Think Before You Generate: Active Panoramic Exploration for Text-to-3D Scenes

Derui Li[1] **Peng Lu**[1,*] **Lujing Cao**[1] **Yuhao Sun**[1] **Wenhao Guo**[1] **Sheng Li**[2]
[1]School of Artificial Intelligence, Beijing University of Posts and Telecommunications, Beijing, China
[2]School of Computer Science, Peking University, Beijing, China
{deruili, lupeng, una, yuhaosun, whguo}@bupt.edu.cn
lisheng@pku.edu.cn
*Corresponding author

Abstract

Creating explorable 3D environments from text is important for immersive content creation, simulation, and interactive design. Recent methods often rely on image or video generative priors by synthesizing views along fixed or manually specified trajectories and lifting them into 3D. However, such passive pipelines cannot adapt view generation to the evolving scene memory, often leaving disoccluded regions incomplete and producing holes, floaters, and inconsistent geometry under free-viewpoint exploration. We argue that the key challenge is not to generate more views, but to decide where to generate useful views according to the current scene state. To this end, we propose ActPano3D, an active text-to-3D scene generation framework based on memory-guided panoramic exploration. ActPano3D formulates trajectory selection as spatial deliberation over the evolving scene memory, where candidate actions are evaluated by their expected utility for exploring unknown regions, repairing uncertain geometry, and improving the future 3D world model. The framework integrates expansion-refinement active planning, memory-routed panoramic generation, and reliability-aware scene memory fusion into a closed-loop generation-and-reconstruction process. The accumulated memory is further optimized into a renderable 3D Gaussian Splatting scene for free-viewpoint exploration. Experiments show that ActPano3D improves scene coverage, reduces rendering holes, and achieves better rendering quality and text alignment than existing text-to-3D scene generation baselines.

1 Introduction

Text-to-3D scene generation aims to create realistic and explorable 3D environments from natural language descriptions, enabling applications in virtual reality, game content creation, embodied simulation, and interactive 3D design [1–3]. Compared with object-level 3D generation [4–6], scene-level generation is more challenging: a generated scene should exhibit realistic local appearance while maintaining coherent spatial layout, complete geometry, and stable rendering under free-viewpoint exploration [7, 8]. An ideal system should allow users to describe a scene in natural language and obtain an immersive 3D environment that can be freely navigated.

Due to the scarcity of large-scale paired text-3D scene data, many recent methods rely on powerful 2D generative priors for 3D scene creation [4, 2]. A common paradigm is to first synthesize images, panoramas, or camera-controlled views using pretrained image or video diffusion models [9, 10], and then lift these 2D observations into 3D representations such as meshes, neural fields, point clouds, or 3D Gaussian Splatting [11]. This paradigm benefits from rich semantic and appearance priors learned

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

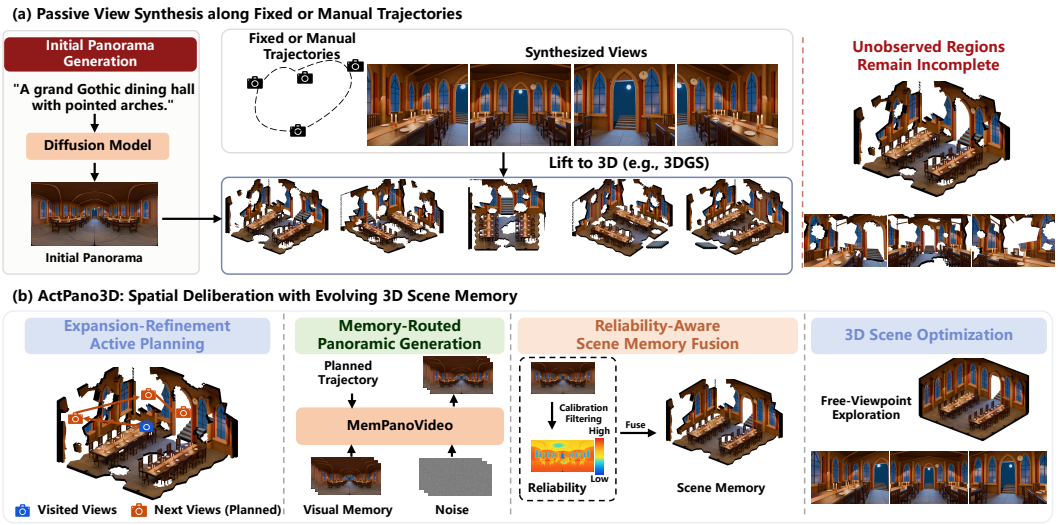

Figure 1: Motivation and overview of ActPano3D. (a) Passive pipelines use fixed trajectories and may leave unseen regions incomplete. (b) ActPano3D actively explores uncertain and unexplored regions, generates new panoramic observations, and reliably fuses them into an evolving 3D scene memory for final 3D Gaussian reconstruction.

from large-scale 2D data, making open-vocabulary text-to-3D scene generation more practical and visually expressive.

However, existing 2D-prior-based pipelines are still largely passive. As illustrated in Fig. 1, they usually generate an initial panorama and then synthesize additional views along fixed, manually specified, or heuristic trajectories [3, 12, 13]. Although such strategies can produce plausible results near selected viewpoints, they do not explicitly deliberate over the evolving reconstruction state of the 3D scene. Consequently, generated views may repeatedly cover already reliable regions while leaving disoccluded or weakly observed areas incomplete, leading to holes, floaters, distorted structures, and inconsistent appearance under free-viewpoint exploration.

We argue that the key challenge is not simply to generate more views, but to determine which views are worth generating according to the current 3D scene state. A useful view should expose missing regions, provide informative parallax, reduce geometric uncertainty, and help refine unreliable structures. This requires a shift from passive view synthesis to reconstruction-aware generation: the evolving 3D representation should actively determine where to explore next, while the generative model synthesizes new observations conditioned on the planned trajectory and existing scene memory. This motivation is related to active mapping, next-best-view planning, and uncertainty-aware reconstruction, where view selection is guided by scene completeness, uncertainty, or expected information gain [14, 15].

In this work, we refer to this decision process as *spatial deliberation*. Here, "thinking" does not refer to symbolic reasoning or language-level deliberation, but to an internal decision process over the current spatial belief of the scene. The system evaluates unexplored regions, uncertainty distribution, and future generation utility before acquiring new observations, estimating how much each candidate trajectory may improve coverage, geometry, consistency, and reconstruction reliability.

To this end, we propose **ActPano3D**, an active text-to-3D scene generation framework based on reconstruction-aware panoramic exploration. Starting from a text-generated panorama, ActPano3D estimates depth and builds an initial scene memory, which provides a coarse but incomplete spatial scaffold due to monocular depth ambiguity, occlusion, and limited viewpoint coverage. Instead of following predefined camera paths, ActPano3D uses the evolving scene memory to deliberate where to generate next. It integrates three components: *Expansion-Refinement Active Planning*, which selects trajectories toward unexplored frontiers and weakly reconstructed regions; *Memory-Routed Panoramic Generation*, which synthesizes trajectory-conditioned panoramic observations using retrieved visual memory; and *Reliability-Aware Scene Memory Fusion*, which filters and

fuses generated observations with confidence-aware updates. The accumulated observations and fused scene memory are finally optimized into a high-quality 3D Gaussian scene for free-viewpoint rendering.

Our contributions are threefold. First, we propose **ActPano3D**, which formulates active text-to-3D scene generation as spatial deliberation over an evolving 3D scene memory, enabling future view generation to be conditioned on accumulated scene knowledge. Second, we develop a closed-loop memory-guided generation pipeline that combines expansion-refinement planning, memory-routed panoramic generation, and reliability-aware fusion to improve coverage, consistency, and robustness against unreliable generated evidence. Third, extensive experiments and ablations demonstrate improved scene coverage, fewer rendering holes, and better free-viewpoint rendering quality and text alignment over existing text-to-3D scene generation baselines.

2 Related Work

Text-to-3D Scene Generation. Text-to-3D scene generation is increasingly important for immersive applications such as virtual reality and interactive content creation, as well as embodied intelligence tasks involving spatial perception and interaction [16–18]. Compared with object-level generation, scene-level generation further requires global layout coherence, spatial completeness, and multi-view consistency. Existing methods mainly follow two paradigms: feedforward generation and optimization-based generation. Feedforward methods directly predict 3D scene representations, such as neural fields, triplanes, point clouds, or structured latent spaces, from textual or visual inputs by learning scene-level priors from 3D datasets [19, 8, 20–25]. They enable efficient inference and coherent global structures when sufficient 3D supervision is available, but are limited by the scale and diversity of paired text-3D scene data, making open-vocabulary scene synthesis with realistic textures and fine details challenging.

Optimization-based methods instead leverage powerful 2D generative models [26, 27] to construct 3D scenes [28–33]. They typically synthesize images, panoramas, or videos using pretrained diffusion models and lift these observations into meshes, neural fields, point clouds, or 3D Gaussian Splatting. By exploiting the realism and open-vocabulary capability of large-scale 2D models, these methods have become a practical paradigm for complex scene generation. Recent works further introduce panoramic generation or camera-controlled video generation to improve spatial coverage and long-range expansion [1, 3, 34, 35]. However, most pipelines remain passive: observations are usually produced along fixed camera paths, predefined expansion rules, user-specified trajectories, or heuristic roaming strategies. Such strategies may generate additional views, but they do not explicitly deliberate over the evolving 3D scene belief to determine which future observations would be most useful. As a result, generated scenes may still suffer from holes, floaters, weakly supported geometry, and inconsistent structures under free-viewpoint exploration. In contrast, our method makes text-to-3D scene generation reconstruction-aware by using the current scene memory to actively decide where to generate next, enabling closed-loop expansion and refinement rather than passive trajectory following.

Active Reconstruction and Generative View Planning. Active reconstruction, or next-best-view planning, selects informative viewpoints to improve coverage, reduce uncertainty, and enhance reconstruction quality [36, 37]. Traditional methods estimate view utility from visibility, occupancy, geometric uncertainty, or information gain, while recent neural and Gaussian reconstruction methods further exploit rendering variance, density uncertainty, Fisher information, Gaussian visibility, or observation completeness [38–43, 15, 14]. These studies demonstrate the effectiveness of reconstruction-aware planning, but they usually assume that new observations come from real cameras or RGB-D sensors and can be treated as reliable measurements of the physical world.

Our setting is fundamentally different. In text-to-3D scene generation, new observations are not captured from a real environment but synthesized by generative models, which may introduce hallucinated content, appearance drift, pose inconsistency, and geometric errors. Therefore, view planning in this setting should consider not only geometric informativeness, but also whether a candidate trajectory can be reliably synthesized and safely fused into the evolving scene memory. We interpret this process as *spatial deliberation*: the system evaluates candidate actions over the current scene belief before acquiring new generated observations. The utility of a future action is determined by whether it can expose unknown regions, repair unreliable geometry, expand the scene boundary, and maintain generation stability. Our method thus bridges active reconstruction and generative view

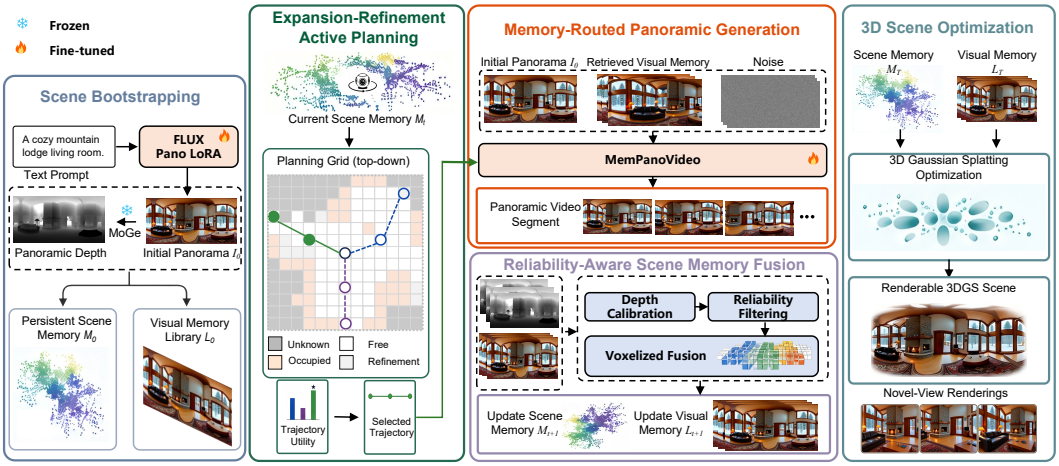

Figure 2: Overview of ActPano3D. Starting from an initial panorama, ActPano3D iteratively plans expansion-refinement trajectories, synthesizes memory-routed panoramic videos, and reliably fuses accepted observations into the scene and visual memories. The final memories are optimized into a 3DGS scene for novel-view rendering.

synthesis by coupling utility-driven trajectory planning, memory-routed panoramic generation, and reliability-aware scene fusion within a closed-loop text-to-3D scene generation framework.

3 Method

Given a text prompt y, our goal is to create an explorable 3D scene that supports free-viewpoint navigation. Instead of synthesizing views along fixed or manually specified trajectories, ActPano3D formulates text-to-3D scene creation as an active generative process, where the current spatial belief determines where new panoramic observations should be generated next. Unlike active reconstruction from real sensors [43, 15, 14], our setting actively selects camera trajectories for a generative model, and therefore must jointly consider geometric utility, long-horizon visual consistency, and generation stability.

We refer to this decision process as *spatial deliberation*. It does not denote symbolic or language-level reasoning, but an internal decision process over the current scene belief, where the system evaluates unexplored regions, reconstruction uncertainty, repair needs, and future generation utility before acquiring new observations. As shown in Fig. 2, ActPano3D maintains a persistent scene memory $\mathcal{M}_t$ for spatial planning and geometric accumulation, and a visual memory library $\mathcal{L}_t$ for long-horizon synthesis and final optimization. Starting from a text-generated panorama, ActPano3D iteratively performs spatial deliberation, memory-routed panoramic video synthesis, and reliability-aware memory expansion, before consolidating all accepted observations into a renderable 3D Gaussian Splatting scene [44].

3.1 Scene Bootstrapping

We initialize the generation process from a text-generated panorama. Given prompt y, we synthesize an equirectangular panorama I_0 using a text-to-panorama generator [7] and estimate panoramic depth D_0 using MoGe [45]. Since each panorama pixel corresponds to a spherical ray, reliable pixels are lifted into a point-based scene memory:

$$\mathcal{M}_0 = \{(D_0(\mathbf{u})\,\mathbf{r}(\mathbf{u}), I_0(\mathbf{u})) \mid \mathbf{u} \in \Omega\}\,, \tag{1}$$

where Ω denotes pixels with reliable depth estimates and $\mathbf{r}(\mathbf{u})$ is the unit spherical ray. Each memory element stores a 3D point and its color, providing an initial spatial proxy for planning and later 3DGS initialization. We also initialize the visual memory library as $\mathcal{L}_0 = \{(I_0, T_0, q_0)\}$, where T_0 is the world origin and $q_0 = 1$ is the initial reliability score.

3.2 Expansion-Refinement Active Planning

At iteration t, ActPano3D selects a camera trajectory for generating the next panoramic video segment. Since panoramic observations cover the full 360° field around each camera center, planning focuses on where and how to move the camera. The planner proceeds through occupancy updating, candidate viewpoint sampling, trajectory construction, virtual visibility estimation, and utility-based selection.

3D occupancy update and 2D planning map. We first update the 3D occupancy evidence using accepted depth observations and the current scene memory $\mathcal{M}_t$. Observed surface points provide occupied-space evidence, while ray casting from accepted camera poses provides free-space evidence. We then aggregate this evidence within a height band around the camera level into a horizontal 2D planning grid $\mathcal{G}_t = \mathcal{O}_t \cup \mathcal{F}_t \cup \mathcal{U}_t$, where $\mathcal{O}_t$, $\mathcal{F}_t$, and $\mathcal{U}_t$ denote occupied, free, and unknown cells, respectively. The planning grid and voxel-level reconstruction reliability together constitute the spatial belief state $\mathcal{B}_t$.

Frontier- and refinement-guided viewpoint sampling. We identify two types of targets from the current spatial belief. Frontiers between observed free space and unknown regions provide expansion targets. Regions with sparse observations, low support, or local inconsistency are identified from voxel-level fusion statistics and form the refinement target set $\mathcal{R}_t$. For both types of targets, we sample candidate camera viewpoints in nearby free space, enabling exploration of unknown regions and revisiting of weakly reconstructed areas.

Candidate trajectory construction. For each candidate viewpoint, we use A* search [46] on the 2D planning grid to find a traversable route from the current camera position. Candidates without a valid route are discarded. The remaining routes are smoothed and resampled to form the candidate trajectory set $\mathcal{T}_t = \{\tau_j\}$.

Virtual visibility estimation. Before generating new observations, we evaluate each candidate trajectory through virtual visibility estimation under the current spatial belief. This produces an estimated observable-cell set $\mathcal{V}(\tau_j)$, accounting for occlusion under the current occupancy estimate. The estimated visibility indicates which unknown cells and weakly reconstructed regions could be observed, providing the basis for evaluating each trajectory's potential contribution to scene expansion and refinement.

Utility evaluation and trajectory selection. We score each candidate trajectory using three normalized benefit terms and a motion cost:

$$U(\tau_j; \mathcal{B}_t) = \alpha I(\tau_j; \mathcal{B}_t) + \beta R(\tau_j; \mathcal{B}_t) + \gamma S(\tau_j; \mathcal{B}_t) - \eta C(\tau_j). \tag{2}$$

Based on the virtual visibility estimates, I measures potential coverage of previously unknown regions, R measures the opportunity to revisit weakly reconstructed regions, and S measures outward expansion of the explored scene boundary. These three terms are normalized before weighting. The motion cost C accounts for route length and deviation from a preferred synthesis distance. The weights α, β, γ, and η balance these four terms.

The next trajectory is selected by

$$\tau_t^\star = \arg\max_{\tau_j \in \mathcal{T}_t} U(\tau_j; \mathcal{B}_t). \tag{3}$$

The selected trajectory is used as the camera-motion condition for the next panoramic video segment, directing generation toward regions with high estimated expansion or refinement utility.

3.3 Memory-Routed Panoramic Generation

Given the selected trajectory $\tau_t^\star$, ActPano3D synthesizes a connected panoramic video segment. As exploration proceeds, each new segment must remain consistent with the initial scene, continuous with the previous segment, and compatible with historical observations. We therefore design a memory-routed trajectory-conditioned panoramic video generator, termed *MemPanoVideo*, built on a pretrained video diffusion model [10] and adapted with lightweight condition adapters [47].

MemPanoVideo uses three visual conditions. The initial panorama I_0 anchors global scene identity and layout; the tail frames I_{tail}^t preserve local temporal continuity; and spatially related memory frames retrieved from $\mathcal{L}_t$ provide long-range appearance and layout references. Each memory frame (I_m, T_m, q_m) is ranked by pose proximity, observable-cell overlap, and reliability:

$$r(I_m, \tau_t^\star) = \beta_p r_{\text{pose}}(T_m, \tau_t^\star) + \beta_o r_{\text{overlap}}(T_m, \tau_t^\star) + \beta_q q_m. \tag{4}$$

The top-ranked frames form the retrieved reference set $\widetilde{\mathcal{R}}_t$. The planned trajectory is encoded as relative pose embeddings and injected into the temporal transformer blocks, while I_0, I^t_{tail}, and $\widetilde{\mathcal{R}}_t$ are encoded as visual reference tokens. The panoramic video segment is synthesized as

$$\{I^t_i\}_{i=1}^{L} = G_{\text{mpv}}\left(y, I_0, \tau^\star_t, I^t_{\text{tail}}, \widetilde{\mathcal{R}}_t\right), \tag{5}$$

where G_{mpv} denotes MemPanoVideo.

3.4 Reliability-Aware Scene Memory Fusion

The generated segment is converted back into the persistent scene memory, serving as a reliability-aware update to the current spatial belief. Since generated frames are pseudo-observations and may contain hallucinated content, monocular depth errors, or trajectory deviations, we update $\mathcal{M}_t$ through keyframe lifting, depth calibration, pose-consistency checking, reliability filtering, and voxelized fusion.

We sample sparse keyframes from the generated segment, estimate panoramic depth with MoGe [45], and lift them into local colored point clouds according to their nominal poses along $\tau^\star_t$. Before fusion, each keyframe depth is calibrated against the current scene memory by projecting $\mathcal{M}_t$ into the keyframe pose and estimating a robust scale-and-shift correction from overlapping valid regions. The same overlap is used to compute a lightweight pose-consistency score, so keyframes with large projection inconsistency or insufficient overlap are assigned low reliability or rejected.

After calibration and consistency checking, unreliable points are filtered using depth validity, projection consistency, local voxel density, and source-frame reliability. The remaining points are fused as

$$\mathcal{M}_{t+1} = \text{Fuse}\big(\mathcal{M}_t, \mathcal{P}^{\text{local}}_t; w_t\big), \tag{6}$$

where w_t combines the reliability cues. Points in the same voxel are merged by reliability-weighted averaging, and voxel statistics such as observation count, accumulated reliability, and local depth variance are updated for later refinement-region discovery. Accepted keyframes are inserted into the visual memory library with their poses and reliability scores, producing $\mathcal{L}_{t+1}$. The loop stops when the maximum iteration budget is reached, when insufficient valid expansion or refinement targets remain, or when the number of newly fused reliable voxels falls below a threshold, indicating that further exploration provides limited additional scene coverage.

3.5 3D Scene Optimization

After active expansion, we convert the final scene memory $\mathcal{M}_T$ and visual memory library $\mathcal{L}_T$ into a renderable 3DGS scene. Gaussian primitives are initialized from valid points in $\mathcal{M}_T$, where point positions and colors define Gaussian centers and appearance, local voxel density estimates scale, and observation support initializes opacity. Weakly supported or inconsistent points are removed before optimization.

Given $\mathcal{L}_T = \{(I_k, T_k, q_k)\}$, we render images and depths from the 3DGS scene at the corresponding poses and optimize

$$\mathcal{L}_{\text{3DGS}} = \sum_k q_k \left(\lambda_{\text{rgb}}\mathcal{L}^k_{\text{rgb}} + \lambda_{\text{depth}}\mathcal{L}^k_{\text{depth}}\right) + \lambda_{\text{reg}}\mathcal{L}_{\text{reg}}. \tag{7}$$

Here, $\mathcal{L}^k_{\text{rgb}}$ enforces appearance consistency with accepted memory frames, $\mathcal{L}^k_{\text{depth}}$ aligns rendered depth with calibrated depth evidence, and $\mathcal{L}_{\text{reg}}$ suppresses floaters, unstable opacity, and oversized Gaussians [48]. Reliability-weighted supervision allows the final optimization to reduce the influence of noisy generated observations and produce a stable renderable 3D scene.

4 Experiments

4.1 Experimental Setup

Datasets. We use LayerPano3D-PanoData [7] and Matterport3D [49] to train the text-to-panorama generator. For trajectory-conditioned panoramic video generation, we follow OmniRoam [50] and

construct a geometrically supervised dataset by rendering 1,000 indoor 3D Gaussian Splatting scenes from InteriorGS [51] with physically feasible camera trajectories. For evaluation, we build a benchmark of 50 diverse indoor prompts, covering common room types, spatial layouts, and object configurations. All evaluation prompts and scenes are disjoint from the training data used for both panorama generation and panoramic video generation. All methods are evaluated on the same prompt set.

Implementation Details. ActPano3D uses two trainable generative modules: a text-to-panorama generator and a trajectory-conditioned panoramic video generator. We adopt FLUX [9] as the base model for panorama generation and fine-tune it with LoRA [47] on panoramic scene data. Given a text prompt, the generator produces an equirectangular panorama, from which we estimate depth using MoGe [45] and initialize a point-based scene memory. For active exploration, we fine-tune Wan2.1-1.3B [10] for trajectory-conditioned panoramic video synthesis with trajectory, tail-frame, and memory-reference conditions. ActPano3D performs $T = 20$ exploration rounds. At each round, the planner selects an expansion-refinement trajectory through spatial deliberation over the current scene belief, and the video generator synthesizes an $L = 81$-frame panoramic segment. We uniformly sample $S = 9$ keyframes per segment for depth estimation, depth calibration, reliability filtering, and scene fusion. The accumulated scene memory and visual memory are finally optimized into a renderable 3DGS scene [44]. All scenes are rendered at 512×512 resolution for evaluation. All experiments are conducted on NVIDIA A6000 GPUs with 48GB memory; the generative modules are fine-tuned using 8 GPUs, and each evaluation scene is processed on a single GPU.

Baselines. We compare ActPano3D with representative text-to-3D and panoramic scene generation methods, including Text2Room [28], FlexWorld [52], DreamScene360 [31], LucidDreamer [29], LayerPano3D [7], and Matrix-3D [3]. Text2Room and LucidDreamer represent text-to-3D scene generation baselines. FlexWorld, DreamScene360, and LayerPano3D are text-to-360-degree or panoramic scene generation baselines. Matrix-3D follows a closely related text-to-panorama-to-3D pipeline. For methods that do not directly output 3DGS scenes, we use their official reconstruction pipeline when available; otherwise, we convert their generated observations into 3DGS using the same reconstruction backend as ActPano3D. All methods use the same prompt set, rendering resolution, camera sampling strategy, and evaluation protocol.

Evaluation Protocol and Metrics. For each prompt, every method generates one 3D scene, which is evaluated from the same $K = 40$ novel viewpoints sampled with a fixed random seed. The viewpoints include both near-initial views and challenging views that expose disoccluded or weakly reconstructed regions. We evaluate scene completeness using Rendering Hole Ratio (RHR), which measures the proportion of low-opacity pixels in novel-view renderings. Given the accumulated opacity map α_k of the k-th rendered view, RHR is computed as

$$\mathrm{RHR} = 100 \times \frac{\sum_{k=1}^{K} \sum_{p} \mathbb{I}\left[\alpha_k(p) < \tau_\alpha\right] M_k(p)}{\sum_{k=1}^{K} \sum_{p} M_k(p)}. \tag{8}$$

Here M_k is the valid evaluation mask and $\tau_\alpha = 0.5$ in all experiments. RHR is reported on a 0–100 scale, with lower values indicating fewer visible rendering holes and better novel-view completeness. For rendered image quality, we report BRISQUE [53], NIQE [54], and Inception Score [55]. For text-scene alignment, we report CLIP-Score [56] between the input prompt and rendered views.

4.2 Comparison with State-of-the-Art

Quantitative Results. As shown in Table 1, ActPano3D achieves the best performance across all metrics. In particular, it reduces RHR from 19.47 to 8.21 compared with the strongest baseline Matrix-3D, indicating substantially fewer visible holes under novel-view rendering. ActPano3D also improves BRISQUE and NIQE, suggesting better perceptual quality and more natural rendered views. The higher CLIP-Score further indicates that active exploration and memory-guided panoramic synthesis preserve semantic consistency with the input prompts while improving scene completeness.

Qualitative Results. Figure 3 compares representative novel-view renderings on challenging indoor prompts. DreamScene360 and LayerPano3D can produce plausible panoramic overviews, but their novel-view renderings often contain missing regions, distorted structures, and inconsistent local

Table 1: Quantitative comparison with state-of-the-art text-to-3D and panoramic scene generation methods. RHR is reported on a 0–100 scale and evaluates novel-view rendering completeness; BRISQUE, NIQE, CLIP-Score, and IS evaluate rendered image quality and text alignment.

Method	RHR↓	BRISQUE↓	NIQE↓	CLIP-Score↑	IS↑
Text2Room [28]	40.35	50.90	6.28	25.04	1.43
FlexWorld [52]	31.34	44.01	5.37	27.21	1.64
DreamScene360 [31]	30.85	43.36	5.54	28.13	1.59
LucidDreamer [29]	38.41	51.25	6.16	26.97	1.49
LayerPano3D [7]	28.76	38.52	5.14	30.40	1.74
Matrix-3D [3]	19.47	31.59	4.71	31.97	1.84
ActPano3D	**8.21**	**24.33**	**3.53**	**33.91**	**1.93**

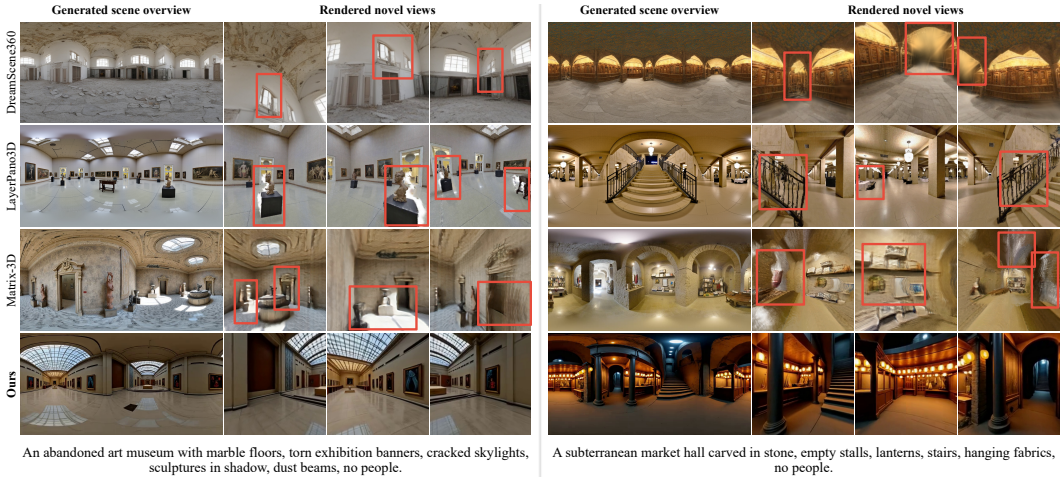

Figure 3: Qualitative comparison with state-of-the-art methods. We show generated scene overviews and rendered novel views for two challenging indoor prompts. Red boxes highlight visible holes, distorted structures, and incomplete regions in baseline results. ActPano3D produces more complete and coherent novel-view renderings.

geometry. Matrix-3D benefits from a panorama-to-video-to-3D pipeline and produces stronger scene layouts, yet visible holes and incomplete regions still appear when the viewpoint moves to disoccluded areas. In contrast, ActPano3D actively plans new trajectories toward weakly reconstructed regions and fuses generated observations into a persistent scene memory, leading to more complete novel-view renderings with fewer visible artifacts. This qualitative improvement is consistent with the lower RHR reported in Table 1.

4.3 Ablation Study

We ablate three key components of ActPano3D: utility-driven spatial deliberation for active planning, memory-routed reference retrieval, and reliability-aware fusion. All variants are evaluated on the same 50-prompt benchmark using the same evaluation protocol. As shown in Table 2, removing any component leads to clear performance degradation.

Effect of spatial deliberation. Without active planning, we remove the proposed utility-driven spatial deliberation and replace it with random feasible trajectory selection, while keeping the video generator, fusion strategy, and 3DGS optimization unchanged. This variant corresponds to a non-deliberative exploration strategy under the same 20-round generation budget. RHR increases from 8.21 to 19.82, showing that simply generating more observations is insufficient. Instead, the system needs to evaluate future actions according to the current spatial belief so that the generation budget is spent on unexplored boundaries, uncertain geometry, and weakly supported regions. This confirms

Table 2: Ablation study of ActPano3D. The variant w/o Active Planning removes the proposed utility-driven spatial deliberation and replaces it with random feasible trajectory selection. RHR is reported on a 0–100 scale and evaluates novel-view rendering completeness; BRISQUE, NIQE, CLIP-Score, and IS evaluate rendered image quality and text alignment.

Variant	RHR↓	BRISQUE↓	NIQE↓	CLIP-Score↑	IS↑
w/o Active Planning	19.82	32.14	4.49	31.86	1.79
w/o Memory Retrieval	12.97	27.42	3.91	32.18	1.86
w/o Reliability-aware Fusion	14.52	30.08	4.16	32.76	1.81
Full Model	**8.21**	**24.33**	**3.53**	**33.91**	**1.93**

Table 3: Exploration efficiency under different generation budgets. We report RHR on a 0–100 scale after different numbers of exploration rounds.

Method	0 rounds	5 rounds	10 rounds	15 rounds	20 rounds
Fixed Trajectory	42.10	31.52	26.73	23.20	21.08
Random Trajectory	42.10	29.84	24.15	21.47	19.82
ActPano3D	42.10	**23.42**	**15.76**	**10.93**	**8.21**

that the proposed expansion-refinement deliberation is essential for improving scene completeness under free-viewpoint rendering.

Effect of memory retrieval. Removing memory retrieval degrades both text alignment and image quality. CLIP-Score drops from 33.91 to 32.18, and BRISQUE increases from 24.33 to 27.42. This suggests that retrieved historical observations provide useful long-range visual context, helping the panoramic video generator maintain consistent appearance, layout, and scene identity during long-horizon exploration.

Effect of reliability-aware fusion. Replacing reliability-aware fusion with naive fusion causes clear degradation, increasing RHR to 14.52 and BRISQUE to 30.08. Generated panoramic frames are pseudo-observations and may contain hallucinated details, depth errors, or local inconsistencies. Directly fusing them introduces noisy geometry and unstable opacity. Depth calibration, pose-consistency checking, and reliability-weighted fusion therefore help build a more stable scene memory and improve the final 3DGS optimization.

4.4 Analysis of Active Exploration

Exploration Efficiency. To evaluate whether spatial deliberation uses the generation budget more effectively, we compare ActPano3D with fixed and random trajectory strategies under different exploration rounds. All variants start from the same initial panorama and use the same panoramic video generator, reconstruction backend, and 3DGS optimization setting; they only differ in trajectory selection. As shown in Table 3, ActPano3D reduces RHR faster than fixed and random exploration, demonstrating that selecting trajectories by utility maximization over the current scene belief is more effective than passively generating additional observations. ActPano3D shows the largest improvement in the early exploration stage. After 5 rounds, it reduces RHR to 23.42, while fixed and random trajectories remain at 31.52 and 29.84, respectively. The gap remains consistent as the budget increases, and the improvement gradually saturates after 15–20 rounds, suggesting that most visible disoccluded regions have been covered. These results show that ActPano3D uses the generation budget more effectively by deliberating over the evolving spatial belief and selecting trajectories that are expected to expand coverage, reduce uncertainty, and repair weakly reconstructed areas.

5 Conclusion

We propose ActPano3D, an active text-to-3D scene generation framework that couples generative view synthesis with an evolving 3D scene memory. Unlike passive pipelines with fixed or manually specified trajectories, ActPano3D performs spatial deliberation before generation, selecting where to explore and refine based on scene belief, reconstruction uncertainty, and expected utility. By

integrating expansion-refinement planning, memory-routed panoramic generation, and reliability-aware scene fusion, ActPano3D improves scene completeness, geometric consistency, and free-viewpoint rendering quality. Experiments and ablations demonstrate the effectiveness of our design over passive generation baselines.

Limitations. ActPano3D is mainly evaluated in indoor and static scene generation settings. Extending it to large-scale outdoor scenes with more complex spatial layouts remains future work. Handling dynamic objects, changing illumination, or time-varying scene content may also require additional temporal modeling and consistency constraints.

Acknowledgments and Disclosure of Funding

This work was supported in part by the National Natural Science Foundation of China (Grant No. 62676046) and in part by the Project of National Key Laboratory of Intelligent Parallel Technology (Grant No. 2025-JCJQ-LC-021-06).

References

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

[2] HunyuanWorld Team, Zhenwei Wang, Yuhao Liu, Junta Wu, Zixiao Gu, Haoyuan Wang, Xuhui Zuo, Tianyu Huang, Wenhuan Li, Sheng Zhang, et al. Hunyuanworld 1.0: Generating immersive, explorable, and interactive 3d worlds from words or pixels. *arXiv preprint arXiv:2507.21809*, 2025.

[3] Zhongqi Yang, Wenhang Ge, Yuqi Li, Jiaqi Chen, Haoyuan Li, Mengyin An, Fei Kang, Hua Xue, Baixin Xu, Yuyang Yin, et al. Matrix-3d: Omnidirectional explorable 3d world generation. *arXiv preprint arXiv:2508.08086*, 2025.

[4] Ben Poole, Ajay Jain, Jonathan T Barron, and Ben Mildenhall. Dreamfusion: Text-to-3d using 2d diffusion. *arXiv preprint arXiv:2209.14988*, 2022.

[5] Chen-Hsuan Lin, Jun Gao, Luming Tang, Towaki Takikawa, Xiaohui Zeng, Xun Huang, Karsten Kreis, Sanja Fidler, Ming-Yu Liu, and Tsung-Yi Lin. Magic3d: High-resolution text-to-3d content creation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 300–309, 2023.

[6] Zhengyi Wang, Cheng Lu, Yikai Wang, Fan Bao, Chongxuan Li, Hang Su, and Jun Zhu. Prolificdreamer: High-fidelity and diverse text-to-3d generation with variational score distillation. *Advances in neural information processing systems*, 36:8406–8441, 2023.

[7] Shuai Yang, Jing Tan, Mengchen Zhang, Tong Wu, Gordon Wetzstein, Ziwei Liu, and Dahua Lin. Layerpano3d: Layered 3d panorama for hyper-immersive scene generation. In *Proceedings of the special interest group on computer graphics and interactive techniques conference conference papers*, pages 1–10, 2025.

[8] Zhaoxi Chen, Guangcong Wang, and Ziwei Liu. Scenedreamer: Unbounded 3d scene generation from 2d image collections. *IEEE transactions on pattern analysis and machine intelligence*, 45(12):15562–15576, 2023.

[9] Black Forest Labs, Stephen Batifol, Andreas Blattmann, Frederic Boesel, Saksham Consul, Cyril Diagne, Tim Dockhorn, Jack English, Zion English, Patrick Esser, Sumith Kulal, Kyle Lacey, Yam Levi, Cheng Li, Dominik Lorenz, Jonas Müller, Dustin Podell, Robin Rombach, Harry Saini, Axel Sauer, and Luke Smith. Flux.1 kontext: Flow matching for in-context image generation and editing in latent space, 2025. URL `https://arxiv.org/abs/2506.15742`.

[10] Team Wan, Ang Wang, Baole Ai, Bin Wen, Chaojie Mao, Chen-Wei Xie, Di Chen, Feiwu Yu, Haiming Zhao, Jianxiao Yang, et al. Wan: Open and advanced large-scale video generative models. *arXiv preprint arXiv:2503.20314*, 2025.

[11] Xiaoyu Li, Qi Zhang, Di Kang, Weihao Cheng, Yiming Gao, Jingbo Zhang, Zhihao Liang, Jing Liao, Yan-Pei Cao, and Ying Shan. Advances in 3d generation: A survey. *arXiv preprint arXiv:2401.17807*, 2024.

[12] Manuel-Andreas Schneider, Lukas Höllein, and Matthias Nießner. Worldexplorer: Towards generating fully navigable 3d scenes. In *Proceedings of the SIGGRAPH Asia 2025 Conference Papers*, pages 1–11, 2025.

[13] Jisheng Chu, Wenrui Li, Rui Zhao, Wangmeng Zuo, Shifeng Chen, and Xiaopeng Fan. Roamscene3d: Immersive text-to-3d scene generation via adaptive object-aware roaming. *arXiv preprint arXiv:2601.19433*, 2026.

[14] Yan Li, Yingzhao Li, and Gim Hee Lee. Active3d: Active high-fidelity 3d reconstruction via multi-level uncertainty quantification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pages 6557–6565, 2026.

[15] Liyan Chen, Huangying Zhan, Kevin Chen, Xiangyu Xu, Qingan Yan, Changjiang Cai, and Yi Xu. Activegamer: Active gaussian mapping through efficient rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16486–16497, 2025.

[16] Yuhao Sun, Ning Cheng, Shixin Zhang, Wenzhuang Li, Lingyue Yang, Shaowei Cui, Huaping Liu, Fuchun Sun, Jianwei Zhang, Di Guo, Wenjuan Han, and Bin Fang. Tactile data generation and applications based on visuo-tactile sensors: A review. *Information Fusion*, 121:103162, 2025.

[17] Yuhao Sun, Shixin Zhang, Wenzhuang Li, Shaobo Yang, Derui Li, Jiatong Du, Di Guo, Jianwei Zhang, and Bin Fang. Tacchi 2.0: A low computational cost and comprehensive dynamic contact simulator for vision-based tactile sensors. *Sensors and Actuators A: Physical*, 405:117856, 2026.

[18] Yuhao Sun, Shixin Zhang, Zixi Chen, Zirong Shen, Fuchun Sun, Cesare Stefanini, Di Guo, Shan Luo, Jianwei Zhang, Jianhua Shan, and Bin Fang. Soft contact simulation and manipulation learning of deformable objects with vision-based tactile sensor. *IEEE Transactions on Automation Science and Engineering*, 22:17618–17630, 2025.

[19] Miguel Angel Bautista, Pengsheng Guo, Samira Abnar, Walter Talbott, Alexander Toshev, Zhuoyuan Chen, Laurent Dinh, Shuangfei Zhai, Hanlin Goh, Daniel Ulbricht, et al. Gaudi: A neural architect for immersive 3d scene generation. *Advances in Neural Information Processing Systems*, 35:25102–25116, 2022.

[20] Seung Wook Kim, Bradley Brown, Kangxue Yin, Karsten Kreis, Katja Schwarz, Daiqing Li, Robin Rombach, Antonio Torralba, and Sanja Fidler. Neuralfield-ldm: Scene generation with hierarchical latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8496–8506, 2023.

[21] Yuheng Liu, Xinke Li, Xueting Li, Lu Qi, Chongshou Li, and Ming-Hsuan Yang. Pyramid diffusion for fine 3d large scene generation. In *European Conference on Computer Vision*, pages 71–87. Springer, 2024.

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

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

[24] Xuanchi Ren and Xiaolong Wang. Look outside the room: Synthesizing a consistent long-term 3d scene video from a single image. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3563–3573, 2022.

[25] Weiyu Li, Xuelin Chen, Jue Wang, and Baoquan Chen. Patch-based 3d natural scene generation from a single example. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16762–16772, 2023.

[26] Derui Li, Qian Qiao, Yuhao Sun, Wenhao Guo, and Peng Lu. Bridging language and spherical space: Object-centric control for text-to-panorama generation. *arXiv preprint arXiv:2608.20691*, 2026.

[27] Wenhao Guo, Zhaoran Zhao, Peng Lu, Sheng Li, Qian Qiao, and RuiDe Li. Casr: A robust cyclic framework for arbitrary large-scale super-resolution with distribution alignment and self-similarity awareness. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2136–2145, 2026.

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

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

[30] Wenrui Li, Fucheng Cai, Yapeng Mi, Zhe Yang, Wangmeng Zuo, Xingtao Wang, and Xiaopeng Fan. Scenedreamer360: Text-driven 3d-consistent scene generation with panoramic gaussian splatting. *arXiv preprint arXiv:2408.13711*, 2024.

[31] Shijie Zhou, Zhiwen Fan, Dejia Xu, Haoran Chang, Pradyumna Chari, Tejas Bharadwaj, Suya You, Zhangyang Wang, and Achuta Kadambi. Dreamscene360: Unconstrained text-to-3d scene generation with panoramic gaussian splatting. In *European Conference on Computer Vision*, pages 324–342. Springer, 2024.

[32] Guo Pu, Yiming Zhao, and Zhouhui Lian. Pano2room: Novel view synthesis from a single indoor panorama. In *SIGGRAPH Asia 2024 conference papers*, pages 1–11, 2024.

[33] Chong Xia, Shengjun Zhang, Fangfu Liu, Chang Liu, Khodchaphun Hirunyaratsameewong, and Yueqi Duan. Scenepainter: Semantically consistent perpetual 3d scene generation with concept relation alignment. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 28808–28817, 2025.

[34] Xuanchi Ren, Tianchang Shen, Jiahui Huang, Huan Ling, Yifan Lu, Merlin Nimier-David, Thomas Müller, Alexander Keller, Sanja Fidler, and Jun Gao. Gen3c: 3d-informed world-consistent video generation with precise camera control. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6121–6132, 2025.

[35] Tianchang Shen, Sherwin Bahmani, Kai He, Sangeetha Grama Srinivasan, Tianshi Cao, Jiawei Ren, Ruilong Li, Zian Wang, Nicholas Sharp, Zan Gojcic, et al. Lyra 2.0: Explorable generative 3d worlds. *arXiv preprint arXiv:2604.13036*, 2026.

[36] Shengyong Chen, Youfu Li, and Ngai Ming Kwok. Active vision in robotic systems: A survey of recent developments. *The International Journal of Robotics Research*, 30(11):1343–1377, 2011.

[37] Andreas Bircher, Mina Kamel, Kostas Alexis, Helen Oleynikova, and Roland Siegwart. Receding horizon "next-best-view" planner for 3d exploration. In *2016 IEEE international conference on robotics and automation (ICRA)*, pages 1462–1468. IEEE, 2016.

[38] Stefan Isler, Reza Sabzevari, Jeffrey Delmerico, and Davide Scaramuzza. An information gain formulation for active volumetric 3d reconstruction. In *2016 IEEE international conference on robotics and automation (ICRA)*, pages 3477–3484. IEEE, 2016.

[39] Yves Kompis, Luca Bartolomei, Ruben Mascaro, Lucas Teixeira, and Margarita Chli. Informed sampling exploration path planner for 3d reconstruction of large scenes. *IEEE Robotics and Automation Letters*, 6 (4):7893–7900, 2021.

[40] Lukas Schmid, Michael Pantic, Raghav Khanna, Lionel Ott, Roland Siegwart, and Juan Nieto. An efficient sampling-based method for online informative path planning in unknown environments. *IEEE Robotics and Automation Letters*, 5(2):1500–1507, 2020.

[41] Yuman Gao, Yingjian Wang, Xingguang Zhong, Tiankai Yang, Mingyang Wang, Zhixiong Xu, Yongchao Wang, Yi Lin, Chao Xu, and Fei Gao. Meeting-merging-mission: A multi-robot coordinate framework for large-scale communication-limited exploration. In *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 13700–13707. IEEE, 2022.

[42] Boyu Zhou, Yichen Zhang, Xinyi Chen, and Shaojie Shen. Fuel: Fast uav exploration using incremental frontier structure and hierarchical planning. *IEEE Robotics and Automation Letters*, 6(2):779–786, 2021.

[43] Liren Jin, Xingguang Zhong, Yue Pan, Jens Behley, Cyrill Stachniss, and Marija Popović. Activegs: Active scene reconstruction using gaussian splatting. *IEEE Robotics and Automation Letters*, 10(5):4866–4873, 2025. doi: 10.1109/LRA.2025.3555149.

[44] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering. *ACM Transactions on Graphics*, 42(4), July 2023. URL `https://repo-sam.inria.fr/fungraph/3d-gaussian-splatting/`.

[45] Ruicheng Wang, Sicheng Xu, Cassie Dai, Jianfeng Xiang, Yu Deng, Xin Tong, and Jiaolong Yang. Moge: Unlocking accurate monocular geometry estimation for open-domain images with optimal training supervision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5261–5271, 2025.

[46] Peter E Hart, Nils J Nilsson, and Bertram Raphael. A formal basis for the heuristic determination of minimum cost paths. *IEEE transactions on Systems Science and Cybernetics*, 4(2):100–107, 1968.

[47] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021.

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

[49] Angel Chang, Angela Dai, Thomas Funkhouser, Maciej Halber, Matthias Niessner, Manolis Savva, Shuran Song, Andy Zeng, and Yinda Zhang. Matterport3d: Learning from rgb-d data in indoor environments. *arXiv preprint arXiv:1709.06158*, 2017.

[50] Yuheng Liu, Xin Lin, Xinke Li, Baihan Yang, Chen Wang, Kalyan Sunkavalli, Yannick Hold-Geoffroy, Hao Tan, Kai Zhang, Xiaohui Xie, et al. Omniroam: World wandering via long-horizon panoramic video generation. *arXiv preprint arXiv:2603.30045*, 2026.

[51] Bingchen Miao, Rong Wei, Zhiqi Ge, Xiaoquan sun, Shiqi Gao, Jingzhe Zhu, Renhan Wang, Siliang Tang, Jun Xiao, Rui Tang, and Juncheng Li. Towards physically executable 3d gaussian for embodied navigation, 2025. URL `https://arxiv.org/abs/2510.21307`.

[52] Luxi Chen, Zihan Zhou, Min Zhao, Yikai Wang, Ge Zhang, Wenhao Huang, Hao Sun, Ji-Rong Wen, and Chongxuan Li. Flexworld: Progressively expanding 3d scenes for flexiable-view synthesis. *arXiv preprint arXiv:2503.13265*, 2025.

[53] Anish Mittal, Anush Krishna Moorthy, and Alan Conrad Bovik. No-reference image quality assessment in the spatial domain. *IEEE Transactions on image processing*, 21(12):4695–4708, 2012.

[54] Anish Mittal, Rajiv Soundararajan, and Alan C. Bovik. Making a "completely blind" image quality analyzer. *IEEE Signal Processing Letters*, 20(3):209–212, 2013. doi: 10.1109/LSP.2012.2227726.

[55] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training gans. *Advances in neural information processing systems*, 29, 2016.

[56] Jack Hessel, Ari Holtzman, Maxwell Forbes, Ronan Le Bras, and Yejin Choi. Clipscore: A reference-free evaluation metric for image captioning. In *Proceedings of the 2021 conference on empirical methods in natural language processing*, pages 7514–7528, 2021.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The claims in the abstract and introduction are consistent with the paper scope. We state the limitations of passive trajectory-based text-to-3D generation, introduce ActPano3D and its three key components, and evaluate the claimed improvements in scene completeness, geometric consistency, and free-viewpoint rendering through quantitative results, ablations, and qualitative comparisons. We restrict our claims to the evaluated panoramic, trajectory-conditioned text-to-3D setting.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: The paper discusses the scope of the current framework, including its focus on indoor environments and static scene generation. We note that extending ActPano3D to large-scale outdoor scenes and dynamic environments with moving objects or time-varying content remains future work.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best

judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [N/A]

   Justification: The paper does not present formal theoretical results, theorems, or proofs. The equations in the method section are used to define the proposed planning, generation, fusion, and optimization procedures rather than to state theoretical claims requiring formal proof.

   Guidelines:

   - The answer [N/A] means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: The paper provides the experimental setup needed to reproduce the main results, including the evaluated datasets and scene settings, baseline methods, evaluation metrics, implementation details, and ablation configurations. We describe the active planning, panoramic generation, scene memory fusion, and final 3DGS optimization procedures used in our experiments. The main claims are supported by quantitative comparisons, qualitative results, and component-wise ablations under the same evaluation protocol.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.

(b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.

(c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).

(d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [No]

   Justification: Our implementation and experiment-specific data have not yet been publicly released. We plan to release the code, model configurations, and data preparation scripts upon publication, subject to the licenses and terms of use of the underlying datasets and third-party assets.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: The paper specifies the main experimental settings needed to understand the results, including the evaluated datasets and scene settings, baseline methods, evaluation metrics, implementation details, and ablation configurations. We describe the training and inference setup of the panoramic generation model, the active exploration protocol, key hyperparameters, and the final 3DGS optimization settings used for evaluation.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.

- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

   Justification: The current paper reports quantitative results averaged over the evaluation scenes but does not include error bars, confidence intervals, or formal statistical significance tests. Since the experiments involve computationally expensive panoramic generation, iterative 3D reconstruction, and 3DGS optimization, repeated full runs with different random seeds are not included in the current submission. We use consistent datasets, evaluation protocols, and ablation settings to support fair comparison across methods.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.
   - It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
   - For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
   - If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: The paper reports the GPU resources used in the experiments, including the GPU type and the compute setting for training and evaluation.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

Answer: [Yes]

Justification: The research conforms to the NeurIPS Code of Ethics. The work does not involve human subjects, private personal data, or sensitive user information, and all datasets and baselines are properly cited.

Guidelines:

- The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [Yes]

Justification: The paper discusses potential positive impacts of text-to-3D scene generation for immersive content creation, simulation, virtual environments, and interactive design.

Guidelines:

- The answer [N/A] means that there is no societal impact of the work performed.
- If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

Answer: [N/A]

Justification: The current submission does not release pretrained generative model weights, scraped datasets, or other high-risk data or models. The paper uses publicly available or properly cited datasets.

Guidelines:

- The answer [N/A] means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.

- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: The paper properly cites the datasets, models, codebases, and baseline methods used in our experiments. We use these assets for academic research and follow their corresponding licenses and terms of use. The relevant assets include publicly available datasets, pretrained generative models, depth estimation models, and 3D reconstruction codebases, all of which are credited in the paper.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [N/A]

    Justification: The current submission does not release new datasets, codebases, model weights, or other standalone assets. The paper describes the constructed evaluation benchmark and experimental protocol, but no new asset is released alongside the submission.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

Justification: The paper does not involve crowdsourcing experiments or research with human subjects. No human participants are recruited for data collection, evaluation, or user studies.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [N/A]

Justification: The paper does not involve crowdsourcing experiments or research with human subjects, and therefore IRB approval or equivalent review is not applicable.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

Answer: [N/A]

Justification: The core method does not use LLMs as an important, original, or non-standard component.

Guidelines:

- The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.