HSM: Hierarchical Scene Motifs for Multi-Scale Indoor Scene Generation

Hou In Derek Pun[1] Hou In Ivan Tam[1] Austin T. Wang[1] Xiaoliang Huo[1]
Angel X. Chang[1,2] Manolis Savva[1]
[1]Simon Fraser University [2]Alberta Machine Intelligence Institute (Amii)

https://3dlg-hcvc.github.io/hsm/

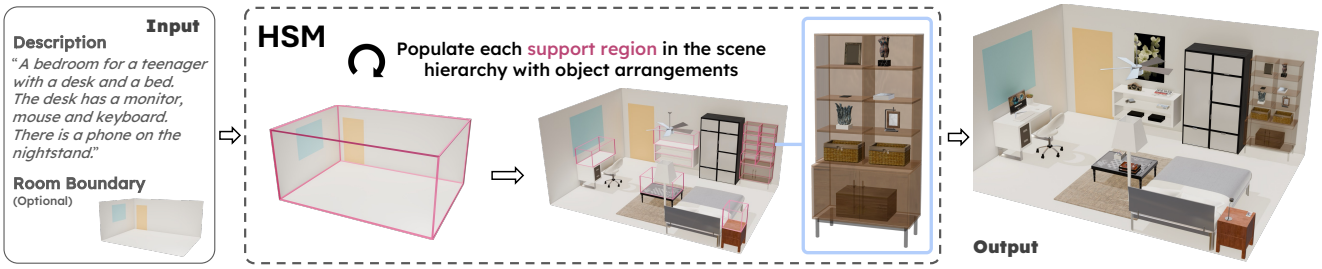

Figure 1. **Overview.** Given a room description and optional room boundary as input, HSM decomposes indoor scenes hierarchically and identifies valid support regions (highlighted in pink boxes) at each level of the hierarchy. The system then populates these regions by generating and optimizing object arrangements in a unified manner across scales, generating scenes with dense object arrangements.

Abstract

Despite advances in indoor 3D scene layout generation, synthesizing scenes with dense object arrangements remains challenging. Existing methods focus on large furniture while neglecting smaller objects, resulting in unrealistically empty scenes. Those that place small objects typically do not honor arrangement specifications, resulting in largely random placement not following the text description. We present Hierarchical Scene Motifs (HSM): a hierarchical framework for indoor scene generation with dense object arrangements across spatial scales. Indoor scenes are inherently hierarchical, with surfaces supporting objects at different scales, from large furniture on floors to smaller objects on tables and shelves. HSM embraces this hierarchy and exploits recurring cross-scale spatial patterns to generate complex and realistic scenes in a unified manner. Our experiments show that HSM outperforms existing methods by generating scenes that better conform to user input across room types and spatial configurations.

1. Introduction

Digital 3D indoor scenes are widely used in domains such as gaming, interior design, virtual training, and simulation for embodied AI and robotics. Consequently, efficient and controllable generation of realistic indoor scenes has been a long-standing research problem. While recent advances have improved scene synthesis through various approaches, most efforts have focused on arranging large furniture pieces, with less attention given to arrangement of smaller objects, such as computer peripherals and place settings. These objects are often treated as an afterthought, placed randomly or based on predefined rules, limiting realism of the generated scenes and downstream applications.

Small objects present a unique challenge in scene generation due to their dependence on larger furniture for support. For instance, a computer monitor is typically placed on a desk, while books are arranged on shelves. These objects must be positioned to respect the physical constraints of their supporting furniture while adhering to user specifications. However, existing methods rarely capture this hierarchical dependency, leading to sparse and unrealistic small object placements. In addition, the lack of large-scale scene datasets that encompass both furniture-level and small object arrangements makes it non-trivial to train models capable of generating complex hierarchical scenes.

In this work, we introduce Hierarchical Scene Motifs (HSM), a hierarchical approach to generating indoor scenes with densely populated objects. We employ a unified approach to placing objects, guided by both text-explicit and implicit relationships. This spans scales from room-level furniture layouts to the placement of small objects on furniture. At each level, HSM first identifies valid *support regions* (i.e., regions in which objects can be placed) and then generates compositional object arrangements within them. This transforms the scene generation problem into equivalent subproblems of arranging objects on surfaces, while

ensuring that the generated scene aligns with user intent.

Our key insight is that object arrangements in indoor scenes exhibit recurring spatial patterns across scales. For instance, dining chairs surrounding a circular table and place settings arranged around a centerpiece follow a common circular pattern. We refer to such spatial patterns as *motifs* — fundamental and reusable object placement structures that are ubiquitous in indoor environments. These motifs can be efficiently learned from a few examples and applied across different scales to generate realistic and coherent object arrangements. By composing these motifs into *scene motifs*, we enable the generation of complex arrangements across the scene hierarchy under a unified framework, facilitating structured and scalable scene generation.

We demonstrate that HSM generates realistic indoor scenes with dense object arrangements that align with user expectations and adhere to physical constraints. In particular, our method excels at plausible and text-consistent small object placement, producing coherent and detailed arrangements that enhance scene realism (e.g. neatly stacked books on shelves and well-organized stationery and accessories on desks). We evaluate HSM against state-of-the-art scene generation methods and show that it more effectively handles complex object arrangements, generating scenes with more realistic object placements and hierarchical structures. Our results highlight the importance of considering hierarchical relationships in indoor scenes and demonstrate that scene motifs are a powerful mechanism for generating high-quality scenes. In summary:

- We present HSM, a hierarchical framework for indoor scene generation that identifies support regions and generates structured object arrangements across different scales in a unified manner, producing realistic scenes.
- We introduce scene motifs, compositional structures that capture recurring spatial patterns in indoor environments, enabling the generation of complex arrangements.
- We show HSM produces realistic scenes with dense object arrangements that adhere to physical constraints and better align with user intent compared to prior work.

2. Related Work

Indoor scene generation. Early approaches primarily relied on rule-based [8, 9, 36, 49, 56, 80] or data-driven [7, 16, 17, 20, 26, 32, 34, 55, 61, 92] methods. The advent of deep learning led to a shift towards learning-based methods [12, 27, 38–40, 50–52, 54, 59, 65, 69, 71, 74, 76, 78, 81, 84, 85, 88, 89, 94]. These methods take various forms of input, including text descriptions, scene graphs, and images, to generate 3D scenes. Of particular interest are methods that generate scenes from text [40, 69, 84, 85], with recent works increasingly incorporating large language models (LLMs) into the process [3, 5, 6, 11, 13, 15, 21, 28, 42–44, 46, 47, 63, 64, 66, 70, 75, 77, 79, 82, 83]. While these methods enable text-conditioned scene generation, most focus solely on arranging large furniture, neglecting the small objects that are ubiquitous in indoor environments. More recent works have begun integrating small objects into the scene generation process [13, 21, 24, 30, 42, 43, 45, 46, 53, 57, 75, 77, 83, 91, 93, 95]. Additionally, some methods attempt to place small objects in existing scenes [1, 29, 48, 87]. However, these methods often treat small objects in a simplified or specialized manner, such as using random placement. This limits both the diversity and controllability of their arrangements. Our work introduces a unified hierarchical framework for scene generation that conditions object placement on precise language descriptions, capturing object relationships at all scales.

Hierarchical scene generation. Exploiting the hierarchical nature of scene generation has been studied for many years [13, 22, 38, 43, 50, 63, 66, 72, 85, 86]. Among recent works, Architect [75] uses a parallel approach to generate large and small objects but relies on 2D inpainting and thus suffers from 3D inconsistencies. Furthermore, Architect does not generate scenes with precise control over object arrangements. SceneFunctioner [44] groups objects and parses relationships within each group, similar to HSM, but their approach realizes relationships using LLM-predicted anchor rules rather than recurring motifs and does not handle small objects. While most such papers primarily capture the semantic hierarchy of objects in scenes, we further capture the repetition of object relationships at each scale, in that small objects can be functionally and spatially arranged equivalently to large objects.

Support region prediction. To place small objects in scenes, it is first beneficial to determine suitable placement regions. Various methods have been proposed to predict support surfaces in indoor environments from 2D images [23, 25, 58, 60, 62, 77]. While these methods are effective, extending them to synthetic 3D scenes is non-trivial, as they rely on sight lines to surface geometry, necessitating camera viewpoint selection and introducing occlusion issues. In contrast, our approach employs geometric reasoning-based support region extraction to identify valid support surfaces on objects, enabling dense and precise small object arrangements in generated scenes.

3. Method

Given a text description T of an indoor scene and an optional room boundary as a list of vertices as input, HSM generates the scene iteratively through a unified hierarchical framework, as shown in Fig. 2. It first uses a vision language models (VLM) to extract a room type and decompose T into a list of required objects at each scale (Sec. 3.2). The scene is then constructed through three key steps at each hierarchical level: support region extraction (Sec. 3.3), scene motif generation (Sec. 3.4), and layout optimization

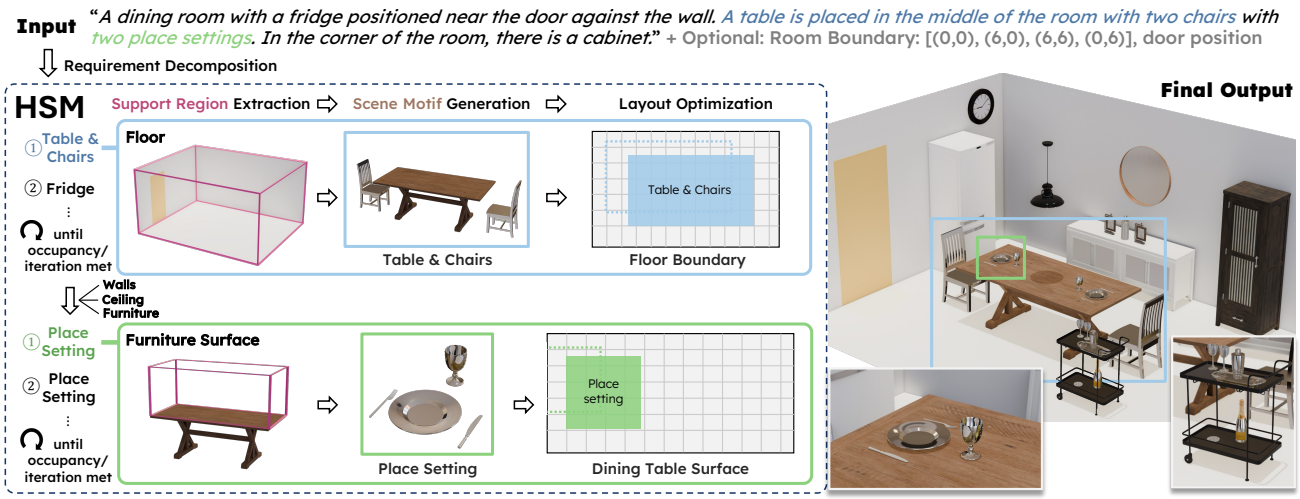

Figure 2. **HSM framework overview.** Given an input text description and an optional room boundary, HSM decomposes the input into requirements at different scales, and generates the scene through a unified three-stage framework: 1) Extract support regions for object placements; 2) Generate appropriate scene motifs for each region; and 3) Optimize scene motif placements within each region. These steps are repeated across scales to generate a scene that aligns to the input with dense small object placements.

(Sec. 3.5). These steps are iteratively applied to an initially empty scene, first placing room-level furniture and then arranging small objects on furniture surfaces. After placing the initial objects, if the occupancy is below a threshold t_{occ} and the number of iterations below t_{iter}, a VLM is prompted for additional objects to add to the scene beyond the input description, conditioned based on the predicted room type. Throughout this process, HSM leverages a library of learned motifs to compose complex spatial relationships as scene motifs (Sec. 3.1), which are instantiated to produce physically valid object arrangements in the scene.

3.1. Scene Motifs

Inspired by SceneMotifCoder (SMC) [67], we define a *motif* as an atomic spatial pattern between objects that can be extracted from a few examples and used as a template for generating new arrangements. For instance, by analyzing arrangements such as "a stack of books" and "a stack of plates", we can learn a `stack` motif that captures the vertical alignment pattern and apply it to new objects to generate novel arrangements. Such motifs are applicable across different scales, from room-scale furniture to small objects.

We extend this concept to capture more complex arrangement structures through *scene motifs*. A scene motif is a composition of one or more learned motifs that represents a set of spatial relationships between objects. For example, a scene motif for a desk setup may consist of a `stack` motif for the books on the desk, a `left_of` motif for placing a lamp to the left of the books, and an `in_front` motif for positioning a laptop in front of the books. By leveraging these compositions, scene motifs dynamically capture a broader range of spatial relationships, enabling the generation of more complex and diverse object arrangements. We describe the generation process of scene motifs in Sec. 3.4.

As in SMC, motifs are implemented as visual programs that encode spatial relationships between objects using programmatic constructs. Each motif is defined as a Python function that takes specific arguments to generate object arrangements when executed. For example, a `stack` motif may be implemented as a function that iteratively stacks objects vertically using a `for` loop, with parameters for count and object type (e.g., `stack(4, book)` to generate "a stack of four books"). HSM assumes a pre-learned library of motifs exists, and uses it to dynamically composes scene motifs. See Appendix A.1 for more details.

3.2. Input Description Requirement Decomposition

Given the input text description, HSM first uses a VLM to extract the room type and an initial list of objects to include in the scene. Objects are grouped by the VLM according to their supporting architectural elements or objects. This breaks the generation task into smaller, more manageable subproblems that can be addressed within a unified framework while minimizing the risk of losing details in subsequent steps (see Fig. 2). For each object, we extract its quantity, appearance, and dimensions from the input description, inferring plausible values when unspecified. If the input lacks a room boundary, we prompt the VLM to generate one including door, window placement and room height. See Appendix D for the VLM prompts used.

3.3. Support Region Extraction

To place objects, we first identify suitable placement areas, referred to as *support regions* S. Support regions are surfaces capable of accommodating object placement. At the room level, these include the floor, walls, and ceiling, while

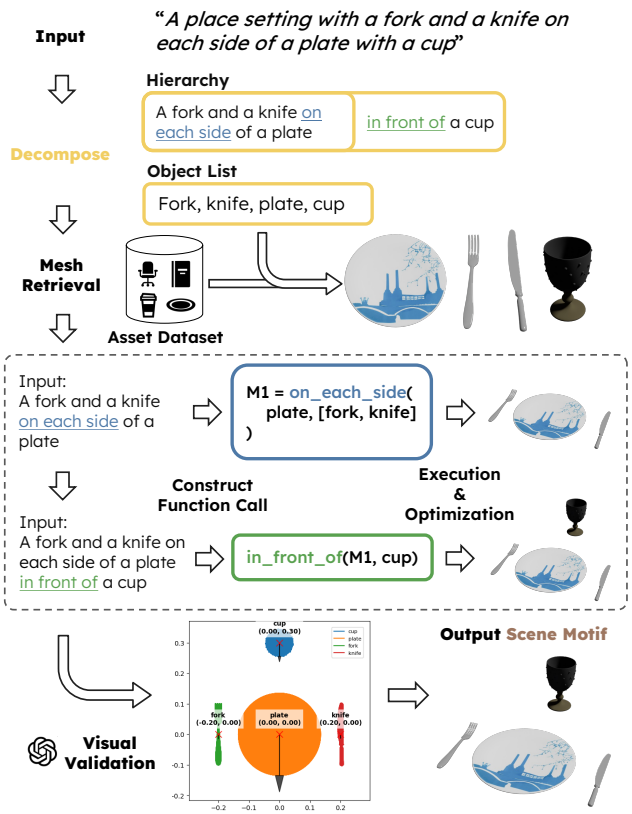

Figure 3. **Scene motif generation process.** An input description is first decomposed into a hierarchy of motifs. We then retrieve the corresponding 3D assets and generate the scene motifs iteratively, starting from the innermost motif (`on_each_side`) and expanding to the outermost motif of the hierarchy (`in_front_of`). The generated scene motif is visually validated with a VLM.

at the furniture level, these correspond to horizontal surfaces such as tabletops and shelves. We parametrize each $s_i \in S$ by a sub-mesh and height clearance h_i. Wall support regions are extracted by projecting scene motifs within t_{wall} onto the wall to exclude blocked areas.

While room-level support regions can be extracted from the room boundary, furniture-level support regions require more complex analysis due to the geometric intricacies of furniture surfaces (e.g., multi-level shelves). We first identify horizontal and vertical surfaces on a furniture mesh with n vertices and p triangle faces $F \in \{1, \ldots, n\}^{p \times 3}$, parameterizing each surface as a mesh subset of F. Inspired by [33], we cluster F into planar surfaces by seeding each cluster c_j with the largest unclustered f_{j_0} by area and adding f_i to $c_j = \{j_k \mid k = 1, \ldots, |c_j|\}$ if its normal is in the same direction as the normals of the first triangle and adjacent triangle. Formally, we check if $n_i \cdot n_{j_0} \geq t_{\text{norm}}$ and $n_i \cdot n_{j_{\text{adj}}} \geq t_{\text{adj}}$, where n_i is the normal of f_i and $f_{j_{\text{adj}}}$ shares an edge with f_i. While building up a cluster, we traverse adjacent faces f_i by area (from largest to smallest).

We identify horizontal and vertical surfaces by fitting a plane to each c_i and thresholding the vertical component of the normal. To ensure functional utility, we compute h_i between horizontal surfaces and discard those which have low clearance ($h_i < t_{\text{clear}}$). For top surfaces without a ceiling, we assign a default clearance of h_{top}, in order to put a reasonable limit on the size of small objects placed on top of furniture. Finally, horizontal surfaces are split by intersecting vertical surfaces thicker than t_{seg}, ensuring continuity within each region. Fig. 4 illustrates this process for a shelf unit. See Appendix A.2 for implementation details.

3.4. Scene Motif Generation

We populate S by generating *scene motifs* M. Given the objects O_i selected for each $m_i \in M$ and the overall room description, HSM prompts a VLM to produce a natural language description of the object layout. The VLM then uses this description to group objects into M based on spatial and functional relationships. For example, given the description, "a room with six chairs around a table and a sofa facing a TV," in the floor support region, the system groups the chairs around the table and the sofa facing the TV.

Each O_i is decomposed into a sequence of one or more motifs m_{ik} that capture the spatial relationships between objects. We first use a VLM to identify a primary motif that serves as the base object arrangement, relative to which the rest of the objects in O_i can be placed. Fig. 3 demonstrates a place setting example, in which the utensil-plate arrangement forms the primary motif, with the cup positioned relative to it. The VLM then predicts an arrangement of the remaining objects around this primary motif based on their spatial and functional relationships. The result is a hierarchical sequence of motifs that define the object arrangements within m_i, validated by the VLM against the input description to ensure completeness and coherence. If validation fails, the reason is included in retry attempts.

The scene motif is instantiated by creating the primary motif and progressively traversing the hierarchy to model additional objects. We first retrieve a suitable 3D mesh for each object from a database based on its category, appearance description, and dimensions (see Appendix A.4 for details on the retrieval process). For each motif, we use a VLM to select a matching program from a pre-learned library of motifs and generate an appropriate function call to instantiate it. This function is then executed to generate an object arrangement using the retrieved meshes. The resulting arrangement is subsequently used as input for the next motif in the hierarchy, iteratively constructing the scene motif until all objects are placed. To ensure physical plausibility between objects, we apply the same spatial optimization used in SMC—resolve collisions, move objects closer, and simulate gravity—while respecting the hierarchy. Motifs are treated as single units during optimization.

To ensure the generated scene motif aligns with the in-

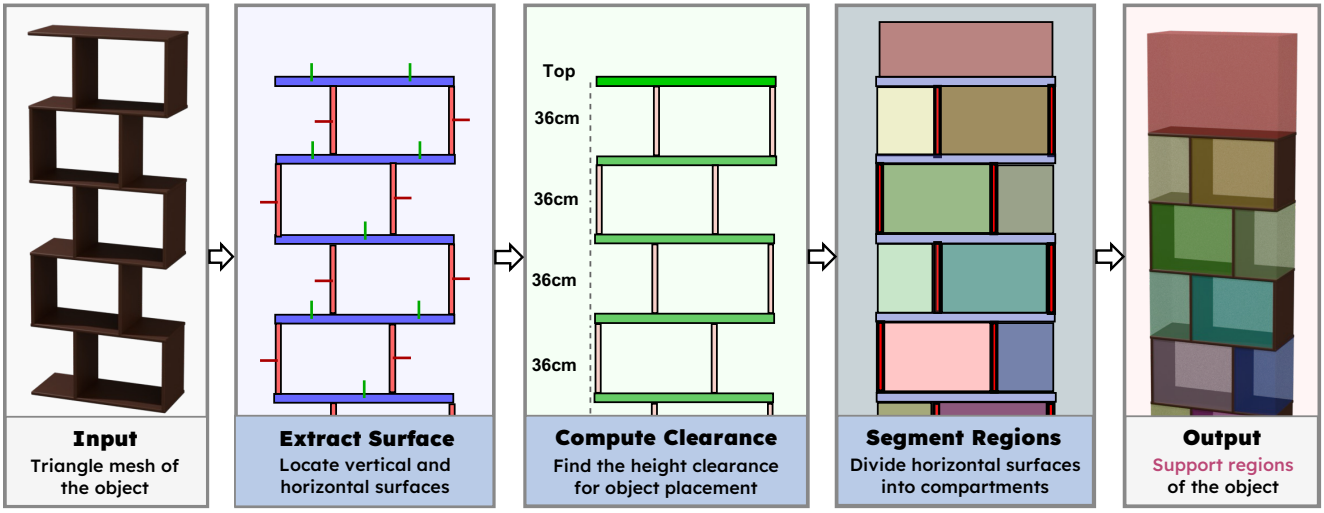

Figure 4. **Support region extraction.** Given a triangle mesh of the object, such as the example shelf unit, we first extract vertical and horizontal surfaces. We then compute the height clearance for each horizontal surface and use the vertical surfaces to segment them into compartments. The result is a set of support regions that can be populated with objects.

put description, we employ a verification process after its generation. We generate top-down and front orthographic projections of the scene motif and prompt a VLM to validate the arrangement against the input description based on these projections. If validation fails, the failure reason is used to guide the VLM in retrying the generation process.

3.5. Layout Optimization

The final step in populating a support region is placing the generated scene motifs within it. This is achieved through a three-stage process. In the first stage, we provide an orthogonal projection of the support region, the bounding boxes of the scene motifs, and the sub-scene description to a VLM, prompting it to suggest initial placements and determine whether the scene motifs should be wall-aligned. The VLM is instructed to consider both explicit constraints from the input description and implicit constraints derived from common arrangements and usage patterns to generate a reasonable initial layout. The second stage refines the layout using an optimization solver. Inspired by Holodeck [83], we employ a grid-based depth-first search (DFS) solver. Starting with the scene motif that has the largest footprint, the solver iteratively refines placements by enforcing the following constraints: 1) scene motifs must be placed within the support region, 2) wall-aligned scene motifs must have their back against a wall and face into the room, and 3) there should be no overlap between scene motifs. The solver returns the first valid layout or otherwise the initial placement if a set time limit is exceeded. Finally, we apply a scene-level spatial optimizer to refine placements by eliminating mesh collisions and ensuring valid support. For each scene motif, we resolve collisions and support issues using relationship-aware rules and raycasting, making minimal adjustments while preserving hierarchy and layout. We provide the details for the layout optimization in Appendix A.3.

These three steps — support region extraction, scene motif generation, and layout optimization — are repeated at each hierarchy level, from room-scale furniture arrangements to fine-grained small object placements on furniture surfaces. This unified framework ensures a structured and efficient generation process, preserving spatial coherence and physical validity across scales to produce complex indoor scenes with realistic object arrangements.

4. Experimental Setup

4.1. Scene Generation

We evaluate HSM on the task of text-conditioned indoor scene generation using 3D assets from the Habitat Synthetic Scenes Dataset (HSSD-200) [35]. We select HSSD as it comprises 211 synthetic indoor scenes with a diverse collection of high-quality 3D assets — both furniture and small objects — making it well-suited for learning motifs and retrieving objects. We use *gpt-4o-2024-08-06* [2] for all VLM usage. For our experiments, we set $t_{\text{occ}} = 0.3$ and $t_{\text{iter}} = 2$ for floor support region, with $t_{\text{occ}} = 0.5$ and $t_{\text{iter}} = 1$ for the others, $t_{\text{wall}} = 1.5\,\text{m}$ for wall projection.

Baselines. We compare HSM against four recent scene generation methods: LayoutGPT [14], InstructScene [40], LayoutVLM [64] and Holodeck [83]. LayoutGPT is the first to leverage an LLM for indoor scene generation. It uses scenes from the 3D-FRONT dataset [18, 19] as in-context examples and prompts an LLM to generate layouts in CSS format. InstructScene employs a graph diffusion model with a semantic scene graph to generate 3D object layouts from text descriptions. LayoutVLM is a framework

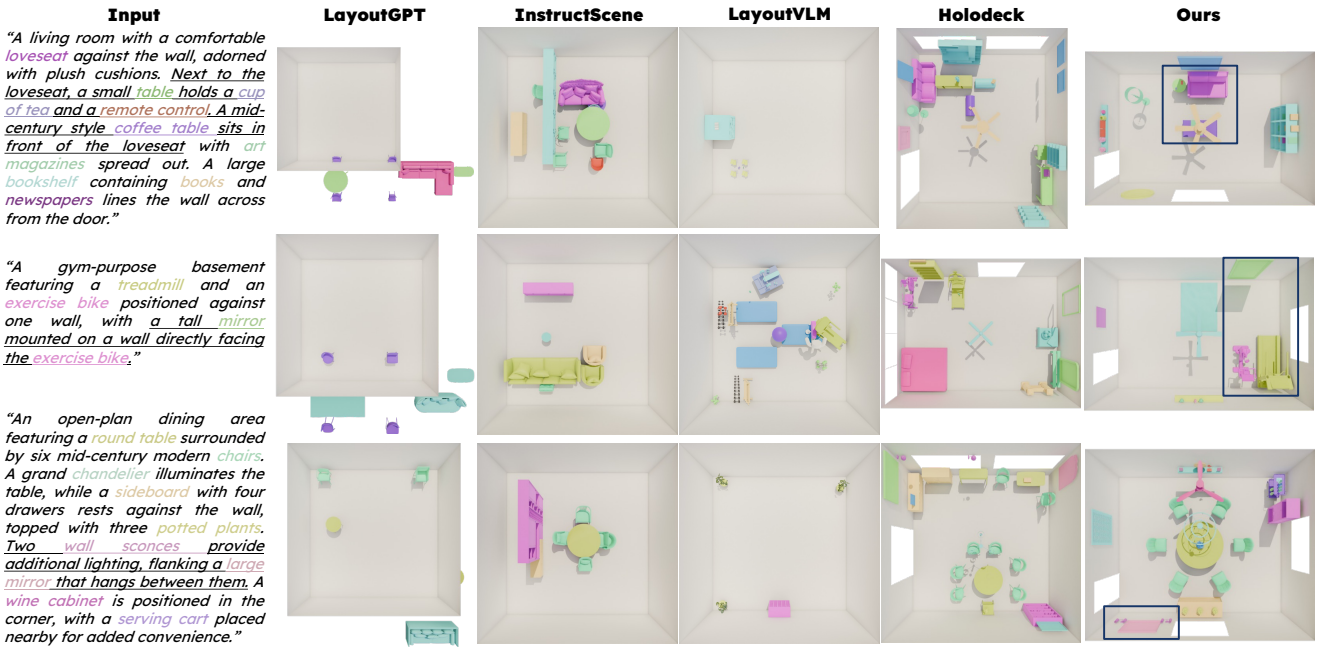

Figure 5. **Qualitative comparisons at the scene level.** Objects and spatial relationships in the input text are highlighted with colors and underlines, and spatial relationships are emphasized using boxes. HSM produces more coherent spatial arrangements and is better aligned to the input compared to existing approaches.

that uses VLM with differentiable optimization to generate 3D object layouts from text descriptions. Holodeck is a comprehensive system that integrates LLM-based generation with optimization steps to produce room boundaries and object placements for embodied AI simulations. We follow the original implementations of these methods and use their respective object databases for object retrieval: 3D-FRONT [18, 19] for LayoutGPT and InstructScene, and Objaverse [10] for LayoutVLM and Holodeck. Since only Holodeck can generate architectural elements, we provide a standardized 6 m × 6 m square floor plan with walls as input for the other methods to ensure fair comparison.

Input Text Descriptions. For all methods, we use the first 100 text descriptions from SceneEval-500, a dataset introduced by SceneEval [68], a recent framework for evaluating text-to-indoor scene generation methods. The descriptions vary in complexity (40 easy, 40 medium, 20 hard) and encompass a wide range of scene types and object arrangements. Each scene description is paired with human-annotated ground truth scene properties (e.g., object counts, attributes, and spatial relationships) to assess how well the text match the generated scenes.

Metrics. We use four fidelity metrics from SceneEval to assess how well the generated scenes align with the input text description: Object Count (**CNT**), Object Attribute (**ATR**), Object-Object Relationship (**OOR**), and Object-Architecture Relationship (**OAR**), using annotations from SceneEval-100 as ground truth. To evaluate object placement, we use five plausibility metrics capturing implicit human expectations: Object Collision (**COL**), Object Support (**SUP**), Scene Navigability (**NAV**), Object Accessibility (**ACC**), and Object Out-of-Bound (**OOB**).

We also conduct a user perceptual study to compare generated scenes from HSM and Holodeck. We select Holodeck as the baseline for comparison due to its strong generation performance and ability to place small objects. The study consists of two parts: scene-level evaluation and small object-level evaluation. At the scene level, we randomly select 25 scene pairs generated by each method for the same input description. Given top-down renderings, participants assess which scene better aligns with the input text descriptions (**Fidelity**) and exhibits more physically realistic object placements (**Plausibility**). At the small object level, we randomly select 30 pairs of furniture populated with small objects and provide close-up renderings. Each pair originates from the same scene description, and participants evaluate the placements based on fidelity and plausibility. The study was conducted with 25 participants, and we report the percentage of participant preferences for each method across both evaluation criteria and levels. See Appendix C for the user study instructions.

We also evaluate alignment between generated scenes and input descriptions by rendering top-down views and computing text-image similarity using BLIP-2 [37], LongCLIP [90] and VQAScore [41]. Similarly, we report alignment between small object arrangements and the corresponding text descriptions using the same metrics and the set of 30 populated furniture from the user study. For meth-

	Text-Image Score			SceneEval Fidelity				SceneEval Plausbility						Avg. #Obj per Scene
	↑ BLIP	↑ CLIP	↑ VQA	↑ CNT$_{\%}$	↑ ATR$_{\%}$	↑ OOR$_{\%}$	↑ OAR$_{\%}$	↓ COL$_{ob\%}$	↓ COL$_{sc\%}$	↑ SUP$_{\%}$	↑ NAV$_{\%}$	↑ ACC$_{\%}$	↓ OOB$_{\%}$	
LayoutGPT	0.0613	0.1670	0.2964	19.54	18.98	2.87	5.24	**12.96**	**30.00**	28.24	**100.00**	47.29	73.11	5.17
InstructScene	0.0845	0.1681	0.4082	25.48	22.26	11.17	10.48	51.18	84.00	75.09	99.53	77.30	22.92	8.07
LayoutVLM	0.0857	0.1612	0.3268	41.19	22.26	8.60	23.29	36.09	69.00	67.96	98.75	85.91	4.14	11.36
Holodeck	0.1230	0.1820	0.5549	44.64	39.42	20.92	49.60	17.32	73.00	62.12	99.45	**90.55**	**1.30**	24.71
HSM (ours)	**0.1748**	**0.1841**	**0.5627**	**61.30**	**59.49**	**40.40**	**70.28**	16.42	61.00	**85.44**	98.97	86.80	2.13	20.65

Table 1. **Evaluation with text-image scores and SceneEval metrics.** Overall, HSM outperforms prior work along metrics measuring the fidelity of object placements relative to the input text (Text-Image Score and SceneEval Fidelity metrics). For plausibility metrics, HSM has the highest support rates. LayoutGPT and InstructScene have better collision and navigation but that is due to placing far fewer objects (last column), most of which are out of the scene (OOB). Bold indicates highest results, underlined denotes second highest.

ods that do not generate small objects, we select the best-matching furniture available for evaluation. The score is set to 0 if no matching furniture of the same category is found.

4.2. Support Region Extraction

As support region extraction from 3D furniture meshes is a key component of our approach and an essential step for placing small objects, we evaluate this step in isolation. To this end, we manually annotate 100 furniture items (e.g., tables, shelves, sofas) from the HSSD-200 dataset [35] with ground truth support regions. Each annotated region is represented by a set of faces defining the support surface and its height clearance. A single furniture piece may contain multiple support regions of varying shapes and sizes (e.g., as in Fig. 4). The 100 objects we annotated have a total of 529 regions, with an average of five regions per object. See the Appendix B for details on the annotation process.

Metrics. We evaluate the extracted support regions against ground truth annotations using two metrics: Intersection over Union (IoU) of the regions' volume, measuring the overlap between predicted and ground truth support regions; and F1-score, which provides a balanced measure between precision (accuracy of predicted regions) and recall (completeness of detected regions) at a threshold of 0.5.

For IoU computation, each ground truth and predicted support surface is projected onto the horizontal plane to simplify calculation of overlapping areas. Subsequently, the intersection volume of the two regions is determined by multiplying the area of intersection of the projected surfaces with the overlap of their heights along the vertical axis. To emphasize the importance of support surface alignment, we apply a height threshold t_d, when computing the IoU. If the vertical distance between ground truth and predicted surfaces exceeds t_d, the IoU is set to 0. We fix t_d to 10 cm to ensure strict correspondence in support surface height alignment. The Hungarian algorithm is applied to these IoU values to determine optimal region-region correspondence, and match predicted and actual support surfaces.

Baselines. We compare our support region extraction approach against a baseline which only predicts support regions on the top surface of an object, thus matching Holodeck [83]'s ray casting approach for object placement.

Level	Method	↑ Fidelity$_{\%}$	↑ Plausibility$_{\%}$
Scene level	Holodeck	23.36	30.24
	HSM (ours)	**76.64**	**69.76**
Small object level	Holodeck	18.40	26.80
	HSM (ours)	**81.60**	**73.20**

Table 2. **User study results.** Holodeck [83] compared to HSM at scene and small object levels. HSM is preferred at both levels.

	↑ BLIP	↑ CLIP	↑ VQA
LayoutGPT	0.0742	0.0144	0.1246
InstructScene	0.1129	0.0413	0.1272
LayoutVLM	0.1432	0.0808	0.1402
Holodeck	0.2183	0.1250	0.1702
HSM (ours)	**0.2497**	**0.1582**	**0.1732**

Table 3. **Evaluation on small object placements.** HSM outperforms prior work across all metrics.

5. Results

5.1. Scene Generation

Fig. 5 presents generated scenes from HSM and the baselines, while Tab. 1 reports the quantitative evaluation results. HSM outperforms the baselines as measured by text-image score and SceneEval fidelity metrics, demonstrating better alignment between generated scenes and input descriptions. Qualitative results further highlight HSM 's ability to capture precise requirements specified in the input. For example, HSM is the only method that accurately generates a small table next to a loveseat with a coffee table in front in the first row, and two wall sconces flanking a mirror in the third row. In contrast, LayoutGPT frequently places objects outside the room. InstructScene and LayoutVLM also place objects outside room boundaries, and exhibit limited object variety and layout diversity. Additional analyses are in the supplement: computational cost and runtime breakdown (Appendix A.6), breakdown of results by difficulty (Tab. 5), and evaluation of HSM using an open-source VLM (Appendix A.7).

Tab. 3 shows that HSM outperform prior work on small object placements. Results from our user study (Tab. 2) further demonstrate that HSM is preferred at both the scene and small object levels in fidelity and plausibility. While

	SceneEval Fidelity				SceneEval Plausbility						Avg. #Obj per Scene
	↑ CNT$_{\%}$	↑ ATR$_{\%}$	↑ OOR$_{\%}$	↑ OAR$_{\%}$	↓ COL$_{ob\%}$	↓ COL$_{sc\%}$	↑ SUP$_{\%}$	↑ NAV$_{\%}$	↑ ACC$_{\%}$	↓ OOB$_{\%}$	
HSM (ours)	**61.30**	59.49	**40.40**	**70.28**	**16.42**	**61.00**	85.44	**98.97**	86.80	**2.13**	20.65
- w/o scene motifs	54.79	53.28	32.66	50.20	24.12	87.00	**87.40**	97.15	84.78	2.26	28.81
- w/o scene spatial optimizer	55.94	**60.95**	30.66	61.85	25.73	74.00	83.98	98.37	**87.70**	2.95	22.04
- w/o DFS solver	52.87	49.64	26.36	54.22	28.96	75.00	75.83	98.78	73.34	12.76	19.20

Table 4. **Ablation study.** Ablations of HSM generate scenes with lower fidelity. Removing scene motifs reduces fidelity, as the VLM must handle individual object placement rather than leveraging grouped structures. Disabling the spatial optimizer reduces plausibility with higher COL. Disabling the DFS solver causes the largest drop across most metrics with lower fidelity and reduced plausibility due to invalid object placements outside support regions. Bold indicates highest results, underlined denotes second highest.

Figure 6. **Qualitative comparison of object placements.** Each row shows close-up views of object arrangements from the input description. HSM better follows the spatial relationships and placement instructions specified in the input text.

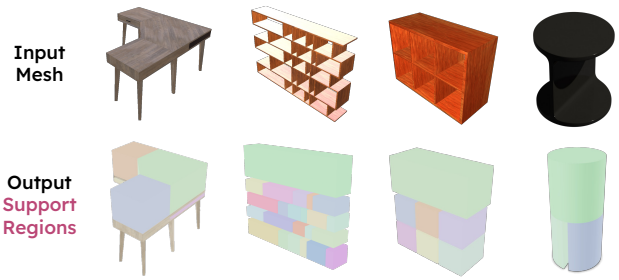

Figure 7. **Support region extraction examples.** Colored boxes are extracted support regions for accurately placing smaller items.

Holodeck can generate small objects, it only considers the top surfaces of furniture items, leaving interiors unrealistically empty (as shown in Fig. 6). In contrast, HSM's support region analysis identifies valid placement regions across all object surfaces, enabling denser and more realistic small object arrangements. By leveraging a hierarchical approach, HSM produces more structured and realistic object placements across all scales within a unified framework. See Fig. 10 and Fig. 11 for more qualitative results.

5.2. Support Region Extraction

Fig. 7 shows examples of extracted support regions. HSM achieves an average IoU of 60.27% and F1-score of 48.54% against ground truth annotations. In contrast, the baseline that considers only top surfaces performs significantly worse, with an IoU of 32.54% and an F1-score of 16.80%.

This demonstrates that HSM more effectively identifies valid support regions within furniture items, which is essential for accurate small object placement in scenes.

5.3. Ablation

To evaluate the contribution of individual components, we ablate key elements of HSM: 1) *w/o scene motif* — removing scene motifs and treating all objects as individual pieces, 2) *w/o scene spatial optimizer* — removing scene spatial optimization and directly using DFS solver positions to place scene motifs, and 3) *w/o DFS solver* — removing DFS solver and directly using VLM-provided positions to place scene motifs. Tab. 4 shows that removing any component results in worse performance. See Appendix A.5 for a detailed analysis and qualitative comparison.

6. Conclusion

We presented Hierarchical Scene Motifs (HSM), a hierarchical framework for indoor 3D scene generation that produces dense object arrangements across spatial scales. Our approach models indoor scenes as a hierarchy of support regions, each to be populated with objects. By leveraging scene motifs, HSM generates object arrangements at multiple scales, from room-level furniture layouts to fine-grained small object placements, exploiting recurring spatial patterns. We believe HSM's unified hierarchical framework represents a significant step toward generating densely populated and realistic indoor environments.

Acknowledgments. This work was funded in part by the Sony Research Award Program, a CIFAR AI Chair, a Canada Research Chair, NSERC DG, and enabled by support from Digital Research Alliance. We thank Jiayi Liu, Weikun Peng, and Qirui Wu for helpful discussions.

References

[1] Ahmed Abdelreheem, Filippo Aleotti, Jamie Watson, Zawar Qureshi, Abdelrahman Eldesokey, Peter Wonka, Gabriel Brostow, Sara Vicente, and Guillermo Garcia-Hernando. PlaceIt3D: Language-guided object placement in real 3D scenes. *arXiv:2505.05288*, 2025. 2

[2] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. GPT-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023. 5

[3] Rio Aguina-Kang, Maxim Gumin, Do Heon Han, Stewart Morris, Seung Jean Yoo, Aditya Ganeshan, R Kenny Jones, Qiuhong Anna Wei, Kailiang Fu, and Daniel Ritchie. Open-Universe indoor scene generation using LLM program synthesis and uncurated object databases. *arXiv preprint arXiv:2403.09675*, 2024. 2

[4] Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, Wenbin Ge, Zhifang Guo, Qidong Huang, Jie Huang, Fei Huang, Binyuan Hui, Shutong Jiang, Zhaohai Li, Mingsheng Li, Mei Li, Kaixin Li, Zicheng Lin, Junyang Lin, Xuejing Liu, Jiawei Liu, Chenglong Liu, Yang Liu, Dayiheng Liu, Shixuan Liu, Dunjie Lu, Ruilin Luo, Chenxu Lv, Rui Men, Lingchen Meng, Xuancheng Ren, Xingzhang Ren, Sibo Song, Yuchong Sun, Jun Tang, Jianhong Tu, Jianqiang Wan, Peng Wang, Pengfei Wang, Qiuyue Wang, Yuxuan Wang, Tianbao Xie, Yiheng Xu, Haiyang Xu, Jin Xu, Zhibo Yang, Mingkun Yang, Jianxin Yang, An Yang, Bowen Yu, Fei Zhang, Hang Zhang, Xi Zhang, Bo Zheng, Humen Zhong, Jingren Zhou, Fan Zhou, Jing Zhou, Yuanzhi Zhu, and Ke Zhu. Qwen3-VL Technical Report. *arXiv preprint arXiv:2511.21631*, 2025. 16

[5] Tongyuan Bai, Wangyuanfan Bai, Dong Chen, Tieru Wu, Manyi Li, and Rui Ma. FreeScene: Mixed graph diffusion for 3D scene synthesis from free prompts. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5893–5903, 2025. 2

[6] Ata Çelen, Guo Han, Konrad Schindler, Luc Van Gool, Iro Armeni, Anton Obukhov, and Xi Wang. I-Design: Personalized LLM interior designer. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 217–234. Springer, 2024. 2

[7] Angel Chang, Manolis Savva, and Christopher D Manning. Learning spatial knowledge for text to 3D scene generation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 2028–2038, 2014. 2

[8] Bob Coyne and Richard Sproat. WordsEye: An automatic text-to-scene conversion system. In *Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques*, pages 487–496, 2001. 2

[9] Matt Deitke, Eli VanderBilt, Alvaro Herrasti, Luca Weihs, Kiana Ehsani, Jordi Salvador, Winson Han, Eric Kolve, Aniruddha Kembhavi, and Roozbeh Mottaghi. ProcTHOR: Large-scale embodied AI using procedural generation. In *Advances in Neural Information Processing Systems*, pages 5982–5994, 2022. 2

[10] Matt Deitke, Dustin Schwenk, Jordi Salvador, Luca Weihs, Oscar Michel, Eli VanderBilt, Ludwig Schmidt, Kiana Ehsani, Aniruddha Kembhavi, and Ali Farhadi. Objaverse: A universe of annotated 3D objects. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 13142–13153, 2023. 6

[11] Wei Deng, Mengshi Qi, and Huadong Ma. Global-local tree search in vlms for 3D indoor scene generation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8975–8984, 2025. 2

[12] Helisa Dhamo, Fabian Manhardt, Nassir Navab, and Federico Tombari. 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*, pages 16352–16361, 2021. 2

[13] 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. *arXiv preprint arXiv:2504.13072*, 2025. 2

[14] 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. In *Advances in Neural Information Processing Systems*, pages 18225–18250, 2023. 5

[15] Weitao Feng, Hang Zhou, Jing Liao, Li Cheng, and Wenbo Zhou. CasaGPT: cuboid arrangement and scene assembly for interior design. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 29173–29182, 2025. 2

[16] Matthew Fisher, Daniel Ritchie, Manolis Savva, Thomas Funkhouser, and Pat Hanrahan. Example-based synthesis of 3D object arrangements. *ACM Transactions on Graphics (TOG)*, 31(6):1–11, 2012. 2

[17] Matthew Fisher, Manolis Savva, Yangyan Li, Pat Hanrahan, and Matthias Nießner. Activity-centric scene synthesis for functional 3D scene modeling. *ACM Transactions on Graphics (TOG)*, 34(6):1–13, 2015. 2

[18] 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 International Conference on Computer Vision (ICCV)*, pages 10913–10922, 2021. 5, 6

[19] Huan Fu, Rongfei Jia, Lin Gao, Mingming Gong, Binqiang Zhao, Steve Maybank, and Dacheng Tao. 3D-FUTURE: 3D furniture shape with texture. *International Journal of Computer Vision (IJCV)*, 129:3313–3337, 2021. 5, 6

[20] Qiang Fu, Xiaowu Chen, Xiaotian Wang, Sijia Wen, Bin Zhou, and Hongbo Fu. Adaptive synthesis of indoor scenes via activity-associated object relation graphs. *ACM Transactions on Graphics (TOG)*, 36(6):1–13, 2017. 2

[21] Rao Fu, Zehao Wen, Zichen Liu, and Srinath Sridhar. AnyHome: Open-vocabulary generation of structured and textured 3D homes. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 52–70, 2024. 2

[22] Lin Gao, Jia-Mu Sun, Kaichun Mo, Yu-Kun Lai, Leonidas J Guibas, and Jie Yang. SceneHGN: Hierarchical graph networks for 3D indoor scene generation with fine-grained geometry. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(7):8902–8919, 2023. 2

[23] Yunhao Ge, Hong-Xing Yu, Cheng Zhao, Yuliang Guo, Xinyu Huang, Liu Ren, Laurent Itti, and Jiajun Wu. 3D copy-paste: Physically plausible object insertion for monocular 3D detection. *Advances in Neural Information Processing Systems*, 36, 2024. 2

[24] 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 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2891–2901, 2025. 2

[25] Ruiqi Guo and Derek Hoiem. Support surface prediction in indoor scenes. In *Proceedings of the IEEE international conference on computer vision*, pages 2144–2151, 2013. 2

[26] Paul Henderson, Kartic Subr, and Vittorio Ferrari. Automatic generation of constrained furniture layouts. *arXiv preprint arXiv:1711.10939*, 2017. 2

[27] Siyi Hu, Diego Martin Arroyo, Stephanie Debats, Fabian Manhardt, Luca Carlone, and Federico Tombari. Mixed diffusion for 3D indoor scene synthesis. *arXiv preprint arXiv:2405.21066*, 2024. 2

[28] 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 *Proceedings of the International Conference on Machine Learning (ICML)*, pages 19252–19282, 2024. 2

[29] Ian Huang, Yanan Bao, Karen Truong, Howard Zhou, Cordelia Schmid, Leonidas Guibas, and Alireza Fathi. Fireplace: Geometric refinements of LLM common sense reasoning for 3D object placement. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 13466–13476, 2025. 2

[30] Rui Huang, Guangyao Zhai, Zuria Bauer, Marc Pollefeys, Federico Tombari, Leonidas Guibas, Gao Huang, and Francis Engelmann. Video perception models for 3D scene synthesis. *arXiv preprint arXiv:2506.20601*, 2025. 2

[31] Gabriel Ilharco, Mitchell Wortsman, Ross Wightman, Cade Gordon, Nicholas Carlini, Rohan Taori, Achal Dave, Vaishaal Shankar, Hongseok Namkoong, John Miller, Hannaneh Hajishirzi, Ali Farhadi, and Ludwig Schmidt. OpenCLIP, 2021. 15

[32] Yun Jiang, Marcus Lim, and Ashutosh Saxena. Learning object arrangements in 3D scenes using human context. *arXiv preprint arXiv:1206.6462*, 2012. 2

[33] Andrej Karpathy, Stephen Miller, and Li Fei-Fei. Object discovery in 3D scenes via shape analysis. In *Proceedings of the International Conference on Robotics and Automation (ICRA)*, pages 2088–2095. IEEE, 2013. 4

[34] Mohammad Keshavarzi, Aakash Parikh, Xiyu Zhai, Melody Mao, Luisa Caldas, and Allen Y Yang. SceneGen: Generative contextual scene augmentation using scene graph priors. *arXiv preprint arXiv:2009.12395*, 2020. 2

[35] Mukul Khanna, Yongsen Mao, Hanxiao Jiang, Sanjay Haresh, Brennan Shacklett, Dhruv Batra, Alexander Clegg, Eric Undersander, Angel X Chang, and Manolis Savva. Habitat synthetic scenes dataset (HSSD-200): An analysis of 3D scene scale and realism tradeoffs for ObjectGoal navigation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 16384–16393, 2024. 5, 7, 13, 16

[36] Kari Anne Høier Kjølaas. *Automatic furniture population of large architectural models*. PhD thesis, Massachusetts Institute of Technology, 2000. 2

[37] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. BLIP-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *International conference on machine learning*, pages 19730–19742. PMLR, 2023. 6

[38] Manyi Li, Akshay Gadi Patil, Kai Xu, Siddhartha Chaudhuri, Owais Khan, Ariel Shamir, Changhe Tu, Baoquan Chen, Daniel Cohen-Or, and Hao Zhang. GRAINS: Generative recursive autoencoders for indoor scenes. *ACM Transactions on Graphics (TOG)*, 38(2):1–16, 2019. 2

[39] Yijie Li, Pengfei Xu, Junquan Ren, Zefan Shao, and Hui Huang. GLTScene: Global-to-local transformers for indoor scene synthesis with general room boundaries. In *Computer Graphics Forum*, page e15236. Wiley Online Library, 2024.

[40] Chenguo Lin and Yadong Mu. InstructScene: Instruction-driven 3D indoor scene synthesis with semantic graph prior. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2024. 2, 5

[41] Zhiqiu Lin, Deepak Pathak, Baiqi Li, Jiayao Li, Xide Xia, Graham Neubig, Pengchuan Zhang, and Deva Ramanan. Evaluating text-to-visual generation with image-to-text generation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 366–384. Springer, 2024. 6

[42] Lu Ling, Chen-Hsuan Lin, Tsung-Yi Lin, Yifan Ding, Yu Zeng, Yichen Sheng, Yunhao Ge, Ming-Yu Liu, Aniket Bera, and Zhaoshuo Li. Scenethesis: A language and vision agentic framework for 3D scene generation. *arXiv preprint arXiv:2505.02836*, 2025. 2

[43] Gabrielle Littlefair, Niladri Shekhar Dutt, and Niloy J Mitra. FlairGPT: Repurposing LLMs for interior designs. In *Computer Graphics Forum*, page e70036. Wiley Online Library, 2025. 2

[44] Jia-Hong Liu, Shao-Kui Zhang, Tianqi Zhang, Jia-Tong Zhang, and Song-Hai Zhang. SceneFunctioner: Tailoring large language model for function-oriented interactive scene synthesis, 2025. 2

[45] Xinhang Liu, Yu-Wing Tai, and Chi-Keung Tang. Agentic 3D scene generation with spatially contextualized VLMs. *arXiv preprint arXiv:2505.20129*, 2025. 2

[46] Xinhang Liu, Chi-Keung Tang, and Yu-Wing Tai. WorldCraft: Photo-realistic 3D world creation and customization via LLM agents. *arXiv preprint arXiv:2502.15601*, 2025. 2

[47] Rui Ma, Akshay Gadi Patil, Matthew Fisher, Manyi Li, Sören Pirk, Binh-Son Hua, Sai-Kit Yeung, Xin Tong, Leonidas Guibas, and Hao Zhang. Language-driven synthesis of 3D scenes from scene databases. *ACM Transactions on Graphics (TOG)*, 37(6):1–16, 2018. 2

[48] Lucas Majerowicz, Ariel Shamir, Alla Sheffer, and Holger H Hoos. Filling your shelves: Synthesizing diverse style-preserving artifact arrangements. *IEEE transactions on visualization and computer graphics*, 20(11):1507–1518, 2013. 2

[49] Paul Merrell, Eric Schkufza, Zeyang Li, Maneesh Agrawala, and Vladlen Koltun. Interactive furniture layout using interior design guidelines. *ACM transactions on graphics (TOG)*, 30(4):1–10, 2011. 2

[50] Wenjie Min, Wenming Wu, Gaofeng Zhang, and Liping Zheng. FuncScene: Function-centric indoor scene synthesis via a variational autoencoder framework. *Computer Aided Geometric Design*, 111:102319, 2024. 2

[51] Wamiq Reyaz Para, Paul Guerrero, Niloy Mitra, and Peter Wonka. COFS: Controllable furniture layout synthesis. In *ACM SIGGRAPH 2023 Conference Proceedings*, pages 1–11, 2023.

[52] Despoina Paschalidou, Amlan Kar, Maria Shugrina, Karsten Kreis, Andreas Geiger, and Sanja Fidler. ATISS: Autoregressive transformers for indoor scene synthesis. In *Advances in Neural Information Processing Systems*, pages 12013–12026, 2021. 2

[53] Nicholas Ezra Pfaff, Hongkai Dai, Sergey Zakharov, Shun Iwase, and Russ Tedrake. Steerable scene generation with post training and inference-time search. In *Proceedings of The 8th Conference on Robot Learning*, pages 1690–1702. PMLR, 2025. 2

[54] Pulak Purkait, Christopher Zach, and Ian Reid. SG-VAE: Scene grammar variational autoencoder to generate new indoor scenes. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 155–171. Springer, 2020. 2

[55] Siyuan Qi, Yixin Zhu, Siyuan Huang, Chenfanfu Jiang, and Song-Chun Zhu. Human-centric indoor scene synthesis using stochastic grammar. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5899–5908, 2018. 2

[56] Alexander Raistrick, Lingjie Mei, Karhan Kayan, David Yan, Yiming Zuo, Beining Han, Hongyu Wen, Meenal Parakh, Stamatis Alexandropoulos, Lahav Lipson, et al. Infinigen Indoors: Photorealistic indoor scenes using procedural generation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 21783–21794, 2024. 2

[57] 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. 2

[58] Zhile Ren and Erik B Sudderth. 3D object detection with latent support surfaces. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 937–946, 2018. 2

[59] Daniel Ritchie, Kai Wang, and Yu-an Lin. Fast and flexible indoor scene synthesis via deep convolutional generative models. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6175–6183, 2019. 2

[60] Denys Rozumnyi, Stefan Popov, Kevis-kokitsi Maninis, Matthias Niessner, and Vittorio Ferrari. Estimating generic 3D room structures from 2D annotations. In *Advances in Neural Information Processing Systems*, pages 37786–37798. Curran Associates, Inc., 2023. 2

[61] Manolis Savva, Angel X Chang, Pat Hanrahan, Matthew Fisher, and Matthias Nießner. PiGraphs: Learning interaction snapshots from observations. *ACM Transactions on Graphics (TOG)*, 35(4):1–12, 2016. 2

[62] Sinisa Stekovic, Shreyas Hampali, Mahdi Rad, Sayan Deb Sarkar, Friedrich Fraundorfer, and Vincent Lepetit. General 3D room layout from a single view by render-and-compare. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XVI 16*, pages 187–203. Springer, 2020. 2

[63] Chong Su, Yingbin Fu, Zheyuan Hu, Jing Yang, Param Hanji, Shaojun Wang, Xuan Zhao, Cengiz Öztireli, and Fangcheng Zhong. 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. 2

[64] 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 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 29469–29478, 2025. 2, 5

[65] Qi Sun, Hang Zhou, Wengang Zhou, Li Li, and Houqiang Li. Forest2Seq: Revitalizing order prior for sequential indoor scene synthesis. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 251–268, 2024. 2

[66] Weilin Sun, Xinran Li, Manyi Li, Kai Xu, Xiangxu Meng, and Lei Meng. Hierarchically-structured open-vocabulary indoor scene synthesis with pre-trained large language model. In *Association for the Advancement of Artificial Intelligence (AAAI)*, pages 7122–7130, 2025. 2

[67] Hou In Ivan Tam, Hou In Derek Pun, Austin T. Wang, Angel X. Chang, and Manolis Savva. SceneMotifCoder: Example-driven Visual Program Learning for Generating 3D Object Arrangements. In *Proceedings of the International Conference on 3D Vision (3DV)*, 2025. 3, 13

[68] Hou In Ivan Tam, Hou In Derek Pun, Austin T. Wang, Angel X. Chang, and Manolis Savva. SceneEval: Evaluating semantic coherence in text-conditioned 3D indoor scene synthesis. In *Proc. of the Winter Conference on Applications of Computer Vision (WACV)*, 2026. 6, 15, 16

[69] 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 Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 20507–20518, 2024. 2

[70] Can Wang, Hongliang Zhong, Menglei Chai, Mingming He, Dongdong Chen, and Jing Liao. Chat2Layout: Interactive 3D Furniture Layout with a Multimodal LLM. *IEEE transactions on visualization and computer graphics*, PP, 2024. 2

[71] Kai Wang, Manolis Savva, Angel X Chang, and Daniel Ritchie. Deep convolutional priors for indoor scene synthesis. *ACM Transactions on Graphics (TOG)*, 37(4):1–14, 2018. 2

[72] Kai Wang, Yu-An Lin, Ben Weissmann, Manolis Savva, Angel X Chang, and Daniel Ritchie. PlanIT: Planning and instantiating indoor scenes with relation graph and spatial prior networks. *ACM Transactions on Graphics (TOG)*, 38(4):1–15, 2019. 2

[73] Weiyun Wang, Zhangwei Gao, Lixin Gu, Hengjun Pu, Long Cui, Xingguang Wei, Zhaoyang Liu, Linglin Jing, Shenglong Ye, Jie Shao, et al. InternVL3.5: Advancing Open-Source Multimodal Models in Versatility, Reasoning, and Efficiency . *arXiv preprint arXiv:2508.18265*, 2025. 16

[74] Xinpeng Wang, Chandan Yeshwanth, and Matthias Nießner. Sceneformer: Indoor scene generation with transformers. In *Proceedings of the International Conference on 3D Vision (3DV)*, pages 106–115. IEEE, 2021. 2

[75] Yian Wang, Xiaowen Qiu, Jiageng Liu, Zhehuan Chen, Jiting Cai, Yufei Wang, Tsun-Hsuan Wang, Zhou Xian, and Chuang Gan. Architect: Generating vivid and interactive 3D scenes with hierarchical 2D inpainting. In *Advances in Neural Information Processing Systems*, pages 67575–67603, 2024. 2

[76] Yao Wei, Martin Renqiang Min, George Vosselman, Li Erran Li, and Michael Ying Yang. Planner3D: LLM-enhanced graph prior meets 3D indoor scene explicit regularization. *IEEE transactions on pattern analysis and machine intelligence*, 2025. 2

[77] Qirui Wu, Denys Iliash, Daniel Ritchie, Manolis Savva, and Angel X Chang. Diorama: Unleashing zero-shot single-view 3d indoor scene modeling. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 8896–8907, 2025. 2

[78] Zijie Wu, Mingtao Feng, Yaonan Wang, He Xie, Weisheng Dong, Bo Miao, and Ajmal Mian. External knowledge enhanced 3D scene generation from sketch. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 286–304. Springer, 2025. 2

[79] Xiao Xia, Dan Zhang, Zibo Liao, Zhenyu Hou, Tianrui Sun, Jing Li, Ling Fu, and Yuxiao Dong. SceneGenAgent: Precise industrial scene generation with coding agent. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics*, pages 17847–17875, 2025. 2

[80] Ken Xu, James Stewart, and Eugene Fiume. Constraint-based automatic placement for scene composition. In *Graphics Interface*, pages 25–34. Citeseer, 2002. 2

[81] Yandan Yang, Baoxiong Jia, Peiyuan Zhi, and Siyuan Huang. PhyScene: Physically interactable 3D scene synthesis for embodied AI. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 16262–16272, 2024. 2

[82] Yixuan Yang, Junru Lu, Zixiang Zhao, Zhen Luo, James JQ Yu, Victor Sanchez, and Feng Zheng. LLplace: The 3D indoor scene layout generation and editing via large language model. *arXiv preprint arXiv:2406.03866*, 2024. 2

[83] 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 Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 16277–16287, 2024. 2, 5, 7

[84] Zhifei Yang, Keyang Lu, Chao Zhang, Jiaxing Qi, Hanqi Jiang, Ruifei Ma, Shenglin Yin, Yifan Xu, Mingzhe Xing, Zhen Xiao, et al. MMGDreamer: Mixed-modality graph for geometry-controllable 3D indoor scene generation. In *Association for the Advancement of Artificial Intelligence (AAAI)*, 2025. 2

[85] Zhaoda Ye, Yang Liu, and Yuxin Peng. MAAN: Memory-augmented auto-regressive network for text-driven 3D indoor scene generation. *IEEE Transactions on Multimedia*, 2024. 2

[86] Lap Fai Yu, Sai Kit Yeung, Chi Keung Tang, Demetri Terzopoulos, Tony F Chan, and Stanley J Osher. Make it Home: Automatic optimization of furniture arrangement. *ACM Transactions on Graphics (TOG)*, 30(4), 2011. 2

[87] Lap-Fai Yu, Sai-Kit Yeung, and Demetri Terzopoulos. The ClutterPalette: An interactive tool for detailing indoor scenes. *IEEE transactions on visualization and computer graphics*, 22(2):1138–1148, 2015. 2

[88] Guangyao Zhai, Evin Pınar Örnek, Shun-Cheng Wu, Yan Di, Federico Tombari, Nassir Navab, and Benjamin Busam. CommonScenes: Generating commonsense 3D indoor scenes with scene graphs. In *Advances in Neural Information Processing Systems*, pages 30026–30038, 2023. 2

[89] Guangyao Zhai, Evin Pınar Örnek, Dave Zhenyu Chen, Ruotong Liao, Yan Di, Nassir Navab, Federico Tombari, and Benjamin Busam. EchoScene: Indoor scene generation via information echo over scene graph diffusion. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 167–184, 2024. 2

[90] Beichen Zhang, Pan Zhang, Xiaoyi Dong, Yuhang Zang, and Jiaqi Wang. Long-CLIP: Unlocking the long-text capability of CLIP. In *Proceedings of the European Conference on Computer Vision (ECCV)*, page 310–325, 2024. 6

[91] Suiyun Zhang, Zhizhong Han, Yu-Kun Lai, Matthias Zwicker, and Hui Zhang. Active arrangement of small objects in 3D indoor scenes. *IEEE transactions on visualization and computer graphics*, 27(4):2250–2264, 2019. 2

[92] Shao-Kui Zhang, Yi-Xiao Li, Yu He, Yong-Liang Yang, and Song-Hai Zhang. MageAdd: Real-time interaction simulation for scene synthesis. In *Proceedings of the 29th ACM International Conference on Multimedia*, pages 965–973, 2021. 2

[93] Shao-Kui Zhang, Jia-Hong Liu, Yike Li, Tianyi Xiong, Ke-Xin Ren, Hongbo Fu, and Song-Hai Zhang. Automatic generation of commercial scenes. In *Proceedings of the 31st ACM International Conference on Multimedia*, pages 1137–1147, 2023. 2

[94] Zaiwei Zhang, Zhenpei Yang, Chongyang Ma, Linjie Luo, Alexander Huth, Etienne Vouga, and Qixing Huang. Deep generative modeling for scene synthesis via hybrid representations. *ACM Transactions on Graphics (TOG)*, 39(2):1–21, 2020. 2

[95] Mengqi Zhou, Xipeng Wang, Yuxi Wang, and Zhaoxiang Zhang. RoomCraft: Controllable and complete 3D indoor scene generation. *arXiv preprint arXiv:2506.22291*, 2025. 2