SCENEPARSER: HIERARCHICAL SCENE PARSING FOR VISUAL SEMANTICS UNDERSTANDING

Anonymous authors
Paper under double-blind review

ABSTRACT

General scene perception has progressed from object recognition toward open-vocabulary grounding, part localization, and affordance prediction. Yet these capabilities are often realized as isolated predictions that localize objects, parts, or interaction points without capturing the structured dependencies needed for interaction-oriented scene understanding. To address this gap, we introduce an interaction-oriented formulation of hierarchical scene parsing that represents scenes as explicit scene→object→part→affordance hierarchies with cross-level bindings. We instantiate this task with SceneParser, a VLM-based parser trained for unified hierarchical generation with structural-completion pseudo labels and curriculum learning. To support training and evaluation, we construct SceneParser-Bench, a large-scale benchmark built with a scalable hierarchical data engine, containing 110K training images, a 5K validation split, 777K objects, 1.14M parts, 1.74M affordance annotations, and 1.74M valid object-part-affordance chain instances. We further introduce Level-1 to Level-3 conditional metrics and ParseRate to evaluate localization, cross-level binding, and hierarchical completeness. Experiments show that existing MLLMs and hierarchy assembly pipelines struggle with hierarchical parsing on our SceneParser-Bench, while SceneParser achieves stronger structure-aware performance. Besides, ablations, evaluations on COCO and AGD20K, and a downstream planning probe demonstrate that our SceneParser is compatible with conventional tasks and provides an actionable representation for visual understanding.

1 INTRODUCTION

Scene perception is a basic computer vision problem with wide downstream applications (Everingham et al., 2010; Lin et al., 2014; Zhou et al., 2017). It is also a fundamental capability for embodied intelligence, supporting robotic navigation, manipulation, and human-robot interaction (Brohan et al., 2024; Majumdar et al., 2024; Wang et al., 2024). Recent progress has shifted visual understanding from closed-set object recognition toward open-world and language-conditioned localization (Liu et al., 2024; Ren et al., 2024b; Cheng et al., 2024; Xiao et al., 2024; Fu et al., 2025; Xiao et al., 2025). Object grounding and referring expression comprehension enable models to localize targets specified by open-vocabulary categories, phrases, or natural-language instructions (Li et al., 2022; Liu et al., 2024; Chen et al., 2023; Zhang et al., 2024; Ma et al., 2024; Jiang et al., 2024; 2025; Wang et al., 2025). Beyond object-level grounding, embodied visual understanding further requires fine-grained and interaction-oriented perception, including part localization and affordance prediction. These advances suggest that modern visual perception is increasingly expected to localize action-relevant entities across multiple semantic levels (He et al., 2022; Ramanathan et al., 2023; Luo et al., 2022; Qian et al., 2024; Yuan et al., 2024; Wang et al., 2026; Sun & Lu, 2025).

However, multi-level localization does not necessarily amount to structured scene understanding. Existing methods may predict object boxes, part regions, and affordance points, but these outputs are often local, fragmented, and weakly bound across levels. An isolated affordance point, for example, does not explicitly encode the object it belongs to, the functional part that supports the action, or its relation to alternative interaction candidates; similarly, standard metrics often score objects, parts, and affordances independently, without verifying whether they form a consistent object-part-affordance chain. This limitation also aligns with a broader observation in agentic and spatial reasoning: downstream decision making often benefits from explicit intermediate representations, such as

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

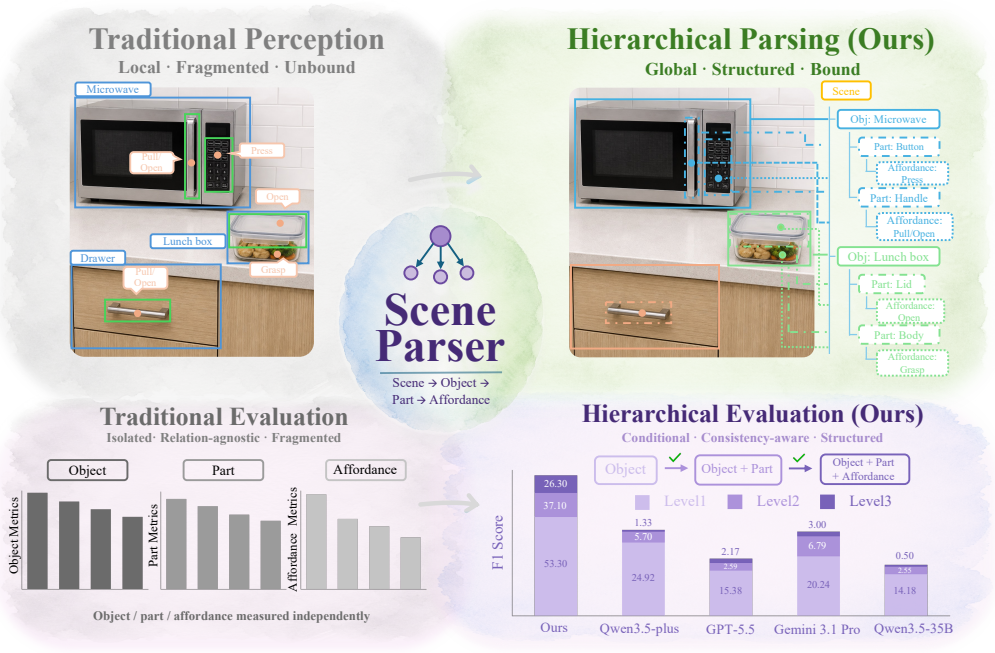

Figure 1: **Overview of SceneParser.** Conventional perception produces local predictions, such as object boxes, part regions, and affordance points, which are often evaluated independently and leave cross-level relations unverified. In contrast, we parse a scene into an explicit *scene* → *object* → *part* → *affordance* hierarchy. Moreover, our hierarchy-aware evaluation conditions lower-level correctness on upper-level matches, jointly assessing detection, cross-level binding, and structural consistency.

structured GUI elements (Lu et al., 2024), spatial layouts, entity registries, or visual primitives (Hua et al., 2026; Lu et al., 2026). For embodied scenes, however, the desired representation should be functional rather than merely spatial, describing objects, their actionable parts, and the affordances grounded on those parts.

Motivated by this perspective, we introduce an interaction-oriented formulation of *hierarchical scene parsing*, as illustrated in Fig. 1. Given an RGB scene image, the goal is to recover a hierarchy of *scene* → *object* → *part* → *affordance*. Each object is represented by a semantic name and a bounding box; each part is represented by a part name and a part bounding box; and each affordance is represented by an action label and an interaction point. Crucially, the representation is not a flat collection of predictions: every part is explicitly attached to its parent object, and every affordance is grounded on a specific functional part. Therefore, instead of only asking where an object, part, or affordance point is located, hierarchical scene parsing asks which objects are present, how they decompose into functional parts, which parts support which actions, and where each action should be grounded.

To instantiate this task as a learnable model, we design **SceneParser**, a VLM-based hierarchical parser for unified object-part-affordance generation. Instead of combining separate object, part, and affordance predictors, SceneParser directly generates a structured hierarchy with explicit parent-child bindings. To better adapt VLM generation to hierarchical targets, we introduce structural-completion pseudo labels to model valid non-expandable cases and reduce broken hierarchies. We further use curriculum learning to progressively balance object-level localization stability, part decomposition, affordance grounding, and hierarchical completeness.

To support training and evaluation, we construct **SceneParser-Bench**, a large-scale benchmark built with a scalable hierarchical data engine. The engine progressively constructs supervision at three levels: scene-level object grounding, object-centric part parsing, and affordance parsing with hierarchy reconstruction. Specifically, it grounds objects from generated names and referring expressions, parses functional parts from object-centric crops, and links generated affordance actions and interaction points to the corresponding object-part hierarchy through textual and geometric matching. SceneParser-Bench contains 110K training images and a 5K validation split, with validated

annotations totaling 777K object instances, 1.14M part instances, 1.74M affordance annotations, and 1.74M valid object-part-affordance chain instances, covering 2,905 object categories, 12,349 part categories, and 85 affordance categories.

To evaluate hierarchical scene parsing, we introduce a structure-aware protocol with Level-1 to Level-3 conditional metrics and ParseRate. Level-1 evaluates object recognition and localization; Level-2 evaluates the object-part chain, counting a part as correct only when both the parent object and the part are correctly matched; and Level-3 evaluates the full object-part-affordance chain, counting an affordance as correct only when the object-part chain is valid, the action label is correct, and the interaction point is spatially valid. By conditioning lower-level correctness on upper-level matches, the protocol measures coherent hierarchical structures beyond independent object, part, and affordance scores. ParseRate further measures whether parse-eligible objects and scenes are structurally completed with the required part and affordance expansions.

Experiments show that hierarchical scene parsing remains challenging for existing MLLMs and hierarchy assembly pipelines: object-level localization does not reliably translate into valid part grounding, affordance grounding, or structural completeness. In contrast, SceneParser achieves stronger hierarchy-aware performance, and ablations show that nested object-part-affordance generation outperforms flat triplet prediction. SceneParser further generalizes to unseen VSIBench images and preserves transferable object detection and affordance grounding on COCO and AGD20K. Finally, providing the parsed hierarchy improves planning completeness across two multimodal planners, supporting the use of hierarchical scene parsing as a structured intermediate representation for interaction-oriented planning. Our contributions are summarized as follows:

- We introduce an interaction-oriented formulation of hierarchical scene parsing that organizes the scene into explicit *scene* $\rightarrow$ *object* $\rightarrow$ *part* $\rightarrow$ *affordance* structures. By preserving cross-level semantic dependencies, it provides a unified representation of interaction-relevant scene information beyond independent semantic grounding.
- We establish **SceneParser-Bench**, a large-scale dataset and benchmark for hierarchical scene parsing, together with structure-aware Level-1–Level-3 metrics and ParseRate to evaluate localization, cross-level binding, and hierarchical completeness.
- We develop **SceneParser**, a VLM-based parser with unified object-part-affordance generation, structural-completion pseudo labels, and curriculum learning. Experiments show stronger hierarchy-aware parsing than general and grounding-oriented VLMs, while further demonstrating the effectiveness for downstream interaction-oriented planning.

2 Related Work

2.1 Object, Part, and Affordance Grounding

Visual grounding has evolved from closed-set object detection to open-vocabulary and language-conditioned localization. Classical and open-vocabulary detectors localize objects under predefined or flexible textual queries (Redmon et al., 2016; Ren et al., 2015; Carion et al., 2020; Zhang et al., 2022; Li et al., 2022; Liu et al., 2024; Ren et al., 2024a; Cheng et al., 2024), while recent MLLMs formulate grounding as coordinate generation conditioned on natural-language prompts (Peng et al., 2023; Chen et al., 2023; You et al., 2023; Ma et al., 2024; Jiang et al., 2024; 2025). Beyond objects, fine-grained perception further localizes semantic or functional parts through part datasets and open-vocabulary part segmentation (He et al., 2022; Ramanathan et al., 2023; Zhou et al., 2017; Sun et al., 2023; Li et al., 2024a; Wan et al., 2025). Affordance grounding extends such perception toward interaction by predicting action-relevant regions or points conditioned on objects, language, or task instructions (Myers et al., 2015; Nguyen et al., 2017; Luo et al., 2022; Qian et al., 2024; Yuan et al., 2024; Wang et al., 2026). These works provide fine-grained interaction cues, but typically predict objects, parts, or affordances separately or under task-specific queries. We instead recover a unified scene-level *object*$\rightarrow$*part*$\rightarrow$*affordance* hierarchy that organizes interaction-relevant information. This structured representation can serve as an intermediate input for downstream interaction-oriented planning.

2.2 Hierarchical and Structured Scene Understanding

Structured scene understanding has been studied through hierarchical parsing and scene graphs. Early work explored explicit hierarchical decomposition of scenes, objects, and parts (Yao et al., 2007; Zhang et al., 2018). More recent work extends structured perception toward dense spatial

grounding and open-vocabulary modeling. Panoptic scene graph generation represents objects with panoptic regions and models their pairwise relations (Yang et al., 2022; Zhao et al., 2023), while PartGLEE explicitly establishes hierarchical correspondences between objects and semantic parts (Li et al., 2024a). Recent vision-language approaches further formulate scene graph prediction as open-vocabulary sequence generation (Li et al., 2024b), with newer benchmarks and models continuing to improve general and large-scale scene graph generation (Wu et al., 2025; Park et al., 2025). These approaches capture compositional, object–part, or inter-object structures, but do not jointly organize scene-level interaction information across objects, functional parts, and grounded actions. We formulate scene parsing as localized *object→part→affordance* dependencies, providing a unified functional representation that facilitates the extraction and organization of interaction-relevant information for downstream planning. Our structure-aware evaluation further measures whether these cross-level dependencies are recovered consistently.

3 SCENEPARSER-BENCH

3.1 PROBLEM FORMULATION

We study *hierarchical scene parsing* for embodied visual understanding. Given an RGB image I, the goal is to predict a structured scene representation that decomposes it into a *scene* $\rightarrow$ *object* $\rightarrow$ *part* $\rightarrow$ *affordance* hierarchy. Unlike conventional perception outputs, the prediction must recover visual entities, locations, and parent-child bindings among objects, parts, and affordances.

Formally, we represent the parsed scene as a rooted hierarchy $\mathcal{H} = \{O_i\}_{i=1}^{N}$, where each object is $O_i = (c_i, b_i, \mathcal{P}_i)$. Here c_i is the object name, $b_i = (x_1, y_1, x_2, y_2)$ is its bounding box, and $\mathcal{P}_i = \{P_{ij}\}_{j=1}^{M_i}$ denotes its parts. Each part is $P_{ij} = (q_{ij}, b_{ij}, \mathcal{A}_{ij})$, where q_{ij} is the part name, b_{ij} is the part bounding box, and $\mathcal{A}_{ij} = \{A_{ijk}\}_{k=1}^{K_{ij}}$ is the set of affordances grounded on this part. Each affordance is $A_{ijk} = (a_{ijk}, u_{ijk})$, where a_{ijk} is an action label and $u_{ijk} = (x, y)$ is the interaction point. Both $\mathcal{P}_i$ and $\mathcal{A}_{ij}$ may be empty, allowing the representation to model whether an object should be decomposed into parts and whether a part supports affordances.

This task is therefore not independent localization at each level. A part is valid only under its parent object, and an affordance is valid only under a valid object-part chain. The expected output is a serialized hierarchy $Y = \text{Serialize}(\mathcal{H})$, not separate object, part, and affordance predictions.

3.2 EVALUATION SUITE

We evaluate hierarchical scene parsing with a structure-aware protocol consisting of Level-1 to Level-3 conditional metrics and ParseRate. Following standard detection evaluation (Everingham et al., 2010; Lin et al., 2014), we report precision, recall, and F1, while enforcing hierarchical validity: parts are evaluated only under matched objects, and affordances only under matched object-part pairs.

Level-1 to Level-3 Metrics. Given an IoU threshold τ, we define three conditional levels:

Level 1 (L1: Object) measures object recognition and localization. Predicted and ground-truth objects are greedily matched one-to-one using identical object names and object-box IoU $\geq \tau$.

Level 2 (L2: Object-Part) measures part parsing under matched objects. Within each matched object pair, predicted and ground-truth parts are greedily matched one-to-one using identical part names and part-box IoU $\geq \tau$. Thus, a part can be counted as correct only if its parent object is also correct.

Level 3 (L3: Object-Part-Affordance) measures affordance grounding conditioned on a valid object-part chain. Within each matched object-part pair, predicted and ground-truth affordances are matched by identical action labels and spatial validity of the interaction point: the point must fall inside the ground-truth affordance box; if no affordance box is recorded, we use the parent part box as the fallback region. Thus, an affordance is correct only when its parent object, parent part, action label, and interaction point are all correct.

ParseRate. Level metrics evaluate the correctness of matched entities, but they do not directly measure whether a parse-eligible object is structurally completed. We therefore define ParseRate to measure hierarchical completeness over ground-truth objects that require part and/or affordance expansion. A matched parse-eligible object at IoU threshold 0.5 is counted as complete if the prediction satisfies its required expansion pattern: parts for objects with parts, affordances for objects with affordances, and both when both are annotated.

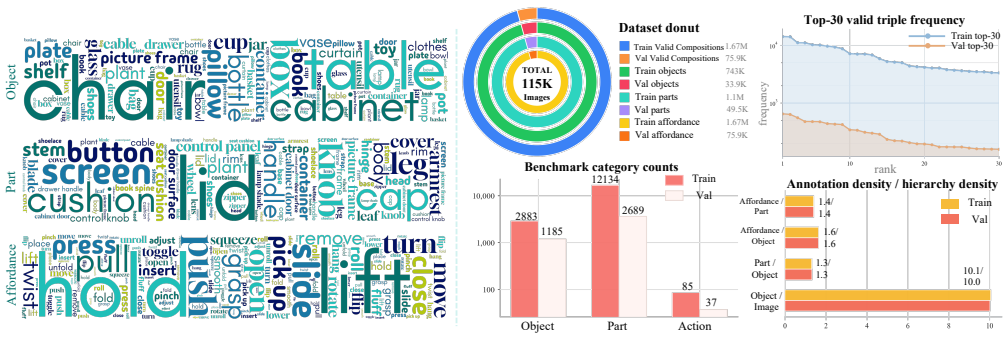

Figure 2: **Statistical overview of SceneParser-Bench.** Left: object, part, and affordance/action word clouds from top to bottom. Right: split scale, category counts, hierarchy density, and top-30 object-part-affordance compositions. The benchmark provides broad coverage, stable hierarchy density, and diverse long-tailed compositions.

Formally,

$$\text{ParseRate} = \frac{\#\{\text{complete parse-eligible GT objects}\}}{\#\{\text{parse-eligible GT objects}\}}.$$

We report ParseRate at both object and scene levels. For scene-level ParseRate, counts are accumulated over all parse-eligible objects across scenes, providing a scene-centric measure of hierarchical coverage. Detailed definitions are provided in Appendix A.

3.3 Data Engine

To support hierarchical scene parsing, we build a three-stage data engine that converts RGB scene images into structured *scene* $\rightarrow$ *object* $\rightarrow$ *part* $\rightarrow$ *affordance* supervision, with the full pipeline shown in Fig. 5. It grounds scene-level objects, parses object-centric parts, extracts action-relevant affordance regions, and reconstructs them into unified hierarchies for structured generation training and hierarchy-aware evaluation.

Stage 1: Scene-level object grounding. Given a scene image, we use GPT-5 (OpenAI, 2025) to generate object names and referring expressions, which are then grounded by localization models, including Grounding DINO (Liu et al., 2024), Rex-Omni (Jiang et al., 2025), and SAM3 (Carion et al., 2025). We apply model-specific confidence thresholds and merge duplicate predictions across models to improve coverage and reduce redundant detections. The resulting object boxes serve as parent anchors for subsequent part and affordance parsing.

Stage 2: Object-centric part parsing. For each grounded object, we crop the corresponding image region to reduce background interference. GPT-5 proposes candidate functional parts for the object, and SAM3 segments the described regions. Part masks are converted into bounding boxes and attached to their parent object, producing object-conditioned part annotations.

Stage 3: Affordance parsing. Affordance supervision is constructed on top of object-centric regions. GPT-5 generates action-oriented region descriptions, such as handles, rims, lids, or buttons, together with the affordance labels. SAM3 segments these described regions, after which we filter low-confidence masks, convert valid masks into affordance boxes, and sample interaction points.

Hierarchy reconstruction and quality control. Since object, part, and affordance annotations are produced by separate stages, we reconstruct them into a strict object-part-affordance hierarchy. Affordances are assigned to parts using textual matching when the interaction part is explicitly specified, and geometric matching otherwise, based on point containment and region overlap. We further apply consistency checks to ensure valid parent-child relations, geometric containment between levels, and removal of highly overlapping duplicates.

3.4 Benchmark Statistics and Diversity

We construct **SceneParser-Bench** based on EgoObject (Zhu et al., 2023) as a large-scale benchmark for hierarchical scene parsing. It contains 110K training and 5K validation images with no image overlap, annotated with explicit hierarchies linking localized objects, semantic parts, and affordances. The train/val splits include 743K/33.9K objects, 1.1M/49.5K parts, and 1.67M/75.9K affordances, totaling 1.74M valid object-part-affordance chain instances.

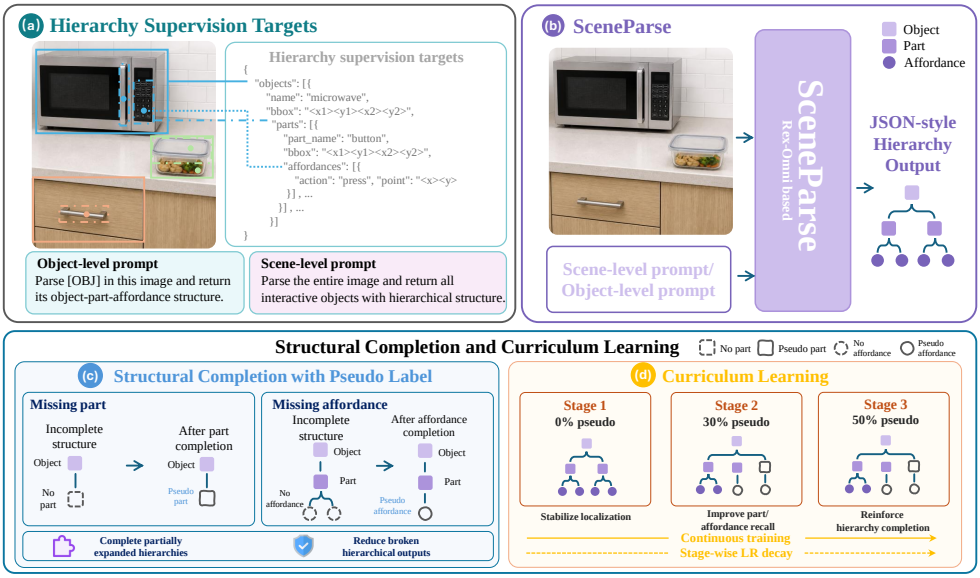

Figure 3: **SceneParser architecture.** SceneParser learns serialized object-part-affordance hierarchy targets and generates JSON-style hierarchies. Structural-completion pseudo labels complete partial hierarchies, while curriculum learning balances grounding reliability and hierarchical completeness.

Fig. 2 summarizes the benchmark statistics. The three word clouds show broad vocabularies at the object, part, and affordance/action levels. The right panel summarizes split scale, category counts, and hierarchy density, showing stable structural density across train and validation splits. The top-30 object-part-affordance composition frequencies further reveal diverse long-tailed combinations beyond frequent head patterns. A human audit confirms reliable annotations, with over 92% node/relation accuracy and 90.4% full-chain accuracy; see Appendix B.

4 SceneParser

SceneParser is a VLM-based hierarchical parser built on Rex-Omni (Jiang et al., 2025). Given an RGB image, it autoregressively generates a unified object-part-affordance hierarchy following Sec. 3.1. Unlike hierarchy assembly pipelines that predict objects, parts, and affordances independently, SceneParser directly decodes a structured hierarchy with explicit parent-child bindings. Spatial outputs reuse Rex-Omni's 1000-bin relative coordinate tokens, with serialization details in Appendix D. As shown in Fig. 3, we adapt this backbone by training it on serialized JSON-style hierarchy targets, together with structural-completion pseudo labels and curriculum learning.

4.1 Structural Completion with Pseudo Labels

Real scenes contain mixed-completeness supervision: some objects have reliable parts and affordances, while others are annotated only at higher levels. Training on such partial hierarchies can produce broken parent-child structures or uncontrolled over-expansion. We therefore introduce structural-completion pseudo labels that convert each target into a valid object-part-affordance tree.

For an object without valid parts, we insert a placeholder part label `__placeholder_part__` with an object-inherited box. For a part without valid affordances, we insert a placeholder affordance label `__placeholder_action__` with a fallback spatial field. These placeholders enforce parent-child consistency without serving as fine-grained semantic ground truth. They encode valid non-expandable cases, regularize the serialized output space, and reduce invalid hierarchical generations. During evaluation, placeholder entries are removed and are not counted as semantic parts or affordances.

4.2 Curriculum Learning

Structural-completion labels define valid branch termination but provide limited fine-grained grounding signals; curriculum learning therefore controls when and how strongly this structural supervision is introduced. Inspired by curriculum learning (Bengio et al., 2009; Kumar et al., 2010; Platanios

Table 1: Hierarchical parsing results across models. Entries are object-level/scene-level percentages. FT denotes fine-tuning on SceneParser-Bench; otherwise zero-shot. Bold and underlined values indicate the best and second-best results.

Method	$L1_{\tau=0.5}$			$L2_{\tau=0.5}$			L3			Parse Rate
	P	R	F1	P	R	F1	P	R	F1	
Closed-source MLLMs										
GPT-5.5 (OpenAI, 2025)	23.8/17.0	19.5/14.1	21.5/15.4	1.3/2.1	3.0/3.5	1.8/2.6	0.4/0.7	1.0/1.3	0.6/1.3	1.8/18.7
Gemini 3.1 Pro (Google DeepMind, 2026)	67.3/47.9	48.4/12.8	56.3/20.2	11.0/11.9	6.4/4.8	8.1/6.8	6.4/5.6	2.8/2.1	3.9/3.0	5.0/16.2
Qwen3.5-plus (Qwen Team, 2026)	25.9/33.5	52.3/19.8	34.6/24.9	3.1/6.4	10.0/5.2	4.7/5.7	0.9/2.4	2.8/2.0	1.4/2.2	5.8/24.5
Open-source MLLMs										
Qwen2.5-VL-72B (Bai et al., 2025)	55.2/37.6	34.1/12.8	42.1/19.1	4.9/4.9	2.4/1.6	3.2/2.4	1.6/1.4	0.7/0.4	0.9/0.6	1.3/13.9
Qwen3-VL-235B (Qwen Team, 2025)	21.2/3.6	41.8/2.3	28.1/2.8	3.1/0.3	8.7/0.2	4.6/0.2	0.5/0.0	1.7/0.0	0.8/0.0	3.5/2.4
Qwen3.5-35B (Qwen Team, 2026)	32.9/19.4	48.9/11.2	39.4/14.2	3.8/2.9	9.6/2.3	5.5/2.6	0.6/0.6	1.4/0.5	0.8/0.5	3.2/13.5
SEED2.0 (ByteDance Seed Team, 2026)	41.6/10.4	34.8/6.6	37.9/8.1	5.7/2.3	4.4/0.7	5.0/1.1	1.0/0.3	0.7/0.1	0.8/0.2	1.6/4.0
Qwen3.5-4B (Qwen Team, 2026)	2.5/0.0	0.0/0.0	0.0/0.0	0.0/0.0	0.0/0.0	0.0/0.0	0.0/0.0	0.0/0.0	0.0/0.0	0.0/0.0
Qwen3.5-4B (FT)	**80.9**/–	34.7/–	48.5/–	37.2/–	21.4/–	27.2/–	26.0/–	8.7/–	13.0/–	10.1/–
Rex-Omni based methods										
Rex-Omni Assembly (Jiang et al., 2025)	78.3/–	24.2/–	37.0/–	**76.1**/–	13.1/–	22.4/–	0.0/–	0.0/–	0.0/–	13.6/–
Rex-Omni (FT)	49.6/–	34.5/–	40.7/–	32.4/–	17.0/–	22.3/–	21.6/–	15.1/–	17.8/–	22.5/–
Ours-3B (FT)	77.0/**59.1**	**70.9/50.7**	**73.8/54.6**	44.6/**38.1**	**42.7/36.9**	**43.6/37.5**	**29.9/26.2**	**28.9/26.4**	**29.4/26.3**	**43.8/48.4**

et al., 2019) and curriculum-style multimodal training (Li et al., 2023; Wu & Chen, 2024), we use a three-stage curriculum: first learn reliable localization from non-pseudo labels, then gradually introduce pseudo-completed targets for structural completion.

Stage 1 uses only non-pseudo supervision to stabilize object localization, part grounding, and affordance pointing. Stage 2 mixes 70% non-pseudo and 30% pseudo-completed samples to improve structural completion while preserving grounding quality. Stage 3 increases the pseudo ratio to 50% to reinforce hierarchy completion and parse consistency. Training continues across stages, with each stage initialized from the previous one and trained with stage-wise learning-rate decay.

5 Experiments

We evaluate SceneParser along four axes: (1) hierarchy-aware performance on SceneParser-Bench; (2) the effect of explicit hierarchy, structural completion, and curriculum learning; (3) transfer to conventional object detection and affordance grounding; and (4) the utility of parsed hierarchies for downstream interaction-oriented reasoning.

5.1 Main Results on SceneParser-Bench

We evaluate two settings: *object-level* parsing, where the model receives an image and a target object category, and *scene-level* parsing, where it parses all visible interactive objects from the image alone. Both require structured hierarchy generation following the schema in Appendix F. We compare closed-source MLLMs, open-source MLLMs, and a Rex-Omni Assembly baseline, where objects, parts, and affordances are predicted separately and then merged into object-part-affordance hierarchies. We further include two fine-tuned baselines: Qwen3.5-4B, trained on the same SceneParser-Bench supervision for hierarchical generation, and Rex-Omni, trained for flat object-part-affordance prediction. We report L1-L3 and ParseRate defined in Sec. 3.2 with training details in Appendix E.

Results. As shown in Table 1, existing MLLMs obtain non-trivial L1 scores but drop sharply at L2 and L3, showing that object grounding alone does not yield valid object-part-affordance chains. ParseRate further shows that structural completion is also difficult: although some MLLMs expand many objects, their low L2/L3 scores indicate incomplete or incorrectly bound part-affordance structures. Rex-Omni Assembly achieves high object-level L1 precision but fails to recover valid L3 chains, while fine-tuning Rex-Omni improves L3 F1 and ParseRate to 17.8 and 22.5, respectively, yet remains substantially below SceneParser. Under matched supervision, Qwen3.5-4B(FT) reaches 48.5/27.2/13.0 L1/L2/L3 F1 and 10.1 ParseRate, remaining below SceneParser and suggesting that the improvement cannot be explained by in-domain fine-tuning alone. SceneParser achieves the strongest L2/L3 performance and the highest ParseRate in both object-level and scene-level settings, demonstrating better cross-level binding and structural completeness. On 1K unseen VSIBench images (Yang et al., 2025), SceneParser also maintains strong hierarchical parsing performance without additional training; see Appendix J.

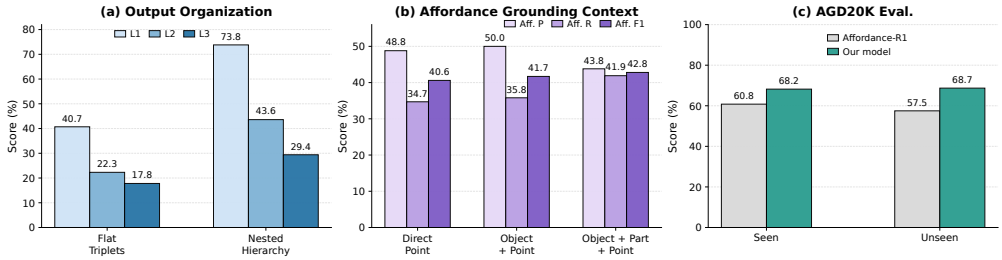

Figure 4: Analysis of hierarchy, affordance context, and cross-benchmark transfer. (a) Nested hierarchy outperforms flat triplets under identical structure-aware L1-L3 evaluation. (b) Adding object and part context improves affordance grounding, especially recall and F1. (c) SceneParser achieves higher point-in-mask accuracy than Affordance-R1 (Wang et al., 2026) on the AGD20K (Luo et al., 2022). Values are percentages.

Table 2: Effect of structural-completion pseudo labels and curriculum learning. The curriculum setting reports mean±std over three random seeds.

Pseudo Labels	Curriculum	L1 F1	L2 F1	L3 F1	ParseRate
×	×	74.1	43.1	29.1	39.6
✓	×	**74.3**	43.1	28.8	40.5
✓	✓	73.8±0.3	**43.4±0.4**	**29.4±0.1**	**43.7±0.5**

5.2 Ablation Studies

We ablate the key representation and training choices of SceneParser, including explicit hierarchical generation, structural-completion pseudo labels, and curriculum learning. We also analyze whether object and part context improves affordance grounding.

5.2.1 Effect of Hierarchical Generation and Affordance Context

Setup. We first isolate the effect of hierarchical output organization by comparing two variants with identical prediction fields: *Flat Triplets*, which predicts object, part, and affordance fields as independent triplets, and *Nested Hierarchy*, which directly preserves object-part-affordance parent-child bindings. For fair evaluation, flat triplets are deterministically converted into object-part-affordance hierarchies, and both variants are evaluated with the same structure-aware L1-L3 protocol as the main benchmark; details are provided in Appendix H.

We then test whether affordance grounding benefits from structural context by comparing *Direct Point*, *Object + Point*, and *Object + Part + Point*. All variants are evaluated with the same point-validity criterion as the main evaluation, and we report affordance precision, recall, and F1.

Results. As shown in Fig. 4(a), *Nested Hierarchy* outperforms *Flat Triplets* under the same structure-aware evaluation, improving L1 from 40.7 to 73.8, L2 from 22.3 to 43.6, and L3 from 17.8 to 29.4. Since both variants predict identical fields, the improvement can be attributed to explicit parent-child organization rather than additional prediction fields, showing that hierarchical organization facilitates coherent object-part-affordance generation.

Fig. 4(b) shows that affordance grounding improves as more structural context is provided. Object context improves affordance F1 from 40.6 to 41.7, while adding part context achieves the best F1 of 42.8 and raises recall from 34.7 to 41.9. This suggests that affordances are better grounded within explicit object-part structures than as isolated points.

5.2.2 Effect of Structural-Completion Curriculum

Setup. We compare no pseudo supervision, fixed pseudo supervision, and curriculum learning with progressively introduced pseudo-completed samples. All variants share the same architecture and evaluation protocol, with placeholders removed before computing L1–L3 and ParseRate.

Results. As shown in Table 2, fixed pseudo supervision improves ParseRate (39.6→40.5) but slightly reduces L3 (29.1→28.8). This suggests that structural-completion labels mainly help the

Table 3: COCO object detection transfer results. Object boxes are evaluated with precision, recall, and F1 at IoU thresholds of 0.5 and 0.95 across detector and MLLM baselines.

Type	Method	R@0.5	P@0.5	F1@0.5	R@.95	P@.95	F1@.95
Closed-set	DINO-R50 (Zhang et al., 2022)	62.6	76.5	68.8	17.8	25.8	21.1
Closed-set	DINO-Swin-L (Zhang et al., 2022)	69.8	82.5	75.6	22.2	29.7	25.4
Open-set	Grounding DINO-Swin-T (Liu et al., 2024)	62.8	78.4	69.8	20.5	26.2	23.0
MLLM	Qwen2.5-VL-7B (Bai et al., 2025)	75.8	55.4	64.0	10.6	15.5	12.6
MLLM	Qwen2.5-VL-3B (Bai et al., 2025)	77.2	55.7	64.7	12.7	18.3	15.0
MLLM	SEED1.5-VL (Guo et al., 2025)	78.6	65.3	71.3	12.7	16.4	14.3
MLLM	Rex-Omni-SFT (Jiang et al., 2025)	70.1	66.4	68.2	14.8	17.0	15.8
MLLM	**Ours**	65.3	68.3	66.8	13.4	12.8	13.1

model recognize valid branch termination and produce structurally complete hierarchies, while providing limited additional supervision for fine-grained grounding.

With curriculum learning, L3 rises to 29.4 ± 0.1 and ParseRate to 43.7 ± 0.5, while L2 remains stable and L1 drops only modestly. The curriculum first stabilizes object, part, and affordance grounding on non-pseudo samples, and then progressively introduces structural-completion supervision. This allows hierarchy completion to be strengthened on top of a more reliable grounding foundation, yielding a better balance between local grounding accuracy and global structural completeness. Pseudo-label ratio sensitivity is reported in Appendix I.

5.3 Cross-Task Transfer

We evaluate cross-task transfer beyond SceneParser-Bench on COCO object detection (Lin et al., 2014) and AGD20K affordance grounding (Luo et al., 2022).

COCO object detection. We extract object boxes from SceneParser's structured outputs and evaluate them using P/R/F1 at fixed IoU thresholds of 0.5 and 0.95. As shown in Table 3, SceneParser achieves 66.8 F1@0.5 from structured outputs, remaining competitive with MLLM-based detectors despite not being optimized as a standalone detector. This suggests that object-part-affordance training preserves transferable object detection.

AGD20K affordance grounding. We evaluate affordance point prediction on AGD20K using point-in-mask accuracy with valid ground-truth masks. As shown in Fig. 4(c), SceneParser achieves 68.2 and 68.7 on the seen and unseen splits, respectively, outperforming Affordance-R1 (60.8/57.5) without AGD20K fine-tuning. These results indicate that SceneParser preserves transferable object detection and affordance grounding while producing richer object-part-affordance structures.

5.4 Downstream Interaction-Oriented Reasoning

We further evaluate whether parsed hierarchies provide useful structure for downstream planning on 200 image-task pairs. We use Complete Plan Rate, which measures the percentage of plans containing all necessary action steps in a valid execution order. Adding the SceneParser hierarchy improves the score from 28% to 43% for GPT-5.5 and from 29% to 71% for Claude Opus 4.7. These gains suggest that explicit object-part-affordance structure provides useful intermediate information for generating more complete interaction-oriented plans. Detailed quantitative results and qualitative examples are provided in Appendix L.

6 Conclusion

We introduced an interaction-oriented formulation of hierarchical scene parsing that represents scenes as explicit *scene* $\rightarrow$ *object* $\rightarrow$ *part* $\rightarrow$ *affordance* hierarchies. Unlike conventional localization outputs, this formulation requires cross-level bindings, grounding each affordance in the functional part and object that support it. We instantiated this task with SceneParser-Bench, a large-scale hierarchical benchmark, and SceneParser, a VLM-based parser trained with unified hierarchical generation, structural-completion pseudo labels, and curriculum learning. Experiments show that existing MLLMs and hierarchy assembly pipelines struggle to recover valid object-part-affordance chains, while SceneParser achieves stronger hierarchy-aware parsing, generalizes to unseen data, preserves transferable grounding ability, and provides useful structured information for downstream interaction-oriented planning.

AI USE STATEMENT

Generative AI models were used as components of the research methodology. GPT-5 was used in the SceneParser-Bench data engine to generate object descriptions, candidate functional parts, and affordance-region descriptions. GPT-5.5 and Claude Opus 4.7 were used as evaluated planners in the downstream interaction-oriented reasoning probe. AI-generated supervision was subject to filtering, hierarchical consistency checks, and human quality auditing. The authors take responsibility for the methodology, analyses, and reported results.

REFERENCES

Shuai Bai, Keqin Chen, Xuejing Liu, et al. Qwen2.5-vl technical report. *arXiv preprint arXiv:2502.13923*, 2025. 7, 9

Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. Curriculum learning. In *Proceedings of the International Conference on Machine Learning*, 2009. 6

Anthony Brohan, Noah Brown, Justice Carbajal, Yevgen Chebotar, Xi Chen, Krzysztof Choromanski, Tianli Ding, Danny Driess, Avinava Dubey, Chelsea Finn, et al. Rt-2: Vision-language-action models transfer web knowledge to robotic control, 2023. *URL https://arxiv. org/abs/2307.15818*, 2024. 1

ByteDance Seed Team. Seed2.0 model card: Towards intelligence frontier for real-world complexity. ByteDance Seed technical report, 2026. 7

Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *European conference on computer vision(ECCV)*, 2020. 3

Nicolas Carion, Laura Gustafson, Yuan-Ting Hu, Shoubhik Debnath, Ronghang Hu, Didac Suris, Chaitanya Ryali, Kalyan Vasudev Alwala, Haitham Khedr, Andrew Huang, Jie Lei, Tengyu Ma, Baishan Guo, Arpit Kalla, Markus Marks, Joseph Greer, Meng Wang, Peize Sun, Roman Rädle, Triantafyllos Afouras, Effrosyni Mavroudi, Katherine Xu, Tsung-Han Wu, Yu Zhou, Liliane Momeni, Rishi Hazra, Shuangrui Ding, Sagar Vaze, Francois Porcher, Feng Li, Siyuan Li, Aishwarya Kamath, Ho Kei Cheng, Piotr Dollár, Nikhila Ravi, Kate Saenko, Pengchuan Zhang, and Christoph Feichtenhofer. Sam 3: Segment anything with concepts, 2025. URL https://arxiv.org/abs/2511.16719. 5

Keqin Chen, Zhao Zhang, Weili Zeng, Richong Zhang, Feng Zhu, and Rui Zhao. Shikra: Unleashing multimodal llm's referential dialogue magic. *arXiv preprint arXiv:2306.15195*, 2023. 1, 3

Tianheng Cheng, Lin Song, Yixiao Ge, Wenyu Liu, Xinggang Wang, and Ying Shan. Yolo-world: Real-time open-vocabulary object detection. In *Proceedings of the IEEE/CVF conference on Computer Vision and Pattern Recognition(CVPR)*, 2024. 1, 3

Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *IJCV*, 2010. 1, 4

Shenghao Fu, Qize Yang, Qijie Mo, Junkai Yan, Xihan Wei, Jingke Meng, Xiaohua Xie, and Wei-Shi Zheng. Llmdet: Learning strong open-vocabulary object detectors under the supervision of large language models. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, 2025. 1

Google DeepMind. Gemini 3.1 pro model card. https://deepmind.google/models/model-cards/gemini-3-1-pro/, 2026. 7

Dong Guo, Faming Wu, Feida Zhu, Fuxing Leng, Guang Shi, Haobin Chen, Haoqi Fan, Jian Wang, Jianyu Jiang, Jiawei Wang, et al. Seed1.5-vl technical report. *arXiv preprint arXiv:2505.07062*, 2025. 9

Ju He, Shuo Yang, Shaokang Yang, Adam Kortylewski, Xiaoding Yuan, Jie-Neng Chen, Shuai Liu, Cheng Yang, Qihang Yu, and Alan Yuille. Partimagenet: A large, high-quality dataset of parts. In *European Conference on Computer Vision(ECCV)*, 2022. 1, 3

Jiacheng Hua, Yishu Yin, Yuhang Wu, Tai Wang, Yifei Huang, and Miao Liu. Unleashing spatial reasoning in multimodal large language models via textual representation guided reasoning. *arXiv preprint arXiv:2603.23404*, 2026. 2

Qing Jiang, Gen Luo, Yuqin Yang, Yuda Xiong, Yihao Chen, Zhaoyang Zeng, Tianhe Ren, and Lei Zhang. Chatrex: Taming multimodal llm for joint perception and understanding. *arXiv preprint arXiv:2411.18363*, 2024. 1, 3

Qing Jiang, Junan Huo, Xingyu Chen, Yuda Xiong, Zhaoyang Zeng, Yihao Chen, Tianhe Ren, Junzhi Yu, and Lei Zhang. Detect anything via next point prediction. *arXiv preprint arXiv:2510.12798*, 2025. 1, 3, 5, 6, 7, 9, 17

M Kumar, Benjamin Packer, and Daphne Koller. Self-paced learning for latent variable models. *Advances in Neural Information Processing Systems(NeuIPS)*, 2010. 6

Chunyuan Li, Cliff Wong, Sheng Zhang, Naoto Usuyama, Haotian Liu, Jianwei Yang, Tristan Naumann, Hoifung Poon, and Jianfeng Gao. Llava-med: Training a large language-and-vision assistant for biomedicine in one day. *Advances in Neural Information Processing Systems(NeuIPS)*, 2023. 7

Junyi Li, Junfeng Wu, Weizhi Zhao, Song Bai, and Xiang Bai. Partglee: A foundation model for recognizing and parsing any objects. In *European Conference on Computer Vision(ECCV)*, 2024a. 3, 4

Liunian Harold Li, Pengchuan Zhang, Haotian Zhang, Jianwei Yang, Chunyuan Li, Yiwu Zhong, Lijuan Wang, Lu Yuan, Lei Zhang, Jenq-Neng Hwang, et al. Grounded language-image pre-training. In *Proceedings of the IEEE/CVF conference on Computer Vision and Pattern Recognition(CVPR)*, 2022. 1, 3

Rongjie Li, Songyang Zhang, Dahua Lin, Kai Chen, and Xuming He. From pixels to graphs: Open-vocabulary scene graph generation with vision-language models. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 28076–28086. IEEE, 2024b. 4

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *ECCV*, 2014. 1, 4, 9

Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Qing Jiang, Chunyuan Li, Jianwei Yang, Hang Su, et al. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. In *European conference on computer vision(ECCV)*, 2024. 1, 3, 5, 9

Ruijie Lu, Yiyang Ma, Xiaokang Chen, Lingxiao Luo, Zhiyu Wu, Zizheng Pan, Xingchao Liu, Yutong Lin, Hao Li, Wen Liu, Zhewen Hao, Xi Gao, Shaoheng Nie, Yixuan Wei, Zhenda Xie, Ting Chen, and Gang Zeng. Thinking with visual primitives. https://github.com/deepseek-ai/Thinking-with-Visual-Primitives, 2026. Technical report. 2

Yadong Lu, Jianwei Yang, Yelong Shen, and Ahmed Awadallah. Omniparser for pure vision based gui agent. *arXiv preprint arXiv:2408.00203*, 2024. 2

Hongchen Luo, Wei Zhai, Jing Zhang, Yang Cao, and Dacheng Tao. Learning affordance grounding from exocentric images. In *Proceedings of the IEEE/CVF conference on Computer Vision and Pattern Recognition*, 2022. 1, 3, 8, 9

Chuofan Ma, Yi Jiang, Jiannan Wu, Zehuan Yuan, and Xiaojuan Qi. Groma: Localized visual tokenization for grounding multimodal large language models. In *European Conference on Computer Vision(ECCV)*, 2024. 1, 3

Arjun Majumdar, Anurag Ajay, Xiaohan Zhang, Pranav Putta, Sriram Yenamandra, Mikael Henaff, Sneha Silwal, Paul Mcvay, Oleksandr Maksymets, Sergio Arnaud, et al. Openeqa: Embodied question answering in the era of foundation models. In *Proceedings of the IEEE/CVF conference on Computer Vision and Pattern Recognition(CVPR)*, 2024. 1

Austin Myers, Ching L Teo, Cornelia Fermüller, and Yiannis Aloimonos. Affordance detection of tool parts from geometric features. In *IEEE international conference on robotics and automation(ICRA)*, 2015. 3

Anh Nguyen, Dimitrios Kanoulas, Darwin G Caldwell, and Nikos G Tsagarakis. Object-based affordances detection with convolutional neural networks and dense conditional random fields. In *IEEE/RSJ International Conference on Intelligent Robots and Systems(IROS)*, 2017. 3

OpenAI. Gpt-5 system card. https://openai.com/index/gpt-5-system-card/, 2025. 5, 7

Jae Sung Park, Zixian Ma, Linjie Li, Chenhao Zheng, Cheng-Yu Hsieh, Ximing Lu, Khyathi Chandu, Quan Kong, Norimasa Kobori, Ali Farhadi, et al. Synthetic visual genome. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9073–9086. IEEE, 2025. 4

Zhiliang Peng, Wenhui Wang, Li Dong, Yaru Hao, Shaohan Huang, Shuming Ma, and Furu Wei. Kosmos-2: Grounding multimodal large language models to the world. *arXiv preprint arXiv:2306.14824*, 2023. 3

Emmanouil Antonios Platanios, Otilia Stretcu, Graham Neubig, Barnabas Poczos, and Tom Mitchell. Competence-based curriculum learning for neural machine translation. In *Proceedings of the conference of the North American chapter of the association for computational linguistics*, 2019. 6

Shengyi Qian, Weifeng Chen, Min Bai, Xiong Zhou, Zhuowen Tu, and Li Erran Li. Affordancellm: Grounding affordance from vision language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition(CVPR)*, 2024. 1, 3

Qwen Team. Qwen3-vl technical report. *arXiv preprint arXiv:2511.21631*, 2025. 7

Qwen Team. Qwen3.5: Towards native multimodal agents. https://qwen.ai/blog?id=qwen3.5, 2026. 7

Vignesh Ramanathan, Anmol Kalia, Vladan Petrovic, Yi Wen, Baixue Zheng, Baishan Guo, Rui Wang, Aaron Marquez, Rama Kovvuri, Abhishek Kadian, et al. Paco: Parts and attributes of common objects. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition(CVPR)*, 2023. 1, 3

Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition(CVPR)*, 2016. 3

Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in Neural Information Processing Systems*, 2015. 3

Tianhe Ren, Yihao Chen, Qing Jiang, Zhaoyang Zeng, Yuda Xiong, Wenlong Liu, Zhengyu Ma, Junyi Shen, Yuan Gao, Xiaoke Jiang, et al. Dino-x: A unified vision model for open-world object detection and understanding. *arXiv preprint arXiv:2411.14347*, 2024a. 3

Tianhe Ren, Qing Jiang, Shilong Liu, Zhaoyang Zeng, Wenlong Liu, Han Gao, Hongjie Huang, Zhengyu Ma, Xiaoke Jiang, Yihao Chen, et al. Grounding dino 1.5: Advance the edge of open-set object detection. *arXiv preprint arXiv:2405.10300*, 2024b. 1

Jianhua Sun and Cewu Lu. Digital gene: Learning about the physical world through analytic concepts. *arXiv preprint arXiv:2504.04170*, 2025. 1

Peize Sun, Shoufa Chen, Chenchen Zhu, Fanyi Xiao, Ping Luo, Saining Xie, and Zhicheng Yan. Going denser with open-vocabulary part segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision(ICCV)*, 2023. 3

Zifu Wan, Yaqi Xie, Ce Zhang, Zhiqiu Lin, Zihan Wang, Simon Stepputtis, Deva Ramanan, and Katia P Sycara. Instructpart: Task-oriented part segmentation with instruction reasoning. In *Proceedings of the Meeting of the Association for Computational Linguistics*, 2025. 3

Hanqing Wang, Shaoyang Wang, Yiming Zhong, Zemin Yang, Jiamin Wang, Zhiqing Cui, Jiahao Yuan, Yifan Han, Mingyu Liu, and Yuexin Ma. Affordance-r1: Reinforcement learning for generalizable affordance reasoning in multimodal large language models. In *Proceedings of the AAAI Conference on Artificial Intelligence(AAAI)*, 2026. 1, 3, 8

Jiacong Wang, Zijian Kang, Haochen Wang, Haiyong Jiang, Jiawen Li, Bohong Wu, Ya Wang, Jiao Ran, Xiao Liang, Chao Feng, et al. Vgr: Visual grounded reasoning. *arXiv preprint arXiv:2506.11991*, 2025. 1

Tai Wang, Xiaohan Mao, Chenming Zhu, Runsen Xu, Ruiyuan Lyu, Peisen Li, Xiao Chen, Wenwei Zhang, Kai Chen, Tianfan Xue, et al. Embodiedscan: A holistic multi-modal 3d perception suite towards embodied ai. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition(CVPR)*, 2024. 1

Biao Wu and Ling Chen. Curriculum learning with quality-driven data selection. *arXiv preprint arXiv:2407.00102*, 2024. 7

Shengqiong Wu, Hao Fei, and Tat-Seng Chua. Universal scene graph generation. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 14158–14168. IEEE, 2025. 4

Bin Xiao, Haiping Wu, Weijian Xu, Xiyang Dai, Houdong Hu, Yumao Lu, Michael Zeng, Ce Liu, and Lu Yuan. Florence-2: Advancing a unified representation for a variety of vision tasks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition(CVPR)*, 2024. 1

Linhui Xiao, Xiaoshan Yang, Xiangyuan Lan, Yaowei Wang, and Changsheng Xu. Towards visual grounding: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence(TPAMI)*, 2025. 1

Jihan Yang, Shusheng Yang, Anjali W Gupta, Rilyn Han, Li Fei-Fei, and Saining Xie. Thinking in space: How multimodal large language models see, remember, and recall spaces. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10632–10643. IEEE, 2025. 7

Jingkang Yang, Yi Zhe Ang, Zujin Guo, Kaiyang Zhou, Wayne Zhang, and Ziwei Liu. Panoptic scene graph generation. In *European conference on computer vision*, pp. 178–196. Springer, 2022. 4

Benjamin Yao, Xiong Yang, and Song-Chun Zhu. Introduction to a large-scale general purpose ground truth database: methodology, annotation tool and benchmarks. In *International Workshop on Energy Minimization Methods in Computer Vision and Pattern Recognition*, pp. 169–183. Springer, 2007. 3

Haoxuan You, Haotian Zhang, Zhe Gan, Xianzhi Du, Bowen Zhang, Zirui Wang, Liangliang Cao, Shih-Fu Chang, and Yinfei Yang. Ferret: Refer and ground anything anywhere at any granularity. *arXiv preprint arXiv:2310.07704*, 2023. 3

Wentao Yuan, Jiafei Duan, Valts Blukis, Wilbert Pumacay, Ranjay Krishna, Adithyavairavan Murali, Arsalan Mousavian, and Dieter Fox. Robopoint: A vision-language model for spatial affordance prediction for robotics. *arXiv preprint arXiv:2406.10721*, 2024. 1, 3

Hao Zhang, Feng Li, Shilong Liu, Lei Zhang, Hang Su, Jun Zhu, Lionel M Ni, and Heung-Yeung Shum. Dino: Detr with improved denoising anchor boxes for end-to-end object detection. *arXiv preprint arXiv:2203.03605*, 2022. 3, 9

Haotian Zhang, Haoxuan You, Philipp Dufter, Bowen Zhang, Chen Chen, Hong-You Chen, Tsu-Jui Fu, William Yang Wang, Shih-Fu Chang, Zhe Gan, et al. Ferret-v2: An improved baseline for referring and grounding with large language models. *arXiv preprint arXiv:2404.07973*, 2024. 1

Ruimao Zhang, Liang Lin, Guangrun Wang, Meng Wang, and Wangmeng Zuo. Hierarchical scene parsing by weakly supervised learning with image descriptions. *IEEE transactions on pattern analysis and machine intelligence*, pp. 596–610, 2018. 3

Chengyang Zhao, Yikang Shen, Zhenfang Chen, Mingyu Ding, and Chuang Gan. Textpsg: Panoptic scene graph generation from textual descriptions. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 2827–2838. IEEE, 2023. 4

Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Scene parsing through ade20k dataset. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition(CVPR)*, 2017. 1, 3

Chenchen Zhu, Fanyi Xiao, Andrés Alvarado, Yasmine Babaei, Jiabo Hu, Hichem El-Mohri, Sean Culatana, Roshan Sumbaly, and Zhicheng Yan. Egoobjects: A large-scale egocentric dataset for fine-grained object understanding. In *Proceedings of the IEEE/CVF international conference on computer vision(ICCV)*, 2023. 5

A Formal Definition of Evaluation Metrics

Overview. We provide the formal definition of the structure-aware evaluation protocol used in Sec. 3.2. The protocol evaluates three conditional levels: L1 for objects, L2 for object-part chains, and L3 for full object-part-affordance chains. At all levels, predictions are matched to ground truth using greedy one-to-one assignment, and precision, recall, and F1 are computed from accumulated true positives, false positives, and false negatives.

Precision, recall, and F1. For each level $\ell \in \{1, 2, 3\}$ and IoU threshold τ, we compute

$$\mathrm{P}_{\ell,\tau} = \frac{\mathrm{TP}_{\ell,\tau}}{\mathrm{TP}_{\ell,\tau} + \mathrm{FP}_{\ell,\tau}}, \quad \mathrm{R}_{\ell,\tau} = \frac{\mathrm{TP}_{\ell,\tau}}{\mathrm{TP}_{\ell,\tau} + \mathrm{FN}_{\ell,\tau}},$$

$$\mathrm{F1}_{\ell,\tau} = \frac{2\mathrm{P}_{\ell,\tau}\mathrm{R}_{\ell,\tau}}{\mathrm{P}_{\ell,\tau} + \mathrm{R}_{\ell,\tau}}.$$

Here $\mathrm{TP}_{\ell,\tau}$, $\mathrm{FP}_{\ell,\tau}$, and $\mathrm{FN}_{\ell,\tau}$ are accumulated over the evaluation set under the hierarchical matching constraints defined below.

L1: object matching. At L1, predicted objects are matched to ground-truth objects. A predicted object is counted as a true positive if its object name matches the ground-truth object name and its object box satisfies

$$\mathrm{IoU}(b_o^{pred}, b_o^{gt}) \geq \tau.$$

Unmatched predicted objects are counted as false positives, and unmatched ground-truth objects are counted as false negatives.

L2: object-part matching. At L2, part matching is performed only within correctly matched object pairs from L1. A predicted object-part chain is counted as a true positive if the parent object is correctly matched, the part name matches, and the part box satisfies

$$\mathrm{IoU}(b_p^{pred}, b_p^{gt}) \geq \tau.$$

Thus, a part can be correct only when its parent object is also correct. Unmatched predicted parts under valid matched objects are counted as false positives, and unmatched ground-truth parts under matched objects are counted as false negatives.

L3: object-part-affordance matching. At L3, affordance matching is performed only within valid object-part matches from L2. A predicted object-part-affordance chain is counted as a true positive if the parent object and part are correctly matched, the action label matches, and the predicted interaction point is spatially valid:

$$u^{pred} \in R_a^{gt},$$

where R_a^{gt} denotes the ground-truth valid region, defined as the ground-truth affordance box when available and otherwise the parent part box. Thus, L3 correctness requires object localization, part localization, action recognition, point grounding, and valid cross-level binding.

ParseRate. ParseRate measures whether a matched object is structurally completed with its required lower-level hierarchy. We define a ground-truth object as *parse-eligible* if it contains parts and/or affordances and therefore requires structural expansion. Given a parse-eligible GT object that is correctly matched at IoU threshold 0.5, its prediction is counted as complete if it satisfies the required expansion pattern: it predicts parts when GT parts exist, affordances when GT affordances exist, and both when both are required.

Let $\mathcal{E}$ denote the set of parse-eligible GT objects in the evaluation set, and let $\mathcal{C} \subseteq \mathcal{E}$ denote the subset whose matched predictions satisfy the required structural expansion. We define

$$\mathrm{ParseRate}_{0.5} = \frac{|\mathcal{C}|}{|\mathcal{E}|}.$$

We report ParseRate under two settings. For object-level evaluation, $\mathcal{E}$ contains parse-eligible target objects specified by the input category. For scene-level evaluation, $\mathcal{E}$ contains all parse-eligible objects across all scenes. Thus, object-level ParseRate measures target-conditioned hierarchy completion, while scene-level ParseRate measures full-scene hierarchical coverage and consistency.

Table 4: Human audit of SceneParser-Bench annotations on 1,000 randomly sampled validation images.

Type	Level / Relation	Precision (%)	Recall / Accuracy (%)
Grounding annotation	Object annotation	93.8	92.1
	Part annotation	94.6	93.0
	Affordance annotation	95.8	96.8
Relation annotation	Object–Part binding	–	92.4
	Part–Affordance binding	–	92.8
	Object–Part–Affordance binding	–	90.4

B Annotation Quality Analysis

We conduct an independent human audit on 1,000 randomly sampled validation images. Five annotators verify object, part, affordance, and hierarchical-relation annotations, with disagreements resolved through adjudication.

Node-level metrics evaluate semantic labels and localization, while relation metrics additionally require correct parent–child bindings. All node and pairwise relation scores exceed 92%, and the complete object–part–affordance chain achieves 90.4% accuracy, demonstrating high annotation reliability.

C Data Engine Illustration

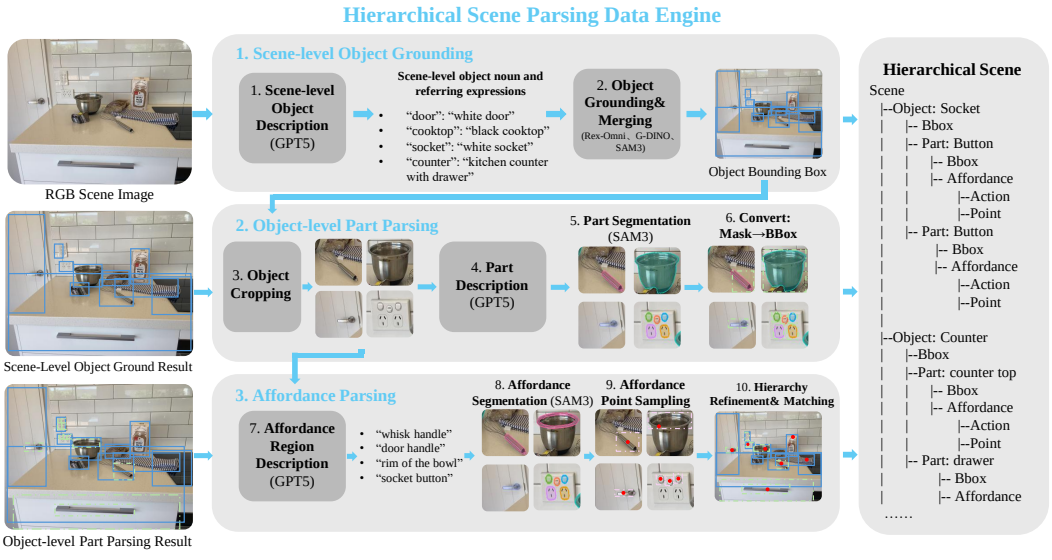

Figure 5: Overview of the proposed data engine. Starting from an RGB scene image, the engine progressively constructs hierarchical supervision through scene-level object grounding, object-level part parsing, affordance parsing, and hierarchy reconstruction. The final output is organized as a unified *scene* → *object* → *part* → *affordance* structure for training and evaluation.

D Output Serialization and Coordinate Tokenization

SceneParser generates a serialized JSON-style hierarchy following the object-part-affordance formulation in Sec. 3.1. A compact training target is represented as:

Serialized Object-Part-Affordance Target

```
{
  "objects": [{
    "name": "microwave",
    "bbox": "<x1><y1><x2><y2>",
    "parts": [{
      "part_name": "door",
      "bbox": "<x1><y1><x2><y2>",
      "affordances": [
        {"action": "open",  "point": "<x><y>"},
      ]
    }, {
      "part_name": "button panel",
      "bbox": "<x1><y1><x2><y2>",
      "affordances": [
        {"action": "press", "point": "<x><y>"}
      ]
    }]
  }, {
    "name": "drawer",
    "bbox": "<x1><y1><x2><y2>",
    "parts": [{
      "part_name": "handle",
      "bbox": "<x1><y1><x2><y2>",
      "affordances": [
        {"action": "pull", "point": "<x><y>"}
      ]
    }]
  }]
}
```

Spatial outputs follow the Rex-Omni coordinate-token protocol (Jiang et al., 2025). For each image, continuous coordinates are first converted to relative image coordinates and quantized into 1000 discrete bins, corresponding to positions from 0 to 999 along each image axis. Each bin is represented by a dedicated coordinate token, so a bounding box is serialized as four spatial tokens (x_1, y_1, x_2, y_2) and an affordance point as two spatial tokens (x, y). These coordinate tokens are inserted directly into the JSON-style hierarchy and generated together with object names, part names, and affordance actions in a single autoregressive sequence. This allows SceneParser to decode a unified hierarchy rather than separate object, part, and affordance predictions.

E Training and Implementation Details

We use the serialization and coordinate-token protocol described in Appendix D. This section provides the exact curriculum schedule and training hyperparameters used for SceneParser.

Table 5: Training schedule of the structural-completion curriculum. Stages are trained sequentially, with each stage initialized from the previous one.

Stage	Supervision	Epochs	Main/MM LR	Vision LR
Stage 1	No-pseudo	3	2×10^{-5}	2×10^{-6}
Stage 2	70% non-pseudo / 30% pseudo	4	1×10^{-5}	1×10^{-6}
Stage 3	50% non-pseudo / 50% pseudo	3	6×10^{-6}	6×10^{-7}

All stages use AdamW with cosine learning-rate decay, warmup ratio 0.03, weight decay 0.01, maximum gradient norm 1.0, bf16 mixed precision, gradient checkpointing, and maximum sequence length 4096. We train with per-device batch size 2 and gradient accumulation 8 on 16 A100-80GB GPUs across two nodes, giving an effective global batch size of 256. The complete three-stage curriculum takes about 24 wall-clock hours on this setup, corresponding to roughly 384 GPU-hours. Training is implemented with `torchrun` and DeepSpeed ZeRO-2. An anonymized supplementary evaluation package is provided to reproduce the reported L1/L2/L3 and ParseRate from saved predictions. We are organizing the full training code, benchmark annotations, and evaluation scripts, which will be released at https://github.com/SceneParser/SceneParser.

F Evaluation Prompts

F.1 Object-Level Prompt

Object-Level Prompt

```
You are given one image.

Target category: [OBJ]

Return JSON only with schema:
{
  "objects": [
    {
      "name": "string",
      "bbox": [x1,y1,x2,y2],
      "parts": [
        {
          "part_name": "string",
          "bbox": [x1,y1,x2,y2],
          "affordances": [
            {"action":"string","point":[x,y]}
          ]
        }
      ]
    }
  ]
}

Rules:
- Focus on [OBJ] only.
- If absent, return {"objects": []}.
- No extra keys. No explanation.
```

F.2 Scene-Level Prompt

Scene-Level Prompt

```
You are given one image.

Return JSON only, using exactly this schema:
{
  "objects":[{
    "name":"string","bbox":[x1,y1,x2,y2],
    "parts":[{
      "part_name":"string","bbox":[x1,y1,x2,y2],
      "affordances":[
        {"action":"string","point":[x,y]}
      ]}
    ]}
  ]
}

Constraints:
1. Parse all visible interactive objects.
2. If no valid object, return {"objects":[]}.
3. Always include parts and affordances arrays (empty if none).
4. Do not output extra keys or explanations.
5. No placeholder literals like "string", "object", "part", "action".
```

G Failure Case Analysis

We further analyze why strong MLLMs achieve low L3 performance. All evaluated models have 0% invalid serialization, ruling out formatting as the main bottleneck. As shown in Table 6, errors mainly arise from object omission, fine-grained part failures, and cross-level error propagation. In particular, part failures block 51.5% and 55.0% of affordances for Gemini 3.1 Pro and Qwen3.5-Plus, respectively, indicating that upstream parsing errors substantially affect complete-chain recovery.

Table 6: Failure analysis of representative models (%). "Obj. omit." and "Part omit." denote missing object and part predictions; "Part mismatch" denotes incorrect part names; "Aff. blocked" denotes affordances that cannot receive credit due to upstream part failures.

Method	Obj. omit.	Part omit.	Part mismatch	Aff. blocked
GPT-5.5	44.6	10.2	12.2	25.9
Gemini 3.1 Pro	32.2	40.7	12.7	51.5
Qwen3.5-Plus	27.5	24.6	31.0	55.0
Ours	11.9	24.3	11.9	40.1

H Flat-Triplet Output and Hierarchy Conversion

Flat-triplet output format. For the flat-output ablation in Sec. 5.2, we remove the explicit parent-child nesting and ask the model to generate a list of object-part-affordance triplets. Each triplet contains the object name and box, part name and box, action label, and affordance point.

Why flat triplets? The flat-triplet format is designed as a controlled counterpart to the nested hierarchy. It preserves the same prediction fields as SceneParser, including object names and boxes, part names and boxes, affordance actions, and interaction points, but removes the explicit object-part-affordance nesting. This allows us to isolate the effect of output organization: any performance gap between flat triplets and nested hierarchy mainly reflects whether explicit parent-child structure helps generation, rather than differences in predicted information.

Flat-Triplet Output Example

```
{
  "triplets": [
    {
      "object": "microwave",
      "object_box": "<x1><y1><x2><y2>",
      "part": "door",
      "part_box": "<x1><y1><x2><y2>",
      "action": "open",
      "affordance_point": "<x><y>"
    },
    {
      "object": "microwave",
      "object_box": "<x1><y1><x2><y2>",
      "part": "button panel",
      "part_box": "<x1><y1><x2><y2>",
      "action": "press",
      "affordance_point": "<x><y>"
    },
    {
      "object": "drawer",
      "object_box": "<x1><y1><x2><y2>",
      "part": "handle",
      "part_box": "<x1><y1><x2><y2>",
      "action": "pull",
      "affordance_point": "<x><y>"
    }
  ]
}
```

Flat-to-hierarchy conversion. To compare flat triplets with nested hierarchy under the same structure-aware evaluation, we convert flat predictions into the standard object-part-affordance hierarchy before scoring. This deterministic post-processing is applied only to predictions, while the ground truth remains unchanged. Each flat triplet contains an object name and box, a part name and box, an action label, and an affordance point. We group triplets by (object name, object box) to create object nodes, then by (part name, part box) within each object to create part nodes. Each (action, affordance point) pair is attached to its corresponding part, with duplicate affordances removed. The converted prediction follows the same `objects` → `parts` → `affordances` structure as SceneParser outputs.

Evaluation after conversion. After conversion, both *Flat Triplets* and *Nested Hierarchy* are evaluated using the main structure-aware L1-L3 protocol in Sec. 3.2. Thus, the ablation isolates output organization: both variants predict the same fields and use the same hierarchical metrics, differing only in whether parent-child bindings are generated natively or reconstructed post hoc.

I Pseudo-Label Ratio Sensitivity

We further vary the Stage-3 pseudo-label ratio from 30% to 50%. L1–L3 remain stable, while ParseRate improves from 41.4 to 43.2, suggesting that stronger structural-completion supervision mainly benefits hierarchy completion.

Table 7: Sensitivity to the Stage-3 pseudo-label ratio.

Pseudo Ratio	L1 F1	L2 F1	L3 F1	ParseRate
30%	73.6	43.0	29.1	41.4
50%	73.5	42.9	29.3	43.2

J External Generalization Evaluation

We further evaluate external generalization on 1,000 unseen VSIBench images. To control for in-domain supervision, we compare SceneParser with Qwen3.5-4B fine-tuned on the same SceneParser-Bench training data. Neither model is further trained or adapted on VSIBench, and both are evaluated using the same normalization and hierarchy-aware evaluation protocol.

Table 8: External generalization results on 1,000 unseen VSIBench images.

Method	L1 F1	L2 F1	L3 F1	ParseRate
Qwen3.5-4B (FT)	38.9	16.2	11.2	13.5
SceneParser	**66.6**	**34.3**	**23.7**	**58.6**

SceneParser retains substantially stronger hierarchical parsing performance on external images, suggesting that its advantage is not limited to the SceneParser-Bench validation distribution.

K Qualitative Visualization

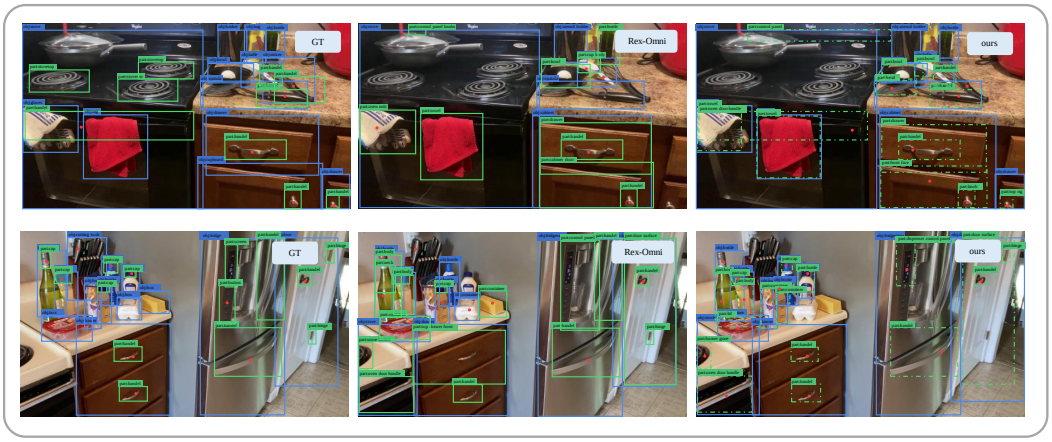

Figure 6: Qualitative comparison among ground truth, Rex-Omni Stitching, and SceneParser. Compared with the Rex-Omni Assembly baseline, SceneParser produces more coherent object-part-affordance hierarchies with better parent-child bindings and affordance grounding.

L Downstream Interaction-Oriented Reasoning Probe

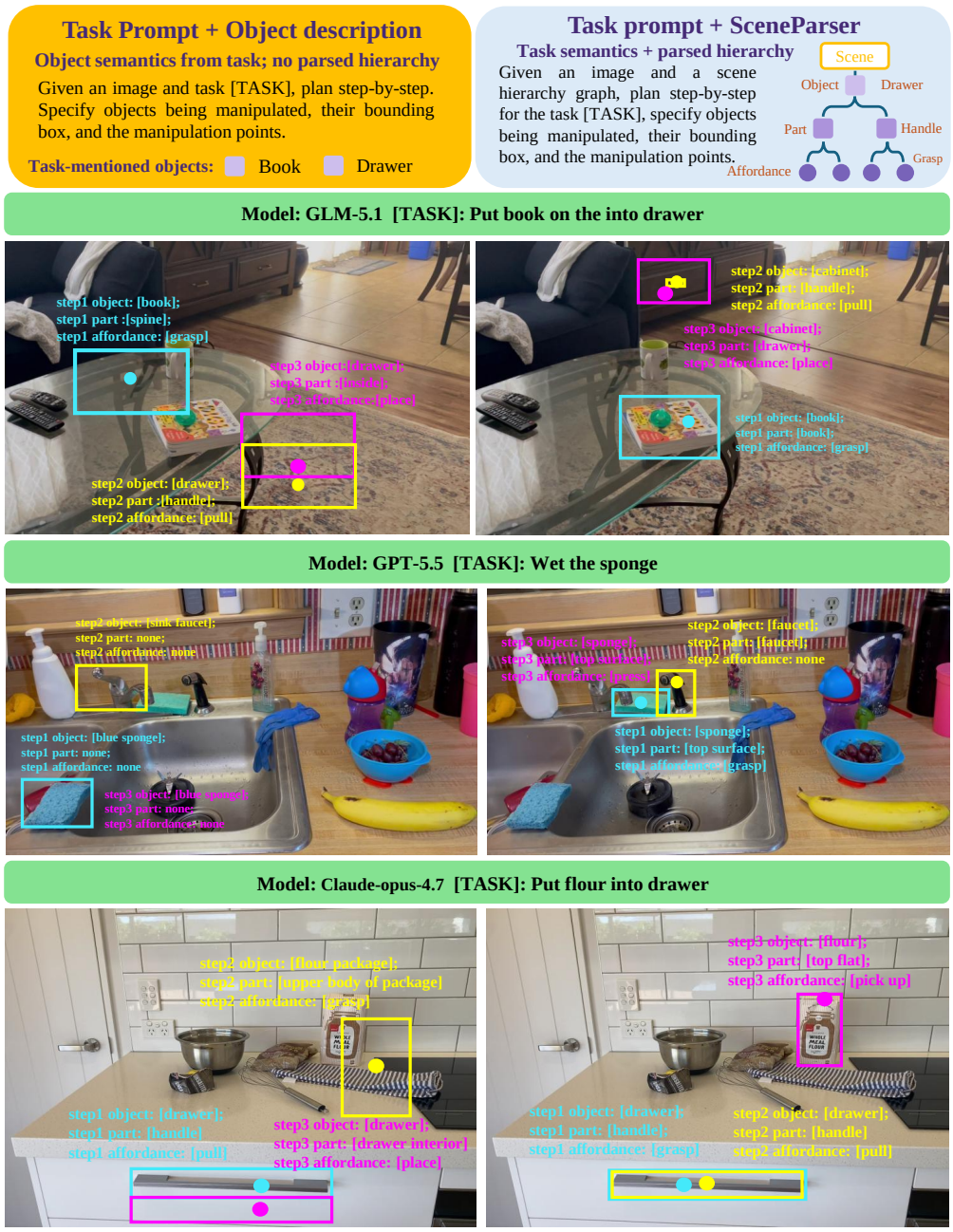

Figure 7: Qualitative downstream interaction-oriented reasoning probe. We compare a task-only prompt, which already provides task-level object semantics, with a structure-augmented prompt that additionally provides the object-part-affordance hierarchy produced by SceneParser. The hierarchy exposes functional parts, affordance labels, and interaction points, helping the planner generate more coherent interaction steps and more accurate object-part-affordance grounding.

We conduct a controlled evaluation on 200 image–task pairs using GPT-5.5 and Claude Opus 4.7 under two prompting conditions. In the *task-only* setting, the planner receives the RGB image and task instruction and generates a step-by-step manipulation plan with target objects, bounding boxes, and interaction points. In the *structure-augmented* setting, the planner additionally receives the SceneParser hierarchy, which explicitly links objects to functional parts, affordance labels, and interaction points. All other prompts and generation settings are kept unchanged.

For evaluation, human annotators specify the required key steps and corresponding object–part–affordance targets for each task. We report *Complete Plan Rate*, defined as the proportion of tasks for which all necessary action steps are generated without omission and arranged in a valid execution order. A plan is counted as complete only when it covers the required procedure and preserves the correct step ordering.

Table 9: Quantitative downstream planning results on 200 image–task pairs. Complete Plan Rate (%) measures the percentage of tasks with all required steps generated in a valid execution order.

Planner	**Task Only**	**+ SceneParser**
GPT-5.5	28	**43**
Claude Opus 4.7	29	**71**

As shown in Table 9, providing the SceneParser hierarchy consistently improves planning completeness for both planners. Complete Plan Rate increases from 28% to 43% for GPT-5.5 and from 29% to 71% for Claude Opus 4.7, indicating that explicit object–part–affordance structure provides useful intermediate information for downstream planning.

Qualitative examples in Fig. 7 further illustrate this effect. Task-only prompting may identify task-relevant objects but miss functional parts, affordance assignments, or necessary intermediate steps. With SceneParser, the planner more consistently grounds actions to explicit object–part–affordance chains, such as selecting the drawer handle for pulling and the drawer interior for placing. These examples complement the quantitative results and illustrate how hierarchical scene representations can support more complete interaction-oriented plans.

M LIMITATIONS

SceneParser remains limited to image-centric 2D representations and does not model 3D geometry, dynamics, force constraints, or manipulation trajectories. SceneParser-Bench may also inherit noise from its automated data engine despite filtering and human quality audits. Moreover, long-tailed object-part-affordance compositions remain challenging, and future work should evaluate the representation in closed-loop robotic interaction.