Spatially Feasible 3D Indoor Scene Generation via Interaction-Oriented Human Proxy

Xinyuan Chen[1] **Mingwen Shao**[2,1*] **Qiao Zhang**[1] **Qinglin Zhan**[1]
Xiang Lv[1] **Boyuan Yang**[2,3] **Jixuan Shi**[2,3] **Lingzhuang Meng**[4]
Chang Liu[5] **Shanshan Song**[1] **Ling Jian**[1]

[1]Shandong Key Laboratory of Intelligent Oil & Gas Industrial Software,
Qingdao Institute of Software, College of Computer Science and Technology,
China University of Petroleum (East China), Qingdao, China

[2]Artificial Intelligence Research Institute,
Shenzhen University of Advanced Technology, Shenzhen, China

[3]Southern University of Science and Technology, Shenzhen, China

[4]College of Information Science and Technology,
Qingdao University of Science and Technology, Qingdao, China

[5]College of Computer Science, Fudan University, Shanghai, China

Abstract

Recent 3D indoor scene generation methods often produce layouts with limited spatial feasibility, where pathways are blocked and objects are difficult to access or use, restricting their applicability in virtual reality and embodied environments. To address this challenge, this paper presents a new paradigm for 3D indoor scene generation that explicitly models interaction feasibility through a carefully designed geometric abstraction, termed the **Interaction-Oriented Human Proxy (IOHP)**. The IOHP serves as a compact yet spatially expressive representation for human interaction occupancy, and can be directly inferred by large language models without task-specific training. Building on IOHP, we formulate layout optimization as a unified objective that resolves geometric conflicts between the proxy and scene elements, while using the proxy as a bridge to preserve functional relationships among furniture, thereby improving overall usability and accessibility. Benefiting from this design, our method eliminates reliance on data-driven human preference learning and complex layout-planning pipelines, producing accessible, usable, and spatially coherent scenes in an efficient and stable manner. Extensive experiments demonstrate that our approach significantly improves layout quality and spatial plausibility, highlighting the role of human-centric interaction modeling in 3D scene generation.

1 Introduction

3D indoor scene generation has recently attracted significant attention due to its broad applications in virtual reality [19, 31], embodied simulation [17, 20, 4], human-centered interaction reasoning [38, 35, 39], and interior design [14, 22, 45]. With the increasing availability of large-scale 3D furniture assets and rapid advances in generative modeling [37], constructing indoor scenes through compositional assembly of these assets has emerged as a scalable and widely adopted strategy. However, achieving truly coherent and usable spatial layouts remains largely unexplored.

*Corresponding author.

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

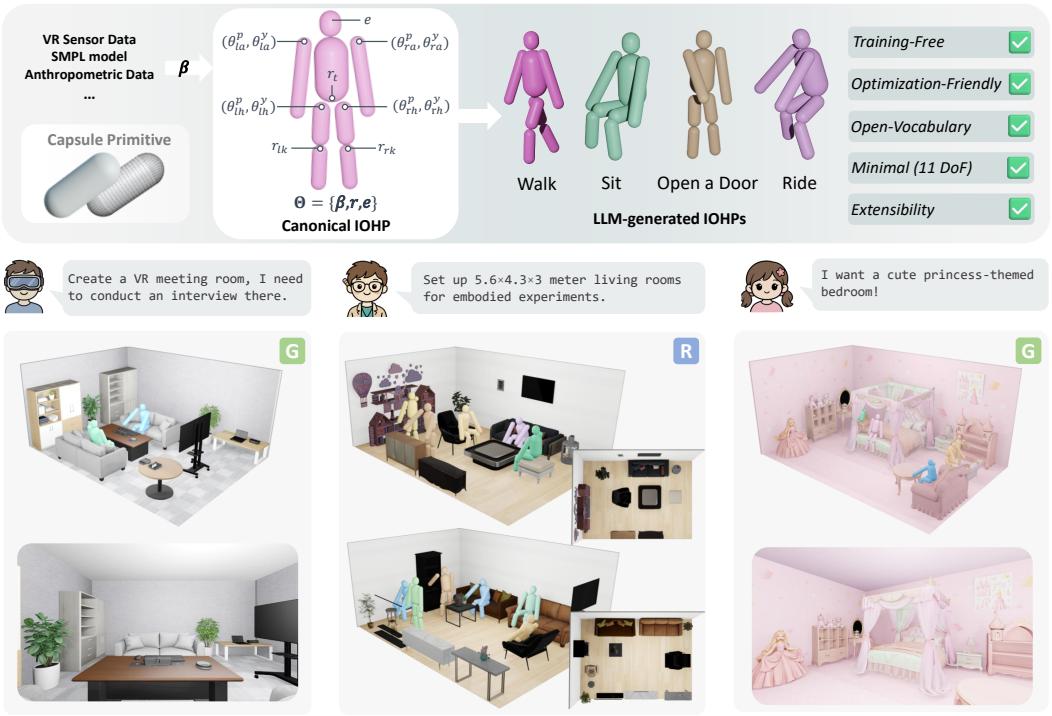

Figure 1: Overview of the Interaction-Oriented Human Proxy (IOHP) and visualizations of IOHP-driven 3D indoor scenes. By modeling interaction feasibility with IOHP, the resulting layouts exhibit strong spatial rationality and functional usability, effectively satisfying user requirements. **G** denotes the fully generative mode, while **R** indicates fast retrieval-based generation. For clarity, we retain only the key IOHPs in the visualizations.

To tackle this problem, existing approaches can be broadly categorized into two paradigms. The first category consists of data-driven generative methods [32, 41, 25, 42, 30], which learn spatial relationships and layout distributions from large-scale furniture arrangement datasets to produce visually plausible scenes. However, indoor layouts exhibit high variability, and training-based approaches often capture dominant patterns in the data, leading to limited diversity in the generated results. Moreover, their reliance on finite datasets limits generalization to open-ended scenarios and complex textual instructions. The lack of explicit spatial constraints, coupled with stochastic generation, further leads to issues such as object interpenetration and inconsistent functional semantics. The second category comprises planning-based methods [29, 9, 43, 12, 2, 40] built upon large language models (LLMs) [24]. By leveraging the strong semantic understanding and reasoning capabilities of LLMs, these approaches formulate scene generation as a structured layout planning problem, improving semantic consistency and interpretability. However, they often rely on multi-stage reasoning pipelines, resulting in high computational overhead and reduced generation stability. In addition, the limited ability of LLMs to model continuous spatial relationships hinders accurate geometric reasoning, often leading to cluttered layouts and blocked pathways.

Overall, ensuring **interaction feasibility** remains a challenge for LLM-based methods, yet this capability is critical for applications in virtual reality and embodied environments. In particular, an interaction-feasible scene should satisfy two key conditions: (1) usability, where at least one collision-free interaction pose exists that allows users to naturally interact with objects (e.g., opening doors, picking up items, or sitting on a chair to watch TV); and (2) accessibility, where unobstructed paths for user movement exist from the scene entrance to target objects.

To address this challenge, we propose a new LLM-based paradigm for 3D indoor scene generation. Our pipeline does not require task-specific training and avoids complex multi-stage layout planning, instead directly optimizing layouts through explicit interaction-aware constraints to produce coherent and usable scenes efficiently. At the core of our method is a novel human-centric representation,

termed the **Interaction-Oriented Human Proxy (IOHP)**, as illustrated in Figure 1. Rather than modeling appearance or pose, IOHP defines a compact geometric abstraction of the space occupied by a human during interaction in the scene, which can be inferred by LLMs through predefined parameterizations and structured formulations. Benefiting from this design, we reformulate interaction feasibility as a unified optimization problem governed by two complementary types of constraints: separation constraints, which resolve geometric conflicts between the proxy and scene elements, and attachment constraints, which establish and preserve the valid contact relationships among them. Specifically, **to ensure usability**, we require that the proxy keeps its interaction regions unobstructed by surrounding objects, while enabling interaction-aware organization of furniture through structured interaction units. **To enhance accessibility**, we promote local free-space expansion by resolving collisions between walking-state proxies and surrounding objects, thereby encouraging clearer pathways throughout the layout. By introducing IOHP, our method enables the generation of layouts in seconds where objects are reachable, usable, and free of spatial conflicts.

Our main contributions are summarized as follows:

- We carefully design the Interaction-Oriented Human Proxy (IOHP), a compact geometric abstraction that models human spatial occupancy during interaction.
- We propose an interaction-aware scene generation framework that unifies accessibility and usability as a geometric resolution problem, enabling direct and continuous optimization of layouts in 3D space.
- Extensive experiments demonstrate that our pipeline produces high-quality 3D scenes with reachable objects, unobstructed interactions, and clear pathways, significantly improving layout quality and scene usability.

2 Related Work

2.1 Data-driven 3D Scene Generation

Most existing 3D generation methods [8, 5, 15] produce monolithic or coupled 3D representations, which limits their applicability. To address this limitation, compositional 3D generation methods [16, 6] generate scenes from images or simple textual prompts. While efficient, these approaches are typically limited to small-scale scenes with a restricted number of objects and simple spatial arrangements, making them less suitable for complex indoor environments. To overcome this limitation, data-driven indoor scene generation methods [21, 32, 44, 7, 33] learn spatial distributions and object relationships from large-scale layout datasets to model more complex scenes. For example, PhyScene [41] incorporates physical guidance to improve plausibility, while OptiScene [42] leverages human preference data to learn spatial patterns. However, these methods largely rely on learning from data distributions, which limits their ability to handle open-set inputs and generalize to novel scene types. Moreover, limited modeling of interaction and physical constraints may compromise the practical usability of the generated layouts. In contrast, our method does not depend on data-driven statistical patterns, but explicitly models human spatial occupancy and interaction requirements, leading to more usable and interaction-consistent layouts.

2.2 Planning-based Indoor Scene Generation

Planning-based methods, such as Holodeck [43], I-Design [2], LayoutGPT [9], and DirectLayout [28] leverage LLMs to formulate indoor scene generation as a structured layout planning problem. More recently, SceneWeaver [40] adopts a reason–act–reflect paradigm to iteratively refine scene configurations, while SceneSmith [27] adopts a multi-agent framework with three agents, Designer, Critic, and Orchestrator, that collaboratively improve semantic consistency and layout rationality. Reason-3D [1] further employs large reasoning models for attribute-aware asset retrieval, constraint-guided object placement, and collision-aware spatial refinement. However, such methods often rely on complex reasoning processes and multi-stage planning pipelines, resulting in high computational cost and slow generation. Furthermore, LLMs have limited ability to capture fine-grained spatial relationships, often resulting in misplaced objects and blocked pathways. Although some methods [29, 36] explicitly optimize object collisions, interaction feasibility remains largely unexplored. In contrast, our method integrates interaction constraints directly into a unified geometric formulation, enabling continuous layout optimization without relying on complex multi-stage planning pipelines. Notably, our method

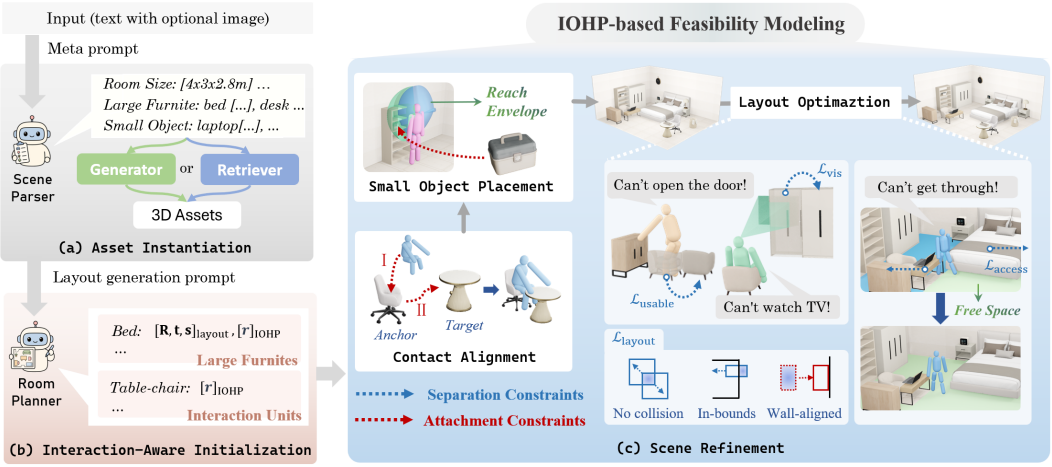

Figure 2: Overview of our proposed pipeline, which consists of three stages: (a) A scene parser processes the input and instantiates 3D assets. (b) A room planner initializes the room layout together with IOHPs. (c) Guided by IOHP-based feasibility modeling, the scene is iteratively refined to produce a usable and accessible 3D indoor environment.

requires only two LLM calls to construct the IOHPs and the initial layout, thereby avoiding issues such as planning conflicts and deadlocks.

2.3 Interaction-driven Scene Generation

Only a few works explicitly incorporate human interaction into scene generation. Wei *et al.* [35] employ a full SMPL-X [26] model to guide layout generation. However, the high-dimensional parameterization of SMPL-X introduces significant complexity, requiring predefined pose templates and making continuous optimization difficult. Jin *et al.* [18] incorporate anthropometric data to constrain layout generation, and SceneTeract [23] evaluates interaction feasibility in existing scenes through agent-specific geometric checks. Nevertheless, such approaches rely on engineered rules, which limit diversity and generalization. In contrast, our proposed IOHP significantly reduces modeling complexity while retaining expressiveness, avoids reliance on template libraries or handcrafted rules, and naturally integrates into a continuous optimization framework for unified interaction feasibility modeling.

3 Method

3.1 Problem Formulation

Given a user input $\mathcal{U}$, our goal is to generate a 3D indoor scene $\mathcal{S}$ through a pipeline $\mathcal{G}$:

$$\mathcal{S} = \mathcal{G}(\mathcal{U}) = \left(\{(o_i, \mathbf{R}_i, \mathbf{t}_i, \mathbf{s}_i)\}_{i=1}^{N},\ \mathcal{E}\right), \tag{1}$$

where o_i denotes the i-th furniture object with rotation $\mathbf{R}_i$, translation $\mathbf{t}_i$, and scale $\mathbf{s}_i$, and $\mathcal{E}$ denotes the scene environment including structural elements such as walls and floors.

3.2 Pipeline Overview

As shown in Figure 2, our pipeline $\mathcal{G}$ consists of three stages:

(a) Asset Instantiation. We first employ an LLM-based *scene parser* to convert $\mathcal{U}$ into structured object lists under a predefined *meta prompt*, and then instantiate the corresponding scene objects:

$$(\mathcal{T}^L, \mathcal{T}^S, \mathcal{E}) = \text{LLM}_{sp}(\mathcal{U}), \quad (\mathcal{O}^L, \mathcal{O}^S) = G_o(\mathcal{T}^L, \mathcal{T}^S), \tag{2}$$

where $\mathcal{T}^L$ and $\mathcal{T}^S$ denote the large furniture and non-floor small-object lists, $\mathcal{O}^L = \{o_i^L\}_{i=1}^{N_L}$ and $\mathcal{O}^S = \{o_k^S\}_{k=1}^{N_S}$ denote the corresponding instantiated object sets together with their metadata. G_o

denotes either an asset generator for fully generative synthesis of high-quality personalized scenes, or an asset retriever for efficient large-scale scene construction via retrieval-based generation.

(b) Interaction-Aware Initialization. At this stage, beyond predicting spatial configurations for objects, a key objective is to infer interaction occupancy represented by IOHPs. Specifically, each large furniture object is associated with an individual IOHP that captures the spatial region required for human interaction. In addition, certain interactions involve multiple objects (e.g., chair–desk). We refer to such combinations as *interaction units*, and assign an additional IOHP to each unit to model the joint interaction space. We employ an LLM-based *interaction planner* to infer structured initialization parameters under a predefined *layout generation prompt*:

$$(\mathcal{P}_{\text{layout}}, \mathcal{P}_{\text{IOHP}}, \mathcal{P}_{\text{IU}}, \mathcal{R}) = \text{LLM}_{ip}(\mathcal{O}^L, \mathcal{O}^S, \mathcal{E}), \tag{3}$$

where $\mathcal{P}_{\text{layout}}$, $\mathcal{P}_{\text{IOHP}}$, and $\mathcal{P}_{\text{IU}}$ denote the predicted layout parameters, object-level IOHP parameters, and interaction-unit IOHP parameters, respectively, and $\mathcal{R}$ denotes the supporting relations for non-floor small objects (e.g., a laptop is supported by a desk). These parameters, together with $\mathcal{E}$, are then used to initialize the scene $\tilde{\mathcal{S}}$ and the IOHP set $\mathcal{H}$. For each small object, only its scale and supporting relation are inferred, rather than requiring explicit absolute placement. Detailed prompts and generation rules are provided in the **Appendix**.

(c) Scene Refinement. Finally, we refine $\tilde{\mathcal{S}}$ via $\mathcal{S} = f(\tilde{\mathcal{S}}, \mathcal{H}, \mathcal{R})$, where f denotes our layout refinement function. In the following, we present the IOHP representation (Sec. 3.3), the IOHP-based feasibility modeling (Sec. 3.4), and the scene refinement process in detail (Sec. 3.5).

3.3 Interaction-Oriented Human Proxy

To enable efficient and reliable modeling of human–scene interaction feasibility, we introduce the Interaction-Oriented Human Proxy (IOHP), a compact geometric abstraction that captures both the spatial occupancy and interaction capability of a human within a scene. Unlike high-dimensional human pose representations (e.g., SMPL-like models) or motion-centric formulations [34] that explicitly model trajectories and dynamics, our design is grounded in an occupancy-centric perspective. We iteratively design and empirically validate IOHP according to the following two principles:

(i) Interaction Completeness. The proxy includes only the body components and degrees of freedom required to approximate human occupancy for interaction. Removing any of them leads to an incomplete coverage of the interaction space and may cause incorrect feasibility estimation.

(ii) Interaction Sufficiency. The proxy avoids unnecessary articulation or geometric detail. Adding extra joints or finer structures brings marginal improvement to interaction modeling, while increasing computational cost and reducing optimization stability.

Overall, IOHP can be viewed as a *minimal yet sufficient* surrogate for interaction-aware occupancy modeling, striking a balance between expressiveness and computational efficiency. We next introduce its formal parameterization and geometric construction.

Parameterization. We represent an IOHP instance by a low-dimensional parameter set:

$$\Theta = \{\boldsymbol{\beta},\, \boldsymbol{r},\, \mathbf{e}\}, \tag{4}$$

where $\boldsymbol{\beta}$ denotes the shape parameters that define the geometric proportions of the proxy, $\boldsymbol{r}$ encodes a set of articulated pose parameters, and $\mathbf{e} \in \mathbb{R}^3$ denotes the eye anchor specifying the origin of visual perception. The local pose $\boldsymbol{r}$ is defined as an 11-DoF articulated parameter set:

$$\boldsymbol{r} = \big(r_{lk},\, r_{rk},\, (\theta^p_{lh}, \theta^y_{lh}),\, (\theta^p_{rh}, \theta^y_{rh}),\, r_t,\, (\theta^p_{la}, \theta^y_{la}),\, (\theta^p_{ra}, \theta^y_{ra})\big), \tag{5}$$

where r_{lk}, r_{rk} denote the left and right knee articulation (single DoF), $(\theta^p_{lh}, \theta^y_{lh})$ and $(\theta^p_{rh}, \theta^y_{rh})$ denote the pitch and yaw rotations of the left and right hips, r_t denotes the trunk articulation (single DoF), and $(\theta^p_{la}, \theta^y_{la})$, $(\theta^p_{ra}, \theta^y_{ra})$ denote the pitch and yaw rotations of the left and right arms. This formulation captures the dominant degrees of freedom required for interaction, while avoiding redundant articulation.

Geometric Representation. We model each body part with a capsule primitive, whose geometric parameters (e.g., length L and radius r) are determined by $\boldsymbol{\beta}$. These parameters can be obtained from anthropometric statistics, VR sensing systems, or parametric human models such as SMPL. The full proxy is defined as the union of all body parts:

$$\mathcal{H}(\Theta) = \bigcup_{i=1}^{K} \mathcal{P}_i(\Theta), \tag{6}$$

where each $\mathcal{P}_i$ corresponds to a semantic body component and is represented as a capsule:

$$\mathcal{P}(L, r) = \{\mathbf{x} \mid \mathrm{dist}(\mathbf{x}, \mathbf{s}) \leq r\}, \tag{7}$$

where $\mathbf{s}$ is a line segment of length L. This representation provides compact geometric primitives with continuous distance queries and isotropic cross-sections, which are well suited for stable collision evaluation and optimization.

Overall, IOHP is a unified representation that captures interaction-relevant human occupancy through a minimal set of degrees of freedom and compact geometric primitives, providing a stable and efficient surrogate for continuous spatial optimization. Although its parameters are inferred by the LLM, the subsequent collision- and contact-based optimization makes the framework robust to imperfect predictions and readily extensible to other human-centric tasks.

3.4 IOHP-based Feasibility Modeling

Given the IOHP representation $\mathcal{H}(\Theta)$, we formulate human–scene interaction feasibility using two complementary geometric constraints. Among them, **separation constraints** resolve geometric conflicts by minimizing overlap between two spatial entities within the scene:

$$\mathrm{occ}(\mathcal{A}, \mathcal{B}) \to 0, \tag{8}$$

where $\mathrm{occ}(\cdot, \cdot)$ denotes a distance-based penetration measure. In addition, **attachment constraints** establish valid contact relationships by translating the source entity toward the target entity until a contact state is reached without interpenetration. With these two constraints, we decompose feasibility modeling into two complementary aspects: *usability* (interaction validity at target locations) and *accessibility* (reachability of those locations).

Usability. To ensure overall usability of the scenario, we consider three cases: **(1) Interaction-unit coherence.** Within each interaction unit, the IOHP should be able to interact with the target object while being properly supported by its corresponding anchor (e.g., sitting on a chair to use a desk). **(2) Small-object reachability.** For a small object o_k^S supported by a large object o_j^L, we evaluate reachability based on the IOHP Θ_j. Specifically, we define a *reach envelope* $\mathcal{E}_{\text{reach}}(\Theta_j)$ [3] as:

$$\mathcal{E}_{\text{reach}}(\Theta_j) = \bigcup_{\boldsymbol{r}' \in \mathcal{N}(\boldsymbol{r}_j)} \mathcal{H}_{\text{arm}}(\boldsymbol{\beta}, \boldsymbol{r}'). \tag{9}$$

The object is considered reachable if $\mathcal{E}_{\text{reach}}(\Theta_j)$ overlaps with any part of o_k^S. **(3) Large-object interactivity.** For each contact-based large object o_i^L with an associated IOHP Θ_i, we define:

$$\mathcal{L}_{\text{usable}} = \sum_i \mathrm{occ}\big(\mathcal{H}(\Theta_i),\ \mathcal{S} \setminus o_i^L\big), \tag{10}$$

which enforces separation constraints between the proxy and all non-target scene elements. When the collision term of an object becomes zero, its associated IOHP is free from unintended collisions, indicating that at least one interaction configuration is feasible. Furthermore, for each vision-based object, we enforce line-of-sight visibility from the eye anchor $\mathbf{e}$. We approximate visibility using a narrow view frustum projected from $\mathbf{e}$ toward the object, and define the visibility term $\mathcal{L}_{\text{vis}}$ as the collision between this frustum volume and the scene, following the same formulation as Eq. 10.

Accessibility. To promote accessibility, instead of relying on explicit path planning [41], we relax the non-differentiable reachability problem into free-space expansion, defined as:

$$\mathcal{F} = \left\{\mathbf{t} \in \mathbb{R}^2 \mid \mathrm{occ}\big(\mathcal{H}(\Theta^{\text{walk}}(\mathbf{t})),\ \mathcal{S}\big) = 0\right\}, \tag{11}$$

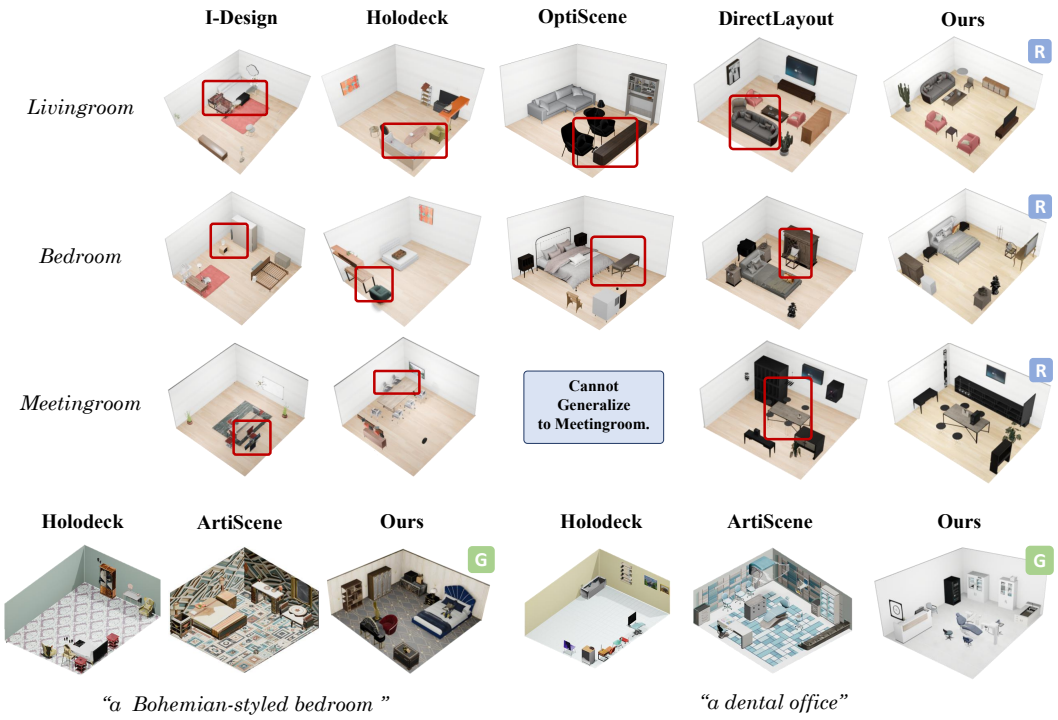

Figure 3: Visual comparisons with existing baselines under two settings: retrieval-based layouts on standard room types (top), and personalized generative scenes (bottom). Regions with unreasonable layouts are highlighted with red bounding boxes. Please zoom in for details.

where $\mathcal{H}(\Theta^{\text{walk}}(\mathbf{t}))$ denotes a walking-state IOHP placed at planar location $\mathbf{t}$. Since accessibility is mainly limited by narrow passages between nearby objects, we place walking-state IOHPs at the ground-plane gap midpoints of nearby object pairs and penalize their collisions with surrounding movable objects using the same separation-based formulation as Eq. 10. The resulting accessibility term $\mathcal{L}_{\text{access}}$ separates obstructing objects, widens passages, and expands $\mathcal{F}$, thereby encouraging more traversable layouts while maintaining minimal yet sufficient clearance for human movement.

3.5 Scene Refinement

The scene refinement is carried out through a three-stage process: (1) **Contact Alignment**: For each large-object IOHP, we establish valid contact with its target via attachment constraints. At the level of an interaction unit, each IOHP is first supported by its corresponding anchor, and then attached to the shared target. Once established, these relationships are fixed during subsequent optimization. (2) **Small Object Placement**: Small objects are placed according to the reachability constraint in Eq. 9 and fixed relative to their supporting entities, preserving these relations during layout optimization; and (3) **Layout Optimization**: All large-object positions are jointly refined by minimizing the following optimization objective:

$$\mathcal{L} = \mathcal{L}_{\text{usable}} + \mathcal{L}_{\text{vis}} + \mathcal{L}_{\text{access}} + \mathcal{L}_{\text{layout}}, \tag{12}$$

where $\mathcal{L}_{\text{layout}}$ enforces physical constraints inspired by [40], including preventing objects from going out of bounds, avoiding inter-object penetration, and enforcing wall-aligned furniture to remain attached to nearby walls. Overall, by introducing IOHP-based feasibility modeling, we cast layout optimization as a process of maintaining object contacts and resolving collision conflicts, offering a concise yet highly expressive formulation.

Table 1: Quantitative comparison of different methods averaged across room types. OOB, OCR, ISR, and UR are reported as percentages. Best results are in **bold**; second-best are underlined. ↑ denotes higher is better, ↓ denotes lower is better.

				GPT Rating				
Method	OOB↓	OCR↓	ISR↑	Inter.↑	Layout.↑	Visual.↑	HR ↑	UR↑
I-Design [2]	7.0	12.7	53.9	6.4	5.9	5.8	4.5	46.7
Holodeck [43]	9.2	9.0	55.3	6.5	6.6	6.3	5.4	50.9
DirectLayout [28]	3.6	5.5	65.5	6.8	6.8	5.3	5.2	52.1
Reason-3D [1]	**1.2**	7.1	70.8	6.9	7.1	5.9	5.6	61.8
Ours	2.0	**3.4**	**76.9**	**7.8**	**8.0**	**7.0**	**6.2**	**72.4**

4 Experiments

4.1 Implementation Details

Our framework is implemented in Python with Blender (`bpy`) for scene construction, optimization, and rendering. We use GPT-5 [24] for scene parsing and interaction planning, TRELLIS.2 [37] for asset generation, and 3D-FRONT [11] for asset retrieval. Experiments run on an NVIDIA A800 GPU (80 GB), with layout initialization and optimization taking approximately 10–30 seconds in total. We compare all methods on 180 matched retrieval-based scenes with diverse AI-generated descriptions, using identical inputs and a shared experimental protocol. All LLM-based methods use GPT-5 for planning. We additionally include 40 generation-mode scenes for ablation studies. More details are provided in the **Appendix**.

4.2 Evaluation Metrics

We assess scene quality using six metrics, following prior work [28, 42] with additional extensions. Out-of-Bounds Rate (OOB) and Object Collision Rate (OCR) measure the proportions of objects exceeding room boundaries and intersecting others, respectively. GPT Rating uses GPT-5.6 Sol to score interactivity, layout rationality, and visual quality on a 10-point scale. Human Rating (HR) measures overall quality based on the same aspects. Each scene is assessed by 10 annotators using the same standardized renderings and scoring guidelines as the GPT evaluator. Usability Rate (UR) is computed by averaging, across all scenes, the proportion of annotators who judge each scene to be physically plausible and interaction-feasible. Finally, Interaction Success Rate (ISR) measures target-object reachability using breadth-first search (BFS) over discretized navigable space.

4.3 Qualitative Evaluation

We qualitatively evaluate our method against several representative baselines, as shown in Figure 3. Under the retrieval-based setting, we compare with I-Design [2], Holodeck [43], OptiScene [42], and DirectLayout [28]. Existing methods often exhibit notable layout deficiencies, including occluded furniture that prevents interaction, overly compact arrangements that waste space, and blocked pathways that hinder movement. Moreover, without explicit interaction modeling, they may produce layouts that appear plausible but violate interaction semantics, such as placing desks and chairs side by side rather than face to face. In contrast, our method significantly improves both spatial utilization and interaction quality through spatial feasibility modeling. Specifically, furniture is more evenly distributed, with sufficient interaction space and accessible pathways, while functionally related objects are arranged in physically and semantically correct configurations. Under the generative setting, methods such as Holodeck and ArtiScene [13] often produce cluttered and less realistic scenes with inconsistent layouts. In contrast, our method leverages an interaction-driven LLM-based pipeline to enable personalized scene synthesis while preserving layout rationality, resulting in scenes that are both functionally plausible and stylistically consistent.

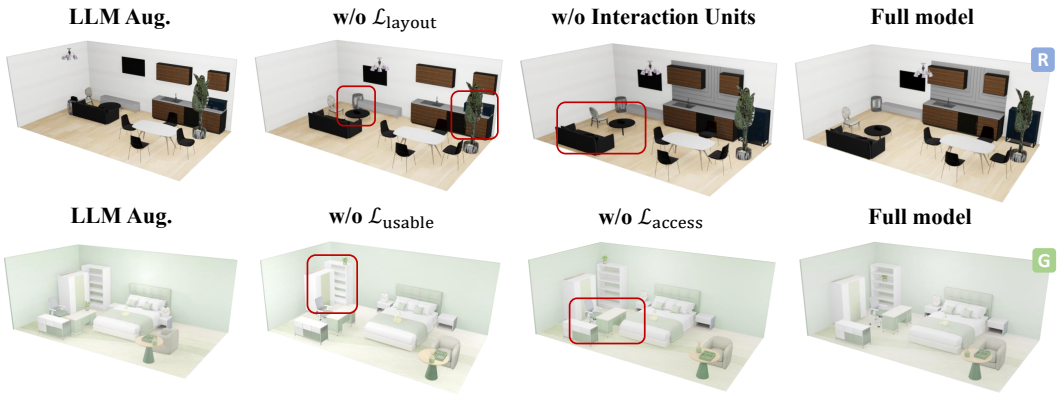

Figure 4: Ablation visualizations of IOHP-based layout optimization. "LLM Aug." denotes the enhanced LLM initialization with carefully designed prompts that explicitly incorporate interaction-related scene priors, serving as our baseline.

Table 2: Ablation study of different components. IU denotes interaction units. GPT Rating† denotes the average GPT Rating over interactivity, layout rationality, and visual quality.

Metric	LLM Aug.	w/o $\mathcal{L}_{\text{layout}}$	w/o $\mathcal{L}_{\text{usable\&vis}}$	w/o $\mathcal{L}_{\text{access}}$	w/o IU	Full model
GPT Rating† ↑	6.2	6.9	7.4	7.2	7.7	**8.0**
HR↑	4.3	4.9	6.1	5.2	5.8	**6.9**
ISR↑	55.7	75.2	73.4	61.3	78.5	**79.7**

4.4 Quantitative Evaluation

As shown in Table 1, we conduct a quantitative comparison with baseline methods using the same version of the LLM for fair evaluation. The compared methods, including, I-Design, Holodeck, DirectLayout, and Reason-3D [1], are all LLM-based approaches that typically require multiple rounds of interaction and refinement. Under the retrieval-based setting, our method requires only two rounds of LLM queries within the entire pipeline, yet consistently outperforms all baselines on both GPT-based and human subjective evaluation metrics. Specifically, it attains a high UR score of 72.4%, indicating improved usability beyond visual plausibility. It also obtains the lowest OCR and highest ISR among the compared methods, supporting the effectiveness of interaction-oriented modeling in improving physical plausibility and interaction feasibility.

4.5 Ablation Study

Qualitative ablation results are shown in Figure 4. The comparisons show that interaction-related prompt augmentation alone is insufficient to guide layout optimization. This limitation stems from the restricted spatial reasoning ability of LLMs and their difficulty in performing precise numerical optimization, making explicit geometric constraints and interaction feasibility modeling essential for producing valid layouts. The term $\mathcal{L}_{\text{layout}}$ serves as the fundamental component for layout optimization; removing it leads to severe physical violations such as object collisions and wall penetrations. The terms $\mathcal{L}_{\text{usable}}$ and $\mathcal{L}_{\text{access}}$ jointly improve interaction feasibility by optimizing usability and accessibility, respectively. Disabling either term results in insufficient interaction space or blocked interaction pathways. In addition, the introduction of interaction units further ensures that high-level interaction semantics among objects are properly captured and maintained. Quantitative results of the ablation study are reported in Table 2. The results show that each component in the scene refinement process contributes significantly to overall layout quality. When all components are enabled, our method achieves the best performance across all metrics, further validating the completeness and effectiveness of the proposed framework.

Robustness to IOHP Inference Errors. Since our framework relies on inferred IOHPs to guide layout refinement, we design two perturbation settings to assess its robustness to imperfect IOHP



Figure 5: Visualization examples of room editing and scene navigation. A walking-state IOHP is used as the agent for navigation.

Table 3: Robustness to IOHP inference errors.

IOHP setting	GPT Rating†↑	HR↑	LIC↓	Validity(%)↑
Cross-scene replacement	6.6	5.2	5.7	22.0
GPT-4o-mini + noise	7.3	6.2	2.5	89.6
GPT-5 (Ours)	**7.7**	**6.4**	**2.1**	**94.9**

predictions: cross-scene replacement substitutes all inferred IOHPs with proxies from other scenes to disrupt interaction associations, while noisy GPT-4o-mini predictions simulate less accurate geometric estimates. We additionally report Layout Issue Count (LIC) and IOHP Validity Rate. LIC is the number of local layout defects averaged across annotators and scenes, while IOHP Validity Rate is the proportion of proxies judged to represent reasonable local interactions by five annotators viewing isolated proxy–furniture renderings. Table 3 shows that cross-scene replacement lowers ratings and increases local defects, whereas noisy predictions remain close to the full method. This suggests greater sensitivity to semantic errors than to pose inaccuracies, consistent with IOHP's role as an interaction-oriented occupancy rather than an exact pose model.

4.6 Applications

The improved layout rationality and interaction feasibility enable better support for downstream applications such as scene editing and LLM-based navigation simulation [42], as presented in Figure 5. After scene editing, our method maintains global consistency through IOHP-based layout optimization, ensuring the updated layout remains interaction-feasible. Moreover, improved global accessibility facilitates navigation reasoning by providing reliable and unobstructed pathways.

5 Conclusion

In this paper, we develop a new LLM-based method for 3D indoor scene generation to address the challenge of interaction feasibility. Unlike prior methods that rely on data-driven priors or complex multi-stage planning, our approach directly optimizes layouts through explicit interaction-aware constraints. Specifically, we carefully design the interaction-oriented human proxy, which provides a compact 3D geometric abstraction for modeling human–scene interaction. Based on this representation, we construct interaction units to organize and relate furniture, and reformulate usability and accessibility as a unified geometric resolution problem, enabling efficient and stable layout optimization. Benefiting from this design, our method generates coherent and usable scenes within seconds, while maintaining strong generalization to open-ended scenarios. Extensive experiments demonstrate that our approach achieves superior performance in terms of scene quality and layout rationality. We believe this work provides a simple yet effective direction for bridging language-based reasoning and geometric optimization, and holds strong potential for applications in virtual reality, embodied AI, and interactive environment design.

6 Acknowledgement

The authors are very indebted to the anonymous referees for their critical comments and suggestions for the improvement of this paper. This work was supported by the National Natural Science Foundation of China (Grant Nos. 62376285 and 61673396).

References

[1] Frédéric Berdoz, Luca A Lanzendörfer, Nick Tuninga, and Roger Wattenhofer. Text-to-scene with large reasoning models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pages 2435–2443, 2026.

[2] 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*, pages 217–234. Springer, 2024.

[3] Don B Chaffin, Gunnar BJ Andersson, and Bernard J Martin. *Occupational biomechanics*. John wiley & sons, 2006.

[4] ChunTeng Chen, YiChen Hsu, YiWen Liu, WeiFang Sun, TsaiChing Ni, ChunYi Lee, Min Sun, and YuanFu Yang. Scenefoundry: Generating interactive infinite 3d worlds. *arXiv preprint arXiv:2601.05810*, 2026.

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

[6] Xingyu Chen, Fu-Jen Chu, Pierre Gleize, Kevin J Liang, Alexander Sax, Hao Tang, Weiyao Wang, Michelle Guo, Thibaut Hardin, Xiang Li, et al. Sam 3d: 3dfy anything in images. *arXiv preprint arXiv:2511.16624*, 2025.

[7] Wenqi Dong, Bangbang Yang, Zesong Yang, Yuan Li, Tao Hu, Hujun Bao, Yuewen Ma, and Zhaopeng Cui. Hiscene: creating hierarchical 3d scenes with isometric view generation. In *Proceedings of the 33rd ACM International Conference on Multimedia*, pages 9783–9792, 2025.

[8] Chuan Fang, Heng Li, Yixun Liang, Jia Zheng, Yongsen Mao, Yuan Liu, Rui Tang, Zihan Zhou, and Ping Tan. Spatialgen: Layout-guided 3d indoor scene generation. *arXiv preprint arXiv:2509.14981*, 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. *Advances in Neural Information Processing Systems*, 36:18225–18250, 2023.

[10] Yao Feng, Vasileios Choutas, Timo Bolkart, Dimitrios Tzionas, and Michael J Black. Collaborative regression of expressive bodies using moderation. In *Proc. of 3D Vision Conference*, pages 792–804, 2021.

[11] Huan Fu, Bowen Cai, Lin Gao, Ling-Xiao Zhang, Jiaming Wang, Cao Li, Qixun Zeng, Chengyue Sun, Rongfei Jia, Binqiang Zhao, et al. 3d-front: 3d furnished rooms with layouts and semantics. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10933–10942, 2021.

[12] 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*, pages 52–70. Springer, 2024.

[13] Zeqi Gu, Yin Cui, Zhaoshuo Li, Fangyin Wei, Yunhao Ge, Jinwei Gu, Ming-Yu Liu, Abe Davis, and Yifan Ding. Artiscene: Language-driven artistic 3d scene generation through image intermediary. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 2891–2901, 2025.

[14] Ziniu Hu, Ahmet Iscen, Aashi Jain, Thomas Kipf, Yisong Yue, David A Ross, Cordelia Schmid, and Alireza Fathi. Scenecraft: An llm agent for synthesizing 3d scenes as blender code. In *Forty-first International Conference on Machine Learning*, 2024.

[15] Tianyu Huang, Wangguandong Zheng, Tengfei Wang, Yuhao Liu, Zhenwei Wang, Junta Wu, Jie Jiang, Hui Li, Rynson Lau, Wangmeng Zuo, et al. Voyager: Long-range and world-consistent video diffusion for explorable 3d scene generation. *ACM Transactions on Graphics (TOG)*, 44(6):1–15, 2025.

[16] 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 *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 23646–23657, 2025.

[17] Baoxiong Jia, Yixin Chen, Huangyue Yu, Yan Wang, Xuesong Niu, Tengyu Liu, Qing Li, and Siyuan Huang. Sceneverse: Scaling 3d vision-language learning for grounded scene understanding. In *European Conference on Computer Vision*, pages 289–310. Springer, 2024.

[18] Semin Jin and Kyung Hoon Hyun. Human-operational 3d indoor layout generation with llm-driven anthropometric simulation. *Automation in Construction*, 184:106846, 2026.

[19] Seok-Young Kim, Dooyoung Kim, Woojin Cho, Hail Song, Suji Kang, and Woontack Woo. Scenelinker: Compositional 3d scene generation via semantic scene graph from rgb sequences. *arXiv preprint arXiv:2602.02974*, 2026.

[20] Sangmin Lee, Sungyong Park, and Heewon Kim. Dynscene: Scalable generation of dynamic robotic manipulation scenes for embodied ai. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 12166–12175, 2025.

[21] Kurt Leimer, Paul Guerrero, Tomer Weiss, and Przemyslaw Musialski. LayoutEnhancer: Generating good indoor layouts from imperfect data. In *SIGGRAPH Asia 2022 Conference Papers*. Association for Computing Machinery, 2022.

[22] Chenguo Lin and Yadong Mu. Instructscene: Instruction-driven 3d indoor scene synthesis with semantic graph prior. *arXiv preprint arXiv:2402.04717*, 2024.

[23] Léopold Maillard, Francis Engelmann, Tom Durand, Boxiao Pan, Yang You, Leonidas Guibas, and Maks Ovsjanikov. SceneTeract: Probing and improving agent-aware activity reasoning in 3D indoor scenes. *arXiv preprint arXiv:2603.29798*, 2026.

[24] OpenAI. Gpt-5. `https://platform.openai.com`, 2025. Accessed via API.

[25] Despoina Paschalidou, Amlan Kar, Maria Shugrina, Karsten Kreis, Andreas Geiger, and Sanja Fidler. Atiss: Autoregressive transformers for indoor scene synthesis. *Advances in neural information processing systems*, 34:12013–12026, 2021.

[26] Georgios Pavlakos, Vasileios Choutas, Nima Ghorbani, Timo Bolkart, Ahmed AA Osman, Dimitrios Tzionas, and Michael J Black. Expressive body capture: 3d hands, face, and body from a single image. In *Proc. of Computer Vision and Pattern Recognition Conference*, pages 10975–10985, 2019.

[27] Nicholas Pfaff, Thomas Cohn, Sergey Zakharov, Rick Cory, and Russ Tedrake. Scenesmith: Agentic generation of simulation-ready indoor scenes. *arXiv preprint arXiv:2602.09153*, 2026.

[28] Xingjian Ran, Yixuan Li, Linning Xu, Mulin Yu, and Bo Dai. Direct numerical layout generation for 3d indoor scene synthesis via spatial reasoning. *arXiv preprint arXiv:2506.05341*, 2025.

[29] 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. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 29469–29478, 2025.

[30] Xiaohao Sun, Divyam Goel, and Angel X Chang. Semlayoutdiff: Semantic layout generation with diffusion model for indoor scene synthesis. *arXiv preprint arXiv:2508.18597*, 2025.

[31] Zhangyao Tan, Qixiang Ma, Runze Fan, Sio Kei Im, and Lili Wang. Interaction-aware shared scene synthesis for vr telepresence. In *Proceedings of the IEEE Conference on Virtual Reality and 3D User Interfaces (IEEE VR)*, 2026.

[32] Jiapeng Tang, Yinyu Nie, Lev Markhasin, Angela Dai, Justus Thies, and Matthias Nießner. Diffuscene: Denoising diffusion models for generative indoor scene synthesis. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 20507–20518, 2024.

[33] Ling Wang, Hao-Xiang Guo, Xinzhou Wang, Fuchun Sun, Kai Sun, Pengkun Liu, Hang Xiao, Zhong Wang, Guangyuan Fu, Eric Li, et al. Scenetransporter: Optimal transport-guided compositional latent diffusion for single-image structured 3d scene generation. *arXiv preprint arXiv:2602.22785*, 2026.

[34] Wenjia Wang, Liang Pan, Zhiyang Dou, Jidong Mei, Zhouyingcheng Liao, Yuke Lou, Yifan Wu, Lei Yang, Jingbo Wang, and Taku Komura. Sims: Simulating stylized human-scene interactions with retrieval-augmented script generation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14117–14127, 2025.

[35] Yao Wei, Matteo Toso, Pietro Morerio, Michael Ying Yang, and Alessio Del Bue. Functional 3d scene synthesis through human-scene optimization. *arXiv preprint arXiv:2502.06819*, 2025.

[36] Hongchi Xia, Chih-Hao Lin, Hao-Yu Hsu, Quentin Leboutet, Katelyn Gao, Michael Paulitsch, Benjamin Ummenhofer, and Shenlong Wang. Holoscene: Simulation-ready interactive 3d worlds from a single video. *arXiv preprint arXiv:2510.05560*, 2025.

[37] Jianfeng Xiang, Xiaoxue Chen, Sicheng Xu, Ruicheng Wang, Zelong Lv, Yu Deng, Hongyuan Zhu, Yue Dong, Hao Zhao, Nicholas Jing Yuan, et al. Native and compact structured latents for 3d generation. *arXiv preprint arXiv:2512.14692*, 2025.

[38] Rongtao Xu, Han Gao, Mingming Yu, Dong An, Shunpeng Chen, Changwei Wang, Li Guo, Xiaodan Liang, and Shibiao Xu. 3d-more: Unified modal-contextual reasoning for embodied question answering. *arXiv preprint arXiv:2507.12026*, 2025.

[39] Pradyumna Yalandur Muralidhar, Yuxuan Xue, Xianghui Xie, Margaret Kostyrko, and Gerard Pons-Moll. Physic: Physically plausible 3d human-scene interaction and contact from a single image. In *Proceedings of the SIGGRAPH Asia 2025 Conference Papers*, pages 1–12, 2025.

[40] Yandan Yang, Baoxiong Jia, Shujie Zhang, and Siyuan Huang. Sceneweaver: All-in-one 3d scene synthesis with an extensible and self-reflective agent. *arXiv preprint arXiv:2509.20414*, 2025.

[41] Yandan Yang, Baoxiong Jia, Peiyuan Zhi, and Siyuan Huang. Physcene: Physically interactable 3d scene synthesis for embodied ai. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16262–16272, 2024.

[42] Yixuan Yang, Zhen Luo, Tongsheng Ding, Junru Lu, Mingqi Gao, Jinyu Yang, Victor Sanchez, and Feng Zheng. Optiscene: Llm-driven indoor scene layout generation via scaled human-aligned data synthesis and multi-stage preference optimization. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025.

[43] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16227–16237, 2024.

[44] Liang Yue, Shao-Kui Zhang, Lin Yuan, Yi-Tao Chen, Zirui Zhou, and Song-Hai Zhang. Synthesizing 3d scenes via diffusion model that incorporates indoor scene characteristics. In *Proceedings of the 33rd ACM International Conference on Multimedia*, pages 9385–9394, 2025.

[45] Mengqi Zhou, Xipeng Wang, Yuxi Wang, and Zhaoxiang Zhang. Roomcraft: Controllable and complete 3d indoor scene generation. *arXiv preprint arXiv:2506.22291*, 2025.

A Extended Method Description

A.1 Separation Constraints

Separation constraints are defined to penalize geometric conflicts between scene elements. Each large object is conservatively approximated by up to eight axis-aligned bounding boxes (AABBs) using structure-aware adaptive splitting. Vertical occupancy changes guide height segmentation, while horizontal splitting removes large empty regions, preserving full geometric coverage and major interaction spaces. For a scene object o_j, the resulting approximation is:

$$\mathcal{B}_j = \bigcup_{k=1}^{K_j} [\mathbf{b}_{jk}^{\min}, \mathbf{b}_{jk}^{\max}], \qquad K_j \leq 8, \tag{13}$$

where $\mathbf{b}_{jk}^{\min}, \mathbf{b}_{jk}^{\max} \in \mathbb{R}^3$ denote the corners of each constituent AABB. Given an IOHP $\mathcal{H}_q \in \mathcal{H}$ and a large furniture object $o_j^L \in \mathcal{O}^L$, we evaluate their geometric conflict based on capsule–AABB proximity. Let $\mathcal{B}_{jk}$ denote the k-th constituent AABB of object o_j. For each capsule primitive $\mathcal{P}_i \subset \mathcal{H}_q$, we compute:

$$d_{ijk} = \mathrm{sdist}(\mathcal{P}_i, \mathcal{B}_{jk}). \tag{14}$$

Penetration occurs when $d_{ijk} < 0$, with depth $-d_{ijk}$. The corresponding separating displacement is:

$$\Delta \mathbf{t}_{ijk} = -d_{ijk}\mathbf{n}_{ijk}, \qquad d_{ijk} < 0, \tag{15}$$

where $\mathbf{n}_{ijk}$ is the unit direction of increasing signed distance with respect to the translation of the element being updated. Displacements are accumulated over colliding capsule–AABB pairs to progressively reduce penetration.

A.2 Attachment Constraints

Attachment constraints are used to establish valid contact relationships between interacting elements by translating objects or IOHPs along a prescribed direction until a contact state is reached without interpenetration. We adopt a unified directional translation formulation. Given an element X (either an object or an IOHP) and a supporting object $\mathcal{Y}$, the translation is parameterized as:

$$\mathbf{t}(\lambda) = \mathbf{t}_0 + \lambda \mathbf{e}_a, \tag{16}$$

where $\mathbf{e}_a$ is a unit direction vector determined by the attachment type (support/attach), and $\lambda \geq 0$ is the translation magnitude. The contact configuration is obtained by finding the first collision-free contact along the prescribed direction:

$$\lambda^\star = \inf \left\{ \lambda \geq 0 \mid d\big(X(\mathbf{t}_0 + \lambda \mathbf{e}_a), \mathcal{Y}\big) = 0,\ \Pi_{\mathcal{Y}}(\mathbf{c}_X) \in \mathcal{A}_{\mathcal{Y}} \right\}, \tag{17}$$

where $\mathbf{c}_X$ is the contact center of X, $\Pi_{\mathcal{Y}}$ projects it onto the supporting object, and $\mathcal{A}_{\mathcal{Y}}$ denotes the valid support region. Collision checks are accelerated using a bounding volume hierarchy (BVH).

The optimal $\lambda^\star$ is approximated using iterative stepping and refined via a coarse-to-fine search to localize the contact boundary. This procedure is used to (i) place small objects onto supporting or attaching surfaces, and (ii) align IOHPs with objects to establish valid interaction configurations. In addition to the common interaction types `attach` and `support`, we introduce a special interaction type, `contain` (e.g., placing an object inside a cabinet). This operation first applies `attach` to move the object into the local interior space, followed by `support` to settle it onto a supporting surface.



Figure 6: A set of examples illustrating small object placement guided by attachment constraints.

We present qualitative results of small object placement in Figure 6. By default, objects are placed within the reachable envelope of the IOHP, ensuring accessibility—an assumption particularly suitable for embodied evaluation and virtual reality scenarios. We also support explicit placement specified by the LLM based on object semantics. Each object undergoes up to 20 random placement attempts. If placement fails (e.g., a tall vase that cannot fit within a storage space), the system iteratively explores alternative placement locations, interaction types, and supporting objects to ensure successful placement within the scene.

A.3 Physical Constraint

The physical layout constraint is implemented using geometric tests on the adaptive AABB approximations:

$$\mathcal{L}_{\text{layout}} = \sum_i \text{vol}(\mathcal{B}_i \setminus \Omega) + \sum_{i<j} \text{vol}(\mathcal{B}_i \cap \mathcal{B}_j), \tag{18}$$

where Ω denotes the room interior, $\mathcal{B}_i$ is the union of AABBs approximating object o_i, and $\text{vol}(\cdot)$ denotes volume. The two terms penalize out-of-bounds volume and inter-object overlap, respectively. For wall-attached furniture, the wall constraint is enforced as a hard constraint. During optimization, the update direction is projected onto the wall-parallel subspace:

$$\Delta \mathbf{t}_i^{wall} = \Delta \mathbf{t}_i - (\Delta \mathbf{t}_i^\top \mathbf{n}_{wall}) \mathbf{n}_{wall}, \tag{19}$$

where $\mathbf{n}_{wall}$ is the wall normal. This removes the perpendicular component, restricting motion to the wall plane. Wall-attached objects are also excluded from the gradient of $\mathcal{L}_{\text{access}}$, as such motion does not improve accessibility and would otherwise lead to undesired dispersion.

B IOHP Representation Details

This section provides a detailed specification of the IOHP representation, including parameter constraints, kinematic structure, geometric primitives, and derived interaction quantities.

Parameter Ranges. All pose parameters are constrained within fixed intervals based on practical requirements of interaction modeling and established ergonomic standards:

$$\begin{aligned} &r_{lk}, r_{rk} \in [0^\circ, 120^\circ], \quad r_t \in [-10^\circ, 60^\circ], \\ &\theta^p_{lh}, \theta^p_{rh} \in [-40^\circ, 120^\circ], \quad \theta^y_{lh} \in [-60^\circ, 100^\circ], \quad \theta^y_{rh} \in [-100^\circ, 60^\circ], \\ &\theta^p_{la}, \theta^p_{ra} \in [-60^\circ, 120^\circ], \quad \theta^y_{la} \in [-60^\circ, 90^\circ], \quad \theta^y_{ra} \in [-90^\circ, 60^\circ]. \end{aligned} \tag{20}$$

All joint angles are defined with respect to the fully extended (canonical) pose, where 0° corresponds to a straight configuration, with positive and negative values indicating opposite anatomical directions of rotation.

Kinematic Dependency. The articulated parameters follow a simplified forward kinematic structure defined on a hierarchical joint tree. All body components are constructed from a set of joints, and each capsule primitive is defined between a pair of joints. Lower-body articulation is driven by hip pitch-yaw parameters, with knee flexion acting as a local coupling on the hip pitch. Specifically, the shank direction is formulated as:

$$\mathbf{d}_{shank} = \mathbf{R}_{yaw}\, \phi_{leg}(\theta^p_h - r_k, \theta^y_h), \tag{21}$$

indicating that knee flexion modulates the effective shank direction rather than acting as an additional global rotation. Upper-limb articulation is defined relative to the torso, with arm directions given by:

$$\mathbf{d}_{arm} = \mathbf{R}_{yaw} \mathbf{R}_t(r_t)\, \phi_{arm}(\theta^p, \theta^y), \tag{22}$$

where $\mathbf{R}_t(r_t)$ denotes the trunk rotation induced by torso articulation, ensuring that arm motion remains consistent with torso orientation. Given the joint positions obtained through forward kinematics, all body parts are instantiated as capsules connecting adjacent joints.

Shape Parameterization. The shape parameter $\boldsymbol{\beta}$ is defined as:

$$\boldsymbol{\beta} = \{(L_i, r_i) \mid i = 1, \ldots, K\}, \tag{23}$$

which determines the geometric proportions of all primitives under a canonical pose. These parameters are deterministically obtained from multiple sources:

- Direct body-part measurements provided by users or sensing systems such as VR devices
- Conversion of monocular-estimated SMPL parameters [10] to a canonical standing pose, followed by body-part measurement
- Skeletal reconstruction from keypoint estimation under a canonical standing pose

Given these inputs, L_i is assigned along the skeletal structure, and r_i is set proportional to local body thickness. In all experiments, a standard adult anthropometric template is used as the default $\boldsymbol{\beta}$, while allowing user-defined customization.

Geometric Primitive Specification. Each primitive $\mathcal{P}_i$ corresponds to a capsule defined in the main text. Here, the underlying segment $\mathbf{s}_i$ is explicitly parameterized as:

$$\mathbf{s}_i = \{\mathbf{c}_i + t\mathbf{d}_i \mid t \in [-L_i/2, L_i/2]\}, \tag{24}$$

where $\mathbf{c}_i$, $\mathbf{d}_i$, and L_i denote the center, unit direction, and length, respectively. This parameterization enables closed-form distance evaluation between a point and the primitive, which is directly used in computing overlap and collision-related quantities.

Visibility Formulation. The eye anchor $\mathbf{e}$ serves as the origin of perception and is by default computed from the head capsule based on standard anthropometric proportions. For a target point $\mathbf{p}$ on object o_{target}:

$$\text{visible}(\mathbf{p}) = \mathbb{I}\Big(\text{ray}(\mathbf{e}, \mathbf{p}) \cap (\mathcal{S} \setminus o_{\text{target}}) = \emptyset\Big). \tag{25}$$

At the object level, this point-level criterion is represented by a view frustum formed from the eye anchor to the target object.

Reachability Envelope. The reachability envelope is formulated as the swept volume of arm primitives over the admissible angular domain. Let

$$\mathcal{A}_{la} = \{(\theta^p_{la}, \theta^y_{la})\}, \quad \mathcal{A}_{ra} = \{(\theta^p_{ra}, \theta^y_{ra})\} \tag{26}$$

denote the valid parameter domains. The reachability region is given by:

$$\mathcal{E}_{\text{reach}}(\Theta) = \mathcal{W}(\mathcal{P}_{la}, \mathcal{A}_{la}) \cup \mathcal{W}(\mathcal{P}_{ra}, \mathcal{A}_{ra}), \tag{27}$$

where $\mathcal{W}(\cdot)$ denotes the swept volume operator. This envelope can be approximated analytically as a sector-like volume in 3D space determined by arm length L_i and angular bounds. In practice, the pitch range used for reachability is restricted to $[10^\circ, 110^\circ]$, which excludes non-interactive configurations (e.g., fully downward or extreme backward poses) and focuses on feasible reaching motions.

External Transformation. The IOHP is embedded into the scene via a rigid transformation:

$$\mathcal{H}_{\text{world}} = \{\mathbf{R}\mathbf{x} + \mathbf{T} \mid \mathbf{x} \in \mathcal{H}(\Theta)\}, \tag{28}$$

where $\mathbf{R} \in SO(3)$ denotes the global rotation and $\mathbf{T} \in \mathbb{R}^3$ denotes the global translation of the IOHP in the scene. The intrinsic geometry $\mathcal{H}(\Theta)$ is determined solely by $\boldsymbol{\beta}$ and $\boldsymbol{r}$.

Extensibility. The formulation supports adaptive extension. Specifically, each primitive can be augmented with two types of parameters: (1) subdivision parameters that control segment partitioning, and (2) local transformation parameters that encode additional offsets or orientations.

C LLM Prompt Design

To enable structured and controllable scene generation, we design a set of prompts that decompose the overall pipeline into modular stages. Specifically, the *Meta Prompt* and the *Layout Generation Prompt* serve as the core components of our framework, responsible for scene parsing and interaction-aware layout planning, respectively. In addition to these prompts, we use the image-generation capability of GPT-5 to support visual asset construction. It generates reference images for 3D asset creation, as well as wall and floor textures that are consistent with the scene context. For efficiency, we generate up to 9 furniture images simultaneously and subsequently process them into the final image set via automatic cropping and rescaling. Empirically, this strategy yields 3D assets with quality comparable to those obtained from individually generated images. Users can also provide their own 3D furniture assets or 2D images. It is worth noting that, to avoid excessive length, we omit parts that are either repetitive with previous sections or primarily designed for output formatting, and mark them as *(omitted)*. The prompts are provided below.

Meta Prompt

You are a professional interior designer. Your task is to parse a user input $\mathcal{U}$ into a structured scene specification for 3D scene generation.
Input Format:
- A textual description $\mathcal{U}^{txt}$
- A reference image $\mathcal{U}^{img}$ (optional)
- A mode flag $m \in \{\texttt{generate}, \texttt{retrieve}\}$
- Optional furniture count constraint η (range or fixed, optional)
If both text and image are provided, jointly reason over them.

Output Format:
[Room Specification: Room Type: ... Room Size: (x, y, z) Door: {wall: ..., position: ...} Environment Constraints: ...]

If $m = \texttt{generate}$:
[Large Furniture List $\mathcal{T}^L$: $\{t_i^L\}$, each as: Type / Description]
[Small Object List $\mathcal{T}^S$: $\{t_k^S\}$, each as: Type / Description]
If $m = \texttt{retrieve}$:
queries = ["category/subcategory", ...]
Each query must be selected from a predefined label vocabulary (see `label.txt`).

Design Principles
1. $\mathcal{T}^L$ must include all large floor-standing furniture that defines the spatial layout (e.g., bed, sofa, table, cabinet).
2. $\mathcal{T}^S$ must include non-floor small objects and wall-mounted objects that are typically supported by large furniture or attached to walls (e.g., books, lamps, decorations, posters).
3. The door orientation is defined as $\in \{\texttt{front}, \texttt{back}, \texttt{left}, \texttt{right}\}$. The door location is parameterized by a scalar $u \in (0, 1)$ along the corresponding wall, measured from left to right for horizontal walls and from bottom to top for vertical walls.
4. All user-specified items must be preserved. Missing but necessary items should be added to ensure functional completeness.
5. In generation mode, each item must include sufficient and diverse attributes (e.g., style, material, structure) to support asset synthesis.
6. In retrieval mode, output label queries that are: - semantically appropriate - sufficient to construct the scene - using appropriate substitutes when exact labels are unavailable (e.g., use a dining table in place of a conference table)
7. Environment constraints $\mathcal{E}$ must only include conditions explicitly specified in the input.
8. The output must be deterministic, structured, and free of ambiguity.
Example output: *(omitted)*

Layout Generation Prompt

You are an expert interior layout planner. Your task is to convert a structured room specification and a set of objects into a scene initialization. The Interaction-Oriented Human Proxy (IOHP) is a geometric representation that helps you design interaction-aware layouts. The documentation and example parameters of IOHP are provided below:
(omitted)
Unless otherwise specified, generate one set of IOHPs and five distinct sets of layout parameters, exploring varied object orientations and spatial configurations. All units are in meters.

Input Format:

- Room specification (geometry, walls, doors, environment constraints)
- (Optional) scene image; if provided, you must refer to the layout shown in the image.
- Large furniture $\mathcal{O}^L$, each represented as $(\texttt{Key}, \texttt{Category}, \texttt{Caption}, L, W, H)$
- Small objects $\mathcal{O}^S$, each represented as $(\texttt{Key}, \texttt{Category}, \texttt{Caption}, L, W, H)$
- (Optional) images of assets, displayed sequentially in a 3×3 grid.
- Mode flag $m \in \{\texttt{generate}, \texttt{retrieve}\}$

Output Format:

- `Pr`: layout of large furniture with their IOHP parameters
- `Se`: small object configuration
- `Re`: relations between objects
- `IU`: interaction units and their IOHPs

1. Definition of `Pr` (Large Furniture Layout with IOHP)
Each entry follows:

`a<ID>|Key|Cx|Cy|R_o|S|Wall|R_h|r|T|N`

- `<ID>`: object index, starting from 0 and increasing sequentially.
- (Cx, Cy): planar position in the room, where the lower-left corner is $(0, 0)$
- R_o: object clockwise orientation angle $\in [0, 359]$, where 0 denotes no rotation and the object faces the front wall by default. Multiples of 90 are recommended.
- S: scaling factor
  - predicted if $m =$ `generate`. When image input is available, predict based on the image, mainly by judging the appropriate object height.
  - fixed to 1 if $m =$ `retrieve`
- $Wall$: whether the object should be placed against a wall, where 1 indicates yes with the back side attached to the wall, 0 indicates no, and a specific wall can be explicitly specified from {`front`, `back`, `left`, `right`}.
- $T \in \{\texttt{S}, \texttt{A}, \texttt{V}\}$: interaction type of the IOHP, including support, attach, and non-contact interaction. For example, `S` can represent sitting on a chair, `A` can represent opening a cabinet, and `V` can represent observing a sculpture.
- R_h: global orientation of the IOHP (`auto`: facing the object for `attach`, and aligned with the object for `support`; can also be specified in degrees)
- $\boldsymbol{r}$: 11-DoF local pose parameters of the IOHP. You should infer appropriate pose parameters that form realistic interactions, such as:
  - sitting for chairs / sofas
  - opening or using cabinets / wardrobes
  - reaching or leaning for desks / tables

  If an existing IOHP corresponds to the same interaction pose as the current one, you may reference it using `a<ID>c` to avoid redundant inference.
- N (optional): when $T =$ `V`, N denotes a reasonable interaction distance between the IOHP and the target object, measured in meters.
- Encourage repeated use of identical furniture items when appropriate (e.g., paired nightstands or multiple identical chairs around a dining table).

2. Definition of `Se` (Small Object Configuration)
Each entry follows:

`b<ID>|Key|R|S|V|Cx|Cy`

- R: orientation constraint, where `0` indicates no orientation constraint (default), and `1` fixes the object orientation to align with its supporting object or attached surface, mainly for functional objects such as printers or laptops.
- V: visibility requirement, where `1` indicates that the object must remain fully visible, and `0` indicates no visibility constraint (default).
- (Cx, Cy) (optional): only used when it is necessary to specify the relative position of a small object (e.g., a wall clock). In this case, (Cx, Cy) represents the normalized coordinates on the attachment surface, where the lower-left corner is $(0, 0)$ and the upper-right corner is $(1, 1)$.

3. Definition of `Re` (Relations)
Each relation follows:

`a<ID>(or wall<ID>)|T|b<ID>`

- T $\in \{\texttt{S}, \texttt{A}, \texttt{C}\}$: `S` for support (on top), `A` for attachment (side or wall), and `C` for containment (inside).
- All small objects must be assigned to a valid supporting object or wall
- Relations must reflect realistic usage while maximizing diversity whenever possible.

4. Interaction Units (`IU`)
For each interaction group:

$$\texttt{a<ID> (target)} \leftarrow \{\texttt{a<ID> (anchor)}_i, \texttt{r}_i\}_{i=1}^{N}$$

- $\texttt{r}_i$ denotes the parameters of the IOHP when supported by anchor $\texttt{a}_i$, where the **anchor** refers to the object that directly supports the human (e.g., chair, sofa), and the **target** denotes the object being interacted with (e.g., desk, table).
- For each IU, identical poses only need to be inferred once, using the first occurrence, and are separated by commas.
- Each IU contains **exactly one target object** and **one or more anchor objects**.
- Identify as many potential interaction units in the scene as possible, particularly human-supporting furniture such as chairs and sofas.

5. Layout and Interaction Principles

- Ensure accessibility and navigable free space, avoid collisions between objects
- Avoid congestion by preventing furniture from clustering too tightly, and distribute objects as evenly as possible
- For wall-attached furniture, prefer distributing them across different walls rather than concentrating on a single wall

6. Example output *(omitted)*

Image Generation Prompt

You are responsible for generating images for the following nine furniture items:
[Furniture List]
Generate a single image arranged as a uniform 3×3 grid of nine equal-sized sub-images. Each sub-image corresponds to one piece of furniture from the list. If fewer than nine items are provided, leave the remaining grid cells blank. The layout must strictly follow the 3×3 grid with consistent sizing and alignment.
If the input contains a reference image, you must extract the appearance of the corresponding furniture from the image and generate each item independently.
Image Requirements
1. The image should be clean and well-organized, with a minimal background. Each furniture item should appear realistic. Light-colored or white backgrounds are preferred.
2. Each sub-image must contain only the furniture itself in its original product form, without any additional objects or decorations.
3. For planar objects such as posters or carpets, generate a clean 2D texture image directly without perspective or surrounding context.
4. If no explicit design description is provided, incorporate diverse combinations of materials, colors, and design styles to enhance visual variety.

Wall and Floor Texture Generation Prompt

You are responsible for generating wall and floor textures for a given indoor scene.

Input Format:

- Room type, room specification, and reference image (optional)
- Texture generation mode $m \in \{\texttt{tile}, \texttt{full}\}$

Output
Generate a single image containing two square sub-images arranged side by side, with the left sub-image representing the wall texture and the right sub-image representing the floor texture.

Texture Requirements
1. The textures must be consistent with the room type, furniture style, and image if provided.
2. Each sub-image must be a planar texture that can be directly used as a texture map.
3. No lighting effects, shadows, text, or additional objects are allowed.
4. While ensuring that the texture remains clean, visually cohesive, and ready for immediate use, you can design a custom texture that complements the room's style.

Mode-Specific Requirements

- **Tile Mode** ($m = \texttt{tile}$): Each sub-image must be a square texture unit. The texture must be seamlessly tileable in both horizontal and vertical directions.
- **Full Mode** ($m = \texttt{full}$): Each sub-image must be a complete texture. The texture will be directly applied without tiling.

D Experimental Implementation Details

Additional Details. Asset generation accounts for the primary GPU memory consumption ($<$ 20 GB VRAM), while bpy-based layout optimization introduces negligible additional memory overhead. We define "small objects" as secondary items placed on, attached to, or contained within furniture or room structures, such as books, cups, lamps, and wall decorations.

The 180 evaluation scenes cover bedrooms, living rooms, public functional spaces, and open-ended room descriptions. AI-generated prompts are designed to maximize diversity in room type, length, and complexity, ranging from basic requests (e.g., "Generate a bedroom") to detailed specifications of style, furniture count, and intended use, without explicitly prescribing furniture arrangements. All methods receive identical inputs and share the same renderer, room shells, wall and floor configurations, and rendering settings. DirectLayout, Reason-3D, and our method use the same per-scene furniture set retrieved from 3D-FRONT. The main quantitative comparison exclusively uses retrieved assets and excludes TRELLIS.2-generated assets. Adapting baselines to the shared GPT-5 API required only JSON-format compatibility fixes.

GPT-5 planning and GPT-5.6 Sol evaluation run through separate API calls with distinct prompts and no shared conversation history, intermediate reasoning, or IOHP parameters. The GPT evaluator receives only standardized scene renderings and the evaluation prompt provided in this supplement. Human annotators use the same renderings and scoring guidelines. Across all 180 scenes, GPT and human scores achieve Spearman and Pearson correlations of 0.84 and 0.80, respectively. Human scores are generally lower, consistent with annotators identifying more subtle layout defects.

Statistical Variability. Table 4 reports scene-level standard deviations for the main comparison in Table 1. HR and UR are first aggregated over the 10 annotators within each scene. These statistics describe variation across the 180 scenes, rather than across repeated generation runs.

Optimization Details. The scene refinement process follows the procedure described in Algorithm 1. For each candidate layout, we perform 200 optimization iterations. For scenes containing more than 10 primary furniture objects, the number of iterations is increased to 350 to allow sufficient refinement in cluttered environments. To improve stability, we first optimize $\mathcal{L}_{\text{layout}}$ alone for 50 warm-up iterations, which enforces basic physical constraints such as boundary validity and object separation. After this stage, all objective terms are jointly optimized.

Table 4: Scene-level standard deviations for Table 1.

Method	OOB	OCR	ISR	Inter.	Layout	Visual	HR	UR
I-Design	4.8	6.2	12.6	1.3	1.5	1.6	2.4	27.6
Holodeck	5.6	5.8	18.8	1.2	1.2	1.4	1.9	19.3
DirectLayout	2.7	3.9	12.4	1.2	1.4	1.3	2.2	26.3
Reason-3D	0.6	4.5	10.8	1.0	1.1	1.3	2.4	23.5
Ours	1.1	2.5	7.7	0.9	0.8	1.1	1.9	22.9

Algorithm 1 Interaction-Oriented Indoor Scene Generation

Require: User input $\mathcal{U}$
Ensure: Refined 3D scene $\mathcal{S}$
$(\mathcal{T}^L, \mathcal{T}^S, \mathcal{E}) \leftarrow \mathrm{LLM}_{sp}(\mathcal{U})$
$(\mathcal{O}^L, \mathcal{O}^S) \leftarrow G_o(\mathcal{T}^L, \mathcal{T}^S)$
$(\mathcal{P}_{\mathrm{layout}}, \mathcal{P}_{\mathrm{IOHP}}, \mathcal{P}_{\mathrm{IU}}, \mathcal{R}) \leftarrow \mathrm{LLM}_{ip}(\mathcal{O}^L, \mathcal{O}^S, \mathcal{E})$
Initialize large-object layout $\tilde{\mathcal{S}}$ from $\mathcal{P}_{\mathrm{layout}}$
Initialize IOHP set $\mathcal{H} = \{\Theta_q\}$ from $\mathcal{P}_{\mathrm{IOHP}}$
for each large object $o_i^L \in \mathcal{O}^L$ **do**
    Associate candidate IOHPs $\{\Theta_{i,k}\}$ with o_i^L
end for
for each interaction unit $u \in \mathcal{P}_{\mathrm{IU}}$ **do**
    Establish relative configurations between anchors, target objects, and IOHPs
end for
for each small object $o_k^S \in \mathcal{O}^S$ **do**
    Retrieve its support entity $q_k \in \mathcal{O}^L \cup \mathcal{E}$ from $\mathcal{R}$
    Place o_k^S on q_k and fix o_k^S; if $q_k = o_j^L$, enforce $o_k^S \cap \mathcal{E}_{\mathrm{reach}}(\Theta_j) \neq \emptyset$ (Eq. 9)
end for
for $t = 1$ to T **do**
    $\mathcal{L} \leftarrow \mathcal{L}_{\mathrm{usable}}(\tilde{\mathcal{S}}, \mathcal{H}) + \mathcal{L}_{\mathrm{vis}}(\tilde{\mathcal{S}}, \mathcal{H}) + \mathcal{L}_{\mathrm{access}}(\tilde{\mathcal{S}}, \mathcal{H}) + \mathcal{L}_{\mathrm{layout}}(\tilde{\mathcal{S}})$ (Eq. 12)
    Update $\tilde{\mathcal{S}}$ by optimizing translations $\{\mathbf{t}_i\}_{i=1}^{N_L}$ to minimize $\mathcal{L}$
end for
$\mathcal{S} \leftarrow \tilde{\mathcal{S}}$
return $\mathcal{S}$

Interaction Success Rate. ISR measures target-object reachability from the room entrance under adult traversal constraints; it does not assess the articulated clearance required for specific actions. We account for traversal clearance by inflating obstacle footprints by $\rho = 0.35\,\mathrm{m}$, requiring a passage width of at least $0.70\,\mathrm{m}$ between opposing obstacles. This empirical margin provides a conservative estimate of adult traversal clearance, whereas smaller margins may classify visually narrow passages as navigable.

We discretize the floor into a binary grid $G \in \{0,1\}^{N_x \times N_y}$, with occupied cells representing the inflated footprints. Starting from an entrance cell s, breadth-first search (BFS) identifies the reachable set $\mathcal{R}_B = \mathrm{BFS}(G, s)$. A target object o_i is considered reachable if at least one of its approach cells belongs to this set, i.e., $\mathcal{T}_i \cap \mathcal{R}_B \neq \emptyset$. ISR is computed as the proportion of reachable target objects. We use the LLM-specified door location as the entrance. For baselines without explicit doors, we instead evaluate candidate entrances at the four room corners and report the maximum ISR.

To examine the sensitivity of ISR to the clearance margin, we vary ρ from 0.29 to $0.38\,\mathrm{m}$. As shown in Table 5, our method consistently outperforms DirectLayout across this range, indicating that the observed accessibility gains are robust to the choice of clearance margin.

Table 5: ISR (%) under different obstacle inflation radii.

ρ (m)	0.29	0.32	0.35	0.38
DirectLayout	67.5	67.1	65.4	65.0
Ours	**78.9**	**78.9**	**78.5**	**77.5**

Editing and Navigation Experiments. For scene editing and navigation tasks, we follow a protocol similar to OptiScene [42]. We first export the current layout of primary furniture and provide it as

structured input to the LLM. For the editing task, the LLM generates an updated layout together with associated IOHPs, after which the layout is further refined using our optimization framework. For the navigation task, the LLM directly generates a simulated agent trajectory, represented as an ordered sequence of walking-state IOHP positions and orientations.

GPT-based Evaluation. Following [42], we employ GPT-5.6 Sol to evaluate the quality of generated layouts from isometric-view renderings. All methods are rendered with the same camera setting, lighting, resolution, and room textures. The prompt template used for evaluation is provided below.

Prompt Template for Room Layout Evaluation

You are an expert in interior design and human-centered spatial planning. Your task is to evaluate the quality of the following room layout renderings. Please assign a numerical score from **0 to 10** (0 = very poor, 10 = excellent) according to the following three aspects:

1. Interactivity

- Can a human or agent move from the room entrance to reach and interact with the main furniture items?
- Are interaction areas reachable without being blocked by other objects?

2. Layout Rationality

- Are furniture pieces arranged logically within the room boundaries?
- Are objects placed according to common layout principles?
- Does the arrangement avoid overlaps or unnatural spatial conflicts?

3. Aesthetics

- Is the overall layout visually balanced and spacious?
- Is the furniture distribution harmonious and visually pleasing?

After evaluating, return your result strictly in the following format:

```
{
  "interactivity_score": X,
  "layout_score": Y,
  "visual_score": Z,
  "comments": "Brief explanation of your judgment."
}
```

Scoring Guidelines

- Provide a short explanation (1–3 sentences) in the "comments" field.
- Focus only on the layout and ignore rendering artifacts such as lighting or texture quality.

E Additional Experiments

E.1 Layout Diversity.

Although our geometric optimization is deterministic, layout diversity arises from independently sampled LLM initializations under the same text input. As shown in Figure 7, even a simple prompt with few primary objects yields distinct object positions, orientations, and groupings. These differences are largely preserved during refinement, as the constraints discourage infeasible configurations without prescribing a unique arrangement.

E.2 Comparison with PhyScene

We additionally compare our method with PhyScene, a data-driven approach, on 15 living rooms and 15 living-dining rooms supported by its publicly available checkpoints. The two methods differ in both their generation settings and interaction representations. PhyScene learns category-specific layout distributions from 3D-FRONT and incorporates collision, boundary, and object-reachability guidance. Our training-free framework supports open-text specifications and explicitly models articulated human occupancy through IOHPs, enabling constraints on interaction-unit configurations,

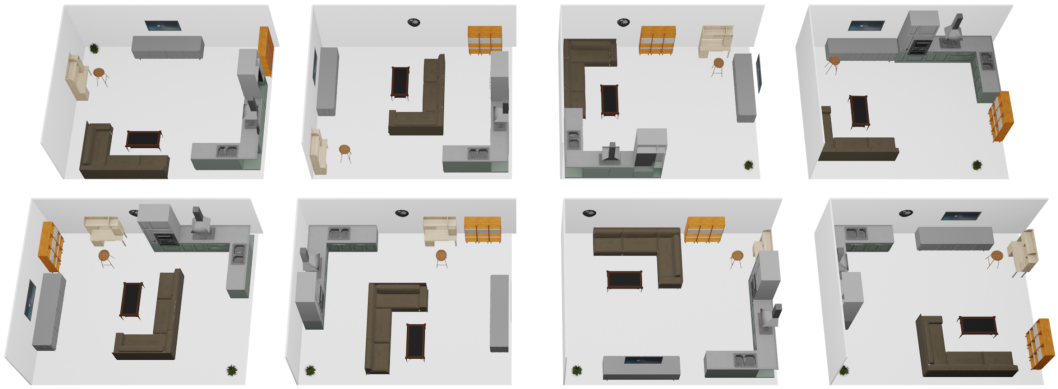

Figure 7: Diverse layouts generated from the same text input.

small-object reachability, and viewpoint-dependent visibility. Table 6 shows that our method achieves lower boundary and collision errors, together with higher reachability and GPT ratings.

Table 6: Comparison with PhyScene on 30 scenes.

Method	OOB↓	OCR↓	ISR↑	GPT Rating†↑
PhyScene	8.1	12.3	74.9	6.4
Ours	**0.0**	**2.3**	**79.1**	**7.5**

E.3 Refinement with an Alternative Initializer

To examine whether IOHP refinement generalizes beyond our initialization procedure, we apply it to DirectLayout outputs on 30 scenes. As shown in Table 7, refinement reduces physical errors and improves ISR by 9.1 percentage points. Our complete pipeline achieves further gains, supporting the benefit of combining interaction-aware semantic initialization with geometric refinement.

Table 7: IOHP refinement with different initializers on 30 scenes.

Initializer	Refinement	OOB↓	OCR↓	ISR↑	GPT Rating†↑
DirectLayout	None	3.0	5.2	65.4	6.6
DirectLayout	IOHP	0.9	2.1	74.5	7.1
LLM Aug.	IOHP	**0.8**	**1.8**	**78.5**	**7.6**

E.4 Contribution of Interaction-Specific Objectives

We isolate the contribution of IOHP-specific objectives by comparing the full model with a variant retaining only basic boundary and collision constraints, under the same setting as the main-paper ablation study. As shown in Table 8, adding usability, visibility, accessibility, and interaction-unit objectives improves both evaluator ratings and ISR. These results indicate that basic physical constraints alone are insufficient to ensure interaction feasibility.

E.5 Robustness to Asset Sources

We further evaluate our method using furniture assets retrieved by Holodeck and assets independently retrieved from Objaverse. Table 9 reports physical metrics and GPT ratings for interactivity, layout rationality, and visual quality. Performance remains broadly consistent with the 3D-FRONT setting, suggesting that the framework is not strongly tied to a single asset source.

Table 8: Contribution of IOHP-specific objectives.

Variant	GPT Rating†↑	HR↑	ISR↑
$\mathcal{L}_{\text{layout}}$ only	6.7	4.7	59.9
Full model	**8.0**	**6.9**	**79.7**

Table 9: Evaluation with alternative asset sources.

Asset setting	OOB↓	OCR↓	ISR↑	Interactivity↑	Layout↑	Visual↑
Holodeck-retrieved	1.6	4.5	74.5	8.0	7.8	7.1
Objaverse retrieval	1.8	3.9	77.6	7.9	8.0	7.3

E.6 Room Editing

We evaluate room editing on 10 scenes, each with five randomly generated editing tasks. Editing Success Rate (ESR) measures the percentage of the 50 tasks judged successful by human evaluators. As shown in Table 10, our method achieves an ESR of 92.0%, with modest increases in collision rate and decreases in reachability after editing.

Table 10: Room-editing results on 50 tasks across 10 scenes.

Stage	OCR↓	ISR↑	ESR↑
Before editing	1.7	80.3	–
After editing	2.4	77.9	92.0

E.7 Optimization Convergence

We assess convergence across scenes of varying complexity, defining convergence as a relative change in total loss below 1% for 20 consecutive iterations. Table 11 reports the mean, standard deviation, and 95th percentile of convergence iterations. Scenes with 5–10 primary objects generally converge within 200 iterations, while denser scenes typically require up to approximately 350 iterations due to more strongly coupled geometric constraints. These observations motivate our iteration budgets. Further iterations yield little change in the evaluated layouts and metrics, making the budgets an empirical efficiency–quality trade-off rather than a strict requirement.

E.8 Runtime Comparison

Table 12 reports runtime statistics measured on a 120-scene subset of the evaluation set. Our method takes 18.35 ± 5.47 seconds on average, substantially less than the compared planning-based methods.

F Discussion

Limitations. While our IOHP-based feasibility modeling significantly improves scene generation quality, the current pipeline still has several limitations. First, our method relies on interaction units as the primary primitives to model semantic relationships. Although this design is more efficient and locally stable than approaches such as semantic scene graphs, it lacks holistic semantic awareness. As shown in Figure 8 (left), an interaction unit (e.g., sofa–coffee table) may be placed in an unreasonable position within the overall layout, leading to suboptimal results. Second, in tightly constrained rooms, limited space for furniture rearrangement may prevent the optimizer from resolving blockages and finding a feasible layout. Sampling multiple initial layouts and selecting the one with the highest feasibility can mitigate this issue. In addition, under the retrieval-based setting, the lack of visual cues may result in inappropriate asset selection and placement. For example, overly small tables or corner-shaped cabinets (Figure 8, right) can produce unrealistic arrangements. Our geometric refinement addresses spatial conflicts but cannot remove redundant objects or correct semantic errors in asset selection, poses, or interaction types. These limitations motivate improved global reasoning,

Table 11: Convergence iterations by scene complexity.

Primary objects	Mean $\pm$ SD	95th percentile
5–10	126.8 ± 27.9	153
> 10	229.4 ± 67.1	315

Table 12: Runtime comparison on 120 scenes.

Method	I-Design	Holodeck	DirectLayout	Reason-3D	Ours
Runtime (s)	167.1 ± 20.4	223.3 ± 29.8	245.5 ± 67.2	182.9 ± 53.2	$\mathbf{18.35 \pm 5.47}$

more robust layout initialization, and better retrieval strategies. Furthermore, inaccuracies in absolute scale estimation and limitations in asset synthesis quality remain open problems.

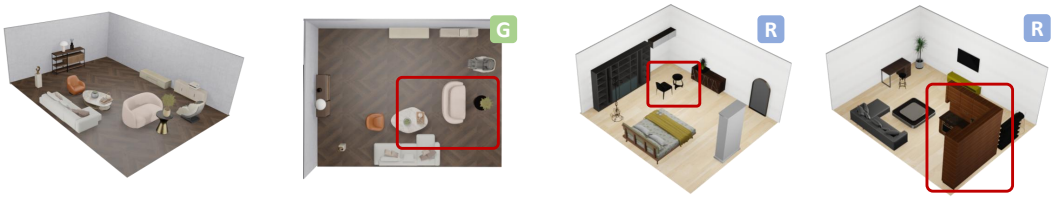

Figure 8: Qualitative examples highlighting the limitations of the proposed method.

Future Directions. Our future work will focus on several directions. (1) We aim to develop a more comprehensive and versatile scene synthesis pipeline, incorporating multi-room compositions as well as additional scene elements such as lighting. (2) We plan to integrate large-scale data synthesis with end-to-end training to enable more efficient and scalable scene reasoning. (3) We will explore the incorporation of physically grounded asset models and articulated object representations to further enhance interaction-aware realism. (4) We intend to extend our human-centric proxy formulation toward broader embodied intelligence and 3D generative modeling tasks, as well as usability-aware scene construction for special populations, such as people with disabilities.

Broader Impacts. Our work has several potential positive societal impacts. By enabling the generation of interaction-feasible 3D indoor scenes, it can support applications in virtual reality, embodied AI, and interior design, thereby improving user experience and the realism of human-centered simulations. In particular, the proposed human-centric modeling paradigm has the potential to facilitate assistive design and accessibility-aware environment planning. At the same time, potential negative impacts include misuse in creating deceptive virtual environments or synthetic data that raise privacy concerns. To mitigate these risks, we emphasize that our framework is intended for academic research, and future work will incorporate safeguards for data privacy and misuse prevention.

G Additional Visualization Results

Figures 9 and 10 present additional qualitative results. The retrieval-based results demonstrate the effectiveness of the proposed method in efficiently generating a large number of spatially feasible and well-structured layouts. The generative results further illustrate the adaptability of our method under diverse input conditions, highlighting its potential for broad applicability across different domains. Figure 11 further presents additional examples of downstream applications enabled by our method.

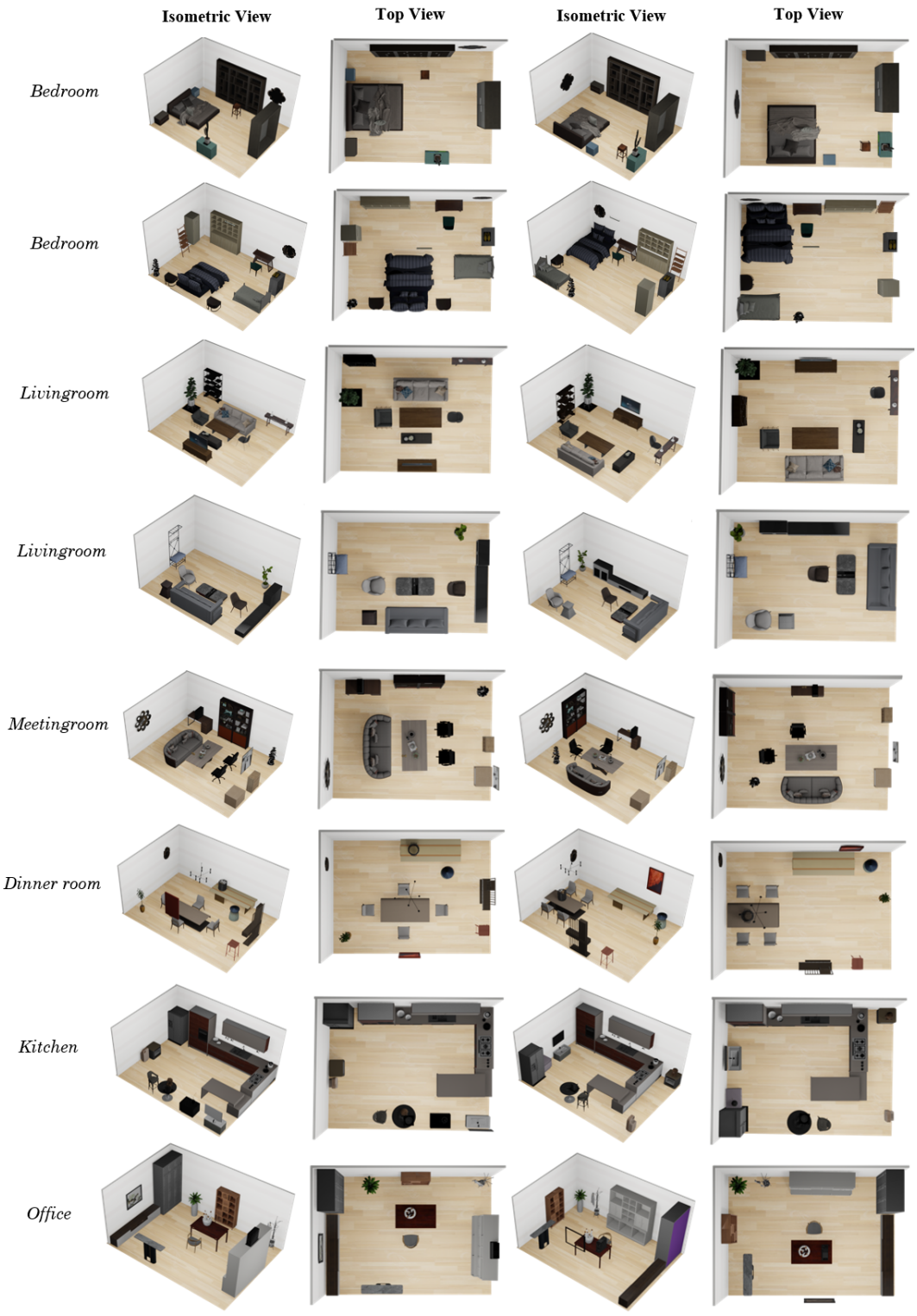

Figure 9: Additional visualizations of retrieval-based scene synthesis. For each input, we present two distinct layout configurations, along with both isometric and top-down views for layout quality assessment.



Figure 10: Additional visualizations of generative scene construction. We present examples under three input modes: concise personalized prompts, complex prompts with furniture type or placement constraints, and image-based inputs provided to guide layout generation.



Figure 11: Additional visualizations of interaction-aware scene editing and navigation.

NeurIPS Paper Checklist

1. **Claims**

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

   Answer: [Yes]

   Justification: The abstract and introduction clearly state the core contributions and are consistent with the proposed methodology and experimental results, while reflecting the intended scope of the work.

   Guidelines:

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

2. **Limitations**

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

   Answer: [Yes]

   Justification: We discuss the limitations in the **Appendix**, where we point out the main weaknesses of the framework and show some visual examples to illustrate them.

   Guidelines:

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

3. **Theory assumptions and proofs**

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

Answer: [N/A]

Justification: The paper does not present formal theoretical results; instead, our contributions focus on introducing interaction modeling into indoor scene generation and proposing a simple yet effective paradigm for resolving spatial feasibility.

Guidelines:

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

4. **Experimental result reproducibility**

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

Answer: [Yes]

Justification: The paper provides the main experimental settings in the **Implementation Details** section, along with additional details in the **Appendix**, providing a reasonable basis for reproducing the main results.

Guidelines:

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

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

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

   Answer: [No]

   Justification: We are currently organizing the data and code used in our experiments to prepare a reproducible public release. We plan to make them available after publication, together with instructions for reproducing the main results.

   Guidelines:

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

6. **Experimental setting/details**

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

   Answer: [Yes]

   Justification: The paper clearly specifies the optimization setup, including data, key parameters, and implementation details, providing sufficient information to understand and interpret the results.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

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

   Answer: [Yes]

   Justification: Table 4 reports standard deviations across the 180 evaluation scenes for all main comparison metrics, with human judgments aggregated within each scene.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.

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

8. **Experiments compute resources**

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

   Answer: [Yes]

   Justification: The paper provides the relevant information on computational resources in the **Implementation Details** section of the main text, including the necessary specifications to reproduce the experiments.

   Guidelines:

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

9. **Code of ethics**

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

   Answer: [Yes]

   Justification: The research complies with the NeurIPS Code of Ethics. It does not involve sensitive personal data, and the human evaluation only concerns subjective ratings of synthetic scene renderings without collecting identifiable information.

   Guidelines:

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

10. **Broader impacts**

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

    Answer: [Yes]

Justification: The paper discusses both potential positive and negative societal impacts in the **Discussion** section of the **Appendix**.

Guidelines:

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

11. **Safeguards**

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

Answer: [N/A]

Justification: The paper does not involve releasing high-risk models or sensitive datasets, and thus does not pose significant risks of misuse requiring specific safeguards.

Guidelines:

- The answer [N/A] means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

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

Answer: [Yes]

Justification: All existing datasets, codebases, and models used in this work are properly credited and cited. We follow their respective licenses and terms of use, and use them only for research purposes.

Guidelines:

- The answer [N/A] means that the paper does not use existing assets.

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

13. **New assets**

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

    Answer: [N/A]

    Justification: The paper does not introduce or release new datasets, models, or other assets.

    Guidelines:

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

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

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

    Answer: [Yes]

    Justification: We conduct a human evaluation for layout quality and usability. Each generated scene is independently evaluated by 10 annotators. The annotators are recruited from volunteer graduate students. They are instructed to evaluate each scene based only on the provided renderings. No personal, sensitive, or identifiable information is collected from the annotators.

    Guidelines:

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

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

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

Answer: [N/A]

Justification: The paper only involves evaluation of synthetic scene renderings and does not collect personal, behavioral, or sensitive information from participants; therefore, IRB approval is not applicable under our setting.

Guidelines:

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

16. **Declaration of LLM usage**

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

Answer: [Yes]

Justification: LLMs are a core component of our method, primarily used to process textual inputs, parse scene specifications, initialize scene layouts and IOHP parameters, and provide prompts or image-generation capability for visual asset construction. The paper clearly describes how they are integrated into the pipeline and how their outputs are further refined by geometry-based optimization.

Guidelines:

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