Global Graph-Validated Optimization for VLM-based 3D Indoor Scene Generation

Jialu Huang[1] Yingxuan You[2] Fei Wang[1] Zheng Dang[3⋆]

[1] National Key Laboratory of Human-Machine Hybrid Augmented Intelligence Institute of Artificial Intelligence and Robotics, Xi'an Jiaotong University, China
[2] École Polytechnique Fédérale de Lausanne (EPFL), CVLab, Switzerland
[3] School of Electronics and Information, Northwestern Polytechnical University and Shaanxi Key Laboratory of Information Acquisition and Processing, China
{huangjialu@stu,wfx@mail}.xjtu.edu.cn
yingxuan.you@epfl.ch
zheng.dang@nwpu.edu.cn

Abstract. We study open-vocabulary 3D indoor layout generation, which synthesizes diverse and physically plausible indoor scenes from unlabeled 3D assets given free-form language instructions. Recent text-guided layout generation methods leverage large language models (LLMs) and vision-language models (VLMs) to synthesize structured scenes directly from text descriptions. However, most existing approaches model inter-asset relations implicitly or rely on local pairwise constraints during local optimization. Such formulations are misaligned with the global and highly non-convex feasible layout space, often producing locally plausible but globally inconsistent or physically infeasible scenes. To address these limitations, we introduce a graph-based intermediate representation that decouples semantic coherence and physical feasibility, and propose a hybrid search-and-refinement strategy to generate indoor layouts with global semantic consistency and physical feasibility. First, we propose Global Semantic Verification (GSV), which represents scenes as structured scene graphs and enforces semantic constraints through rule-based graph verification. This explicit structural validation prunes contradictory configurations and produces a globally consistent semantic scaffold for scene generation. Second, we introduce Global Physical Feasibility Search (GPFS), a hybrid optimization framework that combines evolutionary search for global exploration with gradient-based refinement for local exploitation. GPFS reduces dependence on VLM-proposed initialization and improves robustness in highly non-convex and discontinuous feasible spaces. Together, GSV and GPFS shift layout generation from local relational modeling and initialization-sensitive optimization toward globally consistent reasoning and exploration. Experiments demonstrate that our method achieves state-of-the-art performance on open-vocabulary 3D indoor layout generation, improving both semantic consistency and physical plausibility.

Keywords: Layout Generation · Vision-Language Models · Scene Graphs · Physical Plausibility

⋆ Corresponding author.

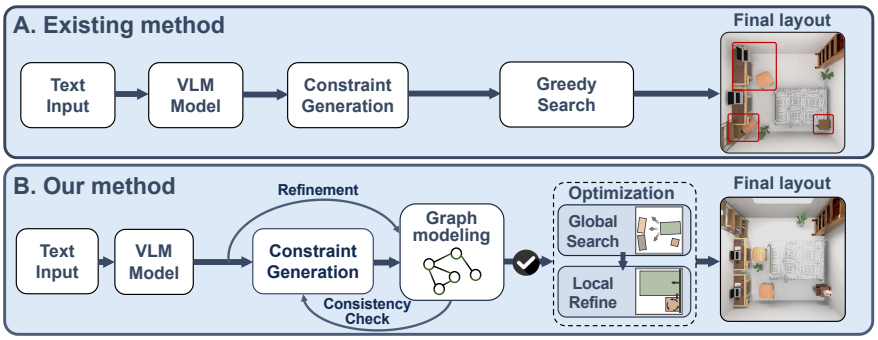

Fig. 1: Existing methods vs. our approach. Prior methods rely on local pairwise consistency and gradient-based refinement, often producing semantically or physically invalid layouts. Our graph-based iterative refinement and optimization yields coherent and physically plausible layouts.

1 Introduction

Spatial reasoning and planning refer to understanding, arranging, and manipulating objects in 3D space while respecting physical constraints. These abilities are crucial for autonomous agents to navigate, plan, and interact with complex environments. Automatically generating realistic simulated scenes has become essential for providing diverse training data to improve agents' spatial reasoning. In this work, we focus on open-vocabulary 3D indoor layout generation, creating diverse indoor scenes from unlabeled 3D assets guided solely by free-form language instructions.

Recent progress in text-guided 3D indoor layout generation leverages large language models (LLMs) [3,7,9,15,22,37] and vision-language models (VLMs) [30] to synthesize structured scenes directly from text descriptions. These methods implicitly model inter-object relations through probabilistic likelihood estimation, and some further enforce pairwise consistency between object pairs during optimization. However, scene layout generation is inherently a globally constrained reasoning problem. The validity of a scene depends not only on local pairwise compatibility, but on the joint satisfaction of semantic and physical constraints across all objects. This misalignment manifests in two critical limitations. Firstly, existing VLM-based layout methods either model relations implicitly or rely on local pairwise regularization, lacking an explicit mechanism to ensure global consistency. As a result, scenes can be locally plausible but globally invalid. Secondly, physical feasibility depends critically on the optimization strategy used to realize layouts. Recent methods predominantly rely on gradient-based optimization initialized from VLM predictions. However, the feasible layout space under semantic and physical constraints is highly non-convex and often exhibits discontinuities due to collision avoidance, support relations, and stability requirements. Gradient descent is inherently local and strongly dependent on initialization, making it prone to being trapped in suboptimal basins. Consequently, the optimization process may converge to configurations that either

violate physical constraints or fail to explore alternative feasible arrangements that better satisfy both semantic and physical requirements.

In this work, we address these two challenges by explicitly disentangling semantic coherence and physical feasibility, and introducing dedicated mechanisms for each. To enforce global semantic coherence, we introduce the Global Semantic Verification (GSV). Instead of promoting consistency through probabilistic alignment, GSV models the scene as a structured scene graph and encodes semantic constraints explicitly. We then perform rule-based graph verification to check relational consistency and prune contradictory configurations. This shifts layout generation from implicit relational fitting to explicit structural validation, producing a scene-level semantically consistent graph that serves as the scaffold for subsequent layout realization.

To improve physical feasibility and mitigate sensitivity to VLM initialization, we propose Global Physical Feasibility Search (GPFS) based on consistent graph generated by GSV for layout generation. GPFS combines evolutionary search for global exploration with gradient-based refinement for local exploitation. The evolutionary component maintains a diverse population of candidate layouts within the semantically constrained space, reducing reliance on a single starting point and helping escape local minima. Notably, GPFS does not require a VLM-proposed initialization and can operate from a diverse population of randomly initialized layouts. The gradient component then refines candidate solutions to better satisfy semantic and physical constraints. Together, our GSV and GPFS move layout generation beyond local consistency checks and neighborhood refinement, enabling globally consistent exploration. This shift from local to global reasoning ensures scene-level coherence and physical validity, improving the structural quality of automatically generated scenes.

Our contribution can be summarized as follows:

- We identify and disentangle two key limitations in VLM-based layout generation: the lack of global semantic coherence and the sensitivity of gradient-based optimization to physical feasibility constraints.
- We propose GSV, a rule-based graph verification mechanism that enforces global semantic coherence through explicit structural validation over scene graphs. To our knowledge, this is the first work to incorporate explicit scene graph modeling and verification into a VLM-based 3D indoor layout generation pipeline.
- We introduce GPFS, a hybrid search framework that integrates evolutionary global exploration with gradient-based local refinement, reducing reliance on VLM initialization and improving reliability to highly non-convex and discontinuous feasible spaces.
- We achieve state-of-the-art performance on open-vocabulary 3D indoor layout generation, with clear improvements in semantic consistency and physical plausibility.

2 Related Work

Open-Vocabulary Indoor Scene Synthesis. Indoor layout generation and scene synthesis have historically relied on manually specified rules and handcrafted statistical priors [4,5,10,14,26,38]. With the availability of large-scale 3D scene datasets [13,18], recent work [8,16,20,21,31,32,34–36] began to learn composition patterns directly from data, marking a transition from early rule-based approaches to deep learning-driven methods. More recently, the emergence of large language models and vision-language models has enabled open-vocabulary 3D scene synthesis, allowing flexible layout generation guided by natural language instructions.

Language-Guided Indoor Layout Generation. To improve flexibility beyond fixed object vocabularies and predefined scene templates, recent work has increasingly incorporated large language models (LLMs) and vision–language models (VLMs) into 3D layout generation pipelines, using natural language as a high-level specification for scene structure. LayoutGPT [9] pioneered this direction by employing GPT to generate room layouts from high-level instructions. Subsequent works such as [1,3,7,15,22,23,29,37] have also integrated LLMs into the synthesis pipeline. Further advancing this line, LayoutVLM [30] employs a vision-language model to define semantic relationships and uses differentiable optimization to produce continuous and semantically consistent layouts. However, they often struggle in scenes with many objects, the relationships generated by LLM become unreliable. In contrast, our method mitigates these limitations by improving the reliability of relational information and enabling robust optimization from scratch, yielding layouts that remain coherent and physically valid.

Scene Graph Representation. Scene graphs, which represent objects as nodes and relationships as edges, have been widely adopted for structured 3D scene representation and support various tasks including scene retrieval [33], comparison [11], and layout generation [25]. This representation has been extended to 3D environments through hierarchical structures [2, 20, 24, 33, 39], parse trees [27], and graph neural networks [17]. Within these approaches, generative methods often incorporate VAEs [19,28] and use message passing for layout synthesis [40]. Recent methods, including Holodeck [37], InstructScene [21], and Graph-to-3D [8], continue this tradition by leveraging scene graphs for semantic control, instruction-driven editing, and geometry-aware generation. However, these methods generally depend on a static scene graph that assumes well-formed inputs. In contrast, our approach maintains an adaptive graph that is refined during generation, providing more reliable relational cues to guide the VLM.

3 Methodology

Problem Definition. We formulate the task as arranging a set of 3D assets within a bounded environment according to natural language instructions. Given a textual layout description ℓ_{layout}, the system identifies a set of all semantically

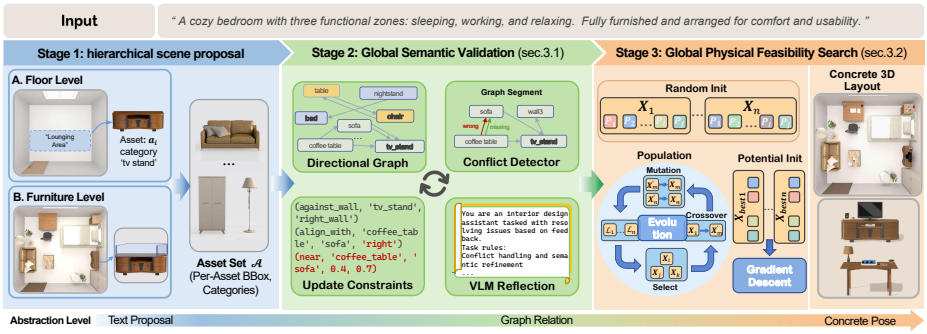

Fig. 2: Overview of the proposed pipeline. Given a textual scene description, (1) the framework first constructs a hierarchical scene proposal mapping functional areas and asset set; (2) semantic constraints are then extracted and refined in a graph via Global Semantic Verification (GSV); finally, (3) a physically valid 3D layout is generated using Global Physical Feasibility Search (GPFS).

relevant 3D assets $\mathcal{A} = \{a_i\}_{i=1}^{N}$, where N is the number of assets and each a_i denotes a 3D asset instance inferred from the text. The objective is to estimate the pose of each asset:

$$P_i = (x_i, y_i, z_i, \theta_i), \tag{1}$$

where (x_i, y_i, z_i) represents the 3D position and θ_i denotes the orientation around the vertical axis. The final output is the scene configuration: $\mathcal{S} = \{(a_i, P_i)\}_{i=1}^{N}$, assigning each asset a specific pose.

Overview. As shown in Fig. 2, our framework consists of three stages:

(1) **Scene Proposal Construction.** Given the input description, we build a hierarchical scene proposal that separates floor- and furniture-level assets. At each level, a VLM infers functional areas and generates asset descriptions, which are mapped via CLIP-based retrieval to form the final asset set $\mathcal{A}$ with per-asset bounding boxes and category labels.

(2) **Semantic Constraint Modeling and Verification.** The scene proposal is translated into structured inter-asset relational constraints and organized as a scene graph. Global Semantic Verification (GSV) enforces explicit global consistency, eliminating contradictory configurations and yielding a semantically coherent scaffold for layout realization.

(3) **Physically Feasible Layout Realization.** Based on the verified semantic graph, Global Physical Feasibility Search (GPFS) computes a concrete arrangement by combining global exploration with local refinement, ensuring physical validity while preserving semantic relations.

Our main contributions lie in the second and third stages of the pipeline, namely Global Semantic Verification (Sec. 3.1) and Gloabl Physical Feasibility Search (Sec. 3.2). We model layout constraints as a graph to capture globally semantically consistent relations, and leverage this relational graph to optimize physically feasible object placements, achieving both search efficiency and solution quality.

3.1 Global Semantic Verification

The Global Semantic Verification (GSV) module aims to enforce global semantic consistency over the relational constraints predicted by the VLM. To achieve this, the verification process is structured into three steps, including scene graph construction and anchor selection, relational consistency verification, and iterative VLM-guided graph refinement. The details of each step are described below.

3.1.1 Step 1: Scene Graph Construction and Anchor Selection

Scene Graph Construction. Building on the hierarchical scene proposal generated by VLM and the asset set $\mathcal{A}$ with their candidate placement regions, the VLM generates a set of implicit pairwise relational constraints in textual form. To obtain a structured representation that enables systematic validation, we formalize these relations as a constraint set and organize them into a directed scene graph. Formally, we define:

$$\mathcal{R} = \{r_{ij} \mid a_i, a_j \in \mathcal{A}\}, \tag{2}$$

where each constraint between assets a_i and a_j is defined as $r_{ij} = (a_i, a_j, \tau_{ij}, \boldsymbol{\psi}_{ij})$, with τ_{ij} denoting the constraint type and $\boldsymbol{\psi}_{ij}$ denoting the associated parameters (see supplementary material for details). Based on this formalization, the constraint set $\mathcal{R}$ induces an initial directed graph $G^{(0)} = (V^{(0)}, E^{(0)})$, where each node $v_i \in V^{(0)}$ represents an asset a_i with pose P_i and each directed edge $e_{ij} \in E^{(0)}$ encodes a pairwise constraint r_{ij}. The poses $\{P_i\}_{i=1}^{N}$ are randomly initialized within the room boundary and updated during optimization in Sec. 3.2.

Anchors Node Selection. Given the initial graph $G^{(0)}$, we partition it to obtain a set of semantically coherent subgraphs $G^{(0)} = \{G_l^{(0)}\}_{l=1}^{N}$, where each $G_l^{(0)} = (V_l^{(0)}, E_l^{(0)})$ corresponds to a localized object group. To establish a clear guidance order for conflict detection, each group is anchored by a single central object. Specifically, we define node centrality as in-degree and choose the node with the largest in-degree as the anchor node v_l^{anchor}. As a local reference, this anchor has the strongest structural influence on its neighbors and guides subsequent conflict detection. More details on anchor selection are provided in the supplementary material.

3.1.2 Step 2: Relational Consistency Verification

Given a refined subgraph $G_l^{(0)} = (V_l^{(0)}, E_l^{(0)})$ with its anchor node $v_l^{anchor} \in V_l^{(0)}$, we perform subgraph-level validation to detect semantic and structural inconsistencies. We introduce a predicate Φ that evaluates both semantic and structural feasibility. Specifically, Φ consists of K validation predicate $\{\Phi_k\}_{k=1}^{K}$, and the overall validity of the subgraph is defined as:

$$\Phi(G_l) = \bigwedge_{k=1}^{K} \Phi_k(G_l), \tag{3}$$

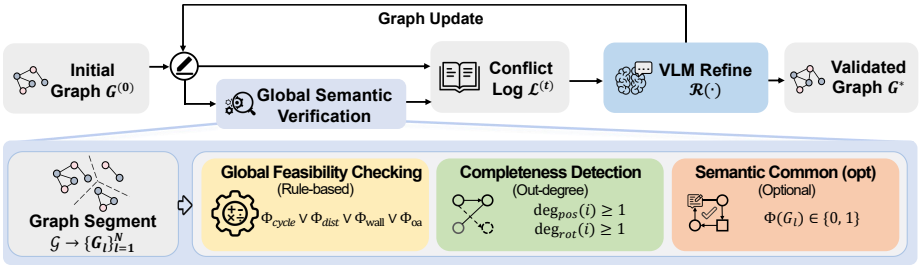

Fig. 3: The Graph-guided Reflection Process. Through iterative updates to scene-graph edges, the model detects and mitigates physical, semantic, and structural inconsistencies, yielding a coherent and physically plausible layout.

where each Φ_k evaluates a semantic criterion on the subgraph $G_l^{(0)}$, $\bigwedge$ denotes logical AND. Consequently, $\Phi(G_l) = 1$ if and only if every Φ_k is satisfied. The K validation predicate $\Phi = \{\Phi_k\}_{k=1}^{K}$ cover three aspects: feasibility verification, completeness verification, and semantic consistency verification, as follows.

Feasibility Verification. This rule-based module verifies structural and physical validity of the subgraph using four constraints: cycle consistency (Φ_{cycle}), distance consistency (Φ_{dist}), wall compliance (Φ_{wall}), XY-plane occupancy (Φ_{oa}) and logic consistency (Φ_{logic}). Detailed definitions are provided in the supplementary material.

Completeness Verification. To ensure relational completeness, we define a completeness predicate Φ_{comp}. This predicate requires that each node v_i must have at least one positional constraint and one orientation constraint. This is enforced by requiring $\deg_{\text{pos}}(v_i) \geq 1$ and $\deg_{\text{rot}}(v_i) \geq 1$.

Semantic Consistency Verification. We further enforce common-sense semantic priors through pairwise compatibility checks Φ_{sem}. Violations are detected by:

$$\Psi(G_l) = \prod_{(i,j)\in V_l} (1 - \Gamma(e_{ij})), \tag{4}$$

where $\Gamma(e_{ij})$ indicates a violated constraint and $\prod$ accumulates all pairwise checks over edges (i, j) in V_l. Detected violations at iteration t are recorded in a conflict log, denoted as $\mathcal{L}^t$, and fed back to the VLM for iterative refinement, producing a conflict-free refined graph G^*.

3.1.3 Step 3: Iterative VLM-Guided Graph Refinement

We introduce an iterative refinement process that examines the scene graph for physical inconsistencies, semantic contradictions, and missing relational constraints. Detected conflicts are provided to the VLM, which proposes refined edges. This loop repeats until no new conflicts arise, producing a self-consistent constraint graph for subsequent optimization, as illustrated in Fig. 3.

Algorithm 1: Hybrid Evolutionary-Gradient Optimizer

Input: Scene graph G^*, asset set $\mathcal{A}$, room bound B, group set $\mathcal{G}$, anchor nodes $\mathcal{C}$, parents size μ, offspring size λ, generations T

Output: Optimized layout $\mathcal{X}^*$

Initialization: Generate μ parent individuals $\{\mathbf{x}_1, \ldots, \mathbf{x}_\mu\}$ via random initialization

Evaluate fitness $f_i = \text{Fitness}(\mathbf{x}_i)$ for all parents

for $t \leftarrow 1$ **to** T **do**

  Recombination and Mutation:

  Select λ parents with probability $\propto e^{-\beta f_i}$, swap random asset pairs

  foreach *offspring* **x do**

    Apply asset-wise and group-wise gaussian mutation to position/orientation of assets

    Apply intra-group attraction toward their center node

    Apply pairwise repulsion to avoid collapse

    Clamp all positions to room bounds B

  Evaluation:

  Combine populations $\mathcal{P} = \{\text{parents} \cup \text{offspring}\}$

  Compute fitness $f(\mathbf{x}_i) = L_{\text{semantic}} + L_{\text{physics}}$

  Selection:

  Retain top-μ elites by fitness

Gradient descent: Perform gradient descent on top-μ elites for N iterations

Return best individual $\mathcal{X}^* = \arg\min_{\mathbf{x}_i} f(\mathbf{x}_i)$

Formally, for $E_l^{(t)}$ at iteration t, edges identified as conflicting are collected in the set $E_{l,\text{conflict}}^{(t)} \subset E_l^{(t)}$. The VLM then suggests refinements to replace or update these edges, resulting in the next iteration's edge set $E_l^{(t+1)}$. This process continues until no conflicts remain, yielding a consistent graph G^* that serves as stronger guidance for subsequent optimization.

3.2 Global Physical Feasibility Search

Given the verified relational graph G^* in Sec. 3.1, we aim to estimate object poses $\mathcal{P} = \{(x_i, y_i, z_i, \theta_i)\}$ that satisfy the semantic constraints encoded in G^* while remaining physically feasible within the scene. To achieve robust layout realization in a highly non-convex solution space, the optimization is organized into two steps: global exploration and local refinement. Algorithm 1 provides a detailed description of the full workflow. The following subsections describe the design and workflow of each step in detail.

3.2.1 Global Exploration

Given G^*, this step performs a population-based global search to generate diverse candidate layouts that serve as initialization for subsequent local refinement. To achieve broad coverage of the highly non-convex solution space, we

introduce a population-based evolutionary search that maintains multiple layout hypotheses and iteratively improves them through stochastic transformations. Specifically, we design two key operations to guide the exploration process: ***swap-based crossover***, which enhances structural diversity by exchanging object poses within a layout, and ***center-guided mutation***, which perturbs object placements while preserving group-level spatial coherence. Together, these operations enable effective exploration of feasible layout configurations before local refinement.

Swap-Based Crossover. The optimization begins with a randomly initialized population $\{\mathbf{x}_1, \ldots, \mathbf{x}_\mu\}$, where μ denotes the number of parent individuals, each individual $\mathbf{x}_i$ represents a complete scene layout as the set of poses for all N objects:

$$\mathbf{x}_i = \{P_j^{(i)} \mid j = 1, \ldots, N\}. \tag{5}$$

At each generation t, a set of λ offspring is generated through crossover and mutation. Traditional crossover can be disruptive in multi-solution settings, as recombination may mix incompatible basins and hinder convergence. We instead use swap-based intra-individual crossover. Given a selected parent $\mathbf{x}_p$, an intermediate offspring $\mathbf{x}'$ is produced by stochastically swapping the poses of two objects within that parent:

$$\mathbf{x}' = \mathrm{Swap}(\mathbf{x}_p; p_{\mathrm{swap}}), \tag{6}$$

where p_{swap} denotes the probability of performing a swap operation. This operation is applied primarily at initialization to enhance population diversity and is also activated when fitness stagnates over several generations, allowing the algorithm to escape local minima and prevent early collapse of the solution space. As such, it serves as a crossover mechanism that promotes both initial exploration and robust recovery from suboptimal configurations.

Center-Guided Mutation. The intermediate offspring $\mathbf{x}'$ then undergoes Gaussian mutation to produce the final offspring $\mathbf{x}_i$. Specifically, each object pose in $\mathbf{x}'$ is perturbed by zero-mean Gaussian noise, followed by an intra-group attraction:

$$\mathbf{x}_i \leftarrow \mathbf{x}_i + w_{\mathrm{cg}}(\mathbf{c}_g - \mathbf{x}_i), \tag{7}$$

where $\mathbf{c}_g$ is the center node of group g, and $w_{\mathrm{cg}} \in [0, 1]$ controls the attraction toward the group center. To mitigate solution collapse and maintain population diversity, a pairwise repulsion term is applied as:

$$\mathbf{x}_i \leftarrow \mathbf{x}_i + \eta \sum_{j \neq i} \frac{\mathbf{x}_i - \mathbf{x}_j}{\|\mathbf{x}_i - \mathbf{x}_j\|^2}. \tag{8}$$

All positions are subsequently clamped to the room bounds B. Center nodes exert stronger influence during optimization, encouraging assets within the same subgraph to form coherent configurations around them.

3.2.2 Local Refinement

After the global exploration step identifies a set of promising candidate layouts, we perform local refinement to further improve solution quality and ensure precise satisfaction of semantic and physical constraints.

Elite Selection. After merging the parent and offspring populations, we retain the top-μ individuals by fitness. Each candidate layout $\mathbf{x_i}$ is evaluated by the fitness function:

$$f(\mathbf{x}_i) = L_{\text{semantic}} + L_{\text{physics}}. \tag{9}$$

where L_{semantic} penalizes violations of inter-object semantic relations and L_{physics} enforces physical feasibility by discouraging collisions and out-of-bound placements. These elites provide initialization for the subsequent refinement step.

Gradient-Based Refinement. The top-μ elite individuals are further refined using gradient descent:

$$\mathcal{X}^* = \arg\min_{\mathbf{x}_i} f(\mathbf{x}_i), \tag{10}$$

This step refines object poses to better satisfy semantic and physical constraints. The resulting configuration $\mathcal{X}^* = \{P_j^* \mid j = 1, \ldots, N\}$ forms the final semantically coherent and physically feasible scene layout.

4 Experiment

4.1 Experimental Setup

Evaluation. To validate our method, we follow the benchmark protocol established in LayoutVLM [30], conducting experiments across **11 room types** (a total of **33 scenes**). We use the same input text instructions as the baseline methods, with assets retrieved via our hierarchical retrieval process, resulting in up to 112 objects per scene.

Evaluation Metrics. We evaluate generated layouts in terms of physical plausibility and semantic coherence with respect to the input textual description. Following LayoutVLM [30], we evaluate generated layouts using Collision-Free Score (CF), In-Boundary Score (IB), Positional Coherency (Pos.), Rotational Coherency (Rot.), and Physically-Grounded Semantic Alignment scores(PSA). Following LayoutVLM [30], we employ GPT-4o as a visual-language evaluator, providing both top-down and side-view renderings alongside the input description to simulate human judgments of plausibility and instruction fidelity. All scores are normalized to the range [0, 100], with higher values indicating stronger semantic-physical consistency.

Baselines. We compare our method against several recent state-of-the-art approaches for open-vocabulary 3D indoor layout generation, including LayoutGPT [9], Holodeck [37], I-Design [3], and LayoutVLM [30].

Methods	Bedroom					Living Room					Dining Room				
	CF	IB	Pos.	Rot.	PSA	CF	IB	Pos.	Rot.	PSA	CF	IB	Pos.	Rot.	PSA
LayoutGPT	100.0	66.7	85.7	85.9	52.2	44.4	11.1	74.7	64.4	9.6	88.9	22.2	76.0	68.9	14.8
Holodeck	88.9	22.2	69.3	67.9	14.1	77.8	0.0	66.3	55.6	0.0	88.9	0.0	38.0	36.6	0.0
I-Design	100.0	77.8	72.1	65.4	51.5	33.3	11.1	62.6	46.7	0.0	88.9	66.7	76.4	66.4	34.8
LayoutVLM	88.9	100.0	82.3	74.9	68.8	22.2	77.8	68.6	54.4	9.6	88.9	100.0	63.4	56.9	51.1
Ours(gpt-4o)	100.0	100.0	83.3	80.0	73.3	100.0	100.0	95.0	62.7	80.0	100.0	100.0	95.0	88.3	80.0
Ours(gpt-4.1)	100.0	100.0	95.0	91.7	**86.7**	100.0	100.0	95.0	91.7	**93.3**	100.0	100.0	85.0	88.3	**86.7**

Methods	Bookstore					Buffet Restaurant					Children Room				
	CF	IB	Pos.	Rot.	PSA	CF	IB	Pos.	Rot.	PSA	CF	IB	Pos.	Rot.	PSA
LayoutGPT	88.9	55.6	80.9	79.4	35.9	100.0	33.3	81.2	83.3	26.9	100.0	0.0	80.9	82.6	0.0
Holodeck	55.6	0.0	65.7	59.0	0.0	77.8	11.1	47.7	42.4	7.4	77.8	22.2	72.7	70.0	18.7
I-Design	66.7	11.1	68.1	69.4	5.2	100.0	55.6	63.5	57.1	35.2	77.8	55.6	78.1	75.1	34.8
LayoutVLM	55.6	100.0	82.0	82.8	49.8	88.9	88.9	74.3	64.8	51.5	100.0	100.0	81.9	88.2	88.5
Ours(gpt-4o)	100.0	100.0	89.3	90.0	**93.3**	100.0	66.7	81.7	70.0	66.7	100.0	100.0	88.3	88.3	80.0
Ours(gpt-4.1)	100.0	66.7	90.0	93.3	53.3	100.0	100.0	88.3	85.0	**86.7**	100.0	100.0	90.0	88.3	**86.7**

Methods	Classroom					Computer Room					Deli				
	CF	IB	Pos.	Rot.	PSA	CF	IB	Pos.	Rot.	PSA	CF	IB	Pos.	Rot.	PSA
LayoutGPT	88.9	0.0	76.3	66.7	0.0	100.0	22.2	87.8	85.2	17.8	88.9	0.0	77.2	77.9	0.0
Holodeck	33.3	0.0	45.2	38.6	0.0	100.0	0.0	66.1	59.7	0.0	88.9	33.3	73.9	63.7	24.4
I-Design	55.6	11.1	50.7	47.0	0.0	88.9	22.2	74.0	70.7	8.9	88.9	22.2	67.8	65.9	10.4
LayoutVLM	77.8	100.0	74.6	68.6	48.3	100.0	88.9	85.4	84.5	77.0	100.0	88.9	83.4	83.4	74.6
Ours(gpt-4o)	100.0	100.0	81.7	88.3	73.3	100.0	100.0	70.0	73.3	66.7	66.7	100.0	81.7	85.0	**83.3**
Ours(gpt-4.1)	100.0	100.0	85.0	78.3	**91.7**	100.0	100.0	96.7	95.0	**96.7**	100.0	66.7	90.0	88.3	60.0

Methods	Florist Shop					Game Room					Average				
	CF	IB	Pos.	Rot.	PSA	CF	IB	Pos.	Rot.	PSA	CF	IB	Pos.	Rot.	PSA
LayoutGPT	66.7	33.3	81.6	80.2	18.3	55.6	22.2	87.0	82.9	6.7	83.8	24.2	80.8	78.0	16.6
Holodeck	63.9	0.0	73.2	64.7	0.0	55.6	22.2	60.7	58.0	0.0	77.8	8.1	62.8	55.6	5.6
I-Design	77.8	0.0	75.5	68.3	0.0	66.7	44.4	62.8	58.9	17.0	76.8	34.3	68.3	62.8	18.0
LayoutVLM	88.9	100.0	83.4	76.4	68.3	88.9	100.0	73.1	70.0	59.5	81.8	94.9	77.5	73.2	58.8
Ours(gpt-4o)	100.0	100.0	85.0	91.7	**93.3**	100.0	100.0	86.7	87.7	93.3	97.0	**97.0**	85.2	82.3	78.5
Ours(gpt-4.1)	100.0	100.0	83.3	75.0	80.0	100.0	100.0	86.7	83.3	**96.7**	**100.0**	93.9	**89.5**	**87.1**	**83.5**

Table 1: Benchmark Performance. Performance across all room types under multiple physical and semantic metrics.

Fig. 4: Qualitative Comparison. Compared to baseline methods, our approach produces layouts with improved semantic and physical consistency.

4.2 Implementation Details

We employ GPT-4o and GPT-4.1 as the VLM backbone for semantic reasoning and constraint generation. For asset retrieval, we use the 3D-FUTURE [12] and Objaverse [6] datasets for floor-level objects, and the HSSD-200 [18] dataset for furniture-level object retrieval. In the evolutionary stage, each scene region is initialized with random object positions. The population size is set to 50, with 5 parent individuals retained per generation. Regions containing fewer than 10 objects are optimized for 30 generations, while larger regions run for 50 generations. The top 5 layouts with the lowest objective energy are selected as initial states for gradient-based refinement, followed by 200 steps of gradient descent. The final layout is the configuration with minimal objective energy.

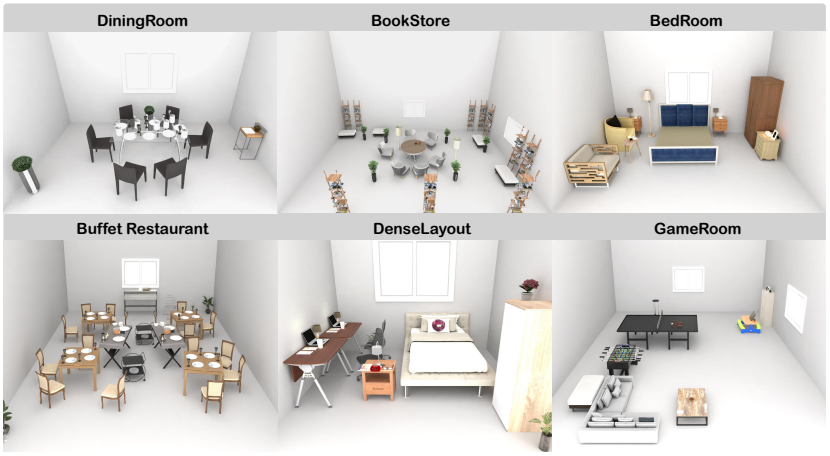

Fig. 5: Results. the results demonstrate that the proposed method can generate dense and collision-free indoor layouts

4.3 Benchmark Performance

Tab. 1 reports comparisons against LayoutGPT, Holodeck, I-Design, and LayoutVLM. Our method achieves higher scores across all metrics (CF, IB, Pos., Rot., PSA) and all room types, indicating stronger semantic coherence and higher physical feasibility across diverse layout scenarios.

The high CF and IB scores mainly arise from the Global Physical Feasibility Search (GPFS) module, which combines global exploration with local refinement to prevent the optimizer from being trapped in local minima, thereby reducing object collisions and placements outside the room. Under this hybrid search strategy, even highly cluttered or entangled initial layouts can be optimized into physically valid configurations, as demonstrated in buffer_restaurant and dense_layout of Fig. 5, where densely packed objects are effectively rearranged without collisions, highlighting GPFS's robustness to challenging initializations.

Improvements in Pos. and Rot. further indicate stronger semantic alignment with textual instructions. This is primarily enabled by our Global Semantic Verification (GSV) module, which explicitly enforces hierarchical and relational semantics at the graph level, preventing inconsistent or implausible object–object relationships early in the process and producing coherent spatial relations. Combined with effective optimization, this mechanism ensures that spatial directives are faithfully realized in the final configuration and promotes functionally coherent arrangements.

The gains in PSA reflect the unified effect of semantic validation and physically grounded optimization. The GSV module guarantees that semantic constraints are globally consistent and feasible, while GPFS ensures that these constraints remain physically realizable throughout optimization. As a result, the generated layouts are simultaneously semantically faithful, functionally plausible, and physically coherent, leading to the highest overall PSA performance.

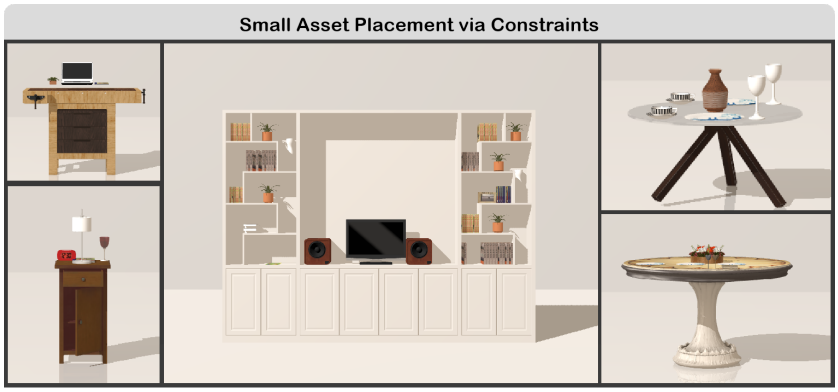

Fig. 6: Furniture surface placement. Small asset placement guided by spatial constraints.

This behavior is consistently observed at both the floor-object and furniture-object levels (see Figs. 5 and 6), where our framework produces correct relational structures with negligible collisions across different object granularities.

Nevertheless, both our GPT-4o and GPT-4.1 variants still exhibit a few failure cases, particularly in the Bookstore and Deli scenes. These scenes are challenging for physical optimization for two main reasons. First, they often involve crowded layouts with relatively large assets and dense object placements on furniture surfaces, making fine-grained collision and boundary optimization difficult. Second, many objects are wall-aligned (e.g., bookshelves) and constrained by near-wall relations, resulting in a highly constrained optimization landscape.

Module	Method	Physics		Semantics		Overall score
		CF	IB	Pos.	Rot.	PSA
GSV	Ours	**100.0**	**100.0**	**91.3**	87.7	**86.7**
	w/o Feasibility	93.3	**100.0**	88.7	**90.3**	82.7
	w/o Completeness	93.3	93.3	89.0	86.7	74.0
	w/o Semantic Consistency	**100.0**	**100.0**	90.0	89.1	85.3
GPFS	w/o GD	13.3	**0.0**	79.7	79.6	0.0
	w/o EA	73.3	**100.0**	86.0	80.5	61.3

Table 2: Ablation study. Quantitative results grouped by global semantic verification (GSV) and global physical feasibility search (GPFS) modules.

4.4 Ablation Study

We conduct ablation studies on a representative subset of scenes, focusing on those that are prone to conflicts or local optima. Specifically, this subset covers five scene types: Bedroom, FloristShop, BuffetRestaurant, LivingRoom, and DiningRoom. These scenes span diverse spatial characteristics and relational complexities, including constrained layouts, functionally coupled object groups,

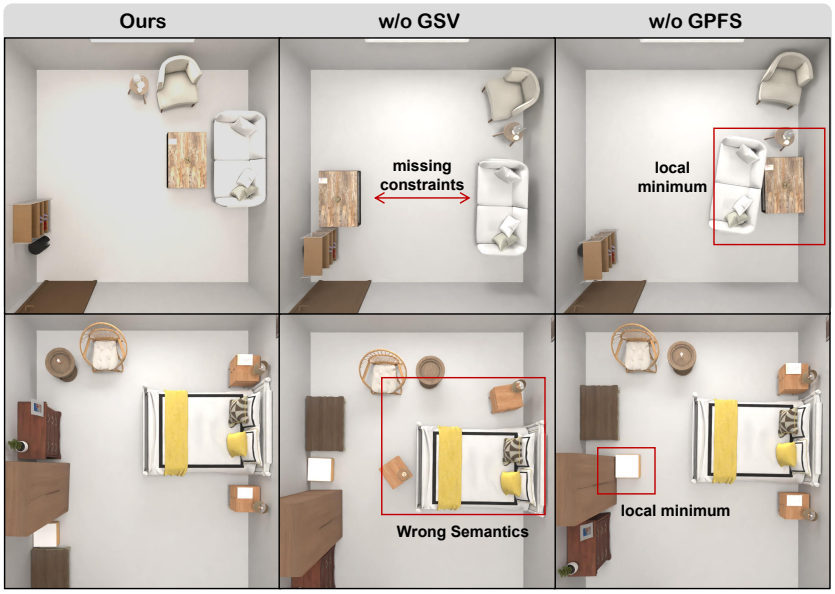

Fig. 7: Ablation study. Comparison of layout results with or without GSV and GPFS module, indicating that GSV ensures semantic coherence and GPFS promotes globally feasible layouts.

and relatively high interaction density. Such diversity ensures that the effects of each component can be clearly and reliably assessed, while avoiding dilution from trivially solvable or conflict-free cases. As shown in Tab. 2, each component distinctly improves semantic quality and physical validity.

Effect of Global Semantic Verification. Removing aspects of the graph module leads to consistent performance degradation. Without feasibility filtering, CF drops from 100.0 to 93.3 and PSA decreases to 82.7, showing that physically infeasible constraints cause unstable interactions during optimization. Removing constraint completeness causes a larger PSA drop (86.7 $\rightarrow$ 74.0), indicating that incomplete relational specifications produce semantically under-constrained layouts. Disabling semantic consistency checking also reduces PSA (to 85.3), highlighting the importance of globally coherent relational structures. These trends are visually confirmed in Fig. 7, where missing or inconsistent constraints cause misaligned object groupings and incorrect relational configurations.

Effect of Global Physical Feasibility Search. The impact of the hybrid evolutionary–gradient optimization is even more pronounced. Removing gradient refinement (w/o GD) causes CF to collapse to 13.3, IB to 0.0 and PSA to 0.0, indicating that evolutionary search alone cannot ensure physically feasible convergence. Conversely, removing evolutionary search (w/o EA) reduces CF to 73.3 and PSA to 61.3, demonstrating that gradient descent without global exploration frequently becomes trapped in poor local minima. As illustrated in Fig. 7, the pure GD variant produces locally entangled layouts, while the absence of GD leaves unresolved collisions. Overall, the ablation results quantitatively and qualitatively confirm that graph-based constraint reasoning ensures

semantic completeness and consistency, while the hybrid evolutionary-gradient strategy is critical for achieving physically valid and globally coherent layouts.

5 Conclusion and Limitation

We revisit VLM-based 3D indoor layout generation from a global reasoning perspective and identify two key limitations: the lack of explicit scene-level semantic verification and the reliance on purely local optimization in a highly non-convex feasible space. To address these issues, we disentangle semantic coherence from physical feasibility through two dedicated mechanisms. Global Semantic Verification (GSV) enforces explicit structural consistency over scene graphs, while Global Physical Feasibility Search (GPFS) combines evolutionary exploration with gradient refinement to enable robust optimization beyond local minima.

Extensive experiments demonstrate state-of-the-art performance in both semantic consistency and physical plausibility. Ablation studies further confirm that explicit graph validation is critical for global coherence, and hybrid global local search is essential for physically valid layout realization. Together, our framework shifts layout generation from local relational fitting to globally consistent and physically grounded reasoning, advancing the reliability of open-vocabulary 3D indoor layout generation.

Despite the promising results, several limitations remain. First, our pipeline still relies on GPT APIs, whose inherent stochasticity affects object category selection, object quantity, and the subsequent evaluation process, leading to variations across different runs. Second, the quality of current 3D asset collections is highly inconsistent. Some assets are oversized, while others may fail to render correctly, which can significantly affect the generated layouts. Although these problematic assets are relatively easy to identify, incorporating a lightweight asset pre-check or filtering stage would substantially improve the robustness and success rate of large-scale scene generation. Reducing pipeline stochasticity and constructing higher-quality 3D asset collections therefore remain important directions for future work.

6 Acknowledgement

This research was supported in part by the National Natural Science Foundation of China (62525115, U2570208) and by the Fundamental Research Funds for the Central Universities (G2026KY05065).

References

1. Aguina-Kang, R., Gumin, M., Han, D.H., Morris, S., Yoo, S.J., Ganeshan, A., Jones, R.K., Wei, Q.A., Fu, K., Ritchie, D.: Open-universe indoor scene generation using llm program synthesis and uncurated object databases. arXiv preprint arXiv:2403.09675 (2024) 4

2. Armeni, I., He, Z.Y., Gwak, J., Zamir, A.R., Fischer, M., Malik, J., Savarese, S.: 3d scene graph: A structure for unified semantics, 3d space, and camera. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 5664–5673 (2019) 4
3. Çelen, A., Han, G., Schindler, K., Van Gool, L., Armeni, I., Obukhov, A., Wang, X.: I-design: Personalized llm interior designer. In: European Conference on Computer Vision. pp. 217–234. Springer (2024) 2, 4, 10
4. Chang, A., Savva, M., Manning, C.D.: Learning spatial knowledge for text to 3d scene generation. In: Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP). pp. 2028–2038 (2014) 4
5. Chang, A.X., Eric, M., Savva, M., Manning, C.D.: Sceneseer: 3d scene design with natural language. arXiv preprint arXiv:1703.00050 (2017) 4
6. Deitke, M., Schwenk, D., Salvador, J., Weihs, L., Michel, O., VanderBilt, E., Schmidt, L., Ehsani, K., Kembhavi, A., Farhadi, A.: Objaverse: A universe of annotated 3d objects. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 13142–13153 (2023) 11
7. Deng, W., Qi, M., Ma, H.: Global-local tree search in vlms for 3d indoor scene generation. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 8975–8984 (2025) 2, 4
8. Dhamo, H., Manhardt, F., Navab, N., Tombari, F.: Graph-to-3d: End-to-end generation and manipulation of 3d scenes using scene graphs. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 16352–16361 (2021) 4
9. Feng, W., Zhu, W., Fu, T.j., Jampani, V., Akula, A., He, X., Basu, S., Wang, X.E., Wang, W.Y.: Layoutgpt: Compositional visual planning and generation with large language models. Advances in Neural Information Processing Systems **36**, 18225–18250 (2023) 2, 4, 10
10. Fisher, M., Ritchie, D., Savva, M., Funkhouser, T., Hanrahan, P.: Example-based synthesis of 3d object arrangements. ACM Transactions on Graphics (TOG) **31**(6), 1–11 (2012) 4
11. Fisher, M., Savva, M., Hanrahan, P.: Characterizing structural relationships in scenes using graph kernels. In: ACM SIGGRAPH 2011 papers, pp. 1–12. ACM (2011) 4
12. Fu, H., Jia, R., Gao, L., et al.: 3d-future: 3d furniture shape with texture. International Journal of Computer Vision **129**(12), 3313–3337 (2021) 11
13. Fu, H., Cai, B., Gao, L., Zhang, L.X., Wang, J., Li, C., Zeng, Q., Sun, C., Jia, R., Zhao, B., et al.: 3d-front: 3d furnished rooms with layouts and semantics. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 10933–10942 (2021) 4
14. Fu, Q., Chen, X., Wang, X., Wen, S., Zhou, B., Fu, H.: Adaptive synthesis of indoor scenes via activity-associated object relation graphs. ACM Transactions on Graphics (TOG) **36**(6), 1–13 (2017) 4
15. Fu, R., Wen, Z., Liu, Z., Sridhar, S.: Anyhome: Open-vocabulary generation of structured and textured 3d homes. In: European Conference on Computer Vision. pp. 52–70. Springer (2024) 2, 4
16. Hu, S., Arroyo, D.M., Debats, S., Manhardt, F., Carlone, L., Tombari, F.: Mixed diffusion for 3d indoor scene synthesis. arXiv preprint arXiv:2405.21066 (2024) 4
17. Johnson, J., Gupta, A., Fei-Fei, L.: Image generation from scene graphs. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 1219–1228 (2018) 4

18. Khanna, M., Mao, Y., Jiang, H., Haresh, S., Shacklett, B., Batra, D., Clegg, A., Undersander, E., Chang, A.X., Savva, M.: Habitat synthetic scenes dataset (hssd-200): An analysis of 3d scene scale and realism tradeoffs for objectgoal navigation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 16384–16393 (2024) 4, 11
19. Kingma, D.P., Welling, M.: Auto-encoding variational bayes. arXiv preprint arXiv:1312.6114 (2013) 4
20. Li, M., Patil, A.G., Xu, K., Chaudhuri, S., Khan, O., Shamir, A., Tu, C., Chen, B., Cohen-Or, D., Zhang, H.: Grains: Generative recursive autoencoders for indoor scenes. ACM Transactions on Graphics (TOG) **38**(2), 1–16 (2019) 4
21. Lin, C., Mu, Y.: Instructscene: Instruction-driven 3d indoor scene synthesis with semantic graph prior. arXiv preprint arXiv:2402.04717 (2024) 4
22. Ling, L., Lin, C.H., Lin, T.Y., Ding, Y., Zeng, Y., Sheng, Y., Ge, Y., Liu, M.Y., Bera, A., Li, Z.: Scenethesis: A language and vision agentic framework for 3d scene generation. arXiv preprint arXiv:2505.02836 (2025) 2, 4
23. Littlefair, G., Dutt, N.S., Mitra, N.J.: Flairgpt: Repurposing llms for interior designs. In: Computer Graphics Forum. p. e70036. Wiley Online Library (2025) 4
24. Liu, T., Chaudhuri, S., Kim, V.G., Huang, Q., Mitra, N.J., Funkhouser, T.: Creating consistent scene graphs using a probabilistic grammar. ACM Transactions on Graphics (TOG) **33**(6), 1–12 (2014) 4
25. Luo, A., Zhang, Z., Wu, J., Tenenbaum, J.B.: End-to-end optimization of scene layout. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 3754–3763 (2020) 4
26. Merrell, P., Schkufza, E., Li, Z., Agrawala, M., Koltun, V.: Interactive furniture layout using interior design guidelines. ACM transactions on graphics (TOG) **30**(4), 1–10 (2011) 4
27. Purkait, P., Zach, C., Reid, I.: Sg-vae: Scene grammar variational autoencoder to generate new indoor scenes. In: European Conference on Computer Vision. pp. 155–171. Springer (2020) 4
28. Sohn, K., Lee, H., Yan, X.: Learning structured output representation using deep conditional generative models. Advances in Neural Information Processing Systems **28** (2015) 4
29. Su, C., Fu, Y., Hu, Z., Yang, J., Hanji, P., Wang, S., Zhao, X., Öztireli, C., Zhong, F.: Chord: Generation of collision-free, house-scale, and organized digital twins for 3d indoor scenes with controllable floor plans and optimal layouts. arXiv preprint arXiv:2503.11958 (2025) 4
30. Sun, F.Y., Liu, W., Gu, S., Lim, D., Bhat, G., Tombari, F., Li, M., Haber, N., Wu, J.: Layoutvlm: Differentiable optimization of 3d layout via vision-language models. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 29469–29478 (2025) 2, 4, 10
31. Sun, Q., Zhou, H., Zhou, W., Li, L., Li, H.: Forest2seq: Revitalizing order prior for sequential indoor scene synthesis. In: European Conference on Computer Vision. pp. 251–268. Springer (2024) 4
32. Tang, J., Nie, Y., Markhasin, L., Dai, A., Thies, J., Nießner, M.: Diffuscene: Denoising diffusion models for generative indoor scene synthesis. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 20507–20518 (2024) 4
33. Wald, J., Dhamo, H., Navab, N., Tombari, F.: Learning 3d semantic scene graphs from 3d indoor reconstructions. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 3961–3970 (2020) 4

34. Wang, X., Yeshwanth, C., Nießner, M.: Sceneformer: Indoor scene generation with transformers. In: 2021 International Conference on 3D Vision (3DV). pp. 106–115. IEEE (2021) 4
35. Wu, Z., Feng, M., Wang, Y., Xie, H., Dong, W., Miao, B., Mian, A.: External knowledge enhanced 3d scene generation from sketch. In: European Conference on Computer Vision. pp. 286–304. Springer (2024) 4
36. Yang, Y., Jia, B., Zhi, P., Huang, S.: Physcene: Physically interactable 3d scene synthesis for embodied ai. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 16262–16272 (2024) 4
37. Yang, Y., Sun, F.Y., Weihs, L., VanderBilt, E., Herrasti, A., Han, W., Wu, J., Haber, N., Krishna, R., Liu, L., et al.: Holodeck: Language guided generation of 3d embodied ai environments. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 16227–16237 (2024) 2, 4, 10
38. Yeh, Y.T., Yang, L., Watson, M., Goodman, N.D., Hanrahan, P.: Synthesizing open worlds with constraints using locally annealed reversible jump mcmc. ACM Transactions on Graphics (TOG) **31**(4), 1–11 (2012) 4
39. Zhao, Y., Zhu, S.C.: Image parsing with stochastic scene grammar. Advances in Neural Information Processing Systems **24** (2011) 4
40. Zhou, Y., While, Z., Kalogerakis, E.: Scenegraphnet: Neural message passing for 3d indoor scene augmentation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 7384–7392 (2019) 4