arXiv:2603.22972v3 [cs.CV] 5 Jul 2026

WorldMesh: Generating Navigable Multi-Room 3D Scenes via Mesh-Conditioned Image Diffusion

Manuel-Andreas Schneider and Angela Dai

Technical University of Munich, Germany
{manuel.schneider, angela.dai}@tum.de
Code & data: https://mschneider456.github.io/world-mesh/

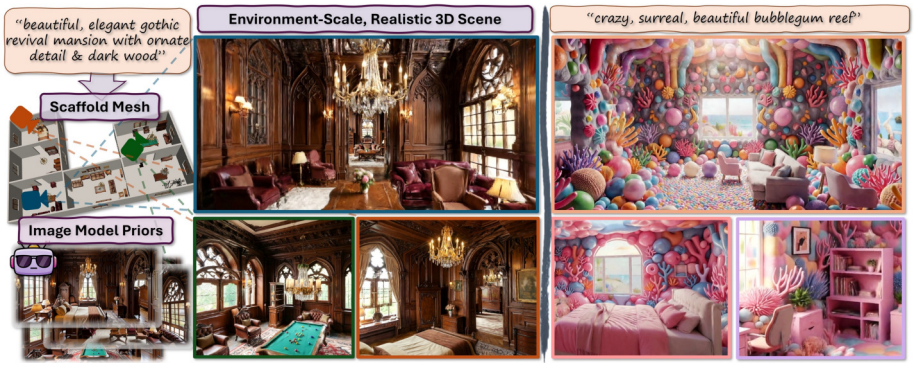


Fig. 1: WorldMesh tackles environment-scale 3D scene synthesis by decoupling this complex problem into structure and appearance. From an input text prompt, we first construct a mesh scaffold that establishes the output scene's layout and geometric structure. This mesh is then used as scaffold for conditioned image synthesis to encourage multi-view consistency across both local (e.g., around objects) and global (e.g., across rooms) scales. Synthesized views are optimized into 3D gaussian splats representing a navigable 3D world through view synthesis.

Abstract. Recent progress in image and video synthesis has inspired their use in advancing 3D scene generation. However, we observe that text-to-image and -video approaches struggle to maintain scene- and object-level consistency beyond a limited environment scale without a persistent, explicit geometric representation. We thus present a geometry-first approach that decouples this complex problem of large-scale 3D scene synthesis into its structural composition, represented as a mesh scaffold, and realistic appearance synthesis, which leverages powerful image synthesis models conditioned on the mesh scaffold. From an input text description, we first construct a mesh capturing the environment's geometry (walls, floors, etc.), and then use image synthesis, segmentation and object reconstruction to populate the mesh structure with objects in realistic layouts. This mesh scaffold is then rendered to condition image

synthesis, providing a structural backbone for consistent appearance generation. This enables scalable, arbitrarily-sized 3D scenes of high object richness and diversity, combining robust 3D consistency with photorealistic detail. We believe this marks a significant step toward generating truly environment-scale, immersive 3D worlds.

Keywords: 3D scene generation · multi-room 3D synthesis · mesh conditioned 3D Gaussian splatting

1 Introduction

Photorealistic 3D environments are central to computer vision and graphics, with applications spanning virtual reality, video games, architecture visualization, and interior design. Creating such environments manually requires significant artistic effort: modeling, texturing, and lighting a single room can take days to weeks for a skilled 3D artist, and scaling to multi-room apartments or larger buildings multiplies this effort dramatically.

Recent advances in generative modeling, in particular in text-to-image and -video generation models [7,14,26,32], have inspired new approaches to 3D scene synthesis. Several recent approaches have leveraged the flexible, photorealistic synthesis capabilities of these image and video generative models to outpaint or synthesize various images of a scene, which can then be used to reconstruct an output 3D scene representation [21, 33, 35]. While these methods can produce impressive single-room synthesis, the very lack of explicit 3D structure in these 2D-based synthesis models, which enables their flexible, powerful synthesis, fundamentally hinders their applicability to world-level, environment-scale scenes. In particular, the absence of 3D structure makes it difficult for such methods to maintain coherent appearance across diverse viewpoints around objects, especially close-up or tightly rotating views. Additionally, in large, multi-room environments these limitations often manifest as inconsistencies across views and rooms, making it largely intractable to scale such methods to arbitrarily large, complex environments.

We thus propose WorldMesh (Fig. 1), a geometry-first approach that decouples the complex task of large-scale 3D scene synthesis into its fundamental challenges: *global spatial consistency* and *local photorealistic appearance*. From an input text prompt, we construct our mesh scaffold representing the output environment's layout. The mesh scaffold is then populated with objects based on image synthesis to suggest object placement and appearance. This enables our scaffolding of image diffusion onto this explicit 3D mesh that maintains inherent spatial coherence across arbitrary camera poses (e.g., close up views, views transitioning across rooms), while preserving the image diffusion model's generative flexibility for local appearance. The generated images can be reconstructed into a navigable 3D scene using 3DGS [24] optimization anchored to the mesh scaffold geometry, maintaining geometric fidelity, multi-room connectivity, and visually diverse, realistic content. This principled separation enables multi-room generation at a scale that remains intractable for purely pixel-space models.

We make the following contributions:

- We present the first approach for scalable, multi-room 3D synthesis from text prompts, supporting diverse visual themes, by constructing a mesh scaffold to represent scene geometry and using it to anchor mesh-conditioned appearance generation. Our approach scales linearly with the number of rooms.
- We introduce mesh-guided appearance synthesis, where the rendered scaffold provides a structural backbone signaling depth, object geometry, and textures, to be used as condition for consistent image synthesis. We reconstruct objects as part of the mesh scaffold based on synthesized images, in order to preserve realistic layouts and high object richness, demonstrating that mesh-conditioning can scale photorealistic 3D scene generation to arbitrarily large, multi-room environments with robust 3D consistency.

2 Related Work

Text-to-Image/Video through Score-based Modeling. Score-based generative models have become the dominant paradigm for high-fidelity image and video synthesis. Early diffusion models [20, 36] generate images by iteratively denoising a stochastic process in raw image pixel space, while latent diffusion models (LDMs) [32] perform this process in a compressed latent space, enabling high-quality generation at practical inference-time computational costs. More recently, transformer-based approaches such as Diffusion Transformers (DiT) [30], along with flow matching [26] training, have improved quality and sampling speed, as adopted by the Flux model family [5]. Additionally, controllable generation has been advanced through ControlNet [47], which injects spatial conditions (e.g., depth maps) into frozen diffusion models, and IP-Adapter [42], which enables image-prompt conditioning. In addition to images, video generation has also seen remarkable progress through score-based generative modeling [7, 9] as well as camera-conditioned control [3, 4, 18], though they typically require substantially more compute than image models. Our approach leverages the generative power and relative efficiency of modern image synthesis models, but rather than relying on unconstrained image generation, we introduce explicit geometric conditioning with a mesh scaffold, enabling image synthesis to produce photorealistic appearance that remains 3D consistent across large-scale 3D scenes.

3D Scene Generation with Image and Video Generative Priors. Inspired by the success of image and video generative models, a growing body of work has explored leveraging these models for text-driven 3D generation. DreamFusion [31] pioneered a text-to-3D approach from 2D image generative models via score distillation into a NeRF [28], showing powerful object-based synthesis but struggling with the complexity of room-scale scenes. Subsequent methods have also proposed to generate many images of a scene in order to reconstruct the 3D scene geometry. Inpainting-based methods like Text2Room [21], WonderJourney [44], and WonderWorld [43] iteratively synthesize and fuse images to expand a camera

trajectory. This can achieve impressive single-room results, but tends to accumulate geometric drift over long trajectories. Multi-view generative approaches, such as MVDiffusion [38] and CAT3D [16], improve view consistency through joint multi-view reasoning, and DreamScene [25] uses panoramic diffusion with 3DGS, but these methods target single objects or bounded scenes.

Video-based approaches including ReconX [27], DimensionX [37], and FlexWorld [11] leverage video diffusion models as strong 3D priors for scene generation. While these methods can improve coherence, they require substantial GPU resources and remain difficult to scale beyond single-room environments. While recent video models increasingly incorporate explicit geometric priors such as camera poses or point-cloud renderings, maintaining consistency over very long, multi-room trajectories remains challenging and computationally demanding. WorldExplorer [33] relies on a camera-conditioned generative model prior [49] to synthesize new views in a scene, thereby constructing navigable 3D scenes. However, because these approaches operate largely in pixel space without an explicit geometric scaffold, maintaining spatial consistency across diverse viewpoints remains challenging. In particular, they often struggle with viewpoints that move close to objects or traverse larger spatial extents (e.g., across multiple rooms), where the limited context of the generative model and the absence of explicit 3D structure can lead to inconsistent geometry and appearance.

Recently, several works have introduced layout guidance to help control the generative process. Methods such as ControlRoom3D [35], SceneCraft [22], and SpatialGen [15] rely on user-provided coarse 3D layouts (e.g., object bounding boxes) to help improve consistency. Our approach also observes that structural priors can help ground consistency; however, we generate the layout from the text prompt and instantiate it as an explicit mesh scaffold that enables more precise geometric anchoring for appearance synthesis.

3D Object Generative Models 3D object generation has also seen remarkable advances, in particular through score-based generative modeling paradigms. Early diffusion-based methods explored diffusion modeling across various 3D latent shape representations [13,45,46,48]. Recently, TRELLIS [40] introduced a structured 3D latent representation for objects, which coupled with flow matching training, produces extremely compelling, high-fidelity 3D shapes from text or image inputs. SAM-3D-Objects [12] employs a similar 3D shape representation to reconstruct objects along with their poses to match an input image. Such methods have shown powerful generative capabilities, but target single objects, which avoids challenges with arbitrary sizes, lack of canonicalization, and large resolutions required for scene-level modeling. We also employ the generative capacity of such object generative models to help instantiate estimated object geometry into our constructed mesh scaffold, which is then used as a geometric anchor for photorealistic appearance synthesis.

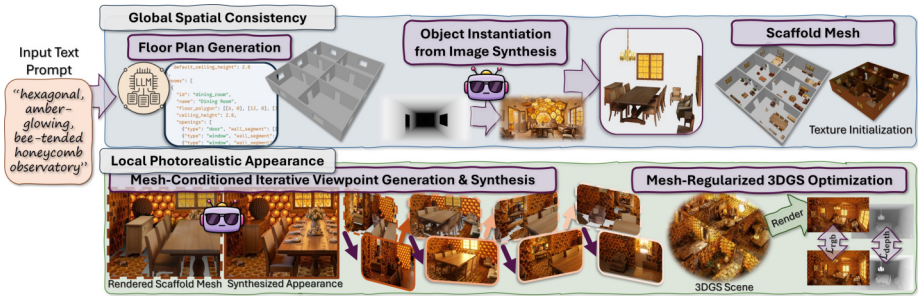


Fig. 2: Overview of WorldMesh. To generate a complex, multi-room 3D scene from a text prompt, we decompose this problem into first constructing the global scene structure as a mesh scaffold (top), and then using the scaffold mesh as anchor for realistic local appearance (bottom). The text prompt is used to generate a text-based floor plan, which we construct in 3D to use as depth conditioning for an image synthesis model Φ, in order to reconstruct estimated 3D objects in each room. The structural elements and 3D objects constitute the scaffold mesh $\mathcal{M}$; its wall textures are not pre-generated but accumulated on-the-fly, by projecting each synthesized view back onto $\mathcal{M}$ during the synthesis loop below. $\mathcal{M}$ then serves as a geometric anchor for iterative image synthesis using Φ to generate images $\{I_i\}$. Finally, the output scene $\mathcal{S}$ is optimized with geometry-regularized 3DGS, against both images $\{I_i\}$ and rendered depth from $\mathcal{M}$.

3 Method

3.1 Overview

Our aim is to generate large, environment-scale, multi-room 3D scenes from a natural language text prompt input $\mathcal{T}$ (e.g., "a cozy Scandinavian apartment"), with an output scene $\mathcal{S}$ represented as 3D gaussian splats [24]. Large-scale multi-room scene generation is challenging: purely 2D or video-based diffusion produces photorealistic local detail but struggles to maintain long-range multi-view and multi-room consistency, particularly around objects and in complex layouts. To address this, we decouple the problem into a geometry-first approach: (i), first, we construct a mesh scaffold $\mathcal{M}$ that encodes the 3D geometric structure of the scene, and (ii) then, we generate photorealistic appearance anchored to this scaffold, enabling both local and global coherence (Fig. 2).

3.2 Mesh Scaffold Construction

To build the mesh scaffold, we first generate from the input text prompt $\mathcal{T}$ a floor plan layout L, represented in JSON text format, to characterize the output 3D scene. The layout L is generated with Claude Opus 4.6 [2] and specifies structural scene metadata such as wall thickness, ceiling height, and room definitions, including floor polygons and openings (doors or passageways). Candidate layouts are sampled and validated against a set of spatial validity rules (e.g., no windows between rooms), and sampling continues until a layout is found that

satisfies these coarse spatial validity constraints. This forms the blueprint for the structural mesh scaffold $\mathcal{M}$.

Based on L, we first instantiate the 3D structure of the scene as $\mathcal{M}_{\text{struct}}$. Each room is defined by a 2D floor polygon, openings, as well as ceiling height. To construct $\mathcal{M}_{\text{struct}}$, each floor polygon is first inset by the wall thickness to define wall cross-sections, which are then extruded vertically to the ceiling height, with shared walls between adjacent rooms represented at half thickness. Matching edges between rooms are detected within a 0.01 m tolerance to ensure continuous walls, and door and window openings are carved using boolean subtraction, with automatic propagation to adjacent rooms sharing the wall. Finally, thin planar extrusions are added to form floors and ceilings, completing the structural mesh $\mathcal{M}_{\text{struct}}$. $\mathcal{M}_{\text{struct}}$ then provides a geometrically consistent layout for the scene and can be exported as a GLB file.

Camera Viewpoint Generation. A set of cameras $\{c_i^r\}$ is generated for each room r and reused across all method stages (i.e., object instantiation and final appearance synthesis). Two *bootstrap cameras*, c_0^r and c_1^r, are placed at eye-level at opposite wall midpoints, on the shorter walls of the rectangular rooms, together spanning all four walls. Approximately 16 *perimeter cameras* at eye-level are uniformly distributed along the room walls with a small wall offset (spaced by arc length, with any perimeter camera within 10° of a bootstrap view pruned to avoid redundancy), each looking toward the room center. Eight *overhead cameras* follow the same perimeter layout but are positioned higher up and look downward for wider spatial coverage. All cameras are nudged away from placed objects to avoid collisions. During initial object placement (Sec. 3.2), only the first coverage camera c_0^r is rendered. During final image generation for 3DGS optimization the full camera set is used.

Object Instantiation While LLM-generated layouts provide reliable structural organization, their textual nature can lead to stale or underspecified object arrangements, lacking the richness and diversity characteristic of real indoor environments. In contrast, modern image synthesis models naturally produce complex and visually coherent object configurations, capturing implicit priors about furniture placement, clutter, and stylistic composition. We therefore leverage image synthesis to populate each room with objects, using the previously constructed floor plan scaffold $\mathcal{M}_{\text{struct}}$ as geometric guidance.

For each room, we render $\mathcal{M}_{\text{struct}}$ from an initial coverage camera pose c_0 located at the wall midpoint of the smallest side of the rectangular room for maximal coverage. This produces a depth map D_0 which is used to condition a photorealistic image generation model (Flux2-Klein [6]). More concretely, generation is guided jointly by D_0, a text prompt combining the room type, global text theme $\mathcal{T}$, and visible architectural context information from L (e.g., door or window location). The architectural context is used to inform the model about the nature of depth maps since it might otherwise interpret the provided openings visible through the depth map as something different (e.g., a TV). The

resulting image I_o^{obj} then establishes the visual style and an initial object layout consistent with the scene scaffold.

We then extract individual 3D object instances from the generated images using SAM3 [10]. In our pipeline these masks are obtained interactively, via a few point prompts per object in SAM3; the segmentation could alternatively be run automatically from a list of text prompts (see appendix). Each segmented object is reconstructed into a 3D representation using SAM-3D-Objects [12], which generates both object meshes in a canonical coordinate system along with their object transforms to the camera coordinate system of the image. As SAM-3D-Objects makes independent predictions per object, we encourage physical plausibility of the outputs by aligning their oriented bounding boxes with the gravity direction of the scene and projecting objects onto the closest support surface below them (or above them in the case of lamps, etc.). The output mesh with composed objects instantiated from the image model prior for all rooms is denoted $\mathcal{M}_{\text{geo}}$, which contains object textures and geometry from SAM-3D-Objects as well as structural floor plan geometry from the layout L.

Mesh Scaffold Texturing We also texture the structural elements of the scaffold (e.g., walls). This wall texturing is not a separate stage that pre-textures the mesh before image generation; it is performed jointly with appearance synthesis, projecting each generated view back onto $\mathcal{M}_{\text{geo}}$ as it is produced. As multi-view consistency of wall appearance between different camera views can be a strong challenge, instead of relying on a 2D image generative model to implicitly preserve color consistency, we explicitly propagate appearance information across views by initializing structural textures through projective texture accumulation. Because the two bootstrap views of each room observe opposite walls and together span all four walls, these first two generations already texture a large portion of the structural surfaces, while the remaining views progressively fill in the rest.

Once an image is generated from a given camera pose c_i, its pixel colors are projected back onto visible wall surfaces of $\mathcal{M}_{\text{geo}}$ using projective texturing [19]. Given the known camera intrinsics (f_x, f_y, c_x, c_y) and camera-to-world transform c_i, each mesh vertex $\mathbf{v}_{\text{world}}$ is projected to its pixel coordinates via the world-to-camera transform c_i^{-1}. The UV coordinates for texture mapping are then $(u = \frac{p_x}{W}, \quad v = 1 - \frac{p_y}{H})$, where W and H are image dimensions and the v-flip follows the glTF convention.

Note that this accumulated mesh texturing provides a strong conditioning signal, but does remain incomplete in various regions due to many object occlusions. The accumulated projections can also introduce seams, and object textures inherited from SAM-3D-Objects are often relatively coarse. Our final appearance synthesis addresses this by enriching the full scene with fine-scale photorealistic detail.

3.3 Mesh Anchored Photorealistic Appearance Synthesis

The constructed mesh scaffold $\mathcal{M}$ now represents the underlying geometric structure of the scene (walls, floors, ceilings, openings, objects). This then serves as a guide for synthesizing photorealistic appearance anchored to $\mathcal{M}$ to enable coherent appearance across views and rooms. Specifically, we employ an image synthesis model Φ conditioned on depth and color renderings of $\mathcal{M}$, using the scaffold to enforce structural fidelity while allowing the model to generate realistic materials, lighting, and fine-grained visual detail. In this way, appearance is not generated freely in 2D, but is instead anchored to the underlying 3D geometry.

Viewpoint Generation for Appearance Synthesis. To synthesize the appearance, we use the set of generated camera viewpoints $\{c_i\}$ to create an output set of multi-view images from a 2D image synthesis model Φ. Note that generating all views independently, even when conditioned on the mesh $\mathcal{M}$, would introduce inconsistencies in color, lighting, and materials due to the 2D-only nature of Φ. We thus design a viewpoint generation strategy that expands coverage progressively while propagating style through the output scene.

For each room r, we begin with the two previously defined bootstrap cameras, c_0^r and c_1^r, to establish the room's global appearance. As c_0^r and c_1^r have been placed at eye-level at opposite wall midpoints, they cover all four walls. These two views enable broad coverage over the room r, which stabilizes appearance conditioning for later view synthesis, reducing the risk of stylistic drift.

After bootstrapping with c_0^r and c_1^r, the remaining camera poses $\{c_i^r\}$ are sampled iteratively. The next camera is selected by greedy nearest-neighbor rotational similarity from the current camera set. At each step, the candidate with the highest quaternion similarity to any previously generated camera is selected:

$$\text{sim}(q_1, q_2) = |\hat{q}_1 \cdot \hat{q}_2|, \tag{1}$$

where $\hat{q}$ denotes a unit quaternion. This creates a gradual outward spiral where each camera has a maximally similar style reference to encourage consistent appearance synthesis. Following generation, the image is projected back to the texture map of $\mathcal{M}$, to obtain the updated $\mathcal{M}_i$. This enables the conditioning signal for the future camera viewpoints to reflect the current appearance information.

Mesh-Conditioned Image Synthesis. To guide image synthesis, we employ a conditioning signal that explicitly separates structural geometry from object appearance while incorporating the estimated wall textures. Note that in the following, we discard the room index r for simplicity.

To construct this, for each camera pose c_i in the generated viewpoints, an image X_i is rendered where objects are rendered with their texture, structural elements are rendered with the alpha-blended wall textures where available, and all other pixels are assigned to their grayscale depth. For the first image conditions X_0 and X_1, corresponding to the bootstrap cameras c_0 and c_1 in the first room

synthesized (no previously generated conditions), the object and wall textures originate from SAM-3D-Objects [12] and our wall texturing, respectively. Each following view (including for additional rooms) then uses the updated texture information from previously synthesized views.

For each camera c_i, we select as style reference the previously generated image $I_{k(i)}$ whose camera pose is rotationally closest, $k(i) = \arg\max_{j<i} \mathrm{sim}(q_j, q_i)$ (Eq. (1)), providing appearance guidance to the synthesis model. Each output image is then synthesized as $I_i = \Phi(X_i,\, I_{k(i)},\, \mathcal{T})$.

Image Verification Even with strong geometric conditioning, image synthesis models may occasionally violate structural constraints. To ensure that a generated image I_i from Φ maintains consistency with the geometry of the mesh scaffold $\mathcal{M}$, we validate I_i by comparing its implied geometry to that of $\mathcal{M}$.

As I_i has been created by an image synthesis model, its underlying geometry is unknown. We thus instead estimate a depth map D_i' with a state-of-the-art depth estimation model [8]. Rather than relying on raw depth estimates, which may contain noise and some depth-scale ambiguity errors, we extract depth edges E_i' through Canny edge detection, which captures the major geometric structures in the scene. These edges are then compared against edges $E_i^{\mathcal{M}}$ extracted from the depth maps rendered from $\mathcal{M}$ to evaluate the structural fidelity of I_i. Specifically, the validation is measured as depth edge recall:

$$\text{Recall} = \frac{|E_i^{\mathcal{M}} \cap \text{dilate}(E_i', \delta)|}{|E_i^{\mathcal{M}}|}, \tag{2}$$

where $E_i^{\mathcal{M}}$ and E_i' are the Canny-detected edge pixel sets and $\delta = 10\,\text{px}$ is the dilation tolerance. Canny edges are extracted with hysteresis thresholds $(0.1, 0.3)$ of the normalized depth range and a 3×3 Sobel aperture. Edge recall measures only *missing* structural edges (false negatives), not extra edges from furniture (false positives), as we allow the image model Φ to imagine plausible objects not present in $\mathcal{M}$. A generated image passes if its recall exceeds the threshold, and is otherwise regenerated with a new seed.

Geometry-Regularized 3DGS Scene Reconstruction. For all successfully validated views $\{I_i\}$, we optimize a 3DGS representation of the output scene $\mathcal{S}$ using the synthesized images I_i, their camera poses c_i, and depth maps D_i rendered from the mesh scaffold $\mathcal{M}$. The Gaussian splats are initialized from a point cloud obtained by back-projecting the mesh depth maps D_i, providing a strong geometric prior for optimization. The training loss combines photometric reconstruction with depth regularization:

$$\mathcal{L} = (1 - \lambda_{\text{s}})\, \mathcal{L}_1(I_i, \hat{I}_i) \;+\; \lambda_{\text{s}}\, \mathcal{L}_{\text{DSSIM}}(I_i, \hat{I}_i) \;+\; \lambda_{\text{d}}\, \|D_i - D_i^{\mathcal{S}}\|_1, \tag{3}$$

where $\hat{I}_i$ and $D_i^{\mathcal{S}}$ are the image and depth rendered from $\mathcal{S}$ at camera c_i, $\mathcal{L}_{\text{DSSIM}} = 1 - \text{SSIM}$, and we set $\lambda_{\text{s}} = 0.2$, $\lambda_{\text{d}} = 0.7$. The depth term prevents geometric drift, a common failure mode in 3DGS under sparse views, and ensures that $\mathcal{S}$ preserves the architectural structure of $\mathcal{M}$.

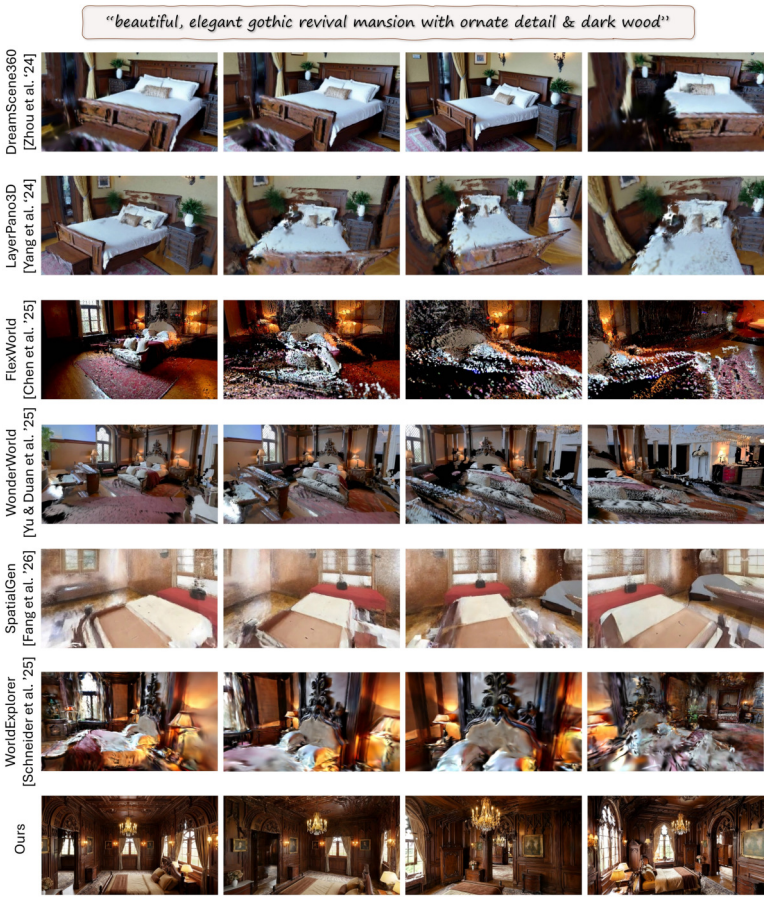


Fig. 3: Qualitative comparison with baselines. We compare with state-of-the-art methods for 3D scene generation leveraging image, panorama, and video generative model priors [11,15,33,41,43,50]. Since these methods focus on a single-room setting, we show our results focusing on one of our generated rooms within our multi-room generation. Most baselines lack explicit 3D structure, leading to view inconsistencies, particularly for challenging viewpoints close to objects (as views are typically synthesized with wider range for easier scene coverage and consistency). SpatialGen [15] mitigates this with 3D box-based layout conditioning, but struggles to maintain realistic local detail. In contrast, our mesh scaffold anchors synthesis geometrically, enabling much stronger multi-view consistency and more realistic appearance across diverse views.

4 Results

Implementation Details We generate various large-scale scenes with different number of rooms, room dimensions and structural layout, each with diverse visual themes. All of our outputs were generated on a single NVIDIA RTX A5000 GPU (24 GB VRAM).

For the images synthesized for object placement we use the Flux2-Klein model locally as the image synthesis model Φ, as we found it the most capable of producing diverse, complex visual results. The images synthesized for object placement use a field of view of 90° for wider coverage; all other images are synthesized with 60° field of view.

Once objects have been placed in the mesh, we generate ≈24 images per room, each of resolution 1376×768. For the first image conditioned on the untextured mesh with textured objects, we use Flux2-Klein locally. For subsequent image generations we use the Nano Banana Pro (NB Pro) [17] model through their API, finding that it is most faithful to the provided mesh priors. Our pipeline is nonetheless backbone-agnostic: it also runs fully locally and open-source with Flux2-klein (9B) on a single RTX A5000, and the layout LLM can likewise be swapped (see the supplementary material).

Baselines. We compare against multiple types of text-to-3D scene, as well as layout guided 3D scene generation methods.

- *Layout-guided 3D Indoor Scene Generation*: most related to our approach, we compare against SpatialGen [15], which takes as input a 3D layout and a reference image to synthesize appearance and geometry from arbitrary viewpoints. Since our method takes only a text prompt as input, we provide SpatialGen our generated room layout and object locations as layout input, along with our initial object placement image as a reference image.
- *Autoregressive Video Diffusion for Navigable 3D Scenes*: we compare against WorldExplorer [33] as the current open-source state-of-the-art for text-driven world generation along pre-defined camera trajectories. We also compare with FlexWorld [11], which generates 360° scenes with video diffusion. Both are evaluated using the same text prompts as our method.
- *Iterative T2I Lifting*: we compare with WonderWorld [43], which constructs 3D scenes via render-refine-repeat using T2I models and monocular depth estimation. We provide one image generated with our text prompt as input to the model for comparison.
- *Panorama-To-3D*: we compare with DreamScene360 [50] and LayerPano3D [41], both of which first generate a 360° panorama from text, which is used to reconstruct a 3DGS [24] scene.

Metrics. For subjective scene quality, we compute CLIP-IQA+ [39] and CLIP Aesthetic [34] scores on frames rendered along trajectories of the final 3DGS scenes. To assess 3D consistency at both environment and object levels, we conduct a perceptual study with 31 participants, who rate object consistency, scene-structure consistency, and overall quality on a 1–5 scale, and give binary preferences between each baseline and our method in randomized order. We evaluate challenging trajectories containing close-up views and rotations across objects.

4.1 Qualitative Results

Fig. 3 shows a qualitative comparison for views rotating around a bed, highlighting the advantages of our mesh-anchored approach. Both WorldExplorer [33]



Fig. 4: WorldMesh generates a diverse range of complex 3D scenes, varying both in spatial layouts and visual themes. We show rendered views from different rooms within each multi-room generated environment, including transitional viewpoints between rooms, demonstrating that stylistic coherence and 3D consistency are maintained throughout.

and FlexWorld [11] produce good results along predefined camera trajectories, but struggle with object consistency in close up views. Panorama-based methods like DreamScene360 [50] and LayerPano3D [41] generate high-fidelity 360° views, but struggle to generate 3D-consistent output beyond their central perspective, causing occlusions and distortions. The iterative WonderWorld [43] builds up accumulated error from inaccurate depth estimates, while SpatialGen [15] partially

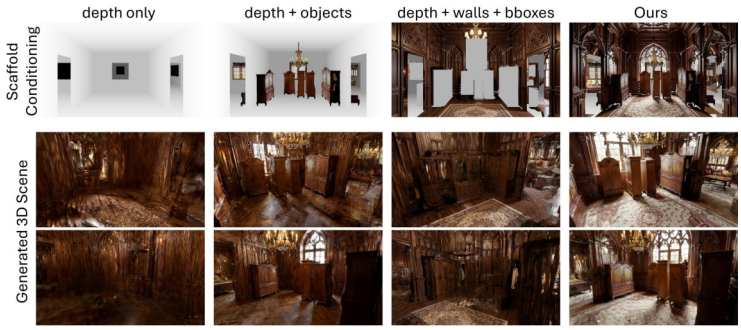


Fig. 5: Ablation of mesh scaffold conditioning. Using only the structural mesh without wall textures (*depth only*) produces large inconsistencies at the local and object level. Using our scaffold mesh with objects but without wall textures (*depth+objects*) preserves object coherency, but still exhibits inconsistencies on structural elements. Structural conditioning with wall texture, but with objects represented as bounding boxes (*depth+walls+bboxes*), does not sufficiently constrain 3D consistency. Our full mesh scaffold with wall texture achieves strong local and global 3D consistency.

mitigates inconsistencies with box-based layout conditioning, but produces less realistic local detail. In contrast, our mesh scaffold provides global spatial consistency, enabling reliable 3D-consistent synthesis even in close-up object views.

We show additional results of our method across a range of realistic and surreal scene types and environments in Fig. 4, as well as in the appendix.

4.2 Quantitative Results

We report automatic image quality metrics and a perceptual study in Tab. 1. CLIP-IQA+ [39] measures technical image quality (sharpness, noise, artifacts), while CLIP Aesthetic [34] evaluates subjective visual appeal (composition, color harmony). Our method leads on both metrics; however, these only assess 2D quality and do not capture 3D consistency.

The perceptual study highlights 3D consistency, with WorldMesh outperforming all baselines across all three dimensions: object consistency across viewpoints (*3D Objects*), spatial coherence of room geometry (*3D Structure*), and overall visual fidelity and 3D coherence (*Quality*). Pairwise comparisons further confirm this, with users preferring our results in 96.2% of cases. This demonstrates that our mesh scaffold provides a superior level of 3D consistency.

We further report per-scene scaling and cost, a navigability analysis, and a 3D-consistency comparison across baselines and image backbones in the supplementary material.

4.3 Ablation Studies

We ablate both the image-conditioning modalities and the 3DGS optimization and validation components in Table 2. Panel (a) reports automatic 3D-

Table 1: Quantitative comparison with state of the art. Perceptual study ratings (1–5 scale, higher is better) across 31 participants. Pairwise preferences (right) from perceptual study show % of participants preferring WorldMesh over each baseline.

	Automatic Metrics ↑		User Study ↑		
Method	IQA+	Aesth.	Quality	3D Objects	3D Structure
WonderWorld	0.4872	4.7795	1.94	2.19	1.84
FlexWorld	0.4097	4.9453	2.05	2.00	2.34
LayerPano3D	0.3949	4.7312	2.27	2.21	2.44
WorldExplorer	0.4053	5.5476	2.58	2.47	2.76
SpatialGen	0.4648	5.0633	2.61	3.00	3.00
DreamScene360	0.3565	4.7270	3.19	3.10	3.37
Ours	**0.5114**	**5.5799**	**4.48**	**4.40**	**4.35**

Ours vs.	Preference (%)
SpatialGen	100.0
WonderWorld	100.0
WorldExplorer	96.8
LayerPano3D	93.5
DreamScene360	93.5
FlexWorld	93.5
Average	96.2

Table 2: Ablation study. **(a)** Automatic metrics over five multi-room scenes: cross-view depth reprojection (MAE-norm, AbsRel), mesh-depth/normal alignment of the final 3DGS to our scaffold (`M-`/`N-`, *Ours* only), and CLIP-IQA+/Aesthetic (separate eval set, so IQA+/Aesth. differ slightly from Tab. 1). **(b)** Perceptual study (1–5, 31 participants) on the gothic-revival scene, for the modality variants (depth+objects = w/o wall texture, depth+walls+bboxes = w/o object recon., depth only = w/o objects or wall texture). **Bold** = best.

(a) 3D consistency and image quality (automatic, 5 scenes)

	3D cons. (Reproj.)		Mesh alignment (*Ours* only)				Quality	
Variant	MAE-norm ↓	AbsRel ↓	M-MAE ↓	M-RMSE ↓	N-Cos ↑	N-Ang ↓	IQA+ ↑	Aesth. ↑
Full (Ours, NB Pro)	**0.1615**	**0.0761**	**0.0383**	**0.1285**	**0.6893**	**39.26**	**0.5529**	**5.5925**
depth+objects	0.2229	0.0997	0.1167	0.2485	0.4033	62.01	0.3870	5.1057
depth+walls+bboxes	0.2722	0.1988	0.3335	0.6387	0.3452	66.17	0.3735	5.2041
depth only	0.1970	0.0888	0.2777	0.5857	0.3780	63.86	0.3818	5.0515
w/o edge recall validation	0.1660	0.0784	0.0478	0.1437	0.6489	42.65	0.5237	5.5482
w/o mesh 3DGS pcd init	0.1706	0.0803	0.0479	0.1406	0.6433	42.93	0.5204	5.5831
w/o mesh 3DGS depth loss	0.3009	0.1641	0.1904	0.3969	0.4574	57.72	0.5312	5.5573
w/o mesh 3DGS init / depth	1.8804	0.6730	2.2549	4.6730	0.2205	74.65	0.4393	5.4270

(b) Perceptual study (1–5 scale, 31 participants, single scene)

Variant	Quality ↑	3D Objects ↑	3D Structure ↑
Full (Ours)	**4.48**	**4.40**	**4.35**
depth+objects	3.35	3.29	3.52
depth+walls+bboxes	2.19	1.94	2.26
depth only	2.10	1.97	2.26

consistency metrics over five multi-room scenes, while panel (b) reports the perceptual study on the same six-room "gothic revival mansion with ornate detail and dark wood" scene, whose conditioning inputs and output frames are shown in Figure 5.

Conditioning modalities. Using only grayscale depth (*depth only*) yields photorealistic images but inconsistent, ghosted furniture across views. Adding textured walls with objects as gray bounding boxes (*depth+walls+bboxes*) improves surface-color consistency but leaves objects unconstrained, whereas reconstructed objects with untextured walls (*depth+objects*) anchor furniture geometry yet force per-frame hallucination of wall appearance. Combining all signals (depth, textured walls, and reconstructed objects) gives comprehensive guidance

for consistent generation, also reflected in the lowest reprojection and mesh-alignment errors in panel (a).

3DGS optimization and validation. The mesh-conditioned 3DGS initialization and depth loss dominate 3D consistency: removing both collapses reprojection accuracy (MAE-norm 1.88 vs. 0.16) while image quality stays largely intact, isolating the consistency gain of mesh conditioning from raw appearance. The edge-recall validation and the point-cloud initialization each add smaller but consistent gains.

Limitations. Our method is currently limited to single-story layouts and does not directly handle multi-level environments, though these could be generated floor by floor with more cohesive layout conditioning. It also relies on SAM-3D-Objects [12] for object reconstruction, which can leave incomplete back faces or missing detail in occluded regions.

5 Conclusion

We presented WorldMesh, an approach for generating navigable multi-room 3D scenes from text by decoupling global spatial consistency from local photorealistic appearance. We first construct an explicit 3D mesh scaffold that captures the scene's geometric layout and maintains spatial coherence across rooms. Image diffusion is then conditioned on this scaffold (through its depth, object geometry, and progressively accumulated wall textures) to synthesize realistic appearance, with a depth-based validation loop ensuring structural fidelity. Our mesh-anchored generation supports consistent appearance across diverse viewpoints, from close-up object views to long-range views spanning multiple rooms, opening a promising path towards truly environment-scale 3D world synthesis.

Acknowledgements

This work has been supported by the ERC Starting Grant SpatialSem (101076253).

References

1. Anthropic: Introducing Claude Haiku 4.5. https://www.anthropic.com/news/claude-haiku-4-5 (2025), accessed: 2026-06-22
2. Anthropic: Claude Opus 4.6 system card. https://www-cdn.anthropic.com/14e4fb01875d2a69f646fa5e574dea2b1c0ff7b5.pdf (2026), accessed: 2026-03-05
3. Bahmani, S., Skorokhodov, I., Qian, G., Siarohin, A., Menapace, W., Tagliasacchi, A., Lindell, D.B., Tulyakov, S.: AC3D: Analyzing and improving 3D camera control in video diffusion transformers. In: IEEE Conf. Comput. Vis. Pattern Recog. (2025)
4. Bahmani, S., Skorokhodov, I., Siarohin, A., Menapace, W., Qian, G., Vasilkovsky, M., Lee, H.Y., Wang, C., Zou, J., Tagliasacchi, A., et al.: VD3D: Taming large video diffusion transformers for 3D camera control. In: Int. Conf. Learn. Represent. (2025)
5. Black Forest Labs: FLUX.1: A suite of text-to-image models. https://bfl.ai (2024), accessed: 2026-06-27
6. Black Forest Labs: FLUX.2 [klein]: Distilled text-to-image model. https://bfl.ai (2026), accessed: 2026-06-27
7. Blattmann, A., Dockhorn, T., Kulal, S., Mendelevitch, D., Kilian, M., Lorenz, D., Levi, Y., English, Z., Voleti, V., Letts, A., et al.: Stable video diffusion: Scaling latent video diffusion models to large datasets. In: arXiv preprint arXiv:2311.15127 (2023)
8. Bochkovskii, A., Delaunoy, A., Germain, H., Santos, M., Zhou, Y., Richter, S.R., Koltun, V.: Depth pro: Sharp monocular metric depth in less than a second. In: Int. Conf. Learn. Represent. (2025)
9. Brooks, T., Peebles, B., Holmes, C., DePue, W., Guo, Y., Jing, L., Schnurr, D., Taylor, J., Luhman, T., Luhman, E., et al.: Video generation models as world simulators. OpenAI Blog **1**(8), 1 (2024)
10. Carion, N., Gustafson, L., Hu, Y.T., Debnath, S., Hu, R., Suris, D., Ryali, C., Alwala, K.V., Khedr, H., Huang, A., Lei, J., Ma, T., Guo, B., Kalla, A., Marks, M., Greer, J., Wang, M., Sun, P., Rädle, R., Afouras, T., Mavroudi, E., Xu, K., Wu, T.H., Zhou, Y., Momeni, L., Hazra, R., Ding, S., Vaze, S., Porcher, F., Li, F., Li, S., Kamath, A., Cheng, H.K., Dollár, P., Ravi, N., Saenko, K., Zhang, P., Feichtenhofer, C.: SAM 3: Segment anything with concepts. In: Int. Conf. Learn. Represent. (2026)
11. Chen, L., Zhou, Z., Zhao, M., Wang, Y., Zhang, G., Huang, W., Sun, H., Wen, J.R., Li, C.: FlexWorld: Progressively expanding 3D scenes for flexible-view exploration. In: Adv. Neural Inform. Process. Syst. (2025)
12. Chen, X., Chu, F.J., Gleize, P., Liang, K.J., Sax, A., Tang, H., Wang, W., Guo, M., Hardin, T., Li, X., Lin, A., Liu, J., Ma, Z., Sagar, A., Song, B., Wang, X., Yang, J., Zhang, B., Dollár, P., Gkioxari, G., Feiszli, M., Malik, J.: SAM 3D: 3Dfy anything in images. In: IEEE Conf. Comput. Vis. Pattern Recog. (2026)
13. Erkoç, Z., Ma, F., Shan, Q., Nießner, M., Dai, A.: HyperDiffusion: Generating implicit neural fields with weight-space diffusion. In: Int. Conf. Comput. Vis. (2023)
14. Esser, P., Kulal, S., Blattmann, A., Entezari, R., Müller, J., Saini, H., Levi, Y., Lorenz, D., Sauer, A., Boesel, F., et al.: Scaling rectified flow transformers for high-resolution image synthesis. In: Int. Conf. Mach. Learn. (2024)
15. Fang, C., Li, H., Liang, Y., Zheng, J., Mao, Y., Liu, Y., Tang, R., Zhou, Z., Tan, P.: SpatialGen: Layout-guided 3D indoor scene generation. In: Int. Conf. 3D Vis. (2026)

16. Gao, R., Holynski, A., Henzler, P., Brussee, A., Martin-Brualla, R., Srinivasan, P., Barron, J.T., Poole, B.: CAT3D: Create anything in 3D with multi-view diffusion models. In: Adv. Neural Inform. Process. Syst. (2024)
17. Google DeepMind: Introducing Nano Banana Pro. `https://blog.google/innovation-and-ai/products/nano-banana-pro/` (2025), gemini 3 Pro Image model. Accessed: 2026-03-05
18. He, H., Yang, C., Lin, S., Xu, Y., Wei, M., Gui, L., Zhao, Q., Wetzstein, G., Jiang, L., Li, H.: CameraCtrl II: Dynamic scene exploration via camera-controlled video diffusion models. In: Int. Conf. Comput. Vis. (2025)
19. Heckbert, P.S.: Survey of texture mapping. IEEE Computer Graphics and Applications **6**(11), 56–67 (1986)
20. Ho, J., Jain, A., Abbeel, P.: Denoising diffusion probabilistic models. In: Adv. Neural Inform. Process. Syst. (2020)
21. Höllein, L., Cao, A., Owens, A., Johnson, J., Nießner, M.: Text2Room: Extracting textured 3D meshes from 2D text-to-image models. In: Int. Conf. Comput. Vis. (2023)
22. Hu, Z., Iscen, A., Jain, A., Kipf, T., Yue, Y., Ross, D.A., Schmid, C., Fathi, A.: SceneCraft: An LLM agent for synthesizing 3D scenes as Blender code. In: Int. Conf. Mach. Learn. (2024)
23. Hui, B., Yang, J., Cui, Z., Yang, J., Liu, D., Zhang, L., et al.: Qwen2.5-Coder technical report. arXiv preprint arXiv:2409.12186 (2024)
24. Kerbl, B., Kopanas, G., Leimkühler, T., Drettakis, G.: 3D Gaussian splatting for real-time radiance field rendering. ACM Trans. Graph. **42**(4) (2023)
25. Li, H., Shi, H., Zhang, W., Wu, W., Liao, Y., Wang, L., Lee, L.H., Zhou, P.: DreamScene: 3D gaussian-based text-to-3D scene generation via formation pattern sampling. In: Eur. Conf. Comput. Vis. (2024)
26. Lipman, Y., Chen, R.T.Q., Ben-Hamu, H., Nickel, M., Le, M.: Flow matching for generative modeling. In: Int. Conf. Learn. Represent. (2023)
27. Liu, F., Sun, W., Wang, H., Wang, Y., Sun, H., Ye, J., Zhang, J., Duan, Y.: ReconX: Reconstruct any scene from sparse views with video diffusion model. IEEE Trans. Image Process. **35**, 2305–2319 (2026)
28. Mildenhall, B., Srinivasan, P.P., Tancik, M., Barron, J.T., Ramamoorthi, R., Ng, R.: NeRF: Representing scenes as neural radiance fields for view synthesis. In: Eur. Conf. Comput. Vis. (2020)
29. OpenAI: gpt-oss-120b & gpt-oss-20b model card. arXiv preprint arXiv:2508.10925 (2025)
30. Peebles, W., Xie, S.: Scalable diffusion models with transformers. In: Int. Conf. Comput. Vis. (2023)
31. Poole, B., Jain, A., Barron, J.T., Mildenhall, B.: DreamFusion: Text-to-3D using 2D diffusion. In: Int. Conf. Learn. Represent. (2023)
32. Rombach, R., Blattmann, A., Lorenz, D., Esser, P., Ommer, B.: High-resolution image synthesis with latent diffusion models. In: IEEE Conf. Comput. Vis. Pattern Recog. (2022)
33. Schneider, M.A., Höllein, L., Nießner, M.: WorldExplorer: Towards generating fully navigable 3D scenes. In: ACM SIGGRAPH Asia (2025)
34. Schuhmann, C.: CLIP+MLP aesthetic score predictor (2022), `https://github.com/christophschuhmann/improved-aesthetic-predictor`, accessed: 2026-06-27
35. Schult, J., Tsai, S., Höllein, L., Wu, B., Wang, J., Ma, C.Y., Li, K., Wang, X., Wimbauer, F., He, Z., et al.: ControlRoom3D: Room generation using semantic proxy rooms. In: IEEE Conf. Comput. Vis. Pattern Recog. (2024)

36. Song, Y., Sohl-Dickstein, J., Kingma, D.P., Kumar, A., Ermon, S., Poole, B.: Score-based generative modeling through stochastic differential equations. In: Int. Conf. Learn. Represent. (2021)
37. Sun, W., Chen, S., Liu, F., Chen, Z., Duan, Y., Zhang, J., Wang, Y.: DimensionX: Create any 3D and 4D scenes from a single image with controllable video diffusion. In: Int. Conf. Comput. Vis. (2025)
38. Tang, S., Zhang, F., Chen, J., Wang, P., Furukawa, Y.: MVDiffusion: Enabling holistic multi-view image generation with correspondence-aware diffusion. In: Adv. Neural Inform. Process. Syst. (2023)
39. Wang, J., Chan, K.C., Loy, C.C.: Exploring CLIP for assessing the look and feel of images. In: AAAI (2023)
40. Xiang, J., Lv, Z., Xu, S., Deng, Y., Wang, R., Zhang, B., Chen, D., Tong, X., Yang, J.: Structured 3D latents for scalable and versatile 3D generation. In: IEEE Conf. Comput. Vis. Pattern Recog. (2025)
41. Yang, S., Tan, J., Zhang, M., Wu, T., Li, Y., Wetzstein, G., Liu, Z., Lin, D.: LayerPano3D: Layered 3D panorama for hyper-immersive scene generation. In: ACM SIGGRAPH (2025)
42. Ye, H., Zhang, J., Liu, S., Han, X., Yang, W.: IP-Adapter: Text compatible image prompt adapter for text-to-image diffusion models. In: arXiv preprint arXiv:2308.06721 (2023)
43. Yu, H.X., Duan, H., Herrmann, C., Freeman, W.T., Wu, J.: WonderWorld: Interactive 3D scene generation from a single image. In: IEEE Conf. Comput. Vis. Pattern Recog. (2025)
44. Yu, H.X., Duan, H., Hur, J., Sargent, K., Rubinstein, M., Freeman, W.T., Cole, F., Sun, D., Snavely, N., Wu, J., Herrmann, C.: WonderJourney: Going from anywhere to everywhere. In: IEEE Conf. Comput. Vis. Pattern Recog. (2024)
45. Zeng, X., Vahdat, A., Williams, F., Gojcic, Z., Litany, O., Fidler, S., Kreis, K.: LION: Latent point diffusion models for 3D shape generation. In: Adv. Neural Inform. Process. Syst. (2022)
46. Zhang, B., Tang, J., Niessner, M., Wonka, P.: 3DShape2VecSet: A 3D shape representation for neural fields and generative diffusion models. ACM Trans. Graph. **42**(4), 1–16 (2023)
47. Zhang, L., Rao, A., Agrawala, M.: Adding conditional control to text-to-image diffusion models. In: Int. Conf. Comput. Vis. (2023)
48. Zhang, X., Li, N., Dai, A.: DNF: Unconditional 4D generation with dictionary-based neural fields. In: IEEE Conf. Comput. Vis. Pattern Recog. (2025)
49. Zhou, J., Gao, H., Voleti, V., Vasishta, A., Yao, C.H., Boss, M., Torr, P., Rupprecht, C., Jampani, V.: Stable virtual camera: Generative view synthesis with diffusion models. In: Int. Conf. Comput. Vis. (2025)
50. Zhou, S., Fan, Z., Xu, D., Chang, H., Chari, P., Bharadwaj, T., You, S., Wang, Z., Kadambi, A.: DreamScene360: Unconstrained text-to-3D scene generation with panoramic gaussian splatting. In: Eur. Conf. Comput. Vis. (2024)

A Video

Please watch our video for additional results, as well as a video comparison of our method. We compare against baselines and ablations. We also provide a diverse set of scene layouts and themes rendered at different trajectories to showcase the quality of our outputs and the robustness of our method.

B Additional Results

We show additional scenes generated by WorldMesh in Fig. A1, highlighting a diverse array of scene arrangement and visual themes.

Additionally, we show a visual comparison of our textured scaffold mesh to our final synthesized scene appearance in Fig. A2, highlighting the impact of our mesh-conditioned appearance synthesis in producing complete, coherent large-scale 3D worlds with rich, high-fidelity detail.

C Additional Implementation Details

Scaffold Mesh Texturing Details Multi-view consistency of 3D objects is ensured through their reconstruction and texturing during the mesh creation phase. To ensure that background information stays similarly consistent, we employ an iterative mesh texturing approach for the room walls based on previously generated images.

To ensure high-quality projections, we apply several geometric filters. A camera-facing constraint excludes back-facing surfaces using the normal–view direction dot product. An occlusion filter compares projected vertex depth with the rendered scene depth (tolerance $\tau = 0.10\,\mathrm{m}$) to prevent projecting foreground object pixels onto background walls. Room-based filtering restricts updates to surfaces belonging to the current room, preventing cross-room texture leakage.

Texture accumulation is progressive: each validated camera view contributes additional coverage to previously untextured surfaces. Over time, this produces an increasingly complete and globally consistent wall texture map directly embedded into the final output $\mathcal{M}$. Importantly, this consistency is enforced at the geometric level, independent of diffusion, allowing future views to inherit accumulated appearance information from the scaffold itself.

The textured mesh in Fig. A2 is the final textured mesh after all images have been generated. Throughout the pipeline the textured mesh is therefore not only input but also output to the image generation model.

Object Segmentation. We use SAM3 [10] in its interactive point-based mode to segment individual objects from the generated images. For each room view, the user provides positive and negative point prompts on the image via a lightweight annotation interface; SAM3 returns a mask prediction after each click, enabling iterative refinement through successive points. Once satisfactory, the mask is

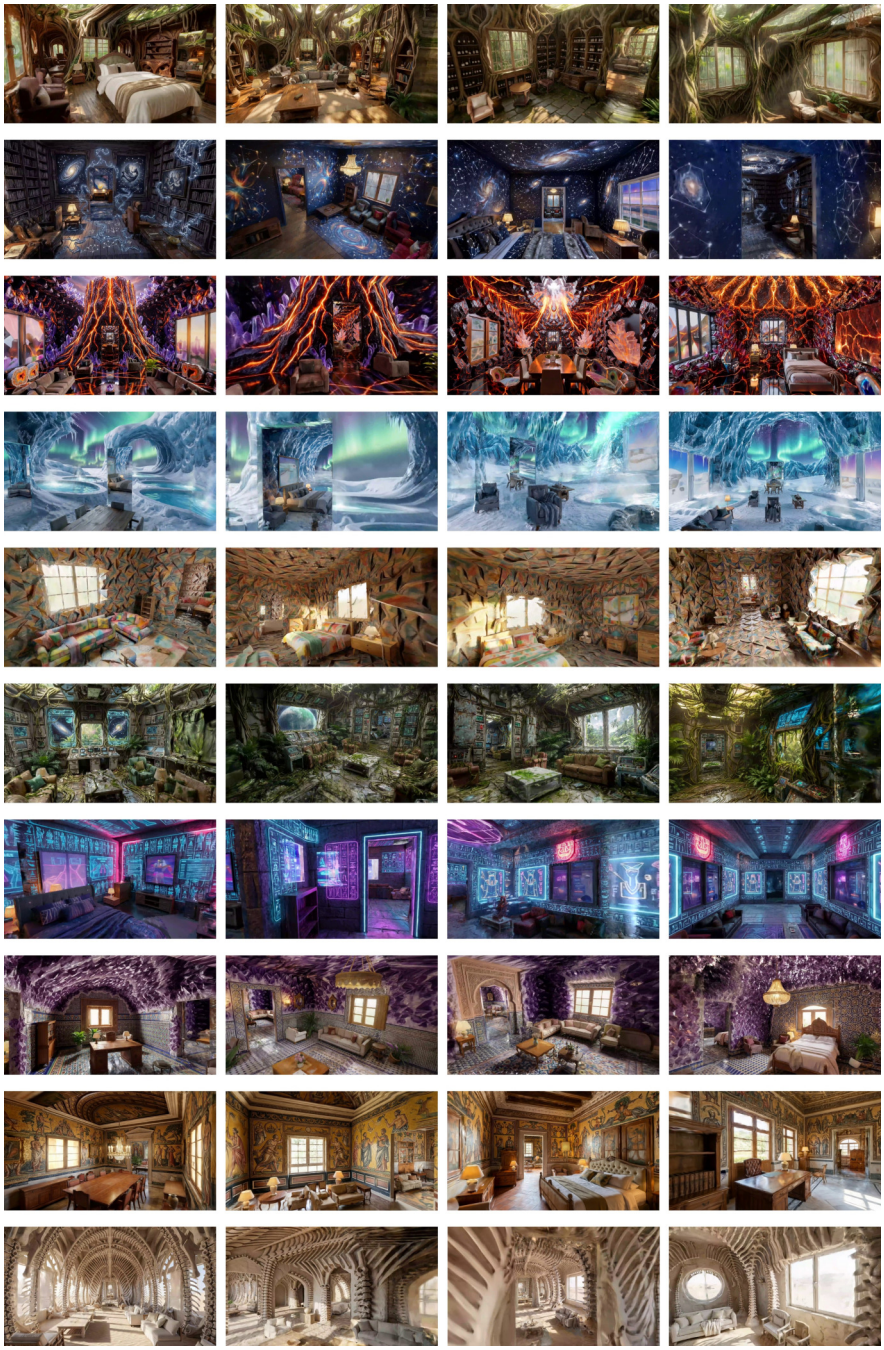


Fig. A1: WorldMesh can synthesize various large, multi-room spaces with varying layouts and diverse visual atmospheres. Each row shows views from a generated 3D world.

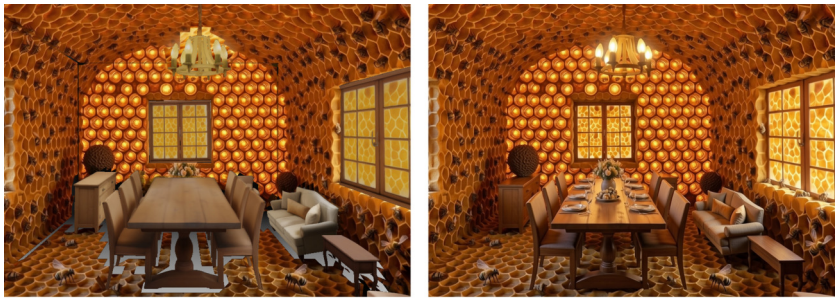


Fig. A2: Comparison of textured scaffold mesh with our final synthesized appearance. The textured scaffold mesh provides a good initialization to encourage multi-view consistent synthesis, but lacks completeness, scene cohesiveness (e.g., seams along wall boundaries, less consistent object appearance between objects), and high-fidelity detail which we provide through our final appearance synthesis.

confirmed and assigned an object label (e.g., "sofa", "lamp"). The resulting per-object binary masks undergo post-processing to discard small detections, merge overlapping regions corresponding to the same object, and split disconnected components into separate instances.

This annotation step can alternatively be automated using a vision-language model to generate text prompts that drive SAM3 in its text-prompted mode, removing the need for manual interaction.

Object Placement. Each object reconstructed by SAM-3D-Objects [12] is produced in camera space. The SAM-3D-Objects pose (rotation and translation) is applied to bring the object into camera coordinates, which are then transformed to world coordinates using the known camera-to-world extrinsics. Once in world space, each object is classified as floor-standing, flat (e.g., rug), wall-mounted, or ceiling-hung based on its label and aspect ratio. Floor-standing objects are leveled by computing their oriented bounding box (OBB), identifying the face most aligned with the ground plane, and rotating the mesh so that face becomes horizontal. They are then dropped to the floor or stacked on previously placed objects using axis-aligned overlap detection. Wall-mounted and ceiling objects are positioned at the appropriate height. Finally, XY conflicts between objects are resolved by shifting overlapping items toward nearby walls while preserving their relative spatial arrangement. All placed objects are merged with the structural room mesh into a single scene.

Iterative Generation Prompt During the final image generation stage of our method, each camera view (except the first bootstrap camera) is generated by conditioning on two inputs: (1) a 3D rendering from the target camera pose, which includes projected wall textures and reconstructed object meshes, and

(2) the photorealistic output from a previous generation in the sequence, which serves as a style reference. Both images are passed to Nano Banana Pro [17] alongside the text prompt shown in Figure A3, where `{theme}` is replaced with the scene description (e.g., "a modern minimalist living room").

D Perceptual Study Details

We conduct a perceptual study to evaluate the performance of WorldMesh and baselines. We ask 31 participants to rate the object consistency across viewpoints (3D Objects), spatial coherence of room geometry (3D Structure), and overall visual fidelity and 3D coherence (Quality) on short trajectory renderings of the final radiance-based outputs, with a particular focus on close-view trajectories and rotations around objects. Please refer to Fig. A4 for an example prompt of our perceptual study and exact question formulation.

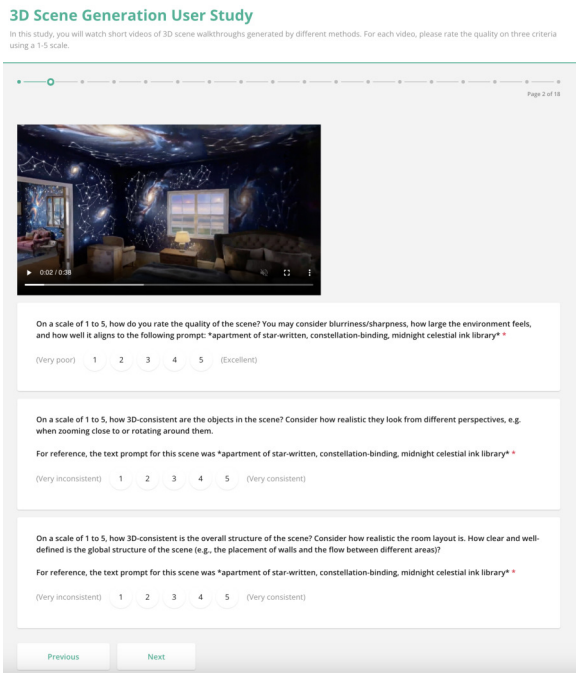


Fig. A4: Example screenshot from our perceptual study. Participants are asked to rank consistency of 3D Objects and 3D Structure, as well as overall Scene Quality on a scale from 1 to 5.

E Additional Baseline Details

Here we provide some additional information on our baseline evaluation setup.

```
The scene depicts a ‘‘{theme}’’

The first image is a 3D rendering showing the correct camera angle,
composition, and object layout for the desired output. However, it is
not photorealistic--the objects are placeholders that show correct
position, scale, and shape but not correct appearance. The rendering
contains texture seams and stretching artifacts from steep-angle
projection that must be removed. Additionally, wall and surface textures
in the rendering are flat 2D projections--the output should depict these
surfaces with proper 3D depth, volume, and parallax-correct detail.

The second image is a photorealistic photograph of the same scene from a
very different camera angle. It defines how the scene should actually
look.

The camera positions differ significantly, so not all objects are
visible in both images. Before generating, establish correspondences
between objects in the rendering and the photograph based on shape,
position, and spatial context.

This image is part of an iterative generation pipeline, so the input
rendering may contain accumulated visual artifacts such as blur, color
degradation, loss of detail, or repeated compression artifacts from
previous iterations. Actively counteract these by producing a clean,
sharp, and visually pleasing output--treat the generation as an
opportunity to restore and enhance visual quality rather than propagate
existing degradation.

Generate a photorealistic image that:
1. Preserves the exact camera angle, composition, and object layout from
the rendering--do not add, remove, or rearrange any objects
2. Transfers photorealistic appearance, materials, lighting, and
textures from corresponding objects in the photograph--not from the
rendering
3. Does not introduce objects from the photograph that are absent in the
rendering
4. Removes all texture seams, stretching artifacts, flat 2D texturing,
placeholder appearances, and any accumulated pipeline
artifacts--surfaces should have realistic 3D depth and detail
5. Is multi-view consistent with the photograph, as if both were real
photos of the same physical scene
6. For objects in the rendering with no match in the photograph, infers
plausible photorealistic appearance consistent with the scene

Use the rendering for geometry and camera only. Use the photograph for
appearance only.
```

Fig. A3: Text prompt used for iterative multi-view generation in Stage 6. The model receives this prompt together with two images: a 3D rendering from the target camera pose and a photorealistic reference from a previous generation.

Layout-guided 3D Indoor Scene Generation. We evaluate SpatialGen [15] on our scenes by converting our textured mesh scaffolds into the required multi-view input format. For each room, we generate 16 views along a circular camera trajectory with a 90° field of view. Per view, we render: (1) an RGB image from the textured mesh, (2) a semantic layout image with per-category color coding (walls, floor, ceiling, and furniture classes), and (3) a layout depth map from the structural mesh. The first RGB view serves as the conditioning image, while the semantic layout and depth maps are used to construct the scene coordinate map that encodes the 3D layout prior. SpatialGen then generates 15 novel views conditioned on these inputs, which are reconstructed into a 3D scene via SpatialGen's custom Sparse-RaDeGS representation.

Autoregressive Video Diffusion for Navigable 3D Scenes. We evaluate FlexWorld [11] by providing one input image generated with the same theme text-prompt as used for our method as input, not changing any of the default hyper-parameters of FlexWorld. We evaluate WorldExplorer [33] by providing the same theme text-prompt and keeping WorldExplorer's predefined trajectories for the scene generation. We then render out the resulting 3DGS scene on trajectories that focus on close-object rotations beyond the initial predefined trajectories used for scene generation, especially in regions not close to the scene center.

Iterative Text-to-Image Lifting. To compare against WonderWorld [43] we provide one image generated with the same theme text-prompt as is used in our method to WonderWorld. Subsequently we choose novel viewpoints following the manner outlined in the original WonderWorld work.

Panorama-To-3D. Initialization of DreamScene360 [50] and LayerPano3D [41] is only possible by providing panorama or text inputs. We compare to these baselines by providing the identical theme text-prompt that is used for our method.

F Extended Quantitative Evaluation and Analysis

This section collects the additional quantitative evaluation referenced in the main paper: a 3D-consistency comparison across methods and image backbones (Tab. A1), per-scene scaling and cost (Tab. A3), layout-LLM statistics (Tab. A2), and a navigability visualization (Fig. A5). Our method does not depend on any single foundation model. The corresponding component ablations, using the same 3D-consistency metrics, are reported in the main paper (Tab. 2).

3D-consistency evaluation. To quantify 3D consistency beyond the perceptual study, Tab. A1 reports, over five multi-room scenes, (i) a cross-view depth-reprojection error that is comparable across methods (a per-method scale-normalized MAE, and the absolute relative error, AbsRel), and (ii) depth and normal alignment of the final 3DGS to the mesh scaffold (`M-`/`N-`). Our method attains the lowest reprojection error of all methods, with both the local Flux2-klein and the NB Pro backbone clearly outperforming the strongest baselines, confirming that our mesh scaffold, rather than any single image model, drives the gain in 3D consistency.

Scaling and cost. The mesh scaffold also scales gracefully: per-scene cost from 4 to 12 rooms (Tab. A3) grows roughly linearly while 3D-consistency and quality remain stable, with high acceptance (*Yield*) at low average reseed counts (*Atm*). With NB Pro, rooms are generated in parallel; the fully-local Flux2-klein path runs sequentially on a single RTX A5000.

Navigability. Rasterizing each scene into a 2D occupancy grid (0.15 m; walls and floor-resting objects as obstacles), the largest 4-connected free-space component covers **96.7%** of free cells on average across the five scenes, confirming near-full traversability (Fig. A5).

Table A1: 3D-consistency comparison across methods and image backbones over 5 scenes (4–8 rooms, different prompts). Reproj. metrics (cross-view depth reprojection: a per-method scale-normalized MAE, and AbsRel) are comparable across methods; `M-`/`N-` are mesh-depth/normal alignment of the final 3DGS to our scaffold (*Ours* only). Computed on a separate evaluation set, so absolute IQA+/Aesth. differ slightly from Tab. 1. The corresponding component ablations are reported in the main paper (Tab. 2). **Bold** = best.

	3D cons. (Reproj.)		Mesh alignment (*Ours* only)				Quality	
Method	MAE-norm ↓	AbsRel ↓	M-MAE ↓	M-RMSE ↓	N-Cos ↑	N-Ang ↓	IQA+ ↑	Aesth. ↑
DreamScene360	0.4336	0.1282	–	–	–	–	0.3153	4.2699
WorldExplorer	0.3221	0.1183	–	–	–	–	0.3633	5.5906
SpatialGen	0.4055	0.1825	–	–	–	–	0.4294	4.4503
Ours (full, Flux2-klein-9B)	0.2173	0.0964	0.1002	0.2311	0.4716	56.74	0.4450	5.5384
Ours (full, NB Pro)	**0.1615**	**0.0761**	**0.0383**	**0.1285**	**0.6893**	**39.26**	**0.5529**	**5.5925**

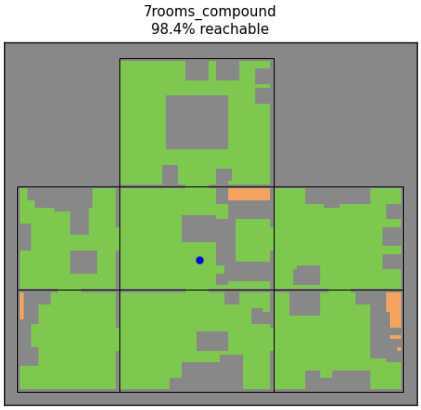


Fig. A5: Navigability. A generated layout rasterized into a 2D occupancy grid (0.15 m): green = reachable free space, gray = obstacles (walls/objects), orange = unreachable free space, dot = seed. Free cells in the largest 4-connected component average 96.7% over our five scenes.

Table A2: Floor-plan layout-LLM backends: mean correction iterations and wall-clock until a spatially-valid layout, over 10 runs of varying scene size. All backends reach valid layouts within a few iterations.

Model	Mean iters	Time
Opus 4.6 [2]	1.30	79 s
Qwen2.5-Coder-32B [23] (local, RTX A5000)	1.40	252 s
gpt-oss-20b [29] (local, RTX A5000)	2.60	229 s
Haiku 4.5 [1]	3.20	101 s

Table A3: Per-scene scaling and cost. Each cell: *Ours–NB Pro* / *Ours–Flux2-klein-9B* (distilled, 4 sampling steps). **Atm**: mean reseed attempts per accepted view; **Yield**: % of intended views retained; **Time**: end-to-end wall-clock per scene. NB Pro is accessed through its API, which lets us generate rooms in parallel, whereas the fully-local Flux2-klein-9B path runs sequentially on a single RTX A5000. 3D-consistency and quality remain stable across room counts and backbones.

#Rooms	Reproj-MAE-norm↓	Reproj-AbsRel↓	M-MAE↓	M-RMSE↓	N-Cos↑	N-Ang↓	IQA+↑	Aesth.↑	Atm↓	Yield↑	Time↓
4 (80 m^2)	0.11 / 0.17	0.07 / 0.10	0.02 / 0.07	0.07 / 0.15	0.75 / 0.50	34.72 / 54.50	0.49 / 0.44	5.56 / 5.74	1.20 / 1.68	74% / 74%	0.8 h / 2.2 h
6 (144 m^2)	0.19 / 0.23	0.10 / 0.11	0.05 / 0.11	0.18 / 0.29	0.61 / 0.39	45.93 / 62.59	0.54 / 0.39	5.54 / 5.52	1.22 / 1.35	86% / 72%	1.0 h / 3.3 h
8 (192 m^2)	0.15 / 0.25	0.05 / 0.08	0.05 / 0.14	0.15 / 0.29	0.63 / 0.44	44.59 / 59.21	0.62 / 0.49	5.56 / 5.27	1.24 / 1.56	71% / 75%	1.2 h / 4.4 h
12 (256 m^2)	0.23 / 0.33	0.08 / 0.11	0.05 / 0.13	0.16 / 0.31	0.61 / 0.43	45.88 / 59.76	0.54 / 0.36	5.55 / 5.54	1.24 / 1.51	82% / 79%	1.5 h / 6.6 h