metzer, Raja Giryes, and Daniel Cohen-Or. 2023. Set-the-Scene: Global-Local Training for Generating Controllable NeRF Scenes . In *2023 IEEE/CVF International Conference on Computer Vision Workshops (ICCVW)*. IEEE Computer Society, Los Alamitos, CA, USA, 2912–2921. doi:10.1109/ICCVW60793.2023.00314

[10] DeeVid AI. 2024. AI Video Generator. https://deevid.ai/.

[11] Haitang Feng, Jie Liu, Jie Tang, Gangshan Wu, Beiqi Chen, Jianhuang Lai, and Guangcong Wang. 2025. Objfiller-3d: Consistent multi-view 3d inpainting via video diffusion models. *arXiv preprint arXiv:2508.18271* (2025).

[12] Rafail Fridman, Amit Abecasis, Yoni Kasten, and Tali Dekel. 2023. SceneScape: text-driven consistent scene generation. In *Proceedings of the 37th International Conference on Neural Information Processing Systems* (New Orleans, LA, USA) *(NIPS '23)*. Curran Associates Inc., Red Hook, NY, USA, Article 1734, 18 pages.

[13] Hao He, Yinghao Xu, Yuwei Guo, Gordon Wetzstein, Bo Dai, Hongsheng Li, and Ceyuan Yang. 2024. Cameractrl: Enabling camera control for text-to-video generation. *arXiv preprint arXiv:2404.02101* (2024).

[14] Peter Hedman, Julien Philip, True Price, Jan-Michael Frahm, George Drettakis, and Gabriel Brostow. 2018. Deep Blending for Free-viewpoint Image-based Rendering. *ACM Transactions on Graphics (ToG)* 37, 6 (2018), 257:1–257:15.

[15] Lukas Höllein, Ang Cao, Andrew Owens, Justin Johnson, and Matthias Nießner. 2023. Text2room: Extracting textured 3d meshes from 2d text-to-image models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 7909–7920.

[16] Xincheng Huang, Dieter Frehlich, Ziyi Xia, Peyman Gholami, and Robert Xiao. 2025. GaussianNexus: Room-Scale Real-Time AR/VR Telepresence with Gaussian Splatting. In *Proceedings of the 38th Annual ACM Symposium on User Interface Software and Technology*. 1–18.

[17] Ying Jiang, Chang Yu, Tianyi Xie, Xuan Li, Yutao Feng, Huamin Wang, Minchen Li, Henry Lau, Feng Gao, Yin Yang, and Chenfanfu Jiang. 2024. VR-GS: A Physical Dynamics-Aware Interactive Gaussian Splatting System in Virtual Reality. In *ACM SIGGRAPH 2024 Conference Papers* (Denver, CO, USA) *(SIGGRAPH '24)*. Association for Computing Machinery, New York, NY, USA, Article 78, 1 pages. doi:10.1145/3641519.3657448

[18] Taehei Kim, Jihun Shin, Hyeshim Kim, Hyuckjin Jang, Jiho Kang, and Sung-Hee Lee. 2025. Voronoi Rooms: Dynamic Visibility Modulation of Overlapping Spaces for Telepresence. *ACM Trans. Graph.* 45, 2, Article 17 (Dec. 2025), 15 pages. doi:10.1145/3777900

[19] Zhengfei Kuang, Shengqu Cai, Hao He, Yinghao Xu, Hongsheng Li, Leonidas J Guibas, and Gordon Wetzstein. 2024. Collaborative video diffusion: Consistent multi-video generation with camera control. *Advances in Neural Information Processing Systems* 37 (2024), 16240–16271.

[20] Vincent Lepetit, Francesc Moreno-Noguer, and Pascal Fua. 2009. EPnP: An accurate O(n) solution to the PnP problem. *International Journal of Computer Vision* 81, 2 (2009), 155–166.

[21] Vincent Leroy, Yohann Cabon, and Jerome Revaud. 2024. Grounding Image Matching in 3D with MASt3R.

[22] Bohan Li, Xin Jin, Jianan Wang, Yukai Shi, Yasheng Sun, Xiaofeng Wang, Zhuang Ma, Baao Xie, Chao Ma, Xiaokang Yang, et al. 2025. OccScene: Semantic occupancy-based cross-task mutual learning for 3D scene generation. *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2025).

[23] Zhiheng Liu, Hao Ouyang, Qiuyu Wang, Ka Leong Cheng, Jie Xiao, Kai Zhu, Nan Xue, Yu Liu, Yujun Shen, and Yang Cao. 2024. Infusion: Inpainting 3d gaussians via learning depth completion from diffusion prior. *arXiv preprint arXiv:2404.11613* (2024).

[24] David G Lowe. 2004. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision* 60, 2 (2004), 91–120.

[25] Ashkan Mirzaei, Tristan Aumentado-Armstrong, Konstantinos G Derpanis, Jonathan Kelly, Marcus A Brubaker, Igor Gilitschenski, and Alex Levinshtein. 2023. Spin-nerf: Multiview segmentation and perceptual inpainting with neural radiance fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 20669–20679.

[26] Norman Müller, Katja Schwarz, Barbara Rössle, Lorenzo Porzi, Samuel Rota Bulo, Matthias Nießner, and Peter Kontschieder. 2024. Multidiff: Consistent novel view synthesis from a single image. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 10258–10268.

[27] Xuanchi Ren, Tianchang Shen, Jiahui Huang, Huan Ling, Yifan Lu, Merlin Nimier-David, Thomas Müller, Alexander Keller, Sanja Fidler, and Jun Gao. 2025. GEN3C: 3D-Informed World-Consistent Video Generation with Precise Camera Control. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

[28] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. 2022. Photorealistic text-to-image diffusion models with deep language understanding. In *NeurIPS*.

[29] Manuel-Andreas Schneider, Lukas Höllein, and Matthias Nießner. 2025. WorldExplorer: Towards Generating Fully Navigable 3D Scenes. In *Proceedings of the SIGGRAPH Asia 2025 Conference Papers (SA Conference Papers '25)*. Association for Computing Machinery, New York, NY, USA, Article 113, 11 pages. doi:10.1145/3757377.3763946

[30] Johannes Lutz Schönberger and Jan-Michael Frahm. 2016. Structure-from-Motion Revisited. In *Conference on Computer Vision and Pattern Recognition (CVPR)*.

[31] Jonas Schult, Sam Tsai, Lukas Hollein, Bichen Wu, Jialiang Wang, Chih-Yao Ma, Kunpeng Li, Xiaofang Wang, Felix Wimbauer, Zijian He, Peizhao Zhang, Bastian Leibe, Peter Vajda, and Ji Hou. 2024. ControlRoom3D: Room Generation Using Semantic Proxy Rooms . In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE Computer Society, Los Alamitos, CA, USA, 6201–6210. doi:10.1109/CVPR52733.2024.00593

[32] Cheng Sun, Jaesung Choe, Charles Loop, Wei-Chiu Ma, and Yu-Chiang Frank Wang. 2025. Sparse Voxels Rasterization: Real-time High-fidelity Radiance Field Rendering. In *Proceedings of the Computer Vision and Pattern Recognition Conference*. 16187–16196.

[33] Shitao Tang, Fuyang Zhang, Jiacheng Chen, Peng Wang, and Yasutaka Furukawa. 2023. MVDiffusion: enabling holistic multi-view image generation with correspondence-aware diffusion. In *Proceedings of the 37th International Conference on Neural Information Processing Systems* (New Orleans, LA, USA) *(NIPS '23)*. Curran Associates Inc., Red Hook, NY, USA, Article 2229, 32 pages.

[34] Jianyuan Wang, Minghao Chen, Nikita Karaev, Andrea Vedaldi, Christian Rupprecht, and David Novotny. 2025. VGGT: Visual Geometry Grounded Transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

[35] Shuzhe Wang, Vincent Leroy, Yohann Cabon, Boris Chidlovskii, and Jerome Revaud. 2024. DUSt3R: Geometric 3D Vision Made Easy. In *CVPR*.

[36] Ethan Weber, Aleksander Holynski, Varun Jampani, Saurabh Saxena, Noah Snavely, Abhishek Kar, and Angjoo Kanazawa. 2024. Nerfiller: Completing scenes via generative 3d inpainting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 20731–20741.

[37] Grzegorz Wilczyński, Mikołaj Zielinski, Bartosz Świrta, Dominik Belter, and Przemysław Spurek. 2026. Mind the Gap: Geometrically Accurate Generative Reconstruction from Disjoint Views. *arXiv preprint arXiv:2605.07550* (2026).

[38] Lihe Yang, Bingyi Kang, Zilong Huang, Zhen Zhao, Xiaogang Xu, Jiashi Feng, and Hengshuang Zhao. 2024. Depth Anything V2. *arXiv:2406.09414* (2024).

[39] Xiuyu Yang, Yunze Man, Jun-Kun Chen, and Yu-Xiong Wang. 2024. SceneCraft: layout-guided 3D scene generation. In *Proceedings of the 38th International Conference on Neural Information Processing Systems* (Vancouver, BC, Canada) *(NIPS '24)*. Curran Associates Inc., Red Hook, NY, USA, Article 2608, 25 pages.

[40] Chandan Yeshwanth, Yueh-Cheng Liu, Matthias Nießner, and Angela Dai. 2023. ScanNet++: A High-Fidelity Dataset of 3D Indoor Scenes. In *Proceedings of the International Conference on Computer Vision (ICCV)*.

[41] Hong-Xing Yu, Haoyi Duan, Charles Herrmann, William T Freeman, and Jiajun Wu. 2025. Wonderworld: Interactive 3d scene generation from a single image. In *Proceedings of the Computer Vision and Pattern Recognition Conference*. 5916–5926.

[42] Hong-Xing Yu, Haoyi Duan, Junhwa Hur, Kyle Sargent, Michael Rubinstein, William T Freeman, Forrester Cole, Deqing Sun, Noah Snavely, Jiajun Wu, et al. 2024. Wonderjourney: Going from anywhere to everywhere. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 6658–6667.

[43] Wangbo Yu, Jinbo Xing, Li Yuan, Wenbo Hu, Xiaoyu Li, Zhipeng Huang, Xiangjun Gao, Tien-Tsin Wong, Ying Shan, and Yonghong Tian. 2024. Viewcrafter: Taming video diffusion models for high-fidelity novel view synthesis. *arXiv preprint arXiv:2409.02048* (2024).

[44] Jingbo Zhang, Xiaoyu Li, Ziyu Wan, Can Wang, and Jing Liao. 2024. Text2NeRF: Text-Driven 3D Scene Generation With Neural Radiance Fields. *IEEE Transactions on Visualization and Computer Graphics* 30, 12 (Dec. 2024), 7749–7762. doi:10.1109/TVCG.2024.3361502

[45] Jensen (Jinghao) Zhou, Hang Gao, Vikram Voleti, Aaryaman Vasishta, Chun-Han Yao, Mark Boss, Philip Torr, Christian Rupprecht, and Varun Jampani. 2025. Stable Virtual Camera: Generative View Synthesis with Diffusion Models. *arXiv preprint arXiv:2503.14489* (2025).