GROUNDBRICKS: A CONFIGURABLE BENCHMARK FOR COMPOSITIONAL VISUAL GROUNDING

Anonymous authors
Paper under double-blind review

ABSTRACT

Visual grounding aims to localize image regions specified by language queries. Existing benchmarks, however, often focus on direct query–object matching, providing limited and systematic control over the diversity and complexity of grounding tasks. Constructing richer benchmarks typically requires additional region-level annotations, which are costly and difficult to scale. We argue that the required localization supervision already exists. Current object detection and instance segmentation datasets contain abundant verified region annotations. What is missing is a mechanism for composing these annotations into diverse and controllable grounding tasks. Based on this insight, we introduce GroundBricks, a configurable benchmark for compositional visual grounding, together with a fully automated construction pipeline. GroundBricks covers five capability categories: referential grounding, set grounding, attribute reading, relational chaining, and knowledge-grounded selection. It also spans three complexity levels. Our pipeline uses existing object detection annotations as its foundation, enriching annotated regions into structured semantic representations that serve as reusable grounding bricks for composing different tasks. The pipeline then applies rule-based query generation and validation. This process requires neither manually written queries nor additional region annotations. GroundBricks provides a fixed test set for reproducible comparison while supporting extensions through new composition rules and query templates. Its evaluation scope can therefore evolve alongside advances in model capabilities.

1 INTRODUCTION

Visual grounding maps language descriptions to specific image regions. It supports tasks such as embodied interaction, graphical user interface agents, and visual question answering (Shridhar et al., 2020; Cheng et al., 2024). Early datasets mainly use short phrases that directly refer to objects. In contrast, real-world requests often involve multiple constraints, such as attributes, spatial relations, sets, and target absence (Wei et al., 2024). Recent benchmarks have introduced reasoning-oriented grounding tasks (Chen et al., 2025; Li et al., 2026), but they are often limited in scale and mainly used for evaluation. Multimodal models enable large-scale data generation (Liu et al., 2023; You et al., 2024), yet ensuring the correctness of generated data remains challenging (Li et al., 2023). Undetected errors can introduce noisy supervision during training.

Our key observation is that the missing piece is not localization supervision, but composability. Detection and segmentation corpora already provide large numbers of localization annotations, such as bounding boxes and masks, which offer reliable spatial supervision. However, these annotations alone lack the attributes and relations needed for compositional reasoning. Therefore we enrich each annotated instance with controlled attributes(Patterson & Hays, 2016), and establish four-directional spatial relations between instance pairs (Krishna et al., 2017). Each enriched instance becomes a **brick** with attributes for reference and relations for connection. Different compositions of these bricks yield tasks involving attribute reference, relation chaining, multi-condition reasoning, and absence-aware grounding. After task generation, we validate each query and answer against all annotations in the image to ensure consistency with the complete scene. We source these annotations from two large-scale corpora, LocateAnything-Data (Wang et al., 2026b) and SenseNova-Vision-Corpus-50M (Han et al., 2026), and reuse their localization annotations as supervision for all generated tasks.

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.

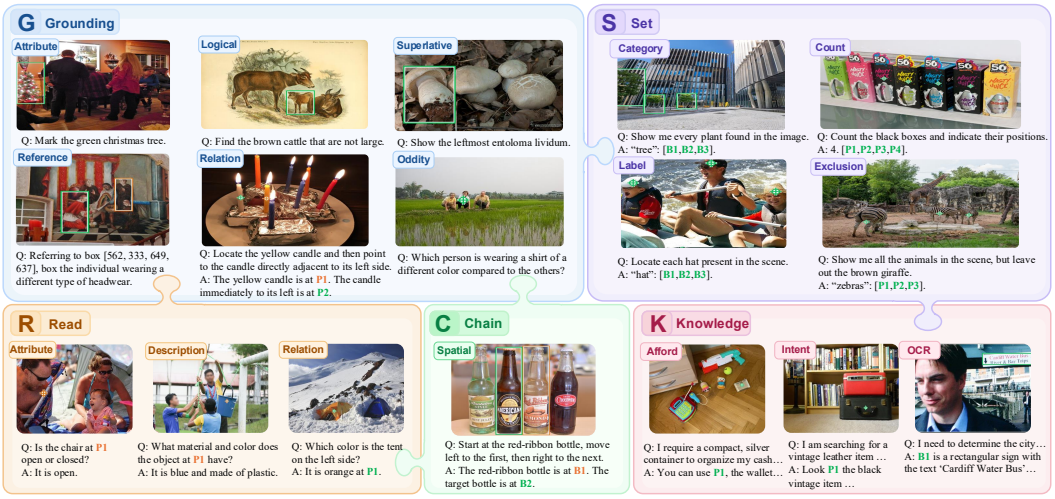


Figure 1: Examples of the five capability categories and their subcategories. Green marks answers and orange marks intermediate anchors. Coordinate-based answers are overlaid as points or bounding boxes; hybrid answers show both text and locations. P/B denote point/bounding box.

This brick-based construction provides reliable supervision, controllable task composition, and broad scalability. Target regions are determined directly from structured conditions, while full-scene validation ensures consistency between generated tasks and the image. Reusable attributes and relations allow new reasoning-oriented grounding tasks to be created through composition rules without additional annotation, while composition parameters control the reasoning requirements and difficulty of each task. The framework is also corpus-agnostic and can reuse localization annotations from other region-annotated datasets.

We instantiate this framework as **GroundBricks**(GB), a large-scale benchmark for compositional and reasoning-oriented visual grounding. GroundBricks contains 145,260 questions over 50,060 images, covering five reasoning capabilities, three difficulty levels, and two composable modifiers. It supports box, point, and text outputs with flexible combinations (Deng et al., 2021; You et al., 2024). As illustrated in Figure 1, the same set of visual bricks can be composed into referring, attribute reading, set grounding, relational chaining, and knowledge-grounded selection tasks. We fine-tune five 4B–35B base models with a unified configuration and evaluate them on ten external grounding benchmarks, covering box-level, point-level, and answer-level evaluation. The three base models gain 13.9, 7.5, and 8.4 points on average, with larger improvements on fine-grained and compositional tasks. Under a controlled comparison with matched images, underlying annotations, and training size, structured task composition still yields an average gain of about 0.75 points, showing that the improvement does not come from data scale alone. The trained models also show consistent difficulty ordering across models, supporting the controllability of the benchmark.

Our contributions are summarized as follows:

- **Brick-based grounding construction.** We introduce a general framework that enriches localization annotations with attributes and relations and composes them into diverse reasoning-oriented grounding tasks with controllable reasoning requirements and difficulty.
- **GroundBricks dataset.** We release GroundBricks with 145,260 questions over 50,060 images, covering five reasoning capabilities, three difficulty levels, and two composable modifiers, together with image-disjoint training and test splits.
- **Systematic evaluation.** We evaluate GroundBricks on five base models and ten external benchmarks, and conduct matched-scale controlled experiments to isolate the contribution of structured task composition from data scale.

2 Related Work

Referring expression comprehension (REC) localizes an image region from a language query (Deng et al., 2021; Kamath et al., 2021). Standard benchmarks include RefCOCO, RefCOCO+ (Yu et al., 2016), and RefCOCOg (Mao et al., 2016). Their aggregate evaluation provides limited capability-level diagnosis and has become less discriminative as model performance approaches saturation (Johnson et al., 2017; Hudson & Manning, 2019; Chen et al., 2023; You et al., 2024). For example, CogVLM reaches 92.44

Several benchmarks introduce compositional reasoning into REC. CLEVR-Ref+ (Liu et al., 2019) generates controllable reasoning chains in synthetic scenes, while Cops-Ref (Chen et al., 2020) and Ref-Reasoning (Yang et al., 2020) compose queries from structured scene annotations. These approaches enable systematic task generation, but their task spaces are constrained by predefined scene structures and relations (Xu et al., 2017). Recent work further studies fine-grained perception, adversarial distractors, spatial reasoning, and abstention (Liu et al., 2024a; Dong et al., 2026; Li et al., 2026; Yu et al., 2026; Kim et al., 2026; Jia et al., 2026). These benchmarks show that strong performance on standard REC does not necessarily imply robust grounding across different reasoning conditions. However, difficulty and additional conditions are often introduced as separate benchmark settings, limiting systematic control over reasoning requirements during task construction. In particular, abstention and multi-image grounding are typically treated as independent task categories, making it difficult to study their interaction with other capabilities.

Object detection and instance segmentation datasets provide large-scale localization annotations, including MSCOCO (Lin et al., 2014), Visual Genome (Krishna et al., 2017), Objects365 (Shao et al., 2019), LVIS (Gupta et al., 2019), SA-1B (Kirillov et al., 2023), and Open Images (Kuznetsova et al., 2020). Their scale, category diversity, and localization supervision make them a natural foundation for grounding data construction. However, these datasets are designed primarily for detection and segmentation and generally lack the attributes, relations, and language queries needed for compositional grounding. Prior work has mainly converted such annotations into instruction data for training (You et al., 2024; Ma et al., 2026), with limited emphasis on controllable reasoning requirements and capability-level evaluation. GroundBricks instead treats existing localization annotations as reusable building blocks and enriches them with structured attributes and relations for compositional task construction.

3 Constructing GroundBricks

3.1 GroundBricks Overview

GroundBricks is built from two large-scale grounding corpora, LocateAnything-Data (Wang et al., 2026b) and SenseNova-Vision-Corpus-50M (Han et al., 2026). We reorganize their localization annotations into instance-level visual bricks with structured attributes and spatial relations. The resulting dataset contains $145{,}260$ questions over $50{,}060$ images. It supports bounding boxes, points, and text as answer components, which can be used individually or combined. The full distribution over task types and answer modalities is provided in Appendix A.

GroundBricks is organized along three axes: **capability**, **difficulty**, and **modifier**. The capability axis specifies what the model needs to deliver. It contains five categories: *Referential grounding* (G) identifies a unique instance; *Set grounding* (S) returns all instances satisfying the conditions; *Attribute reading* (R) reads an attribute value from a given region; *Relational chaining* (C) requires reasoning over multi-hop spatial relations; and *Knowledge-grounded selection* (K) uses knowledge beyond visual appearance.

The difficulty axis describes the structural depth of a query. L1 contains a single attribute predicate, L2 introduces one relation or comparison, and L3 contains conjunction, negation, or multi-hop composition. Difficulty is determined directly by the construction rules rather than by human assessment or model accuracy, making it an intrinsic property of the query structure. The cross-distribution of capability and difficulty is shown in Table 1. The three empty cells are also structural consequences of the construction rules: Attribute reading has no L3 because reading one attribute value does not form a composite condition; Relational chaining only appears at L3 because multi-hop relations are

Table 1: GroundBricks composition by capability and complexity level.

Capability	L1 Direct	L2 Relational	L3 Compositional
Referential grounding	35,014	42,254	9,316
Set grounding	7,115	6,712	3,872
Attribute reading	21,011	4,586	-
Relational chaining	-	-	3,525
Knowledge-grounded selection	-	7,991	3,864
Total	**63,140**	**61,543**	**20,577**

inherently compositional; and Knowledge-grounded selection has no L1 because it requires at least one mapping from usage to an attribute.

The modifier axis introduces two composable operators that can be applied to base tasks without defining new task types. +REJ replaces a condition value with a value absent from the image, requiring the model to reject the query rather than rely on visual priors. +XIMG introduces multiple images and requires the model to first identify the image containing the target. The two modifiers can be applied independently or jointly. Representative cases are shown in Appendix A.

3.2 PROBLEM FORMULATION

Each image I is converted into an intermediate scene representation

$$\mathcal{S}(I) = (\mathcal{O}, \mathcal{E}),$$

where each instance $o = (b, \lambda, \alpha) \in \mathcal{O}$ forms a visual brick with region b, category λ, and attribute mapping α. The relation structure $\mathcal{E}$ captures spatial relations among instances, with relation types $\{\textit{left}, \textit{right}, \textit{above}, \textit{below}\}$. Local relations of an instance can be directly queried from $\mathcal{E}$, while $\mathcal{E}$ jointly represents the relational structure of the entire scene. A grounding task operates on an image set $\mathcal{I} = \{I_1, \ldots, I_n\}$ and is represented as

$$q = (\mathcal{I}, x, y, c, \ell, m),$$

where x is the query, y is the answer, and c, ℓ, and m denote capability, difficulty, and modifier. Given the scene representation and optional external knowledge $\mathcal{K}$, a structured selector φ determines the target set:

$$T = \{o \in \mathcal{O} : \varphi(o; \mathcal{S}, \mathcal{K}) = 1\}, \qquad x = \rho(\varphi),$$

where ρ renders the selector as a natural-language query. The selector can encode attribute constraints, spatial relations, comparisons, conjunctions, negation, and knowledge-grounded conditions. The answer is represented as

$$y = (a_{\text{loc}}, a_{\text{text}}),$$

where a_{loc} is an optional localization output in the form of a bounding box or point, and a_{text} is an optional textual output. This formulation supports box-only, point-only, text-only, box-text, and point-text answers. An answer renderer γ maps the selected targets to the required answer components:

$$y = \gamma(T, \varphi).$$

The three axes are determined by the structure of the selector and its operators: capability specifies the target and reasoning type, difficulty is determined by structural depth, and modifiers apply additional operators to the selector. New task types can therefore be introduced by defining new selectors φ and composition rules, without requiring additional region annotations.

3.3 CONSTRUCTION PIPELINE

The overall construction pipeline is shown in Figure 2; all model-based operations use Qwen3.8-27B. Starting from existing region annotations, the pipeline progressively enriches instances into visual bricks, connects them through spatial relations, and composes them into reasoning-oriented grounding tasks. Detailed construction procedures and implementation rules are provided in Appendix B, and all prompts are listed in Appendix C. Table 5 summarizes the operators, outputs, and scales of the five stages.

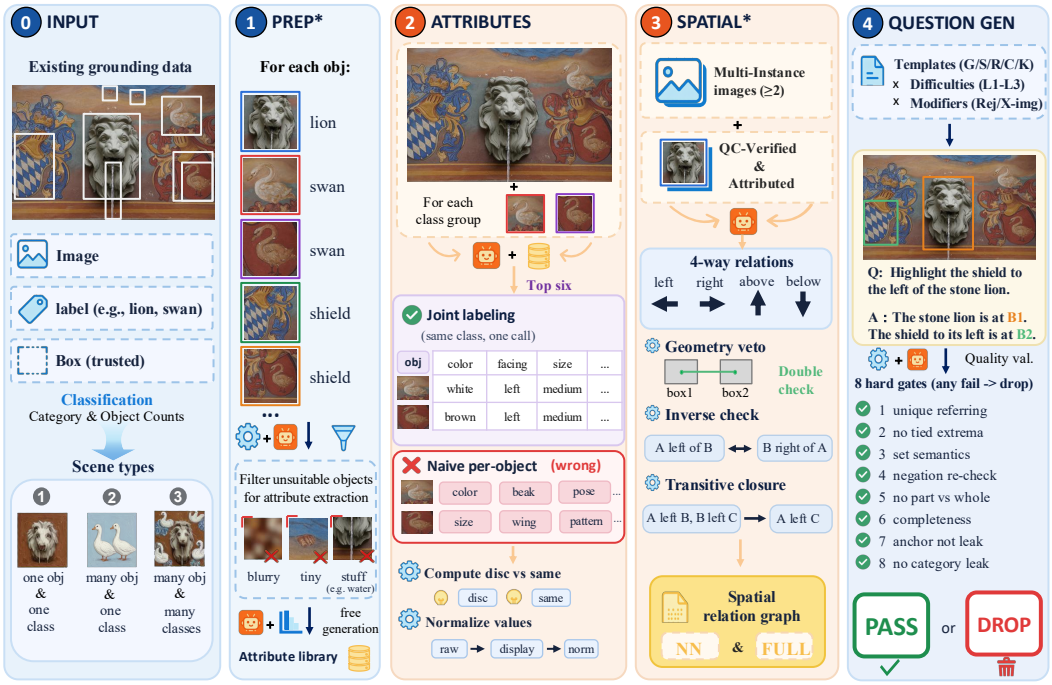


Figure 2: GroundBricks construction pipeline with five stages.

Stage 0: Input. We start from existing region annotations and flatten them into (I, b_i, λ_i) tuples, where I, b_i, and λ_i denote the image, instance region, and category. These annotations provide the localization supervision for subsequent task construction. Then we classify images into three scene types according to their category and instance counts: single-category single-instance, single-category multi-instance, and multi-category multi-instance. This stratification initially determines the task types that can be constructed from each scene.

Stage 1: Preparation. We build a controlled attribute vocabulary $\mathcal{V}_{\text{attr}}$ from 6,000 randomly sampled images, where the model extracts at most six suitable attributes per instance. Invalid attribute types and instances with extreme bounding-box areas are filtered before subsequent annotation. The resulting hierarchy are described in Appendix B.2.

Stage 2: Attributes Enrichment. For images containing multiple instances of the same category, we provide both the full image and cropped instance regions for joint annotation. The extracted attributes are stored in raw, display, and norm forms to preserve the original output, standardize expressions for generation, and support programmatic comparison, respectively. Only valid and confirmed absent values are retained, while uncertain values are discarded. The value representations are described in Appendix B.2.

Stage 3: Spatial Relation Construction. We construct a complete four-directional relation graph through geometric validation, inverse-consistency checks, and transitive closure. We additionally derive a nearest-neighbor graph under a distance-gap constraint. The complete graph supports multi-hop spatial reasoning, while the nearest-neighbor graph supports unique referring. Details are given in Appendix B.3.

Stage 4: Question Generation. We compose 21 task types using five capabilities, three difficulty levels, and two composable modifiers. Before generation, templates pass eight structural checks to ensure valid targets, sufficient discrimination, and no unintended shortcuts; the checks and their admission criteria are listed in Table 7 (Appendix B.4). After generation, the full image, query, answer, and all instances are jointly provided to the model for correctness verification; incorrect cases are discarded. Accepted questions are then lightly polished with coordinate placeholders protected.

Table 2: Grounding accuracy before and after fine-tuning on GroundBricks. Models are divided into base and non-base groups. GB refers GroundBricks. Bold and underlined values indicate the best and second-best results within each group, respectively.

Model	Box-level Acc					Point-level Acc			Answer-level Acc		Avg.
	RefCOCO	Ref-L4	Ref-Adv	PIO	FineC.	RefSp.	PointA.	OSW-G	TreeB.	GME	
Base models											
Qwen3.5-4B-Base	29.21	83.11	59.11	28.80	57.25	9.03	58.04	10.99	32.84	28.46	39.68
Qwen3.5-4B-Base + GB	74.92	88.22	61.30	34.40	76.03	28.88	62.63	48.40	29.14	32.04	53.60
Qwen3.5-9B-Base	48.91	89.04	62.35	35.80	68.72	12.64	**64.77**	23.58	41.48	**43.08**	49.04
Qwen3.5-9B-Base + GB	**79.00**	88.98	61.30	39.00	74.36	34.66	63.65	58.87	29.14	36.32	56.53
Qwen3.5-35B-A3B-Base	47.13	88.70	61.82	42.40	67.61	20.94	58.45	31.56	**47.16**	29.45	49.52
Qwen3.5-35B-A3B-Base + GB	67.43	**89.98**	**64.62**	**45.80**	**76.22**	**41.88**	54.58	**63.65**	36.79	37.81	**57.88**
Non-base models											
Qwen3.5-4B-Instruct	87.35	71.65	51.14	36.40	68.80	32.49	58.15	52.84	25.19	31.54	51.56
Qwen3.5-4B-Instruct + GB	76.52	87.31	53.59	38.00	73.95	27.08	**61.71**	58.69	23.95	34.33	53.51
Qwen3.8-27B	**91.44**	**89.87**	**65.85**	**54.20**	72.01	52.09	57.23	39.01	**33.09**	**50.15**	60.49
Qwen3.8-27B + GB	84.10	85.45	65.67	48.20	**75.36**	**54.15**	55.50	**68.44**	28.40	39.90	**60.52**

4 EXPERIMENTS

4.1 EXPERIMENT SETUP

We fine-tune five models spanning 4B–35B parameters: Qwen3.5-4B-Base, Qwen3.5-4B-Instruct, Qwen3.5-9B-Base, Qwen3.8-27B, and Qwen3.5-35B-A3B-Base. All models use the same LoRA configuration (rank 32, learning rate 10^{-4}, 2 epochs, and global batch size 64).

We evaluate on ten external grounding benchmarks with no image overlap with the training set. Depending on output granularity, we report box-level, point-level, and answer-level accuracy. All metrics follow the general form

$$\mathrm{Acc} = \frac{1}{N}\sum_{i=1}^{N} \mathbf{1}[\mathrm{Correct}(\hat{y}_i, y_i)]\,,$$

where $\hat{y}_i$ and y_i denote the predicted and reference outputs, respectively. For box-level evaluation, a prediction is correct when its IoU with the reference box is at least 0.5; for point-level evaluation, the predicted point must fall within the reference region; and for answer-level evaluation, the predicted answer must exactly match the reference answer.

We use box-level accuracy (Acc@0.5) for RefCOCO(Yu et al., 2016), Ref-L4(Chen et al., 2025), Ref-Adv(Dong et al., 2026), PIO(Xue et al., 2025), and FineCops(Liu et al., 2024a); point-level accuracy for RefSpatial(Zhou et al., 2025), PointArena(Cheng et al., 2025), and OSWorld-G(Xie et al., 2026); and answer-level accuracy for TreeBench(Wang et al., 2026a) and GroundingME(Li et al., 2026). We normalize model outputs into a unified representation before evaluation. TreeBench uses micro-averaged accuracy, and GroundingME combines localization and refusal samples into a single accuracy.

4.2 MAIN RESULT

Table 2 summarizes the performance of five backbone models before and after fine-tuning on GroundBricks. We evaluate box-level localization, point-level localization, and answer-level prediction on ten external grounding benchmarks with no image overlap with the training data. The results assess the transfer of supervision constructed from existing region annotations to diverse grounding tasks.

GroundBricks consistently improves base models. The average scores of the three base models increase from 39.68, 49.04, and 49.52 to 53.60, 56.53, and 57.88, yielding gains of $+13.92$, $+7.49$, and $+8.36$ points, respectively. The improvements hold across model scales from 4B to 35B and span both box- and point-level evaluation. The gains also transfer to diverse downstream grounding tasks.

Table 3: Effect of GroundBricks with matched training-data volume. The base model is trained on the same amount of source data with either raw annotations or GroundBricks. Bold and underlined values indicate the best and second-best results, respectively. #box denotes the mean number of boxes per question.

Training data	Box-level Acc					Point-level Acc			Answer-level Acc		#box	Avg.
	RefCOCO	Ref-L4	Ref-Adv	PIO	FineC.	RefSp.	PointA.	OSW-G	TreeB.	GME		
Qwen3.5-4B-Base	29.21	83.11	59.11	28.80	57.25	9.03	58.04	10.99	<u>32.84</u>	<u>30.85</u>	**0.85**	39.92
+ LA+SN (raw)	**78.15**	<u>83.88</u>	<u>60.51</u>	**37.20**	<u>75.31</u>	<u>23.47</u>	<u>60.39</u>	**50.14**	**34.26**	27.74	3.56	<u>53.11</u>
+ GroundBricks	<u>74.92</u>	**88.22**	**61.30**	<u>34.40</u>	**76.03**	**28.88**	**62.63**	<u>48.40</u>	29.14	**34.70**	<u>0.97</u>	**53.86**
Δ (GB − raw)	-3.23	+4.34	+0.79	-2.80	+0.72	+5.41	+2.24	-1.74	-5.12	+6.96	-2.59	+0.76

The largest gains appear on fine-grained and spatially compositional grounding. FineCops improves by $+18.78$, $+5.64$, and $+8.61$ points across the three base models, while RefSpatial improves by $+19.85$, $+22.02$, and $+20.94$ points. FineCops emphasizes fine-grained candidate disambiguation, while RefSpatial focuses on spatial reasoning. GroundBricks composes objects, attributes, sets, relations, and knowledge constraints into structured queries. The strong gains on these benchmarks suggest that compositional supervision improves multi-constraint grounding. Gains on direct grounding benchmarks further show that this supervision preserves basic localization ability.

GroundBricks provides a scalable way to reuse existing region annotations. The performance gains transfer across grounding benchmarks with different query distributions, output granularities, and evaluation protocols. This result supports the central design of GroundBricks: existing detection and segmentation annotations provide reliable localization primitives, while structured enrichment and rule-based composition turn them into diverse grounding supervision. Non-base models also benefit, but their gains are more concentrated on fine-grained and spatially compositional tasks. We therefore use base models for subsequent ablations and analyses. This setting provides a more controlled initialization and facilitates clearer attribution of gains to GroundBricks supervision.

5 Analysis

GroundBricks does not introduce new region annotations but reorganizes existing annotations into diverse grounding tasks. To isolate the effect of task composition from data scale, Table 3 compares two training settings with the same $145{,}260$ training samples derived from the same underlying annotations. The raw annotations cover natural images, GUI, and OCR, whereas GroundBricks uses only natural images. Thus, the comparison favors the raw setting in scene coverage, while GroundBricks isolates the additional effect of structured task composition.

5.1 Composition, Rather Than Data Scale, Drives the Gains

Composition improves performance under matched training scale. Training on raw annotations increases the average score from 39.92 to 53.11, demonstrating the value of large-scale region supervision. GroundBricks further raises the average score to 53.86. Compared with raw-annotation training, GroundBricks improves Ref-L4, Ref-Adv, FineCops, and RefSpatial by $+4.34$, $+0.79$, $+0.72$, and $+5.41$ points, respectively, and improves PointArena and GroundingME by $+2.24$ and $+6.96$ points. These gains span different output granularities and extend to tasks closely related to GUI and OCR scenarios, despite GroundBricks being trained only on natural images. This result further supports the value of structured task composition beyond direct region supervision.

Compositional supervision also improves target selection. After GroundBricks training, the average number of predicted boxes per query decreases from 3.56 to 0.97, alongside a $+6.96$-point gain on GroundingME. This result suggests more accurate target selection under compositional supervision.

5.2 Controllable Complexity Enables Quantifiable Capability Gains

GroundBricks assigns each sample a verified capability type and complexity level, enabling controlled analysis of model capabilities beyond training. Figure 3(a) reports results on a held-out test set of $6{,}210$ samples. Before and after training, all three base models show a consistent accuracy

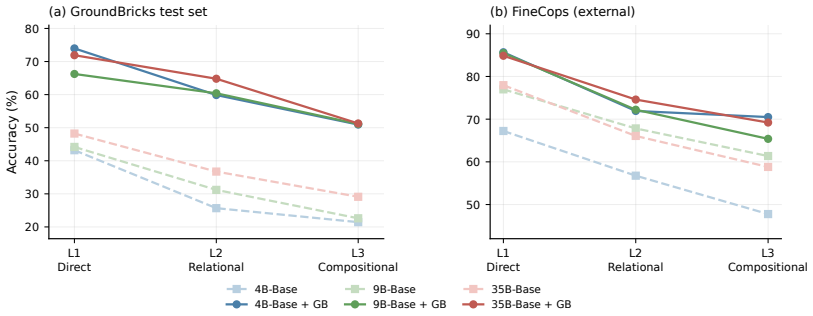


Figure 3: Complexity levels in GroundBricks and their transfer to FineCops. (a) Results on the GroundBricks test set. (b) Results on FineCops. Dashed and solid lines denote pre- and post-training results, respectively.

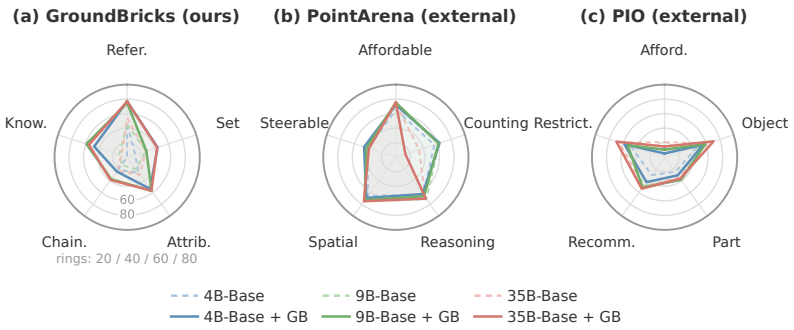


Figure 4: Capability decomposition and external transfer of GroundBricks. (a) Five capability categories on the GroundBricks test set. (b)–(c) Subcategory-level results on PointArena and PIO.

decline from L1 (direct localization) to L2 (relational reasoning) and L3 (compositional reasoning), with gaps of 15–23 points between levels. This consistent ordering across models indicates that the predefined complexity levels capture the reasoning demands of the queries rather than model-specific preferences. After GroundBricks training, the three models still differ by about 7.7 points at L1, but converge to within 0.3 points at L3, suggesting that high-complexity compositional tasks remain a common bottleneck across models.

The difficulty ordering also transfers to an independently constructed benchmark.Figure 3(b) group FineCops into direct, relational, and compositional levels using its own annotations, without reference to GroundBricks. FineCops exhibits the same monotonic accuracy decline across levels. GroundBricks further improves performance by $+18.4$, $+15.2$, and $+22.7$ points on the three levels, respectively, with the largest gain on the most difficult compositional level. The consistent ordering across two independently defined difficulty schemes, together with the larger gains on external compositional tasks, suggests that GroundBricks captures transferable relational and multi-constraint reasoning ability rather than dataset-specific templates or distributions.

5.3 Capability Decomposition Reveals Differential Transfer Gains

GroundBricks provides capability labels that decompose model performance into five distinct dimensions rather than a single aggregate score. Figure 4(a) shows that the accuracy gap across capabilities reaches 52.6 points within the same model, with different responses to model scale. Referential grounding (Refer.) already reaches 74.7–76.9 with limited variation across model sizes. Chain reasoning (Chain.) remains the most challenging capability, but improves by 14.4 points from 4B to 35B, showing strong sensitivity to model capacity. In contrast, Set grounding does not improve consistently with scale, suggesting that target cardinality and multi-target selection require more than simply increasing model capacity. These differences indicate distinct capability bottlenecks and provide finer-grained discrimination between models with similar overall performance.

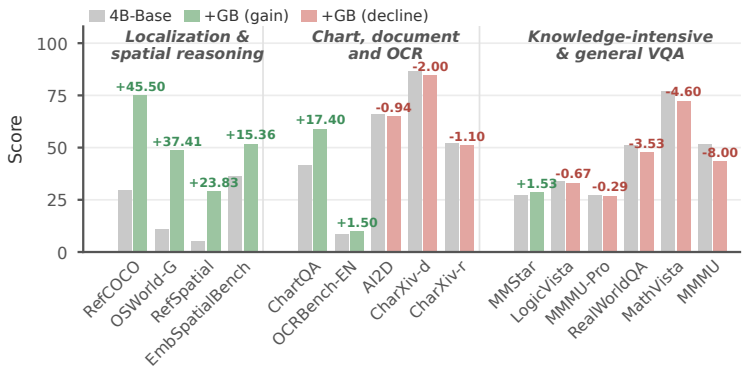


Figure 5: Performance changes on general multimodal benchmarks before and after GroundBricks training. Gray bars denote the 4B base model, while green and red bars denote gains and drops after training, respectively.

The same pattern appears on external benchmarks with subcategory annotations. As shown in Figure 4(b)–(c), gains concentrate on subcategories related to localization, relational parsing, and constraint selection. On PointArena, Affordable improves by 1.5–7.0 points, while the 4B model gains 7.7 and 3.6 points on Counting and Spatial, respectively. On PIO, Object improves by 8.2–9.0 points across the three models, with additional gains on Part, Recommendation, and Restriction. In contrast, Reasoning, Steerable, and Affordance show smaller changes. This pattern is consistent with the capabilities emphasized by GroundBricks, suggesting that its gains transfer most clearly to tasks involving visual localization and compositional constraints.

Capability decomposition also identifies remaining bottlenecks. Chain reasoning is most sensitive to model scale, while Set grounding and Counting show greater variation. The limited changes on Affordance and Steerable point to capabilities that are less covered by the current supervision. Thus, the capability taxonomy not only measures overall improvement, but also identifies where gains occur and where additional supervision may be needed.

Transfer extends beyond localization while preserving most general capabilities. Beyond localization and spatial reasoning, Figure 5 shows that GroundBricks improves ChartQA (Masry et al., 2022) by +17.40 points, with additional gains on OCRBench (Liu et al., 2024b) and MMStar (Chen et al., 2024). These results indicate that compositional supervision can transfer beyond coordinate prediction to chart understanding, text recognition, and general multimodal reasoning. Meanwhile, changes on AI2D (Kembhavi et al., 2016), LogicVista (Xiao et al., 2024), and MMMU-Pro (Yue et al., 2025) are all below 1 point, suggesting that most general capabilities remain stable after training. Limited changes mainly occur on benchmarks involving long-form generation or knowledge-intensive reasoning, including CharXiv (Wang et al., 2024), RealWorldQA[1] , MathVista (Lu et al., 2024), and MMMU (Yue et al., 2024). These tasks emphasize capabilities that differ from the concise prediction and compositional reasoning reinforced by GroundBricks. Overall, GroundBricks substantially improves its target capabilities while preserving performance on most non-target tasks.

6 Conclusion

We introduce GroundBricks, a configurable framework for constructing reasoning-oriented grounding tasks from existing region annotations. By enriching instances with attributes and spatial relations and composing them through explicit rules, GroundBricks turns existing localization supervision into diverse and controllable tasks without additional region annotation. Extensive experiments across five base models and ten external benchmarks show consistent gains, particularly on spatial and fine-grained grounding. Controlled comparisons further demonstrate that task composition provides additional benefits beyond data scale, while the predefined difficulty levels consistently reflect model performance. We hope GroundBricks will facilitate future research on scalable construction and systematic evaluation of grounding capabilities.

[1]Released by xAI alongside the Grok-1.5 Vision announcement; it contains 765 real-world photographs with multiple-choice questions and is available at `https://huggingface.co/datasets/xai-org/RealWorldQA`.

AI USE STATEMENT

In this work, we used generative AI tools to generate synthetic datasets and support research execution. Specifically, vision-language models were used to annotate attributes of image regions, which were then used to instantiate predefined templates for constructing grounding queries and answers. We further used generative AI tools to review the generated data. We also used generative AI tools to edit the manuscript for readability. All AI-assisted outputs were reviewed by the authors, and we take responsibility for the final content of this work, including the dataset, text, and other artifacts produced with the aid of generative AI.

ETHICS STATEMENT

GroundBricks is constructed from two publicly available, open-source region-annotated datasets and does not involve human subjects, user data collection, or the release of personally identifiable information. We follow the licenses and usage terms of the source datasets and acknowledge their original authors and contributors.

Model-based operations are used to enrich existing annotations and construct candidate attributes, relations, and queries. Generated content is filtered through controlled vocabularies, programmatic validation, and full-scene consistency checks before inclusion in the benchmark. Model outputs are therefore not treated as ground truth without verification. We acknowledge that automated attribute extraction may introduce semantic biases or errors. Our filtering and validation procedures reduce such issues but cannot eliminate them completely.

REPRODUCIBILITY STATEMENT

GroundBricks is built from two publicly available corpora, and the benchmark construction process is described in Section3.3. The evaluation pipeline is detailed in Section4. Dataset statistics and all task definitions are provided in Appendix A. The code, data splits, and evaluation scripts will be released publicly upon acceptance.

REFERENCES

Jierun Chen, Fangyun Wei, Jinjing Zhao, Sizhe Song, Bohuai Wu, Zhuoxuan Peng, S.-H. Gary Chan, and Hongyang Zhang. Revisiting referring expression comprehension evaluation in the era of large multimodal models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 513–524, 2025.

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.

Lin Chen, Jinsong Li, Xiaoyi Dong, Pan Zhang, Yuhang Zang, Zehui Chen, Haodong Duan, Jiaqi Wang, Yu Qiao, Dahua Lin, and Feng Zhao. MMStar: Are multimodal large language models ready for real-world challenges? In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

Zhenfang Chen, Peng Wang, Lin Ma, Kwan-Yee K. Wong, and Qi Wu. Cops-Ref: A new dataset and task on compositional referring expression comprehension. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10083–10092, 2020.

Kanzhi Cheng, Qiushi Sun, Yougang Chu, Fangzhi Xu, Li YanTao, Jianbing Zhang, and Zhiyong Wu. SeeClick: Harnessing GUI grounding for advanced visual GUI agents. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (ACL)*, pp. 9313–9332, 2024.

Long Cheng, Jiafei Duan, Yi Ru Wang, Haoquan Fang, Boyang Li, Yushan Huang, Elvis Wang, Ainaz Eftekhar, Jason Lee, Wentao Yuan, et al. Pointarena: Probing multimodal grounding through language-guided pointing. *arXiv preprint arXiv:2505.09990*, 2025.

Jiajun Deng, Zhengyuan Yang, Tianlang Chen, Wengang Zhou, and Houqiang Li. TransVG: End-to-end visual grounding with transformers. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 1749–1759, 2021.

Qihua Dong, Kuo Yang, Lin Ju, Handong Zhao, Yitian Zhang, Yizhou Wang, Huimin Zeng, Jianglin Lu, and Yun Fu. Ref-Adv: Exploring MLLM visual reasoning in referring expression tasks. In *The Fourteenth International Conference on Learning Representations (ICLR)*, 2026. URL https://openreview.net/forum?id=iEBgrepR9i.

Agrim Gupta, Piotr Dollar, and Ross Girshick. LVIS: A dataset for large vocabulary instance segmentation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5351–5359, 2019.

Xiaoyang Han, Jianhua Li, Kewang Deng, Zukai Chen, Xuanke Shi, Sihan Wang, Boxuan Li, Linyan Wang, Siyi Xie, Xin You, et al. Vision as unified multimodal generation. *arXiv preprint arXiv:2607.06560*, 2026.

Drew A. Hudson and Christopher D. Manning. GQA: A new dataset for real-world visual reasoning and compositional question answering. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6693–6702, 2019.

Mengdi Jia, Zekun Qi, Shaochen Zhang, Wenyao Zhang, XinQiang Yu, Jiawei He, He Wang, and Li Yi. OmniSpatial: Towards comprehensive spatial reasoning benchmark for vision language models. In *International Conference on Learning Representations (ICLR)*, volume 2026, pp. 35634–35670, 2026.

Justin Johnson, Bharath Hariharan, Laurens van der Maaten, Li Fei-Fei, C. Lawrence Zitnick, and Ross Girshick. CLEVR: A diagnostic dataset for compositional language and elementary visual reasoning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2901–2910, 2017.

Aishwarya Kamath, Mannat Singh, Yann LeCun, Gabriel Synnaeve, Ishan Misra, and Nicolas Carion. MDETR – modulated detection for end-to-end multi-modal understanding. In *IEEE/CVF international conference on computer vision (ICCV)*, pp. 1760–1770, 2021.

Aniruddha Kembhavi, Mike Salvato, Eric Kolve, Minjoon Seo, Hannaneh Hajishirzi, and Ali Farhadi. A diagram is worth a dozen images. In *European conference on computer vision*, pp. 235–251, 2016.

Minkyu Kim, Sangheon Lee, and Dongmin Park. VLM-SubtleBench: How far are VLMs from human-level subtle comparative reasoning? In *The Fourteenth International Conference on Learning Representations (ICLR)*, 2026. URL https://openreview.net/forum?id=pBTXsu1i77.

Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. In *IEEE/CVF international conference on computer vision (ICCV)*, pp. 3992–4003, 2023.

Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, et al. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *International Journal of Computer Vision (IJCV)*, 123(1):32–73, 2017.

Alina Kuznetsova, Hassan Rom, Neil Alldrin, Jasper Uijlings, Ivan Krasin, Jordi Pont-Tuset, Shahab Kamali, Stefan Popov, Matteo Malloci, Alexander Kolesnikov, et al. The open images dataset V4: Unified image classification, object detection, and visual relationship detection at scale. *International Journal of Computer Vision (IJCV)*, 128(7):1956–1981, 2020.

Rang Li, Lei Li, Shuhuai Ren, Hao Tian, Shuhao Gu, Shicheng Li, Zihao Yue, Yudong Wang, Wenhan Ma, Zhe Yang, et al. GroundingME: Exposing the visual grounding gap in MLLMs through multi-dimensional evaluation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2412–2422, 2026.

Yifan Li, Yifan Du, Kun Zhou, Jinpeng Wang, Wayne Xin Zhao, and Ji-Rong Wen. Evaluating object hallucination in large vision-language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 292–305, 2023.

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 *European Conference on Computer Vision (ECCV)*, pp. 740–755, 2014.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 36, pp. 34892–34916, 2023.

Junzhuo Liu, Xuzheng Yang, Weiwei Li, and Peng Wang. FineCops-Ref: A new dataset and task for fine-grained compositional referring expression comprehension. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 15440–15457, 2024a.

Runtao Liu, Chenxi Liu, Yutong Bai, and Alan L Yuille. CLEVR-Ref+: Diagnosing visual reasoning with referring expressions. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4180–4189, 2019.

Yuliang Liu, Zhang Li, Mingxin Huang, Biao Yang, Wenwen Yu, Chunyuan Li, Xu-Cheng Yin, Cheng-Lin Liu, Lianwen Jin, and Xiang Bai. Ocrbench: on the hidden mystery of ocr in large multimodal models. *Science China Information Sciences*, 67(12):220102, 2024b.

Pan Lu, Hritik Bansal, Tony Xia, Jiacheng Liu, Chunyuan Li, Hannaneh Hajishirzi, Hao Cheng, Kai-Wei Chang, Michel Galley, and Jianfeng Gao. MathVista: Evaluating mathematical reasoning of foundation models in visual contexts. In *International Conference on Learning Representations (ICLR)*, volume 2024, pp. 23439–23554, 2024.

Ziyang Ma, Ruiyang Xu, Zhenghao Xing, Yunfei Chu, Yuxuan Wang, Jinzheng He, Jin Xu, Pheng-Ann Heng, Kai Yu, Junyang Lin, et al. Omni-Captioner: Data pipeline, models, and benchmark for omni detailed perception. In *International Conference on Learning Representations (ICLR)*, volume 2026, pp. 132277–132304, 2026.

Junhua Mao, Jonathan Huang, Alexander Toshev, Oana Camburu, Alan L. Yuille, and Kevin Murphy. Generation and comprehension of unambiguous object descriptions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 11–20, 2016.

Ahmed Masry, Jia Qing Tan, Shafiq Joty, Enamul Hoque, et al. ChartQA: A benchmark for question answering about charts with visual and logical reasoning. In *Findings of the Association for Computational Linguistics (ACL)*, pp. 2263–2279, 2022.

Genevieve Patterson and James Hays. COCO attributes: Attributes for objects, people, and scenes. In *European Conference on Computer Vision (ECCV)*, pp. 85–100, 2016.

Shuai Shao, Zeming Li, Tianyuan Zhang, Chao Peng, Gang Yu, Xiangyu Zhang, Jing Li, and Jian Sun. Objects365: A large-scale, high-quality dataset for object detection. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 8429–8438, 2019.

Mohit Shridhar, Jesse Thomason, Daniel Gordon, Yonatan Bisk, Winson Han, Roozbeh Mottaghi, Luke Zettlemoyer, and Dieter Fox. Alfred: A benchmark for interpreting grounded instructions for everyday tasks. In *IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pp. 10737–10746, 2020.

Haochen Wang, Xiangtai Li, Zilong Huang, Anran Wang, Jiacong Wang, Tao Zhang, Sule Bai, Zijian Kang, Jiashi Feng, Zhuochen Wang, et al. Traceable evidence enhanced visual grounded reasoning: Evaluation and methodology. In *International Conference on Learning Representations (ICLR)*, volume 2026, pp. 148769–148794, 2026a.

Shihao Wang, Shilong Liu, Yuanguo Kuang, Xinyu Wei, Yangzhou Liu, Zhiqi Li, Yunze Man, Guo Chen, Andrew Tao, Guilin Liu, et al. LocateAnything: Fast and high-quality vision-language grounding with parallel box decoding. In *European Conference on Computer Vision*, pp. 336–357, 2026b.

Zirui Wang, Mengzhou Xia, Luxi He, Howard Chen, Yitao Liu, Richard Zhu, Kaiqu Liang, Xindi Wu, Haotian Liu, Sadhika Malladi, et al. CharXiv: Charting gaps in realistic chart understanding in multimodal LLMs. In *Advances in Neural Information Processing Systems (NeurIPS) Datasets and Benchmarks Track*, volume 37, pp. 113569–113697, 2024.

Fangyun Wei, Jinjing Zhao, Kun Yan, Hongyang Zhang, and Chang Xu. A large-scale human-centric benchmark for referring expression comprehension in the LMM era. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 37, pp. 69566–69587, 2024. Introduces HC-RefLoCo.

Yijia Xiao, Edward Sun, Tianyu Liu, and Wei Wang. Logicvista: Multimodal llm logical reasoning benchmark in visual contexts. *arXiv preprint arXiv:2407.04973*, 2024.

Tianbao Xie, Jiaqi Deng, Xiaochuan Li, Junlin Yang, Haoyuan Wu, Jixuan Chen, Wenjing Hu, Xinyuan Wang, Yuhui Xu, Zekun Wang, et al. Scaling computer-use grounding via user interface decomposition and synthesis. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 38, 2026.

Yunqiu Xu, Linchao Zhu, and Yi Yang. Scene graph generation by iterative message passing. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 17675–17687, 2017.

Haotian Xue, Yunhao Ge, Yu Zeng, Zhaoshuo Li, Ming-Yu Liu, Yongxin Chen, and Jiaojiao Fan. Point-it-out: Benchmarking embodied reasoning for vision language models in multi-stage visual grounding. *arXiv preprint arXiv:2509.25794*, 2025.

Sibei Yang, Guanbin Li, and Yizhou Yu. Graph-structured referring expression reasoning in the wild. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9949–9958, 2020.

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. In *International Conference on Learning Representations (ICLR)*, volume 2024, pp. 57153–57180, 2024.

Hongtao Yu, Yuxin Peng, Serge Belongie, and Xiu-Shen Wei. Benchmarking large vision-language models on fine-grained image tasks: A comprehensive evaluation. In *International Conference on Learning Representations (ICLR)*, volume 2026, pp. 44677–44700, 2026.

Licheng Yu, Patrick Poirson, Shan Yang, Alexander C Berg, and Tamara L Berg. Modeling context in referring expressions. In *European Conference on Computer Vision (ECCV)*, pp. 69–85, 2016.

Xiang Yue, Yuansheng Ni, Kai Zhang, Tianyu Zheng, Ruoqi Liu, Ge Zhang, Samuel Stevens, Dongfu Jiang, Weiming Ren, Yuxuan Sun, et al. MMMU: A massive multi-discipline multimodal understanding and reasoning benchmark for expert AGI. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9556–9567, 2024.

Xiang Yue, Tianyu Zheng, Yuansheng Ni, Yubo Wang, Kai Zhang, Shengbang Tong, Yuxuan Sun, Botao Yu, Ge Zhang, Huan Sun, et al. MMMU-Pro: A more robust multi-discipline multimodal understanding benchmark. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (ACL)*, pp. 15134–15186, 2025.

Enshen Zhou, Jingkun An, Cheng Chi, Yi Han, Shanyu Rong, Chi Zhang, Pengwei Wang, Zhongyuan Wang, Tiejun Huang, Lu Sheng, et al. RoboRefer: Towards spatial referring with reasoning in vision-language models for robotics. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 38, pp. 28404–28481, 2025.

A Task Types and Answer Modalities

Table 4 lists the number of samples and difficulty levels for all 21 task types. The upper part contains 17 base task types (123,198 samples, 84.8%), while the lower part contains 4 variants derived by applying modifiers (22,062 samples, 15.2%).

Table 4: Base question types and variants obtained by applying composition modifiers.

Capability	Question type	Level	#Questions	Share (%)
Base types				
G	`Ground.Attribute`	L1	25,251	17.4
G	`Ground.Reference`	L2	15,731	10.8
G	`Ground.Relation`	L2	12,268	8.4
G	`Ground.Superlative`	L2	7,400	5.1
G	`Ground.Oddity`	L2	1,884	1.3
G	`Ground.Logical`	L3	1,988	1.4
S	`Set.Labeling`	L1	7,115	4.9
S	`Set.Counting`	L2	3,623	2.5
S	`Set.Category`	L2	3,089	2.1
S	`Set.Exclusion`	L3	3,872	2.7
R	`Read.Attribute`	L1	10,598	7.3
R	`Read.Description`	L1	10,413	7.2
R	`Read.Relation`	L2	4,586	3.2
C	`Chain.Spatial`	L3	3,525	2.4
K	`Know.Afford`	L2	5,541	3.8
K	`Know.OCR`	L2	2,450	1.7
K	`Know.Intent`	L3	3,864	2.7
Subtotal			*123,198*	*84.8*
Variants with modifiers				
G	`Ground.Attribute` +Rej	L1	9,763	6.7
G	`Ground.Attribute` +Ximg	L2	4,971	3.4
G	`Ground.Reference` +Ximg	L3	3,701	2.5
G	`Ground.Reference` +Ximg +Rej	L3	3,627	2.5
Subtotal			*22,062*	*15.2*

The task distribution is determined by construction feasibility rather than uniform quotas. Modifier-derived samples are likewise constrained by the availability of valid absent conditions or suitable image pairs.

Figure 6 shows the distribution of answer modalities. Modality is an independent construction axis and can vary within the same task type. GroundBricks supports box, point, and text outputs, as well as box+text and point+text combinations. Text-only outputs mainly correspond to attribute reading, while text-augmented outputs combine localization with additional reasoning. Refusal samples provide negative answers for queries with no valid target. The two modifiers can be applied independently or jointly. And Figure 7 illustrates the two modifiers effects on the base task, including the individual and jointly applied cases.

B Construction Details

This appendix provides the implementation details needed to reproduce GroundBricks. The organization follows the construction stages in Section 3.3.

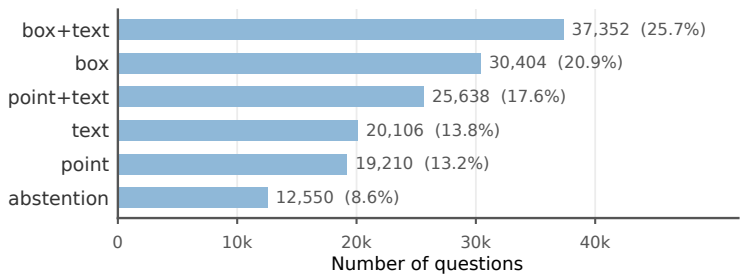


Figure 6: Distribution of answer modalities.



Figure 7: Effects of the two modifiers applied to the base task.

B.1 Pipeline by Stage

Table 5 summarizes the five construction stages, their main operations, intermediate products, and scales. The following details clarify implementation choices that are not specified in the main text.

Model-based operations. The model is used for attribute discovery, joint attribute annotation, spatial relation extraction, and final quality control. Model outputs are restricted to structured fields and are subsequently checked by deterministic rules. The model does not generate grounding queries or answers directly. Invalid or unparseable outputs are discarded.

Rejected instances. An instance rejected during attribute annotation is not removed from the image-level object set. Such instances are excluded only from attribute-based construction when their attributes cannot be reliably determined. Their original bounding boxes remain available and are included in the final scene during quality control. This distinction allows attribute uncertainty to be separated from localization validity.

B.2 Attribute Library

Stage 1 constructs the attribute vocabulary $\mathcal{V}_\mathrm{attr}$, which is then fixed for production annotation.

Vocabulary construction. We randomly sample (6,000) images for attribute discovery. For each annotated instance, Qwen3.8-27B is asked to select at most six visually observable attributes without being given candidate keys. The discovered keys are normalized, merged across synonymous forms, and filtered by frequency and validity before forming $\mathcal{V}_\mathrm{attr}$.

The vocabulary contains three levels: generic, category-cluster, and scene-specific attributes. Generic and category-cluster attributes are considered first; scene-specific attributes are used when the former do not adequately describe an instance. The complete vocabulary and its hierarchical organization are shown in Table 6. Each key is associated with an applicable category scope.

We remove three classes of unsuitable attributes: part-level attributes, image-level descriptions, and object-level attributes that cannot support reliable comparison or referring. We additionally remove instances whose bounding boxes are excessively small or large according to predefined area thresholds. The discovery stage produces a blacklist of invalid keys that is applied during production annotation. The final vocabulary is fixed before Stage 2.

Joint annotation. For images containing multiple instances of the same category, the instance crops and the original image are provided together to the model. The model uses the same candidate keys for all instances in the group and assigns values independently to each instance. This shared-key structure is required for subsequent comparison and discriminative referring.

Each value is stored in three forms. *Raw* preserves the original model output. *Display* maps semantically equivalent expressions to a consistent surface form for query generation. *Norm* maps values to canonical forms for programmatic comparison. Discriminative keys are recomputed from normalized values rather than taken directly from the model output.

Key semantics and cardinality. Each key in $\mathcal{V}_\mathrm{attr}$ represents a single semantic axis. Mixed-axis keys are split into separate keys before production annotation. Each key is also assigned a cardinality: single-valued or set-valued. Single-valued keys with multiple conflicting predictions are treated as uncertain, while set-valued keys use set semantics for comparison. Partially overlapping sets are excluded from comparison because neither equality nor inequality is well defined.

The final vocabulary contains (88) keys, including (76) single-valued and (12) set-valued keys. These constraints prevent lexical variation and ambiguous attribute semantics from producing invalid comparisons.

B.3 Spatial Relation Construction

Stage 3 extracts four directional relations between instances. The model predicts only direct pairwise assertions; it is not asked to infer relation chains. Candidate relations are subsequently checked against bounding-box geometry and inverse consistency, after which transitive closure is applied deterministically.

We construct two graph representations. The complete relation graph retains all verified directional edges and supports multi-hop spatial reasoning. The nearest-neighbor graph retains only the nearest uniquely identifiable candidate in each direction under a predefined distance-gap constraint and is used for unambiguous referring.

B.4 Admission Checks

Before instantiating a template, we apply the eight structural checks listed in Table 7. The checks are implemented as deterministic predicates over the structured scene representation and selector (φ). A candidate that fails any applicable check is discarded rather than repaired.

The checks cover three types of constraints.

Uniqueness constraints ensure that referring and extremum conditions identify a single target. For extremum queries, the difference between the first and second candidates must exceed (30/999).

Semantic constraints ensure that attribute comparisons are well defined. Set-valued attributes must be equal or disjoint; partially overlapping sets are excluded. Negative conditions must correspond to verified absent values. Part–whole pairs are excluded from direct attribute comparison.

Completeness and leakage constraints ensure that enumeration and counting queries use complete same-category annotations and that auxiliary descriptions do not directly reveal the answer. Multi-hop queries expose only the information required by the intended reasoning chain.

All applicable checks must pass before a candidate is instantiated. The resulting rejection is final; no failed candidate is repaired by the model. This makes the acceptance decision deterministic and keeps answer correctness tied to the construction process.

Table 5: Construction pipeline of GroundBricks.

#	Stage	By	Operation	Output
0	Scene stratification	rule	Instance-level flattening; three scene strata; one image used once	50,752 images
1	Attribute library	rule + VLM	Size and stuff filters; free-form key discovery, then a closed library	17 general + 6 cluster + 4 scene keys
2	Attribute labeling	VLM	Joint labeling per class group; tri-state values; discriminative and same keys re-derived in code	200,879 instances
3	Spatial relations	rule + VLM	Four directions; geometry veto, inverse check, transitive closure; nearest-neighbor reduction	99,240 relations on 22,792 of 30,807 images
4	Question generation	rule + VLM	21 types $\times$ 3 levels $\times$ 2 modifiers; eight gates; audit against all objects	145,260 questions, 50,060 images

Table 6: Attribute vocabulary and its hierarchical organization.

Tier	Applies to	#keys	Example keys
General	any object	16	color, shape, material, size, surface, pattern
Cluster: person	72 labels	23	hair_color, shirt_color, sleeve_length, expression
Cluster: animal	93 labels	18	fur_color, beak_shape, ear_shape, tail_length
Cluster: plant	39 labels	13	leaf_color, petal_count, bloom_stage, bark_texture
Cluster: vehicle	43 labels	10	body_style, wheel_type, livery, landing_gear
Cluster: building	44 labels	8	wall_material, roof_style, floors, facade
Cluster: food	63 labels	4	topping, contents, filling, container
Scene fallback	isolated object	–	position, background, setting, surrounding
Total		**88**	

Table 7: The eight gates a question must pass before it is emitted.

Gate	Error it prevents	Admission criterion
Unique referring	Two cars both red	The attribute value used to refer must be unique among candidates
No tied extrema	Two objects tie for leftmost	The gap between first and second must exceed $30/999$
Set semantics	Partially overlapping value sets	Equal sets means same, disjoint means different, overlap is dropped
Negation re-check	The target violates its own negated condition	The negated value is taken from another candidate and is disjoint from the target
No part vs whole	Comparing a person with their jacket	A part-of table blocks comparison within attached pairs
Completeness	"box all dogs" misses one	The labeler verifies that no same-class instance is unlabeled; incomplete groups yield no enumeration or counting question
Anchor does not leak	"the one with the same color as the red car"	The anchor is described on an axis other than the compared one
Category does not leak	A chained question names the target class	The question says "the object"; the class appears only in the answer

C PROMPTS

The construction pipeline uses several model prompts. These prompts are shown in Figures 8–11, respectively. They are used only for structured annotation and verification, while query construction and relation composition are handled by deterministic rules. All prompts require structured JSON outputs, and invalid or unparseable responses are discarded.

```
This crop shows a "{label}". Describe its visible visual attributes.
Rules: only attributes directly observable; no guessing; at most 6;
attribute names are up to you - pick whatever fits THIS object best
(short lowercase names); each value is a short phrase of 1-3 words,
no sentences, no parentheses, no explanations;
use a JSON list when an attribute has multiple values.
Reply JSON only: {{"attributes":{{"<name>": "<value or [values]>", ...}}}}
If nothing reliable, reply {{"attributes":{{}}}}.
```

```
This image contains a "{label}" at point {pt} (normalized coords [0,999]).
Describe the visible visual attributes of that specific "{label}".
Rules: only attributes directly observable; no guessing; at most 6;
attribute names are up to you - pick whatever fits THIS object best
(short lowercase names); each value is a short phrase of 1-3 words,
no sentences, no parentheses, no explanations;
use a JSON list when an attribute has multiple values.
Reply JSON only: {{"attributes":{{"<name>": "<value or [values]>", ...}}}}
If nothing reliable, reply {{"attributes":{{}}}}.
```

Figure 8: Prompt for free discovery of attribute keys (Stage 1).

```
This image has {n} annotated objects (normalized coords [0,999], box=[x1,y1,x2,y2], point=[x,y]):
{obj_list}
For each object, decide whether it is worth extracting visual attributes.
Worth = clearly visible, distinct appearance (color/material/shape/state), the annotation localizes ONE specific instance, and the object looks distinctive enough that attributes could tell it apart from similar objects.
Not worth = large background region (sky, grass, road, water); scene-level stuff (pier, pathway, mountain, steps); too small/blurry; heavily occluded; fragmentary; the box encloses MULTIPLE similar instances (group/herd/crowd under one box); or a generic indistinguishable instance whose attributes could not uniquely identify it (one of many similar trees in a forest, one of many identical items).
If the image shows a DENSE PACK of many same-kind instances piled together (a heap/basket/patch where individuals are numerous and barely distinguishable, with no clear spatial arrangement), reject ALL of them.
If several instances of the SAME kind appear in the image, keep ONLY the most prominent, front-most, unobstructed one or two; reject occluded, partial, or background duplicates.
Reply JSON only: {{"objects":[{{"id":1,"ok":true}},{{"id":2,"ok":false,"reason":"..."}}]}}
```

Figure 9: Prompt for instance-level quality control before Stage 2.

```
You will analyze {n} instance(s) of "{label}" in ONE image.
The image is followed by {n} crop(s), one per instance (crop A = instance 1, crop B = instance 2, ...).
Instances (normalized coords [0,999]):
{obj_list}

Tasks:
A. COMPLETENESS - besides these annotated instance(s), are there other clearly visible "{label}" instances in the image that are NOT covered by the boxes? If yes, how many?
B. KEY SELECTION - choose 3 to {max_keys} attribute keys from the list below that best describe these instances AND help tell them apart. The SAME keys must be used for every instance.
{key_list}
C. ATTRIBUTES - for EACH instance and EACH chosen key, give exactly one of:
  - a concrete value (strictly from the key's vocabulary when a vocab is given),
  - "none"    = the instance definitely does NOT have this attribute (e.g. no pattern, not wearing a hat),
  - "unknown" = it cannot be determined (occluded, too small, blurry).
D. KEY ANALYSIS - from the chosen keys: "discriminative_keys" = keys whose values differ across instances ("none" counts as a value); "same_keys" = keys with the same value in every instance.

Rules: only what is directly observable, no guessing; each value is a short phrase of 1-3 words, no sentences; single-value keys take exactly one value, multi-value keys take a JSON list.
Reply JSON only:
{{"complete": true, "missing_count": 0,
  "keys": ["color", "shape"],
  "instances": [{{"id": 1, "attributes": {{"color": "red", "shape": "none"}}}}],
  "discriminative_keys": ["color"], "same_keys": ["shape"]}}
```

Figure 10: Prompt for joint attribute annotation (Stage 2).

```
This image has {n} verified objects (normalized coords [0,999]):
{obj_list}
Task: find CLEAR spatial relations (left_of / right_of / above / below) between pairs of these objects, from the VIEWER's perspective looking at the image.
Rules:
- only relations that are obvious and unambiguous; when in doubt, omit the pair;
- above/below is ONLY for real vertical arrangement: stacked, mounted, hanging, flying, or riding/support (a person ON a motorcycle, a cat ON a mat, a cup ON a table all count as above, even when their boxes overlap);
- objects that are merely FARTHER AWAY (depth/perspective makes them appear higher in the image, e.g. people standing on the same ground further from the camera) are NOT above - omit such pairs;
- judge each pair directly, do not infer chains (a left of b and b left of c does NOT imply a left of c);
- skip pairs that overlap or touch without a clear side relation;
- outputting no pairs is perfectly fine.
Reply JSON only: {{"pairs":[{{"a":2,"rel":"left_of","b":3}}]}} meaning object a is on the left side of object b.
```

Figure 11: Prompt for spatial relation extraction (Stage 3).