

Video Perception Models for 3D Scene Synthesis

Rui Huang^{1*} Guangyao Zhai^{2,4*} Zuria Bauer³ Marc Pollefeys^{3,5}
Federico Tombari² Leonidas Guibas⁶ Gao Huang^{1†} Francis Engelmann⁶

¹Tsinghua University ²Technical University of Munich ³ETH Zurich
⁴Munich Center for Machine Learning ⁵Microsoft ⁶Stanford University

<https://vipscene.github.io>

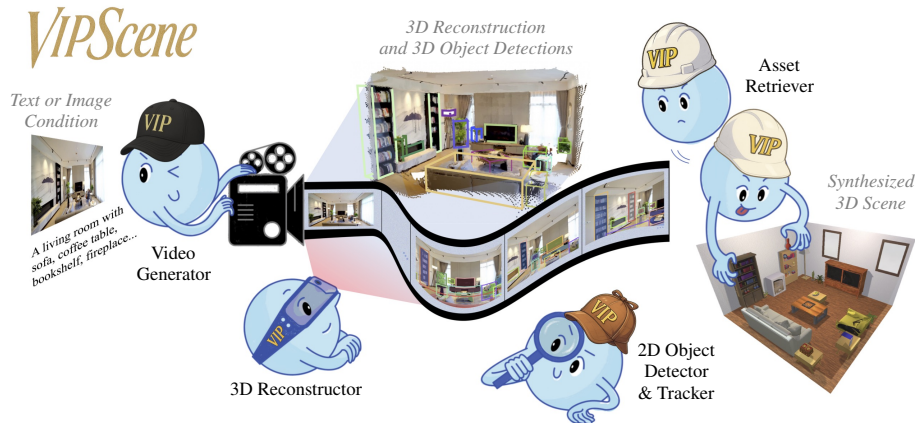


Figure 1: **Schematic of VIPSCENE.** Conditioned on text or image prompts, VIPSCENE generates scenes by leveraging the commonsense priors of video generation models for scene layout and object placements. From the generated video, we reconstruct the 3D scene and extract individual objects. The final scene is synthesized by replacing detected objects with high-quality 3D assets from an object database.

Abstract

Automating the expert-dependent and labor-intensive task of 3D scene synthesis would significantly benefit fields such as architectural design, robotics simulation, and virtual reality. Recent approaches to 3D scene synthesis often rely on the commonsense reasoning of large language models (LLMs) or strong visual priors from image generation models. However, current LLMs exhibit limited 3D spatial reasoning, undermining the realism and global coherence of synthesized scenes, while image-generation-based methods often constrain viewpoint control and introduce multi-view inconsistencies. In this work, we present **Video Perception models for 3D Scene synthesis (VIPSCENE)**, a novel framework that exploits the encoded commonsense knowledge of the 3D physical world in video generation models to ensure coherent scene layouts and consistent object placements across views. VIPSCENE accepts both text and image prompts and seamlessly integrates video generation, feedforward 3D reconstruction, and open-vocabulary perception models to semantically and geometrically analyze each object in a scene. This enables flexible scene synthesis with high realism and structural consistency. For a more sufficient evaluation on coherence and plausibility, we further introduce **First-Person View Score (FPVSCORE)**, utilizing a continuous first-person perspective to capitalize on the reasoning ability of multimodal large language models. Extensive experiments show that VIPSCENE significantly outperforms existing methods and generalizes well across diverse scenarios.

*Equal contribution. †Corresponding author.

1 Introduction



Figure 2: **Diverse Synthesized Environments.** From text prompts, VIPSCENE generates (a) a bookstore and (b) a children’s room. From an image prompt (c), VIPSCENE synthesizes a corresponding outdoor scene. Across settings, objects are semantically aligned with the prompts and arranged in physically plausible layouts.

Recent advancements in 3D scene synthesis have sparked significant interest across multiple domains, including gaming [22], augmented reality [1, 50], and robotics [6, 69]. Deep learning-based methods have enabled the automatic generation of 3D scenes; however, they are limited by the insufficient diversity of available datasets. Recent progress in language and image generative models [2, 18, 53–55, 11, 37] has further expanded the possibilities for synthesizing more diverse and plausible 3D scenes. A crucial step in this progress is the generation of spatially coherent layouts, which serve as the foundational structure for building lifelike 3D environments.

Despite impressive progress, existing methods still face notable limitations in generating realistic layouts. One promising direction involves leveraging multimodal large language models (MLLMs) [69, 48, 9, 27, 8, 4, 3]. Among them, pure large language models (LLMs), which typically translate linguistic priors into layout constraints, often produce incomplete spatial specifications and treat scene synthesis as an optimization problem, which can result in the loss of spatial commonsense [69]. Vision-language models (VLMs) [18, 30] attempt to address this issue by incorporating visual context to enhance spatial reasoning. However, their reliance on fixed image viewpoints limits their ability to generalize to 3D layout understanding [48, 34]. Alternatively, image-based approaches [60, 15] adopt specific viewpoints and apply recurrent inpainting strategies to directly exploit visual priors for commonsense layout generation. These methods, however, often suffer from hallucinations and spatial inconsistencies due to viewpoint dependency and the iterative nature.

Moreover, existing automated evaluation protocols often overlook spatial inconsistencies and unrealistic layout distributions. Metrics such as CLIPScore [13] and VQAScore [28], based on VLMs, typically rely on a single top-down view [69, 60]. This perspective can obscure important object details and impede accurate semantic interpretation (see Fig. 7), thus hindering effective assessment of layout coherence. Additionally, top-down views are also likely underrepresented in VLM training data, further reducing their ability to interpret such perspectives. Consequently, these metrics alone can not reliably reflect scene generation quality (see Tab. 2).

In this work, we approach the problem from two complementary perspectives: 3D scene synthesis and evaluation, with a focus on commonsense reasoning and spatial coherence. For scene synthesis, we introduce *Video Perception Models for 3D Scene Synthesis* (VIPSCENE), which leverage rich visual priors from video generation models to capture coherent scene layouts and object placements across views. As shown in Fig. 1, conditioned on either a text or image prompt or both, VIPSCENE first generates an egocentric video of a scene, based on which the feed-forward 3D reconstruction recovers global geometry and open-vocabulary perception segments object instances. The instances enable a large-scale asset retriever to substitute editable 3D assets, followed by a global optimization step to refine object poses, ensuring physical plausibility and avoiding collisions. VIPSCENE enables the generation of realistic, semantically faithful, and spatially coherent 3D scenes (see Fig. 2). Complementing this, we also propose a novel evaluation protocol, the *First-Person View Score* (FPVSCORE), which leverages advanced MLLMs like GPT-4o [18] and Gemini [53, 54] to assess generated scenes. A virtual camera captures 360° first-person views, which are concatenated into visual summaries and analyzed by MLLMs through structured prompts. This approach offers a more interpretable and human-aligned evaluation of layout realism, spatial consistency, and semantic fidelity. Experiments show that VIPSCENE surpasses state-of-the-art baselines across standard metrics, while FPVSCORE outperforms existing metrics and aligns closely with human judgments.

In summary, our contributions are threefold: **(i)** We present VIPSCENE, a novel approach for realistic 3D scene synthesis that leverages video-based commonsense layout understanding, enriched with semantic and geometric cues obtained through consistent 3D reconstruction and perception. **(ii)** We introduce FPVSCORE, a first-person view-based evaluation protocol that leverages MLLMs for more comprehensive, interpretable, and human-aligned assessment of spatial coherence and semantic fidelity. **(iii)** We show that VIPSCENE outperforms state-of-the-art baselines across all metrics, demonstrating the effectiveness of video-grounded priors and a modular decomposition pipeline for generating physically plausible 3D scenes.

2 Related Work

Indoor 3D Scene Synthesis. Indoor 3D scene synthesis has gained attention for applications in robotics [72, 32] and augmented reality [1, 50]. Existing methods generate scenes from language [51, 69, 48, 39, 10], graph-based instructions [71, 73, 26], or images [17, 60], producing either object-level layouts [71, 70] or complete mesh scenes [15, 47]. However, models trained from scratch often suffer from dataset biases [71, 73, 67, 51]. Recent work thus leverages foundation models for broader generalization. While large language models (LLMs) can capture inter-object commonsense from text, they often yield incomplete or ambiguous spatial layouts due to the lack of visual grounding [9, 69, 4, 3, 68]. In contrast, vision-language models (VLMs) incorporate visual cues to enhance spatial reasoning, but their reliance on fixed viewpoints limits their ability to infer holistic 3D scene layouts [48, 8]. Image-based approaches [60, 15, 29] attempt to exploit commonsense visual priors for scene layout generation, but those based on single or multi-view images often struggle with limited viewpoint coverage and inconsistent spatial alignment across views. In this work, we leverage large-scale pre-trained video diffusion models [33] to generate long-horizon clips from textual or image conditions, ensuring consistent viewpoints while enriching visual details.

Video Models. Building on the success of image synthesis [38], recent research has shifted toward video generation, which introduces challenges like temporal consistency and dynamic content modeling. Early efforts mainly relied on GAN-based frameworks [56, 43] that, while producing plausible results in controlled settings, often suffered from mode collapse and temporal incoherence. Recent approaches have extended diffusion models to the video domain [14, 65], leveraging their robustness to generate temporally coherent sequences. With the rise of Diffusion Transformers (DiT)[35], methods now produce highly consistent photorealistic videos[21, 25, 31, 42, 33, 19], revolutionizing applications in film, robotics, and other downstream tasks. In this work, we employ advanced video models to synthesize scenes with consistent views and broad viewpoint coverage, providing a strong foundation for generating 3D scenes with coherent layouts.

3D Geometric Reconstruction. Early approaches primarily focused on Structure from Motion (SfM) techniques, where keypoint detection and matching formed the basis for estimating camera poses and sparse point clouds [44, 45]. More recently, learning-based approaches [57, 49, 52, 58, 64], including DUST3R [59] and MAST3R [23], aim to eliminate iterative optimization and complex post-processing by directly performing multiple 3D tasks through a feedforward network. These methods typically integrate data-driven priors to mitigate ambiguities that classical techniques encounter, demonstrating strong generalization capabilities. In this work, we rely on Fast3R[64] for 3D reconstruction, and MAST3R[23] for tracking of objects from the generated video.

3 Method

The goal of VIPSCENE is to generate a realistic and physically plausible 3D scene from a user-specified prompt. Formally, given an image- or text-prompt, VIPSCENE generates a 3D scene $S = \{o_1, \dots, o_N\}$, where each object $o_i = (c_i, s_i, l_i, \theta_i)$ is represented by its category $c_i \in \mathcal{C}$, size $s_i \in \mathbb{R}^3$, position $l_i \in \mathbb{R}^3$, and orientation $\theta_i \in \mathbb{R}$ around the gravity axis. Conditioned on the prompt, VIPSCENE first generates a first-person view video of a 3D indoor scene, leveraging commonsense knowledge on scene layout as well as object placements embedded in the video generation model. From this video, we employ recent 3D reconstruction and visual perception models to recover the full 3D scene geometry and decompose it into its individual objects (Sec. 3.1). The scene is then recomposed by retrieving the most similar 3D assets from an object database. To ensure physical plausibility and resolve collisions, an additional optimization step refines object placements (Sec. 3.2). The overall framework is illustrated in Fig. 1, and a detailed algorithm is included in Algorithm 1. In summary, VIPSCENE effectively integrates commonsense knowledge into generated 3D scenes, capturing both the input prompt semantics and the physical plausibility of spatial layouts.

3.1 Scene Understanding

Scene Reconstruction. Given an input prompt, a conditional video generator (Cosmos [33]) produces a high-fidelity video. For 3D reconstruction, we first sample frames at 2 fps, yielding $\{I_1, \dots, I_T\}$, where each frame $I_j \in \mathbb{R}^{3 \times H \times W}$ for $j = 1, \dots, T$, representing diverse views of the scene. Trained on web-scale videos, the generator captures commonsense layout priors and spatial relationships, naturally extending the perceptual field beyond image-based methods like Architect [60]. Next, we process all unposed frames in parallel using a multi-view 3D reconstruction method to produce the 3D scene reconstruction R , implemented with Fast3R [64]. For metric 3D reconstruction, we estimate metric depth for each frame using the monocular predictor UniDepth [36], and rescale the reconstructed scene accordingly.

Object Detection. Next, we aim to detect all 3D objects in the reconstructed scene R . While off-the-shelf 3D object detectors are a natural choice, we found them to perform poorly on the noisy reconstructed point cloud R , leading to inaccurate object categorization and size estimation, as shown in Fig. 8. Instead, we adopt an image-based approach. Specifically, we apply Grounded-SAM [41] to detect and segment objects of interest independently in each frame, and then use MAST3R [23] to track and associate 2D detections across frames using its strong multi-view pixel-correspondence estimation capabilities. In this way, we can assign a unique identifier i to each object in the 3D scene. Specifically, for an object i in frame t , we store its binary 2D object mask M_t^i . We then use the masks across all views, $M_1^i, \dots, M_T^i$, to extract the corresponding points from the reconstructed point cloud R . Given the high degree of noise in R , we propose an adaptive erosion scheme that filters out artifacts while preserving the object geometry. Specifically, we apply morphological erosion to each binary object mask M to suppress edge noise, with the erosion strength scaled by object size: larger objects undergo more aggressive denoising, while smaller objects are subject to gentler erosion. This allows us to obtain a clean and accurate point cloud P_i for each object i .

3.2 Scene Assembly

3D Asset Retrieval. In this stage, the goal is to replace the object point clouds P_i with actual 3D assets. Towards that end, we pick the most similar asset from a large-scale object database based on estimated object properties. Specifically, we extend beyond prior approaches (e.g., Holodeck [69]) that rely primarily on visual similarity, textual relevance, and size similarity. Instead, we additionally adopt a point cloud registration-based retrieval strategy to identify the most suitable asset candidates. Given an object point cloud P_i , we first estimate its orientation θ_i . We apply Principal Component Analysis (PCA) to compute the principal axes and use the direction of greatest variance to approximate the orientation. A tight bounding box is then aligned with θ_i , from which position l_i and size s_i are derived. As PCA cannot distinguish between 0 and π , each object yields two symmetric poses. For each candidate asset, we compute a rigid transformation $\mathbf{T} = [\mathbf{R}|\mathbf{t}]$, with rotation $\mathbf{R} \in \mathbb{R}^{3 \times 3}$ and translation $\mathbf{t} \in \mathbb{R}^3$, by minimizing:

$$[\mathbf{R}^*, \mathbf{t}^*] = \underset{\mathbf{R}, \mathbf{t}}{\operatorname{argmin}} \sum_{q \in Q_j} \left(\min_{p \in P_i} \|\mathbf{R}q + \mathbf{t} - p\|^2 \right) + I_{\text{SO}(3)}(\mathbf{R}), \quad (1)$$

where p is a point in the object’s point cloud P_i and q is the closest point in the candidate asset Q_j . The term $I_{\text{SO}(3)}(\mathbf{R})$ ensures that the rotation matrix $\mathbf{R}$ remains within the special orthogonal group $\text{SO}(3)$ as defined in [74]. To solve this optimization problem, we use the Iterative Closest Point (ICP), initialized with the estimated poses. To identify the most suitable asset for each object, we select the candidate with the lowest root mean square error (RMSE), resulting in a scene composed of geometrically well-aligned assets. This step ensures accurate geometric alignment of the assets.

Object Pose Refinement. To address potential collisions caused by size mismatches between retrieved assets and objects, we introduce an optimization step to refine object placements in a physically plausible manner. This ensures that objects avoid overlap, stay within room boundaries (if specified), and remain close to their initial position. We define the total loss as a weighted sum of three losses, the position loss $\mathcal{L}_p$, the overlap loss $\mathcal{L}_o$, and the optional boundary loss $\mathcal{L}_b$:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_p + \lambda_o \mathcal{L}_o + \lambda_b \mathcal{L}_b, \quad (2)$$

where λ_o and λ_b are weighting parameters, and the individual loss terms are defined as:

$$\mathcal{L}_p = \sum_{i=1}^n |l_i - l_i^{\text{orig}}|_2^2, \quad \mathcal{L}_o = \sum_{i \neq j} \text{Area}(\text{BBox}_i \cap \text{BBox}_j), \quad \mathcal{L}_b = \sum_{i=1}^N \text{Area}(\text{BBox}_i \setminus \text{Room}).$$

The position loss $\mathcal{L}_p$ encourages minimal deviation from the objects’ original locations l . The overlap loss $\mathcal{L}_o$ penalizes intersecting object pairs based on the area of their bounding boxes. The boundary loss $\mathcal{L}_b$ penalizes any part of an object that lies outside the room bounds, if provided. During optimization, object positions are iteratively updated along the gradient of $\mathcal{L}_{\text{total}}$ until convergence. The process terminates once overlap and boundary violations are eliminated or when no significant improvements are observed. This results in a collision-free, spatially coherent scene layout.

4 First-Person View Score – FPVSCORE

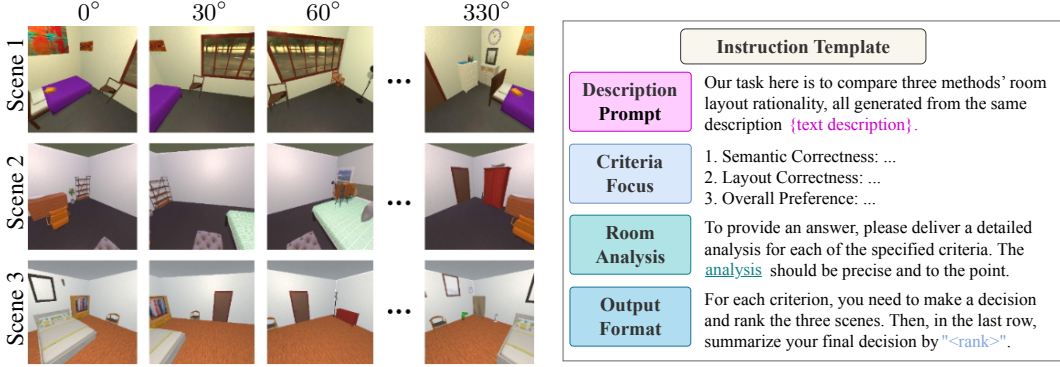


Figure 3: **Illustration of First-Person View Score.** Rather than relying on a *single top-down view*, our metric uses a *sequence of first-person view images* for each generated scene (*left*). A multimodal language model (MLLM) then analyzes and ranks the sequences based on multiple evaluation criteria (*right*).

User studies remain the gold standard for evaluating the quality of generated 3D scenes, as they accurately reflect human judgment. However, they are expensive, time-consuming, and difficult to scale. Recent works [69, 60] have explored *automated* evaluation using vision-language models (VLMs) like CLIP [40] and BLIP [24], which assess how well a *top-down* rendering aligns with a given text prompt. For example, CLIPScore[13] computes the cosine similarity between the text encoding of the prompt and the image encoding of the rendered image.

Yet, top-down views may be underrepresented in the training distributions of VLMs, potentially limiting their ability to interpret such inputs accurately. Capturing the full scene layout, including fine-grained geometry and object semantics, in a single image embedding is inherently difficult. These views may also obscure key visual details, further reducing alignment fidelity. As a result, the scores produced by these metrics are often unreliable and hard to interpret meaningfully.

Instead, we introduce an alternative evaluation protocol, called FPVSCORE, that uses multiple *first-person* views, which better reflect the training distributions of foundation models and offer improved scene coverage. Crucially, rather than using a single similarity metric, we exploit the perceptual and reasoning capabilities of multimodal models such as GPT-4o [18] and Gemini [53, 54] to rank the outputs of 3D scene generation models and to explain the rationale behind each ranking.

Metric Details and Prompt Design. Fig. 3 illustrates our first-person view metric. For each scene, a virtual camera is placed at the center and simulates a 360-degree rotation, rendering frames at 30-degree intervals to ensure comprehensive coverage and sufficient overlap. These frames are horizontally concatenated to form a compact visual summary. To enable consistent comparison across methods, we stack the summaries for all scenes and input them into the multimodal large language models (MLLMs) simultaneously, avoiding inconsistencies that could arise when evaluating them in isolation. Building on prior work in multimodal model-based 3D object assessment [61], we design a structured prompt to enable comparative evaluation of scene generations. As illustrated in Fig. 3, the prompt comprises three key elements: (1) task-specific instructions defining the multi-scene comparison goal, (2) a clear list of evaluation criteria, and (3) formatting guidelines to ensure consistent output. The prompt guides the model to assess scenes along dimensions such as semantic correctness, layout accuracy, and overall coherence. We further instruct the model to justify its ratings with brief explanations, enabling verification of its reasoning and increasing trust in the evaluation. This design facilitates a more holistic, interpretable, and scalable evaluation protocol for 3D scene generation, addressing the limitations of traditional top-down metrics by aligning more closely with human-like reasoning patterns (see Tab. 2).

5 Experiments

Experimental Details. We utilize Cosmos [33] for video generation and adopt Fast3R [64] for 3D reconstruction. For open-vocabulary segmentation, we employ Grounded-SAM [41], and UniDepth [36] is applied for monocular depth estimation. For comparison with baselines, we follow prior work [69] and evaluate on four types of scenes *living room*, *bedroom*, *bathroom*, and *kitchen*. We ask GPT-4o [18] to produce 25 text prompts for each room type. Each prompt consists of a description of a room type and the desired items. Based on these prompts, we generate 100 rooms using each method under evaluation. We set $\lambda_o = \lambda_b = 10$. Consistent with prior work, Holodeck [69], we retrieve 3D models from a high-quality subset of Objaverse [7] to ensure realistic and diverse object representations in the scene. Please refer to the appendix for more details.

Baselines. We compare our method with the most recent state-of-the-art approaches for 3D scene synthesis: Holodeck [69] is a comprehensive system that integrates LLM-based scene generation with optimization steps to jointly produce room layouts and object placements. Architect [60] is a generative framework that creates interactive 3D scenes through diffusion-based 2D inpainting, relying on visual priors extracted from single images.

Metrics. To evaluate the quality of generated scenes, we report scores using the proposed first-person view metric FPVSCORE (Sec.4) and other automatic metrics used in prior work [60]: (1) CLIPScore [13], which measures image-text similarity via CLIP embeddings; (2) BLIPScore, which evaluates image-caption alignment using the matching head of BLIPv2 [24]; (3) VQAScore [28], which uses a visual question answering model to score how likely an image depicts the given caption; and (4) GPT-4o Ranking [18], which prompts GPT-4o to rank top-down rendered views.

User Study. We conduct a user study to compare scenes generated by our method against baseline approaches. Participants are shown a 360-degree video captured from the center of each scene, along with a top-down rendered image, allowing them to assess both global structure and fine details. Thirty graduate students rated the scenes on a 3-point scale (1 = lowest, 3 = highest) across three criteria: Prompt Adherence (PA), “*To what extent does the generated scene align with the input prompt?*”, Layout Correctness (LC), “*Are the object placements physically plausible and functionally sensible?*”, and Overall Preference (OP).

5.1 Quantitative Results

Tab.1 presents quantitative results comparing our VIPSCENE to the Holodeck[69] and Architect [60] baselines. We report 2D image-based metrics using both first-person and top-down views, along with user study outcomes. VIPSCENE outperforms both baselines across all metrics. The user study, our most reliable evaluation, indicates that VIPSCENE better captures prompt semantics, produces more realistic scene layouts, and is the overall preferred method. This trend is also reflected in our proposed first-person view metric across three different VLMs (Gemini 2.0, GPT-4o, GPT 4.1). Top-down view scores offer a much less clear picture. According to these metrics, performance across methods is nearly indistinguishable, a finding not supported by the more reliable user study. This supports our intuition that top-down view metrics are poorly suited for evaluating 3D scene generation; we examine this further in Sec. 5.3. Another notable finding from our study is that Architect generally underperforms compared to Holodeck on the 100 generated scenes. This contrasts with the results reported in [60], but is partially supported by the qualitative examples in Fig. 4, which illustrate that Architect often produces unusual and sometimes impractical object placements.

Method	First-Person View Scores			User Study			Top-Down View Scores			
	Gemini 2.0↑	GPT-4o↑	GPT 4.1↑	PA↑	LC↑	OP↑	CLIP↑	BLIP↑	VQAScore↑	GPT-4o↑
Holodeck [69]	1.92	2.02	1.94	2.31	2.06	2.05	29.17	51.27	81.43	1.98
Architect [60]	1.77	1.62	1.76	2.05	1.94	1.98	29.95	49.72	78.34	1.90
VIPSCENE (Ours)	2.32	2.45	2.43	2.52	2.51	2.39	29.98	54.36	82.13	2.12

Table 1: **First-Person View, User Study, and Top-Down View Scores.** We report scores across different evaluation metrics. For First-Person View Scores and GPT-4o-based metrics, we report average ranking where 3 is the best and 1 is the worst. Prompt Adherence (PA), Layout Correctness (LC), Overall Preference (OP).

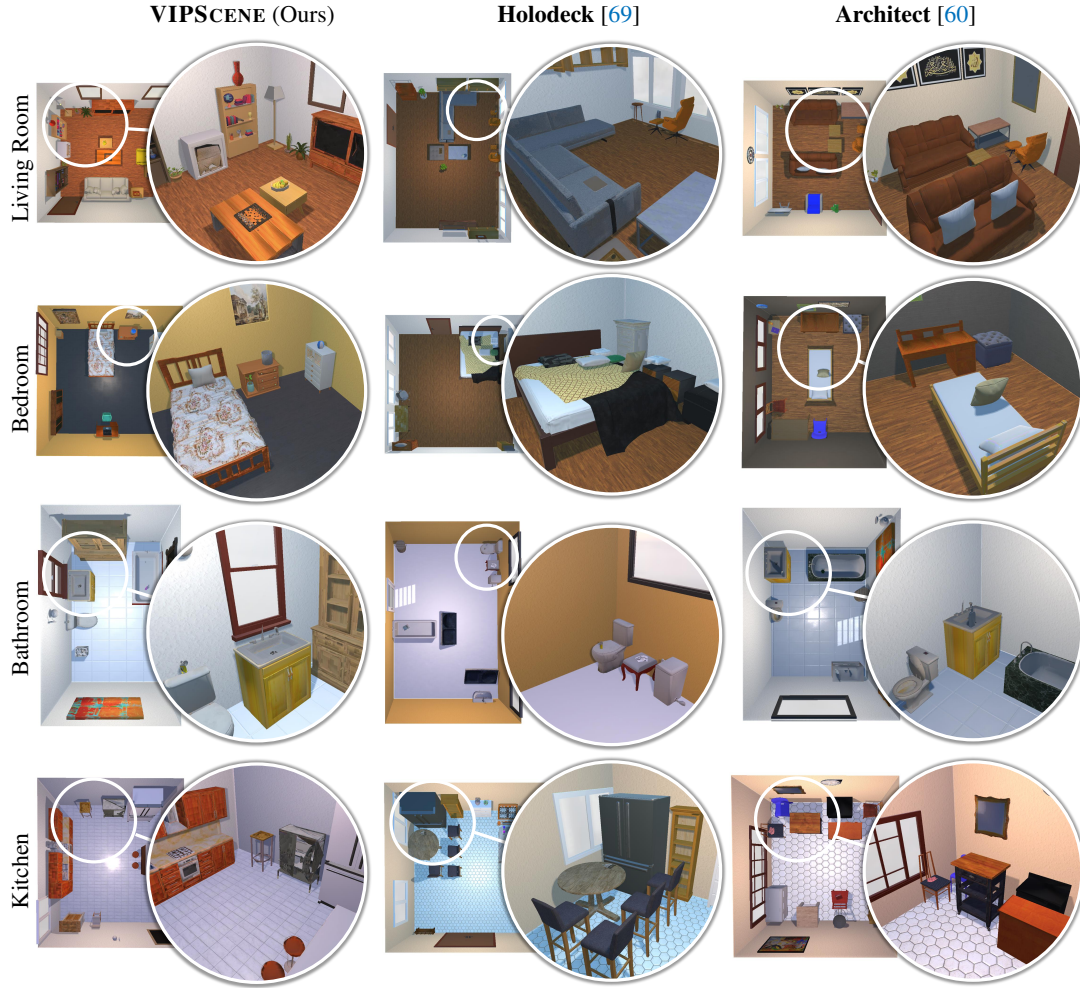


Figure 4: **Qualitative Results.** We present top-down and close-up views of our VIPSCENE, comparing it against Holodeck [69] and Architect [60] (*columns*) across four room types (*rows*). In the figures, Holodeck clearly leaves large areas unused while over-cluttering others, whereas Architect produces implausible arrangements that are impractical and rarely seen in real environments. VIPSCENE generates room layouts that are overall more realistic and natural.



Figure 5: **Image Prompting and Scene Realism.** *Left:* Examples of image-based prompting: given an input image, VIPSCENE generates a video and reconstructs a full 3D scene. Note that based on the information from the first frame, the generated video can plausibly hallucinate objects beyond the original field of view, such as the white sofa near the observer in the living room or the side window in the bedroom. *Right:* Additional detailed comparison of scene layouts. Unlike Holodeck [69] (*top*), which handles object placement and window layout separately, our method (*bottom*) jointly reasons about their spatial relationships, avoiding window occlusions and yielding more coherent, realistic arrangements.

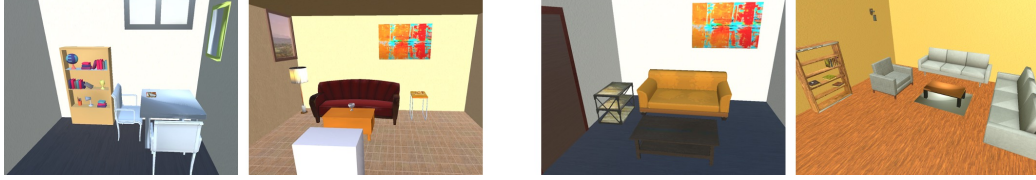


Figure 6: **Results from Complex Inputs.** *Left:* Our model can effectively generate coherent videos and corresponding 3D scenes from detailed spatial prompts, such as "a desk placed under a window with a bookshelf to its left" and "a modern living room with a red sofa facing a fireplace, a coffee table in between, and a floor lamp placed diagonally behind the sofa." *Right:* To further enhance realism, we utilize CLIP features from cropped object regions during retrieval. This improves the alignment of textures and materials with input prompts like "a yellow sofa in the center of the room, with a dark rectangular coffee table in front and a metal shelf next to the sofa" and "the room has two light-colored sofas, an upholstered armchair nearby, a modern brown coffee table in the center, and a wooden bookshelf against the wall."

5.2 Qualitative Results

We present qualitative results in Fig. 4 to 6. For additional results, including those from multimodal inputs, please refer to the appendix. Fig. 4 specifically compares our VIPSCENE with Holodeck [69] and Architect [60], displaying randomly selected outputs from text prompts for living room, bedroom, bathroom, and kitchen scenes. Holodeck, relying solely on LLMs for spatial constraints and object relationships, often produces implausible absolute placements despite semantically correct relative pairings (e.g., chairs around a table). This is because LLMs exhibit limited 3D spatial reasoning, leading to objects being too close for passage while large room areas remain unused. Architect employs multi-view image diffusion and inpainting to hierarchically populate corner views, yielding a more natural mix of object scales. However, its resulting layouts are often incoherent, with issues like multiple sofas facing the same direction and implausible furniture placements. This incoherence likely stems from view inconsistencies and inpainting limitations.

Fig. 5 (left) shows scenes generated from image inputs by replacing the text-to-video model with an image-to-video model [21], while keeping the rest of the pipeline unchanged. The generated scenes demonstrate strong realism and coherence, with objects arranged in a manner consistent with the overall video context. It is worth noting that, relying solely on the first frame, the generated video can reasonably infer objects beyond the original field of view, such as the white sofa near the observer in the living room and the side window in the bedroom. Fig. 5 (right) showcases how existing methods overlook the joint spatial relationship between objects and windows, whereas our method explicitly models their relative placement. By leveraging spatial patterns in video data, our approach achieves greater logical consistency and enhanced visual plausibility.

We further evaluate our model using more complex textual descriptions, as illustrated in Fig. 6. The video generation model successfully interprets these detailed spatial instructions, producing coherent and realistic layouts that reflect the specified relationships among objects. These generated videos are subsequently used to reconstruct the corresponding 3D scenes, as shown in Fig. 6 (left). Our initial retrieval strategy primarily focused on object category and point cloud geometry to ensure overall scene plausibility. To better align retrieved assets with the textures and materials implied by the input prompt, we conduct additional experiments leveraging CLIP image features extracted from cropped object regions in the video frames. This enhancement enables the system to retrieve assets that more accurately reflect the semantic and visual cues specified in the text, as demonstrated in Fig. 6 (right).

5.3 Metrics Evaluation

A key challenge in evaluating generated 3D scenes is the lack of a suitable metric. Prior work [69, 60] typically uses 2D image-based metrics such as CLIPScore [13], BLIPScore [24], or VQAScore [28]; however, motivated by the surprisingly similar scores across methods observed in Sec. 5.1 (Tab. 1), we investigate how well these metrics actually align with human preferences. Specifically, we compute Kendall’s τ correlation [20] between metric-generated scores and reference scores from hu-

Metrics	τ ($\uparrow$)
CLIPScore (single top-down) [13]	0.06
BLIPScore (single top-down) [24]	0.07
VQAScore (single top-down) [28]	0.13
GPT-4o (single top-down)	0.27
GPT-4o (FPV SCORE, first-person views)	0.39

Table 2: **Metrics Evaluation.** How well do automated metrics agree with human ratings? Scores are Kendall’s τ correlations. Perfect agreement is $\tau = 1$, no association is $\tau = 0$, perfect disagreement is $\tau = -1$.

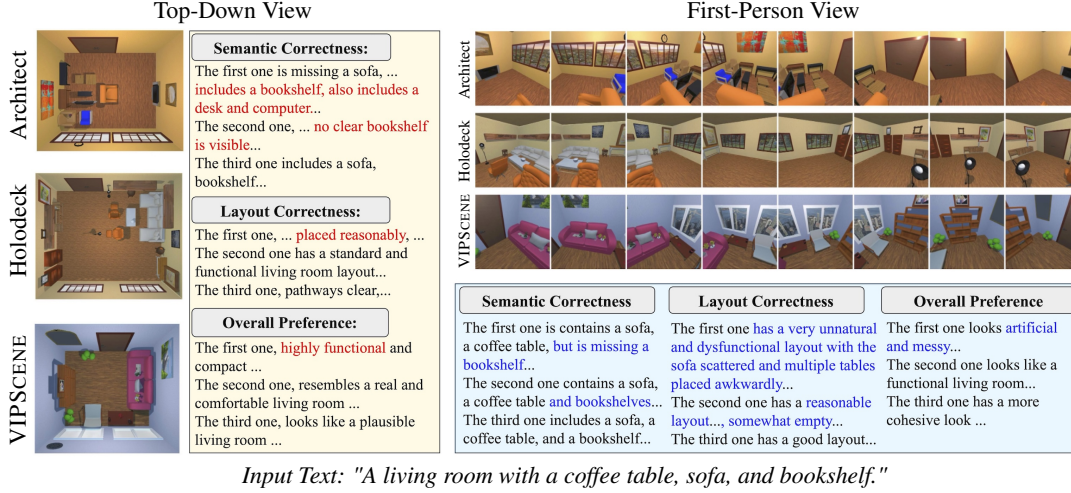


Figure 7: **GPT-4o Output Analysis.** Exemplary GPT-4o output: top-down view (left) and first-person view (right). Red text highlights implausible results; blue text marks reasonable ones.

man evaluators. Tab. 2 reports the correlations, showing how closely each metric’s predictions align with human judgments. Both CLIPScore and BLIPScore exhibit almost no association with human preference, suggesting that these metrics are not suitable for automated evaluation. Although overall correlations are relatively low (reflecting the subjectivity of the task and variability in human judgments), our proposed first-person view metric, FPVSCORE, shows the strongest alignment with human preferences, suggesting that first-person views provide richer semantic cues leading to more reliable evaluations. Fig. 7 illustrates a comparison of the same model’s reasoning for top-down and first-person views. Top-down views frequently obscure key object details, hindering semantic understanding and impairing layout evaluation.

Model	τ	Model	τ	Prompt	τ
Gemini 2.0	0.61	Gemini 2.0 & GPT-4o	0.54	w/o criteria	0.31
GPT-4o	0.74	Gemini 2.0 & GPT-4.1	0.47	and analysis	
GPT-4.1	0.71	GPT-4.1 & GPT-4o	0.60	Ours	0.39
(a) τ vs. MLLMs		(b) Agreement across MLLMs		(c) Prompt design	

Table 3: **FPVScore Consistency Evaluation using Kendall’s τ correlation:** Single-model stability, inter-model agreement, and prompt variants.

We further analyze the consistency of FPVSCORE to validate FPVScore’s reliability, focusing on three aspects: (i) *Consistency Across Tries*. We repeatedly query each model and compute the average Kendall’s τ correlation. Results (Tab. 3a) show MLLMs yield stable outputs, with GPT-4o demonstrating the highest consistency. (ii) *Consistency Across Models*. We assess agreement among different MLLMs by measuring pairwise Kendall’s τ correlations of their scene rankings. Results (Tab. 3b) indicate models exhibit similar relative judgments. (iii) *Prompt Design*. Removing specific prompt instructions (Tab. 3c) reduces Kendall’s τ correlation, suggesting structured prompts enhance human alignment beyond MLLMs’ inherent reasoning.

5.4 Ablation Study

In this section, we present ablation studies on a randomly selected subset of scenes to evaluate the contribution of individual model components. For each variant with one component removed, we compare it against the full VIPSCENE model in a user preference test. Fig. 8 reports the percentage of times VIPSCENE was favored over the model variant.

2D vs. 3D Perception Models. After reconstructing the scene’s point cloud, one straightforward approach is to directly apply a 3D perception model for scene decomposition. However, in practice, we observe that using a 3D model, such as Mask3D [46], yields suboptimal results. The model suffers from significant classification errors across various object categories, and its segmentation masks are often inaccurate, leading to poor scene composition. We believe this issue stems from the higher levels of noise present in the point cloud generated by the video generation model, which differs significantly from the distribution of the

Frames	FPVSCORE
5	2.02
10	2.50
20	2.48

Table 4: **Ablation on Frame Quantity in Perception.**

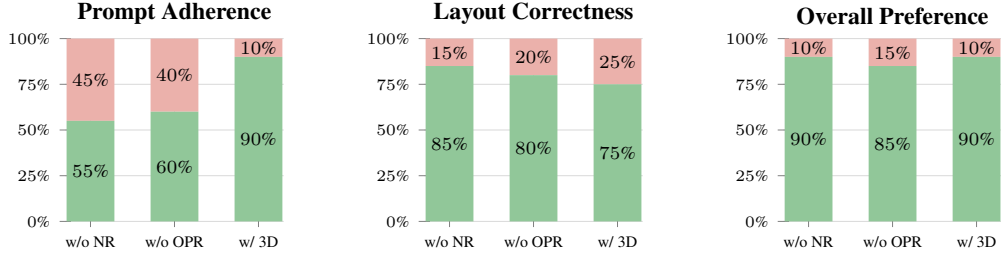


Figure 8: **Ablation Study.** Win ratio of VIPSCENE versus variants measured by prompt adherence (PA), layout correctness (LC), and overall performance (OP). Variants include the original model without noise reduction (NR), without object pose refinement (OPR), and using a 3D instead of a 2D object detector. A 50% win ratio indicates equal performance, while 100% means the full model always outperforms the variant.

training data used for the 3D perception model. In contrast, our approach begins with 2D perception models, which are more robust in accurately identifying object categories in images. This is further enhanced by point cloud denoising techniques, which reduce the noise in the reconstructed point cloud. As a result, our method achieves more reliable and precise scene decompositions, ultimately leading to higher-quality scene synthesis. To assess 2D perception efficiency, we analyze the trade-off between frame count and accuracy. As shown in Tab. 4, enlarging the temporal window improves performance but quickly saturates. In practice, sampling at 2 fps and using ~ 10 frames strikes a favorable balance between cost and accuracy.

Noise Reduction (NR). Due to potential motion blur and other artifacts in the generated video, as well as the 3D reconstruction model itself, the reconstructed point clouds may exhibit significant noise, especially at the border of object masks, i.e., at depth discontinuities. The adaptive erosion method we introduce effectively filters out these artifacts while preserving the integrity of the target objects, facilitating more precise scene decomposition and improving the asset retrieval process. Beyond the full-module ablation, we also compare fixed-kernel variants. The adaptive setting consistently outperforms others (Tab. 5).

Erosion Method	FPVSCORE
Small Kernel	2.30
Large Kernel	2.05
Adaptive	2.67

Table 5: **Ablation on Adaptive Erosion.**

Object Pose Refinement (OPR). As demonstrated in Fig. 8, this refinement step effectively mitigates object collisions that may arise from size mismatches between the retrieved assets and the target objects. Following this step, users observe enhanced scene coherence, where objects are maintained close to their original placements while avoiding collisions with each other. This results in a more realistic and visually pleasing scene. We further conduct additional ablation experiments to better isolate the impact of each component in the object pose refinement loss. The results are summarized in Tab. 6. It demonstrates that each loss term makes a meaningful contribution to enhancing the physical realism and stability of the final scenes.

Loss Variant	Win ratio (%)
w/o position loss $\mathcal{L}_p$	60
w/o overlap loss $\mathcal{L}_o$	85
w/o boundary loss $\mathcal{L}_b$	75

Table 6: **Ablation on Refinement Loss.**

6 Conclusion

In this work, we present VIPSCENE, a novel framework that leverages video perception models for 3D scene synthesis. By integrating video generation, 3D reconstruction, open-vocabulary object detection and tracking, as well as 3D asset retrieval, VIPSCENE bridges the gap between multimodal prompts and coherent, editable 3D scenes. Our method addresses existing challenges in spatial reasoning and multi-view consistency that limit current approaches based on language and image generation models. Furthermore, we introduce a new evaluation metric FPVSCORE that aligns better with human judgment of 3D scenes, offering a more reliable measure of semantic and spatial correctness in generated scenes. Through extensive experiments and user studies, VIPSCENE demonstrates superior performance across diverse scene types, both qualitatively and quantitatively. The obtained results highlight the importance of commonsense knowledge from video data and point toward a promising direction for future research in realistic and interpretable 3D scene generation.

Acknowledgment This work is supported in part by the National Key R&D Program of China under Grant 2024YFB4708200, the National Natural Science Foundation of China under Grants U24B20173 and 42327901, and the Scientific Research Innovation Capability Support Project for Young Faculty under Grant ZYGXQNJSKYCXNLZCXM-I20.

References

- [1] Hassan Abu Alhaija, Siva Karthik Mustikovela, Lars Mescheder, Andreas Geiger, and Carsten Rother. Augmented reality meets computer vision: Efficient data generation for urban driving scenes. *International Journal on Computer Vision (IJCV)*, 2018.
- [2] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.
- [3] Rio Aguina-Kang, Maxim Gumin, Do Heon Han, Stewart Morris, Seung Jean Yoo, Aditya Ganeshan, R Kenny Jones, Qihong Anna Wei, Kailiang Fu, and Daniel Ritchie. Open-universe indoor scene generation using llm program synthesis and uncured object databases. *arXiv preprint arXiv:2403.09675*, 2024.
- [4] Ata Çelen, Guo Han, Konrad Schindler, Luc Van Gool, Iro Armeni, Anton Obukhov, and Xi Wang. I-design: Personalized llm interior designer. In *European Conference on Computer Vision (ECCV)*, 2025.
- [5] Zilong Chen, Yikai Wang, Wenqiang Sun, Feng Wang, Yiwen Chen, and Huaping Liu. Meshgen: Generating pbr textured mesh with render-enhanced auto-encoder and generative data augmentation. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025.
- [6] Matt Deitke, Eli VanderBilt, Alvaro Herrasti, Luca Weihs, Kiana Ehsani, Jordi Salvador, Winson Han, Eric Kolve, Aniruddha Kembhavi, and Roozbeh Mottaghi. Procthor: Large-scale embodied ai using procedural generation. *International Conference on Neural Information Processing Systems (NeurIPS)*, 2022.
- [7] Matt Deitke, Dustin Schwenk, Jordi Salvador, Luca Weihs, Oscar Michel, Eli VanderBilt, Ludwig Schmidt, Kiana Ehsani, Aniruddha Kembhavi, and Ali Farhadi. Objaverse: A universe of annotated 3d objects. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.
- [8] Wei Deng, Mengshi Qi, and Huadong Ma. Global-local tree search in vlms for 3d indoor scene generation. *arXiv preprint arXiv:2503.18476*, 2025.
- [9] Weixi Feng, Wanrong Zhu, Tsu-jui Fu, Varun Jampani, Arjun Akula, Xuehai He, Sugato Basu, Xin Eric Wang, and William Yang Wang. Layoutgpt: Compositional visual planning and generation with large language models. *International Conference on Neural Information Processing Systems (NeurIPS)*, 2023.
- [10] Rao Fu, Zehao Wen, Zichen Liu, and Srinath Sridhar. Anyhome: Open-vocabulary generation of structured and textured 3d homes. In *European Conference on Computer Vision (ECCV)*, 2024.
- [11] Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, et al. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.
- [12] Zebin He, Mingxin Yang, Shuhui Yang, Yixuan Tang, Tao Wang, Kaihao Zhang, Guanying Chen, Yuhong Liu, Jie Jiang, Chunchao Guo, et al. Materialmvp: Illumination-invariant material generation via multi-view pbr diffusion. *arXiv preprint arXiv:2503.10289*, 2025.
- [13] Jack Hessel, Ari Holtzman, Maxwell Forbes, Ronan Le Bras, and Yejin Choi. Clipscore: A reference-free evaluation metric for image captioning. *arXiv preprint arXiv:2104.08718*, 2021.
- [14] Jonathan Ho, Tim Salimans, Alexey Gritsenko, William Chan, Mohammad Norouzi, and David J Fleet. Video diffusion models. *International Conference on Neural Information Processing Systems (NeurIPS)*, 2022.
- [15] Lukas Höllein, Ang Cao, Andrew Owens, Justin Johnson, and Matthias Nießner. Text2room: Extracting textured 3d meshes from 2d text-to-image models. In *International Conference on Computer Vision (ICCV)*, 2023.
- [16] Xin Huang, Tengfei Wang, Ziwei Liu, and Qing Wang. Material anything: Generating materials for any 3d object via diffusion. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025.
- [17] Zehuan Huang, Yuan-Chen Guo, Xingqiao An, Yunhan Yang, Yangguang Li, Zi-Xin Zou, Ding Liang, Xihui Liu, Yan-Pei Cao, and Lu Sheng. Midi: Multi-instance diffusion for single image to 3d scene generation. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025.
- [18] Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, et al. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*, 2024.

- [19] Bingyi Kang, Yang Yue, Rui Lu, Zhijie Lin, Yang Zhao, Kaixin Wang, Gao Huang, and Jiashi Feng. How far is video generation from world model: A physical law perspective. *arXiv preprint arXiv:2411.02385*, 2024.
- [20] Maurice G Kendall. A new measure of rank correlation. *Biometrika*, 1938.
- [21] Kuaishou. Kling, 2025. <https://klingai.com/global/>.
- [22] Vikram Kumaran, Jonathan Rowe, Bradford Mott, and James Lester. Scenecraft: automating interactive narrative scene generation in digital games with large language models. In *Association for the Advancement of Artificial Intelligence (AAAI)*, 2023.
- [23] Vincent Leroy, Yohann Cabon, and Jérôme Revaud. Grounding image matching in 3d with mast3r. In *European Conference on Computer Vision (ECCV)*, 2024.
- [24] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *International Conference on Machine Learning (ICML)*, 2023.
- [25] Bin Lin, Yunyang Ge, Xinhua Cheng, Zongjian Li, Bin Zhu, Shaodong Wang, Xianyi He, Yang Ye, Shenghai Yuan, Liuhan Chen, et al. Open-sora plan: Open-source large video generation model. *arXiv preprint arXiv:2412.00131*, 2024.
- [26] Chenguo Lin and MU Yadong. Instructscene: Instruction-driven 3d indoor scene synthesis with semantic graph prior. In *International Conference on Learning Representations (ICLR)*, 2024.
- [27] Yiqi Lin, Hao Wu, Ruichen Wang, Haonan Lu, Xiaodong Lin, Hui Xiong, and Lin Wang. Towards language-guided interactive 3d generation: LLMs as layout interpreter with generative feedback. *arXiv preprint arXiv:2305.15808*, 2023.
- [28] Zhiqiu Lin, Deepak Pathak, Baiqi Li, Jiayao Li, Xide Xia, Graham Neubig, Pengchuan Zhang, and Deva Ramanan. Evaluating text-to-visual generation with image-to-text generation. In *European Conference on Computer Vision (ECCV)*, 2024.
- [29] Lu Ling, Chen-Hsuan Lin, Tsung-Yi Lin, Yifan Ding, Yu Zeng, Yichen Sheng, Yunhao Ge, Ming-Yu Liu, Aniket Bera, and Zhaoshuo Li. Scenethesis: A language and vision agentic framework for 3d scene generation. *arXiv preprint arXiv:2505.02836*, 2025.
- [30] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *International Conference on Neural Information Processing Systems (NeurIPS)*, 2023.
- [31] Luma. Dream machine, 2024. <https://lumalabs.ai/dream-machine>.
- [32] Ajay Mandlekar, Soroush Nasiriany, Bowen Wen, Ireteyo Akinola, Yashraj Narang, Linxi Fan, Yuke Zhu, and Dieter Fox. Mimicgen: A data generation system for scalable robot learning using human demonstrations. In *Conference on Robot Learning (CoRL)*, 2023.
- [33] Nvidia. Cosmos, 2024. <https://www.nvidia.com/en-us/ai/cosmos/>.
- [34] Linfei Pan, Dániel Baráth, Marc Pollefeys, and Johannes L Schönberger. Global structure-from-motion revisited. In *European Conference on Computer Vision (ECCV)*, 2024.
- [35] William Peebles and Saining Xie. Scalable diffusion models with transformers. In *International Conference on Computer Vision (ICCV)*, 2023.
- [36] Luigi Piccinelli, Yung-Hsu Yang, Christos Sakaridis, Mattia Segu, Siyuan Li, Luc Van Gool, and Fisher Yu. Unidepth: Universal monocular metric depth estimation. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.
- [37] Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. Sdxl: Improving latent diffusion models for high-resolution image synthesis. *arXiv preprint arXiv:2307.01952*, 2023.
- [38] Yifan Pu, Yiming Zhao, Zhicong Tang, Ruihong Yin, Haoxing Ye, Yuhui Yuan, Dong Chen, Jianmin Bao, Sirui Zhang, Yanbin Wang, et al. ART: Anonymous region transformer for variable multi-layer transparent image generation. In *CVPR*, 2025.
- [39] Hou In Derek Pun, Hou In Ivan Tam, Austin T Wang, Xiaoliang Huo, Angel X Chang, and Manolis Savva. Hsm: Hierarchical scene motifs for multi-scale indoor scene generation. *arXiv preprint arXiv:2503.16848*, 2025.

- [40] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning (ICML)*, 2021.
- [41] Tianhe Ren, Shilong Liu, Ailing Zeng, Jing Lin, Kunchang Li, He Cao, Jiayu Chen, Xinyu Huang, Yukang Chen, Feng Yan, et al. Grounded sam: Assembling open-world models for diverse visual tasks. *arXiv preprint arXiv:2401.14159*, 2024.
- [42] Runway. Gen 3, 2024. <https://runwayml.com/research/introducing-gen-3-alpha>.
- [43] Masaki Saito, Eiichi Matsumoto, and Shunta Saito. Temporal generative adversarial nets with singular value clipping. In *International Conference on Computer Vision (ICCV)*, 2017.
- [44] Johannes Lutz Schönberger and Jan-Michael Frahm. Structure-from-motion revisited. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.
- [45] Johannes Lutz Schönberger, Enliang Zheng, Marc Pollefeys, and Jan-Michael Frahm. Pixelwise view selection for unstructured multi-view stereo. In *European Conference on Computer Vision (ECCV)*, 2016.
- [46] Jonas Schult, Francis Engelmann, Alexander Hermans, Or Litany, Siyu Tang, and Bastian Leibe. Mask3d: Mask transformer for 3d semantic instance segmentation. In *International Conference on Robotics and Automation (ICRA)*, 2023.
- [47] Jonas Schult, Sam Tsai, Lukas Höllein, Bichen Wu, Jialiang Wang, Chih-Yao Ma, Kunpeng Li, Xiaofang Wang, Felix Wimbauer, Zijian He, et al. Controlroom3d: Room generation using semantic proxy rooms. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.
- [48] Fan-Yun Sun, Weiyu Liu, Siyi Gu, Dylan Lim, Goutam Bhat, Federico Tombari, Manling Li, Nick Haber, and Jiajun Wu. Layoutvlm: Differentiable optimization of 3d layout via vision-language models. *arXiv preprint arXiv:2412.02193*, 2024.
- [49] Jiaming Sun, Yiming Xie, Linghao Chen, Xiaowei Zhou, and Hujun Bao. Neuralrecon: Real-time coherent 3d reconstruction from monocular video. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.
- [50] Tomu Tahara, Takashi Seno, Gaku Narita, and Tomoya Ishikawa. Retargetable ar: Context-aware augmented reality in indoor scenes based on 3d scene graph. In *2020 IEEE International Symposium on Mixed and Augmented Reality Adjunct (ISMAR-Adjunct)*, 2020.
- [51] Jiapeng Tang, Yinyu Nie, Lev Markhasin, Angela Dai, Justus Thies, and Matthias Nießner. Diffuscene: Denoising diffusion models for generative indoor scene synthesis. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.
- [52] Zhenggang Tang, Yuchen Fan, Dilin Wang, Hongyu Xu, Rakesh Ranjan, Alexander Schwing, and Zhicheng Yan. Mv-dust3r+: Single-stage scene reconstruction from sparse views in 2 seconds. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025.
- [53] Gemini Team, Rohan Anil, Sebastian Borgeaud, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, Katie Millican, et al. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*, 2023.
- [54] Gemini Team, Petko Georgiev, Ving Ian Lei, Ryan Burnell, Libin Bai, Anmol Gulati, Garrett Tanzer, Damien Vincent, Zhufeng Pan, Shibo Wang, et al. Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context. *arXiv preprint arXiv:2403.05530*, 2024.
- [55] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023.
- [56] Sergey Tulyakov, Ming-Yu Liu, Xiaodong Yang, and Jan Kautz. Mocogan: Decomposing motion and content for video generation. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.
- [57] Hengyi Wang and Lourdes Agapito. 3d reconstruction with spatial memory. *arXiv preprint arXiv:2408.16061*, 2024.
- [58] Jianyuan Wang, Minghao Chen, Nikita Karaev, Andrea Vedaldi, Christian Rupprecht, and David Novotny. Vggt: Visual geometry grounded transformer. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025.

- [59] Shuzhe Wang, Vincent Leroy, Yohann Cabon, Boris Chidlovskii, and Jerome Revaud. Dust3r: Geometric 3d vision made easy. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 20697–20709, 2024.
- [60] Yian Wang, Xiaowen Qiu, Jiageng Liu, Zhehuan Chen, Jiting Cai, Yufei Wang, Tsun-Hsuan Johnson Wang, Zhou Xian, and Chuang Gan. Architect: Generating vivid and interactive 3d scenes with hierarchical 2d inpainting. *International Conference on Neural Information Processing Systems (NeurIPS)*, 2025.
- [61] Tong Wu, Guandao Yang, Zhibing Li, Kai Zhang, Ziwei Liu, Leonidas Guibas, Dahua Lin, and Gordon Wetzstein. Gpt-4v (ision) is a human-aligned evaluator for text-to-3d generation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 22227–22238, 2024.
- [62] Tianhao Wu, Chuanxia Zheng, Frank Guan, Andrea Vedaldi, and Tat-Jen Cham. Amodal3r: Amodal 3d reconstruction from occluded 2d images. *arXiv preprint arXiv:2503.13439*, 2025.
- [63] Jianfeng Xiang, Zelong Lv, Sicheng Xu, Yu Deng, Ruicheng Wang, Bowen Zhang, Dong Chen, Xin Tong, and Jiaolong Yang. Structured 3d latents for scalable and versatile 3d generation. *arXiv preprint arXiv:2412.01506*, 2024.
- [64] Jianing Yang, Alexander Sax, Kevin J Liang, Mikael Henaff, Hao Tang, Ang Cao, Joyce Chai, Franziska Meier, and Matt Feiszli. Fast3r: Towards 3d reconstruction of 1000+ images in one forward pass. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025.
- [65] Ruihan Yang, Prakhar Srivastava, and Stephan Mandt. Diffusion probabilistic modeling for video generation. *Entropy*, 2023.
- [66] Xianghui Yang, Huiwen Shi, Bowen Zhang, Fan Yang, Jiacheng Wang, Hongxu Zhao, Xinhai Liu, Xinzhou Wang, Qingxiang Lin, Jiaao Yu, et al. Hunyuan3d 1.0: A unified framework for text-to-3d and image-to-3d generation. *arXiv preprint arXiv:2411.02293*, 2024.
- [67] Yandan Yang, Baoxiong Jia, Peiyuan Zhi, and Siyuan Huang. Physcene: Physically interactable 3d scene synthesis for embodied ai. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.
- [68] Yixuan Yang, Junru Lu, Zixiang Zhao, Zhen Luo, James JQ Yu, Victor Sanchez, and Feng Zheng. Lplace: The 3d indoor scene layout generation and editing via large language model. *arXiv preprint arXiv:2406.03866*, 2024.
- [69] Yue Yang, Fan-Yun Sun, Luca Weihs, Eli VanderBilt, Alvaro Herrasti, Winson Han, Jiajun Wu, Nick Haber, Ranjay Krishna, Lingjie Liu, et al. Holodeck: Language guided generation of 3d embodied ai environments. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.
- [70] Zhifei Yang, Keyang Lu, Chao Zhang, Jiaying Qi, Hanqi Jiang, Ruifei Ma, Shenglin Yin, Yifan Xu, Mingzhe Xing, Zhen Xiao, Jieyi Long, Xiangde Liu, and Guangyao Zhai. Mmgdreamer: Mixed-modality graph for geometry-controllable 3d indoor scene generation. In *Association for the Advancement of Artificial Intelligence (AAAI)*, 2025.
- [71] Guangyao Zhai, Evin Pinar Örnek, Shun-Cheng Wu, Yan Di, Federico Tombari, Nassir Navab, and Benjamin Busam. Commonsences: Generating commonsense 3d indoor scenes with scene graphs. In *International Conference on Neural Information Processing Systems (NeurIPS)*, 2023.
- [72] Guangyao Zhai, Xiaoni Cai, Dianye Huang, Yan Di, Fabian Manhardt, Federico Tombari, Nassir Navab, and Benjamin Busam. Sg-bot: Object rearrangement via coarse-to-fine robotic imagination on scene graphs. In *International Conference on Robotics and Automation (ICRA)*, 2024.
- [73] Guangyao Zhai, Evin Pinar Örnek, Dave Zhenyu Chen, Ruotong Liao, Yan Di, Nassir Navab, Federico Tombari, and Benjamin Busam. Echoscene: Indoor scene generation via information echo over scene graph diffusion. In *European Conference on Computer Vision (ECCV)*, 2024.
- [74] Juyong Zhang, Yuxin Yao, and Bailin Deng. Fast and robust iterative closest point. *Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 2021.
- [75] Yuqing Zhang, Yuan Liu, Zhiyu Xie, Lei Yang, Zhongyuan Liu, Mengzhou Yang, Runze Zhang, Qilong Kou, Cheng Lin, Wenping Wang, et al. Dreammat: High-quality pbr material generation with geometry-and light-aware diffusion models. *ACM Transactions on Graphics (TOG)*, 2024.
- [76] Zibo Zhao, Zeqiang Lai, Qingxiang Lin, Yunfei Zhao, Haolin Liu, Shuhui Yang, Yifei Feng, Mingxin Yang, Sheng Zhang, Xianghui Yang, et al. Hunyuan3d 2.0: Scaling diffusion models for high resolution textured 3d assets generation. *arXiv preprint arXiv:2501.12202*, 2025.

Appendix

A Detailed Algorithm

This section provides a comprehensive breakdown of the proposed algorithm, outlining its modules, their interactions, and the data representations used throughout the process.

Algorithm 1: VIPSCENE: Prompt-to-Scene Generation

Input: Prompt π (text / image), video generator $\mathcal{G}$, reconstructor $\mathcal{R}$ (Fast3R), monocular depth $\mathcal{D}$ (UniDepth), detector $\mathcal{S}$ (Grounded-SAM), tracker $\mathcal{T}$ (MASt3R), assets $\mathcal{A}$, weights λ_o, λ_b

Output: Final metrically scaled collision-free scene $S = \{o_i\}_{i=1}^N$, where each object $o_i = (c_i, s_i, l_i, \theta_i)$ is represented by its category $c_i \in \mathcal{C}$, size $s_i \in \mathbb{R}^3$, position $l_i \in \mathbb{R}^3$, and orientation $\theta_i \in \mathbb{R}$ around the gravity axis.

```

1 (A) Prompt  $\rightarrow$  Video frames
2  $V \leftarrow \mathcal{G}(\pi)$ 
3  $\{I_t\}_{t=1}^T \leftarrow$  Sample frames from  $V$  with fps = 2
4 (B) Video  $\rightarrow$  Metric 3D reconstruction
5  $R \leftarrow \mathcal{R}(\{I_t\}_{t=1}^T)$  // Globally consistent 3D (unposed inputs)
6  $\{D_t\}_{t=1}^T \leftarrow \mathcal{D}(\{I_t\}_{t=1}^T)$  // Metric depths
7  $R \leftarrow$  Rescale  $R$  with metric  $\{D_t\}_{t=1}^T$  // Enforce metric scale
8 (C) Scene decomposition & object extraction
9 for  $t = 1$  to  $T$  do
10    $\{(c, M_t^{(k)})\}_k \leftarrow \mathcal{S}(I_t)$  // Per-frame 2D instance masks with
   categories
11    $\{M_t^{(k)}\}_k \leftarrow$  AdaptiveErode( $\{M_t^{(k)}\}_k$ ) // Size-aware morphological
   denoising
12    $\left\{ \left\{ M_t^i \right\}_{t=1}^T \right\}_{i=1}^N \leftarrow \mathcal{T}(\{I_t, \{M_t^{(k)}\}_k\}_{t=1}^T)$  // Temporal association / IDs
13   for  $i = 1$  to  $N$  do
14      $P_i \leftarrow$  Segment points from  $R$  by  $\{M_t^i\}_{t=1}^T$  // Per-object point cloud
15      $c_i \leftarrow$  Majority label( $\{M_t^i\}_{t=1}^T$ ) // Object category from detections
16 (D) 3D asset retrieval & alignment
17 for  $i = 1$  to  $N$  do
18    $(s_i, l_i^{\text{init}}, \theta_i^{\text{init}}) \leftarrow \text{PCAInit}(P_i)$ 
19    $\mathcal{C}_i \leftarrow$  Retrieve candidates from  $\mathcal{A}$  according to  $c_i$  best  $\leftarrow \emptyset$ ,  $\text{rmse}_{\min} \leftarrow +\infty$ 
20   foreach  $Q \in \mathcal{C}_i$  do
21     foreach  $\theta \in \{\theta_i^{\text{init}}, \theta_i^{\text{init}} + \pi\}$  do
22        $(\mathbf{R}^*, \mathbf{t}^*), \text{rmse} \leftarrow \text{ICPAlign}(P_i, Q; l_i^{\text{init}}, \theta)$  // Eq. (1),  $\mathbf{R} \in \text{SO}(3)$ 
23       if  $\text{rmse} < \text{rmse}_{\min}$  then
24          $\text{rmse}_{\min} \leftarrow \text{rmse}$ 
25         best  $\leftarrow (Q, \mathbf{R}^*, \mathbf{t}^*, \theta)$ 
26    $(Q_i, \mathbf{R}_i, \mathbf{t}_i, \theta_i) \leftarrow \text{best}, l_i \leftarrow \mathbf{t}_i$ 
27 (E) Final scene refinement
28  $l_i^{\text{orig}} \leftarrow l_i \quad \forall i$  //  $l_i$  denotes position variables
29 repeat // Gradient-based optimization
30    $\mathcal{L}_p = \sum_{i=1}^N \|l_i - l_i^{\text{orig}}\|_2^2$ 
31    $\mathcal{L}_o = \sum_{i \neq j} \text{Area}(\text{BBox}_i(l_i, s_i) \cap \text{BBox}_j(l_j, s_j))$ 
32    $\mathcal{L}_b = \sum_{i=1}^N \text{Area}(\text{BBox}_i(l_i, s_i) \setminus \text{Room})$ 
33    $\mathcal{L}_{\text{total}} \leftarrow \mathcal{L}_p + \lambda_o \mathcal{L}_o + \lambda_b \mathcal{L}_b$ 
34    $\{l_i\} \leftarrow \text{Update}(\{l_i\}, -\eta \nabla_{\{l_i\}} \mathcal{L}_{\text{total}})$  if  $\text{Overlap}(\{\text{BBox}_i\}) = 0$  and  $\Delta \mathcal{L}_{\text{total}} < \varepsilon$  then
35   break
36 until converged
37 return  $S = \{(c_i, s_i, l_i, \theta_i)\}_{i=1}^N$ 

```

B Computational Complexity

We have profiled both the inference latency and peak GPU memory consumption of each major stage in our pipeline in Tab. 7. Among all stages, video generation is the most computationally expensive, requiring around 380s per video and 74GB GPU memory on an H100. 3D reconstruction, object detection and tracking, as well as asset retrieval, are significantly more efficient, each taking only a few seconds per scene with much lower memory usage. The total latency for generating a complete scene is 400s, compared to 200s in Holodeck. Although our pipeline is more computationally intensive in the generation stage, it yields significantly higher scene quality, which we believe justifies the trade-off.

Table 7: Runtime and Memory Footprint per Stage.

Stage	Time Cost	Memory Usage
Video Generation	~380 s	~74 GB
3D Reconstruction	~2 s	~8 GB
Object Detection & Tracking	~15 s	~8 GB
3D Asset Retrieval	~5 s	—
Pose Refinement	~1 s	—

C Additional Qualitative Results

We provide additional qualitative results, including experiments on scene synthesis from multimodal and text-only inputs, as shown in Fig. 9 and Fig. 10. Specifically, our video generation model utilizes both image and text inputs as complementary modalities for 3D scene generation. The input image defines the original field of view, while the accompanying text guides the model to plausibly infer and complete scene elements beyond the visible region. This multimodal setting enhances the flexibility and robustness of the generation process, resulting in more accurate and diverse 3D scenes.

D Adaptive Erosion Method

We apply an adaptive erosion strategy to remove noise and retain high-confidence points in the reconstructed point clouds. As shown in Fig. 11, this approach effectively reduces artifacts in the initial per-object point clouds, resulting in cleaner and more coherent geometries. These improvements further benefit the subsequent asset retrieval process and enhance the overall scene quality.



Figure 11: **Before-and-After Visualization of the Adaptive Erosion Strategy.** The method effectively removes noisy points and preserves high-confidence regions, producing cleaner and more consistent geometries.

E Failure Cases

We show representative failure cases in Fig. 12. Typical errors include incorrect category recognition by the detector and missing detections caused by occlusion. When objects are dense, the layout optimization is not realistic enough. In the left example, a misclassified fireplace results in duplicated instances, whereas in the right example, the placement of the wooden table is not reasonable enough.

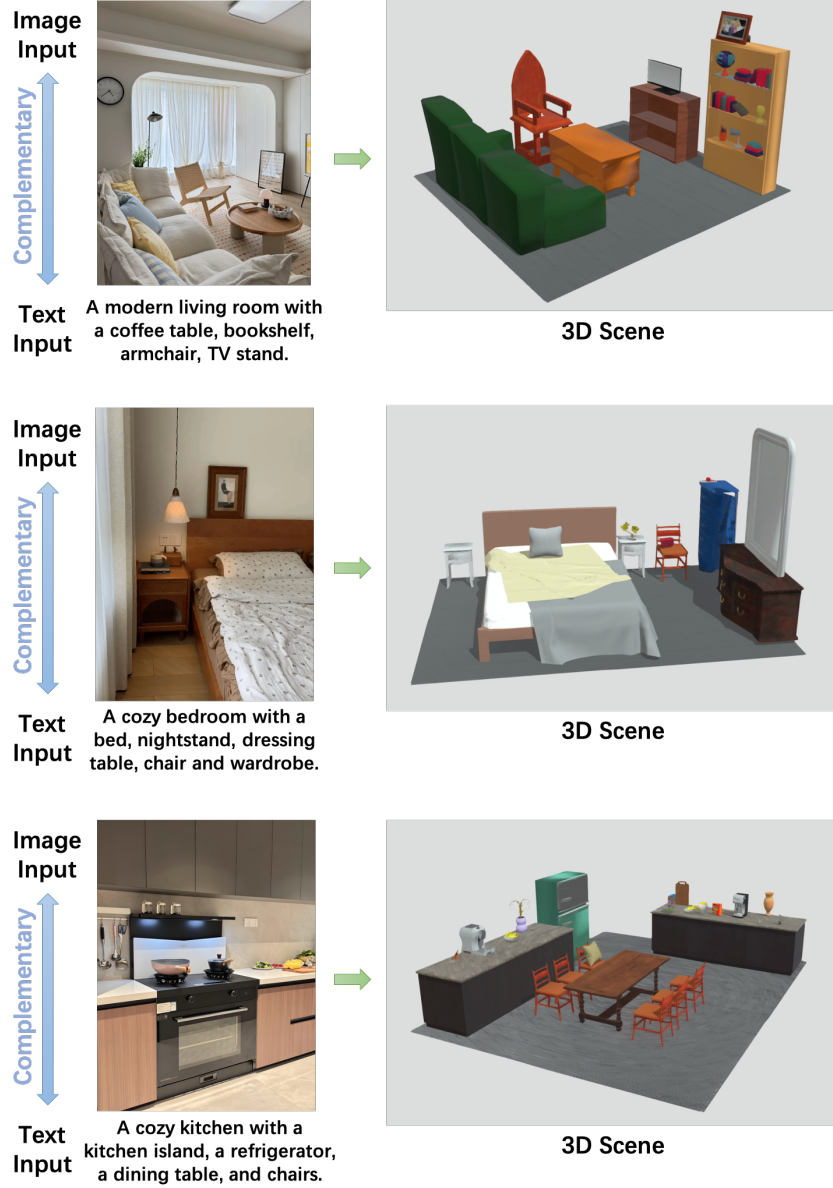


Figure 9: **Qualitative Results from Multimodal Inputs.** The generated video respects the original field of view provided by the input image while leveraging the accompanying text to plausibly infer and complete scene elements beyond the visible area.

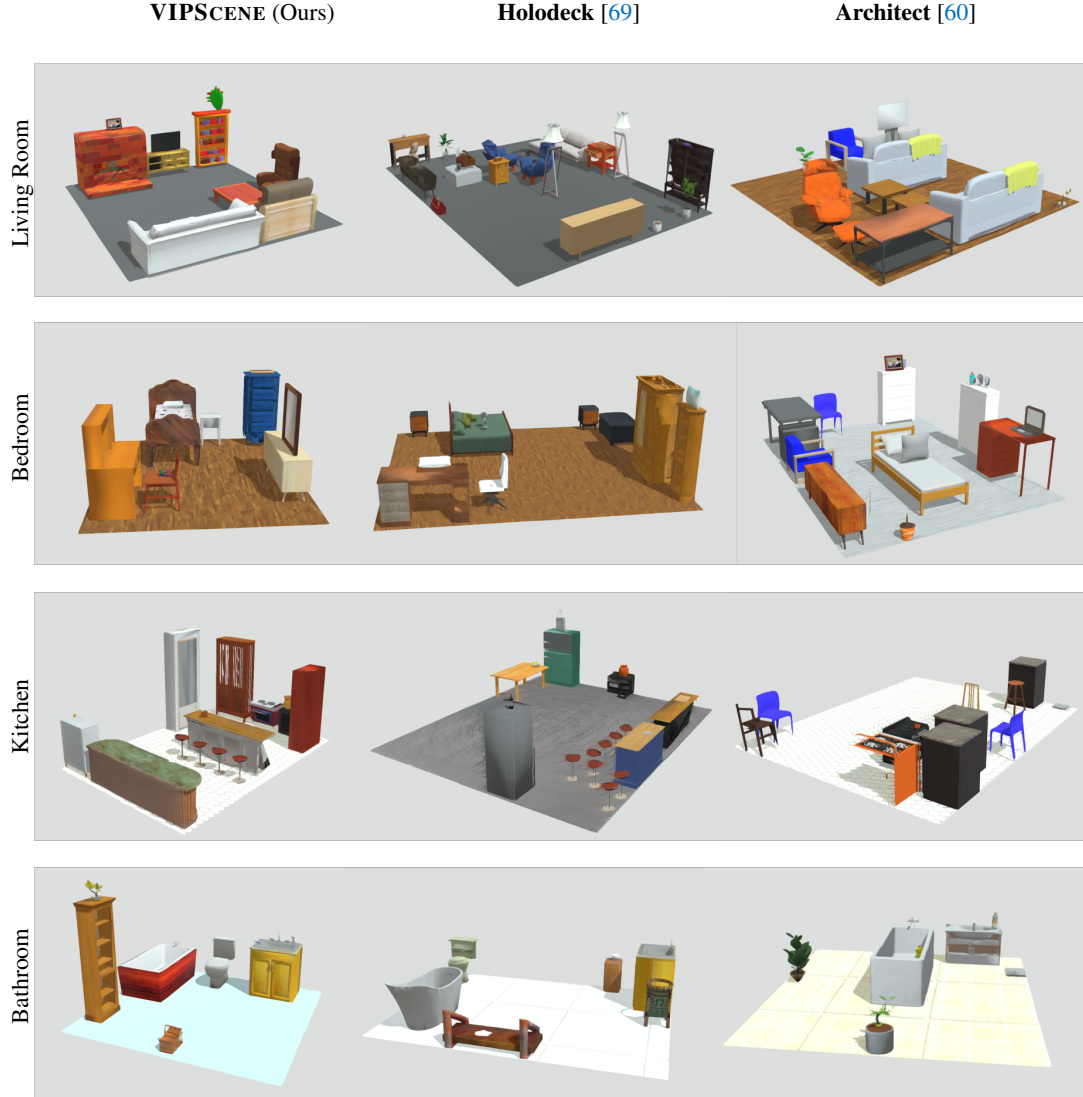


Figure 10: **Additional Qualitative Results.** We present results of VIPSCENE, comparing it against Holodeck [69] and Architect [60] (*columns*) across four room types (*rows*). For better visibility, ceilings and walls are removed.

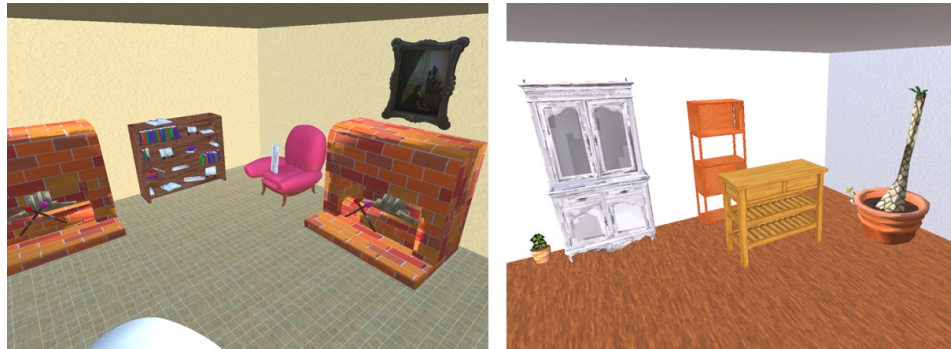


Figure 12: **Failure Cases.** *Left:* Misclassification by the detector results in duplicated fireplaces. *Right:* Inaccurate placement of a wooden table due to layout optimization issues in dense scenes.

F Implementation Details

Rescaling the Scene. To refine the overall scale of the reconstructed scene, we estimate depth maps for each view using UniDepth [36], and compare them against the corresponding reconstructed point clouds. For each point, we compute the ratio between the estimated depth and the depth derived from the reconstructed geometry. The global scale factor is then determined as the median of these per-point ratios across all views, providing a robust estimate that mitigates the influence of outliers. This median-based scaling approach ensures consistency across views and improves the alignment of the reconstructed scene with real-world metric dimensions.

Orientation Estimation of the Object. Without loss of generality, we assume that object bounding boxes are aligned with the ground plane. To estimate their orientation, we first determine the ground plane equation. This process begins by extracting the ground point cloud, using a method analogous to object extraction. Specifically, we prompt Grounded-SAM [41] with the label “ground” for outdoor scenes or “floor” for indoor scenes to generate ground masks. These masks are then used to extract the corresponding ground points from the reconstructed scene, and a least-squares fitting is applied to estimate the ground plane. With the ground plane established, each object’s point cloud P_i is transformed into a new coordinate system that retains the origin of the original camera coordinate system C , but aligns its horizontal plane with the estimated ground plane. The transformed point cloud is then projected onto the ground plane, and Principal Component Analysis (PCA) is applied to identify the principal axes of the point distribution. The direction of greatest variance is taken as an approximation of the object’s orientation θ_i . A tight bounding box is subsequently aligned with this estimated direction.

G Prompts Details

Prompts for Scene Synthesis. We utilize GPT-4o [18] to generate text prompts for four types of indoor scenes: living room, bedroom, kitchen, and bathroom. Each prompt specifies the room type along with the objects intended to furnish the space. For example, “A bedroom with a large bed, two nightstands, a floor lamp, a wardrobe, and a big window.”

I’m working on an interior design project and would like to generate video scenes of a {room type} using a text-to-video model. Please help me create detailed prompts to feed into the model.

Guidelines:

1. Based on the typical function and layout of a {room type}, list the furniture, appliances, decorations, and other items commonly found in the space. 2. Prompts should describe the room’s contents clearly and in detail.

Example: “A bedroom with a large bed, two nightstands, a floor lamp, a wardrobe, and a big window.”

Prompts for FPVSCORE. To facilitate consistent and goal-driven evaluations in FPVSCORE, we design structured prompts that include: (1) task-specific instructions for multi-scene comparison, (2) clearly defined evaluation criteria, and (3) standardized formatting requirements. These prompts guide the model to assess each scene in terms of semantic fidelity, spatial layout accuracy, and overall coherence, while also requiring concise justifications to support its ratings and enhance transparency.

Task: Compare the room layout rationality of three methods, all generated from the same text description. From top to bottom, the video sequences display a 360-degree view of each method’s generated scene. Decide which method performs best according to the criteria below.

Text Description: {text_description}

Instructions:

1. Semantic Correctness

Does the generated layout accurately reflect the text description?

Check whether all described objects are present and correctly represented.

2. Layout Correctness

Is the room design physically plausible and functional?

Evaluate if the layout supports practical use, space efficiency, and proper object functionality. Consider object positions, orientations, and user convenience.

3. Overall Preference

Does the room layout look realistic and natural?

Consider the visual coherence and harmony of the scene.

Evaluation process:

Carefully examine the multi-view images of all three 3D scenes. Focus on one criterion at a time and make independent judgments for each.

Output format:

Provide a clear, concise analysis for each criterion. Avoid vague terms like “realistic” or “spacious.” Instead, specify exact issues or strengths. For example:

- For Semantic Correctness, indicate which objects are missing or inaccurately depicted.
- For Layout Correctness, specify which objects are misplaced or poorly oriented, and explain how this impacts usability or functionality.

After the analyses, assign ranks (1–3) to each method per criterion (1 = best, 3 = worst).

Summarize your final ranking in the format: <rank for criterion 1> <rank for criterion 2> <rank for criterion 3>

for each method.

Example:

Analysis:

1. Semantic Correctness: The first one ...; The second one ...; The third one ...
2. Layout Correctness: The first one ...; The second one ...; The third one ...
3. Overall Preference: The first one ...; The second one ...; The third one ...

Final answer:

The first one: x x x

The second one: x x x

The third one: x x x

(where x denotes ranks 1–3)

(Please strictly follow the format above. Do not include extra symbols like **, quotation marks, or bullet points.)

Prompts for Top-Down View Scores. Following the approach of Architect [60], we design targeted prompts to guide GPT-4o in evaluating room layouts based solely on top-down views. To ensure a fair comparison, the prompts also emphasize spatial structure, semantic fidelity, and functional usability, consistent with our own.

Task: Compare the room layout rationality of three methods, all generated from the same text description. The top-down views of the scenes produced by the three methods are presented from left to right. Identify which method performs best based on the criteria below.

Text Description: {text_description}

Instructions:

1. Semantic Correctness

Does the generated layout accurately reflect the text description?

Check whether all described objects are present and correctly represented.

2. Layout Correctness

Is the room design physically plausible and functional?

Evaluate if the layout supports practical use, space efficiency, and proper object functionality.

Consider object positions, orientations, and user convenience.

3. Overall Preference

Does the room layout look realistic and natural?

Consider the visual coherence and harmony of the scene.

Provide only your final ranking of the three methods in the format below:

Final answer:

x x x

(where x denotes ranks from 1 to 3)

H User Study Details

We conducted a thorough user study to evaluate the quality of the generated scenes, involving thirty participants. All participants took part voluntarily and received no compensation. At the start of the study, participants were given five minutes to read through the instructions, as illustrated in Fig. 13. An example evaluation page presented to the participants is shown in Fig. 14.

Text-to-Room Layout Algorithm Evaluation

In this study, we invite you to evaluate three different text-to-room layout algorithms. You will see a series of room layout images and videos generated by different algorithms and compare them based on the following three metrics:

Prompt Adherence (PA)

"To what extent does the generated scene align with the input prompt?"
Check if the generated content represents the room description.

Layout Correctness (LC)

"Do the object placements make sense both physically and functionally?"
Evaluate if the layout meets practical requirements, optimizes space usage, and ensures objects can be used as intended.

Overall Preference (OP)

"Does this solution resemble a real scene?"
Rate your overall satisfaction with the layout design. Does the room layout appear realistic and natural?

Evaluation Process

Click the "Start Evaluation" button to enter the main evaluation page.
Please read the background description and the text to be depicted at the top of each page. Then, based on your assessment of the three images and corresponding videos below, rate the different metrics. The higher the score, the more it meets the requirements.
Use the "Previous" and "Next" buttons to continue evaluating the next set of samples.
On the summary page, click the "Download Results" button to download a JSON file and send it to us.

Key Tips

We strongly recommend using a computer as the terminal, as you can zoom the page for the best experience.
This webpage supports resuming the evaluation from where you left off. If you accidentally close the page, don't worry, just reopen it in the same browser to continue.
The order of scenes generated by different methods has been randomized.
Thank you very much for your support of this experiment!

Figure 13: **User Study Instructions.** This page was shown to participants at the beginning of the study to explain the task, interface, and evaluation criteria.

I Limitations

VIPSCENE currently generates spatially coherent scene layouts, retrieving furniture from Objaverse [7]. Although objects are richly annotated, some textures lack photorealistic quality. Future work will improve object quality by adopting advanced 3D generation techniques like text-to-3D and image-to-3D methods [62, 63, 66, 76], and by incorporating state-of-the-art physically-based rendering (PBR) techniques [5, 12, 16, 75] for realistic material representations and lighting. These improvements aim to enhance both the diversity and realism of generated scenes.

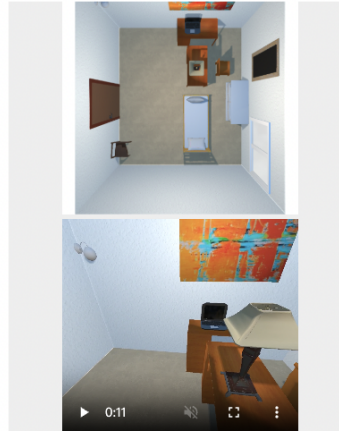
A bedroom with a bed, wardrobe, chair, dressing table, and armchair.



Method 1



Method 2



Method 3

Prompt Adherence: To what extent does the generated scene align with the input prompt?

Method 1

1
2
3

Method 2

1
2
3

Method 3

1
2
3

Layout Correctness: Is the room design both physically plausible and functional?

Method 1

1
2
3

Method 2

1
2
3

Method 3

1
2
3

Overall Preference: Does the room layout appear realistic and natural?

Method 1

1
2
3

Method 2

1
2
3

Method 3

1
2
3

Previous
Home
Next

Figure 14: **Example Page.** Participants were shown a 360-degree video captured from the center of each scene, along with a top-down rendered image. This setup allowed them to evaluate both the global structure and fine details. Each scene was rated on a 3-point scale (1 = lowest, 3 = highest) across three criteria.

22

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: We have included our claims and contributions in the introduction/abstract.

Guidelines:

- The answer NA 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 NA 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: We have discussed the limitations throughout the paper when necessary.

Guidelines:

- The answer NA 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: [\[NA\]](#)

Justification: Our paper does not include theoretical results.

Guidelines:

- The answer NA 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: We have disclosed the implementation information necessary for reproducibility in the main paper and the appendix as thoroughly as possible.

Guidelines:

- The answer NA 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: [Yes]

Justification: Code will be released.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (<https://nips.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://nips.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, hyper-parameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

Answer: [Yes]

Justification: We have provided experimental details in the experiments section of our paper. More details can be found in the appendix.

Guidelines:

- The answer NA 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 available resources limit our ability to achieve statistical significance in the experiment.

Guidelines:

- The answer NA 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: We have indicated the compute resources required for this paper.

Guidelines:

- The answer NA 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: We understand and conform to the NeurIPS Code of Ethics.

Guidelines:

- The answer NA 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: We discussed the detailed broader impact in the appendix.

Guidelines:

- The answer NA means that there is no societal impact of the work performed.
- If the authors answer NA 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., pretrained language models, image generators, or scraped datasets)?

Answer: [NA]

Justification: We believe no such risk is involved.

Guidelines:

- The answer NA 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: We have credited and referenced the papers of the codes and models used in producing this work. We have read and followed the licenses of the works.

Guidelines:

- The answer NA 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: [\[Yes\]](#)

Justification: We will release well-documented assets.

Guidelines:

- The answer NA 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: [\[Yes\]](#)

Justification: We conducted a user study to evaluate our method. The full text of instructions given to participants is included in the appendix. All participants were volunteers, and no monetary or other compensation was provided.

Guidelines:

- The answer NA 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: [\[Yes\]](#)

Justification: We adhere to the NeurIPS Code of Ethics and the guidelines. No IRB approvals are needed.

Guidelines:

- The answer NA 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 rigorousness, or originality of the research, declaration is not required.

Answer: [NA]

Justification: We do not involve LLM in the core of the research.

Guidelines:

- The answer NA 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 (<https://neurips.cc/Conferences/2025/LLM>) for what should or should not be described.