OpenGround: Planning-based Online Perception for Open-World 3D Visual Grounding

Wenyuan Huang[1], Zhenyu Zhang[1*], Zhao Wang[2], Zhou Wei[2], Ting Huang[1], Fang Zhao[1], and Jian Yang[1]

[1] Nanjing University, School of Intelligent Science and Technology, Nanjing, China
[2] China Mobile Zijin Innovation Institute, Nanjing, China
*Corresponding Author
wenyuan_huang@smail.nju.edu.cn, zhenyuzhang@nju.edu.cn, wangzh8@js.chinamobile.com, zhouwei9@js.chinamobile.com, huangting0403@sues.edu.cn, zhaofang0627@gmail.com, csjyang@nju.edu.cn
Project Page: https://why-102.github.io/openground.io/

Abstract. 3D visual grounding aims to locate objects based on natural language descriptions in 3D scenes. Existing supervised methods are limited by generalization and recent zero-shot methods typically rely on a predefined Object Lookup Table (OLT) to query Visual Language Models (VLMs) for reasoning about object locations via a single step grounding, which limits the applications in scenarios with undefined targets and complex queries. To address these problems, we present OpenGround, a novel zero-shot framework for open-world 3D visual grounding that remains compatible with recent zero-shot methods. OpenGround integrates Task-Chain Planning to decompose a query into a plan of context-to-target sub-goals for progressive grounding, and Context-Guided Perception to perceive novel objects online under context guidance from the task chain. We also propose a new dataset named OpenTarget, which contains over 7000 object-description pairs to mimic open-world evaluation. Extensive experiments demonstrate that OpenGround achieves competitive performance on Nr3D, state-of-the-art on ScanRefer, and delivers a substantial 17.6% improvement on OpenTarget.

Keywords: 3D Visual Grounding · Vision-Language Model · Zero-shot Scene Understanding

1 Introduction

3D Visual Grounding (3DVG) aims to locate target objects in 3D scenes from natural language queries, with broad applications in AR/VR [6,35,51,65], vision-language navigation [13, 17, 64] and robotic perception [7, 23, 32]. A practical 3DVG approach should not only reason over natural language queries but also flexibly adapt to targets from coarse object categories to fine-grained parts, and generalize to previously unseen real world environments.

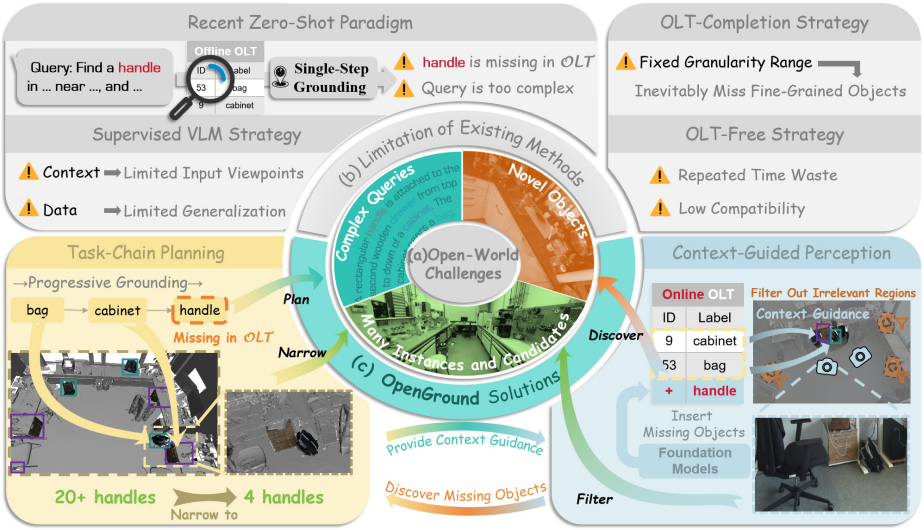


Fig. 1: Overview of Introduction. (a) Open-world 3D visual grounding faces challenges from complex queries, novel objects, and massive instances and candidates. (b) Existing methods are limited: supervised 3D-VLMs suffer from data scarcity and restricted input views; recent zero-shot pipelines rely on an offline OLT and break when the target is missing or the query is compositional, and either OLT-Completion Strategy is impractical or OLT-Free Strategy repeatedly scans the whole scene and is less compatible with OLT-based backbones. (c) Our proposed OpenGround framework addresses these limitations through Task-Chain Planning, which progressively grounds sub-goals to narrow the search space, and Context-Guided Perception, which dynamically discovers missing objects and turns a static offline OLT into an online OLT.

Research in 3DVG has made significant progress, encompassing both supervised and zero-shot methods. While supervised methods [11, 14, 16, 24, 27, 39, 68, 73], trained on diverse datasets [1, 4, 6, 10, 47, 63], and zero-shot methods [25, 26, 34, 46, 56, 58, 62, 71] leveraging the power of VLMs, both achieve high accuracy on existing benchmarks. However, in practical deployment, they are facing open-world challenges: as illustrated in Fig. 1-(a), there are long and complex queries, novel and fine-grained objects, and massive instances and candidates.

Under such challenges, supervised methods face two practical constraints, in Fig. 1-(b) "Supervised VLM Strategy": first, supervision scarcity limits real-world generalization; second, context window of supervised 3D-VLM limits its number of input views, probably missing critical evidence in large scenes. Further, recent zero-shot methods [25, 26, 34, 46, 62, 71] adopt a paradigm in Fig. 1-(b) "Recent Zero-Shot Paradigm": they use 2D VLMs [3, 37, 54] for single-step grounding with a predefined object lookup table ($\mathcal{OLT}$). However, they are fundamentally limited by the predefined offline OLT which is built on a fixed granularity range before queries and inevitably misses novel and fine-grained objects, even though a stronger segmentor is provided by AffordBot [56] in **OLT-Completion Strat-**

egy. At the other extreme, **OLT-Free Strategy**, such as VLM-Grounder [58], abandons $\mathcal{OLT}$ entirely but suffers from repeated searching whole scene for common objects and less compatible with recent OLT-based methods. Moreover, single-step formulation is brittle for complex queries which require multi-step evidence gathering. These suggest an intuitive solution: instead of exhaustively completing $\mathcal{OLT}$ or abandoning it in a single-step grounding, a practical open-world 3DVG system should plan and ground progressively to resolve complex queries, and discover novel objects to extend $\mathcal{OLT}$ online.

In this paper,we propose OpenGround, a zero-shot framework designed to address the open-world challenges of 3D visual grounding through two key modules, while remaining compatible with existing OLT-based pipelines. As illustrated in Fig. 1-(c), given a query, Task-Chain Planning decomposes it into a plan with a sequence of sub-goals. By grounding through the plan progressively, OpenGround narrows the search space for the final target. Complementing the task chain, Context-Guided Perception (CGP) discovers missing objects by leveraging previously grounded objects as context guidance. When a sub-goal does not exist in $\mathcal{OLT}$, CGP filters out irrelevant regions to perceive the missing object and inserts it into the table online. It effectively transforms a static offline OLT into a dynamic online OLT. As a result, subsequent single-step grounding can continue on the updated online $\mathcal{OLT}$, enabling efficient grounding of both predefined and previously unseen objects under complex open-world queries.

To evaluate our framework, we also construct a novel benchmark named OpenTarget to simulate open-world evaluation. The benchmark takes ScanNet++ [60] as the basement, and integrate fine-grained part-level segmentation results from Articulate3D [12] as the targeted but undefined 3D objects to simulate the open-world setting. With the automatic description generating and rigorous quality filtering, we obtain 7,724 query-object pairs which are sufficiently comprehensive to support reliable validation experiments. Overall, our contributions are summarized as follows:

- We define a novel task for zero-shot 3DVG: grounding objects outside the predefined object lookup table ($\mathcal{OLT}$) in 3D environments using natural language descriptions. We further propose a benchmark named OpenTarget to enable rigorous evaluation of the open-world 3DVG task. The dataset uses VLM-generated natural language descriptions, and ensures final quality through manual filtering, resulting in 7,724 high-quality query-object pairs.
- We extend the existing zero-shot 3DVG paradigm by integrating the Context-Guided Perception (CGP), which enables grounding objects beyond $\mathcal{OLT}$.
- We propose Task-Chain Planning for progressive grounding to resolve complex queries. It also restricts CGP application to regions around already grounded context objects, balancing efficiency and precision.

2 Related Work

Supervised 3DVG. Supervised 3D visual grounding began with ScanRefer [6] and Referit3D [1], and can be categorized into two paradigms: two-stage and

single-stage methods. Two-stage approaches [1, 6, 18, 24, 63, 66, 72] first generate object proposals via 3D segmentation [20, 33, 44, 53], then match them to textual queries. In contrast, single-stage methods [11, 14, 16, 27, 39, 40, 52, 57, 68] employ end-to-end architectures that jointly learn 3D and language representations for direct grounding. Recent advances [8, 16, 39, 59, 70] further integrate VLMs to enhance 3D understanding, forming 3D-VLMs. Notably, GS-Reasoner [8] leverages chain-of-thought reasoning to better handle long and complex queries; however, such 3D-VLM solutions limit number of input views and involve expensive, hard-to-control CoT inference, which restricts evidence gathering and practical deployment. Moreover, despite their strong performance, particularly 3D-R1 [16] which achieves SoTA through large-scale GRPO [45] training, these methods remain constrained by training data and fail to generalize in open-world grounding.

Zero-Shot 3DVG. Zero-shot 3DVG aims to improve generalization beyond supervised 3DVG by leveraging pre-trained VLMs. Early attempts [19, 38] follow CLIP-based paradigms, where query is matched against 3D proposals or fused scene features via cross-modal similarity, enabling zero-shot grounding but suffering from multiple candidates. Recent trend [25, 26, 34, 46, 62, 71] uses an **offline object lookup table ($\mathcal{OLT}$)** to assist VLMs: methods first retrieve candidates from the $\mathcal{OLT}$ (from 3D segmentation or CLIP-based detection), then perform VLM-based single-step confirmation to ground the target, achieving strong accuracy and improved efficiency in closed-set settings. However, the reliance on a predefined $\mathcal{OLT}$ fundamentally limits open-world deployment, since undefined or long-tailed objects (especially fine-grained parts) are not covered. Existing remedies fall into two extremes: AffordBot [56] densifies $\mathcal{OLT}$ with a stronger segmentor but still inevitably misses novel objects; VLM-Grounder [58] abandons $\mathcal{OLT}$ at the cost of repeated full-scene scanning and lower compatibility with modular OLT-based pipelines. Moreover, they remain single-step and struggle with complex queries, which is common in open-world deployment.

Advances in VLMs and Visual Perception. Recent VLMs [2, 3, 9, 28–30, 48–50, 54, 55], such as the Qwen-VL series, show strong cross-domain generalization and fine-grained spatial understanding [48, 55]. Meanwhile, 3D perception frameworks like PointGroup [20], Mask3D [44], and Point-SAM [69] have advanced but still lack open-world generalization due to limited 3D data. In contrast, 2D foundation models [15, 22, 31, 42, 43], such as SAM [22, 42], achieve strong open-world segmentation and grounding. The convergence of spatially aware VLMs and generalizable 2D perception models makes open-world 3D grounding promising.

3 Dataset

To evaluate the open-world grounding capability of OpenGround, we construct a novel dataset named OpenTarget to **mimic** open-world settings, based on ScanNet++ [60] and Articulate3D [12]. Existing 3DVG benchmarks [1, 6] focus on object-level instances with limited category diversity, failing to simulate the open-world scenarios where fine-grained, undefined objects (e.g., sink handles, cabinet doors) are pervasive. In contrast, OpenTarget introduces objects from

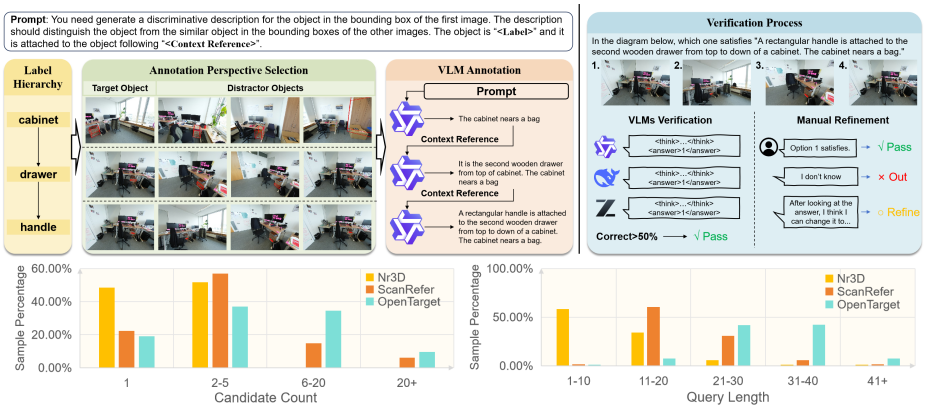


Fig. 2: OpenTarget dataset construction and statistics. Top: A progressive VLM-assisted pipeline builds discriminative queries for part-level targets from Articulate3D on ScanNet++ scenes, using a label hierarchy (e.g., cabinet→drawer→handle), target–distractor viewpoint selection, and context-aware description generation conditioned on parent annotations, followed by VLM voting and human refinement. **Bottom:** Compared with Nr3D and ScanRefer, OpenTarget shows larger candidate pools and longer, more compositional queries, reflecting harder grounding.

Articulate3D absent in the $\mathcal{OLT}$ built on ScanNet++. These fine-grained parts **mimic** unforeseen objects in open-world scenarios, providing a benchmark for open-world grounding. OpenTarget provides totally 7,724 object-pairs for segments in Articulate3D, across 50 object classes and 70 part classes. We outline the dataset construction and partial statistics below and details in **Appendix**.

Annotation. As illustrated in Fig. 2, we adopt a progressive VLM-based strategy for annotation, involving three stages: Label Hierarchy, Annotation Perspective Selection, and VLM Annotation. First, we utilize the hierarchical labels from Articulate3D (e.g., cabinet→drawer→handle), starting from top-level objects in subsequent processes. Second, we collect the target and its distractors, and select suitable perspectives for each. Third, we prompt VLMs with these perspectives and labels to generate queries that distinguish the target from distractors. For child objects, we repeat the process with parent object annotations as contextual references, ensuring progressively detailed and discriminative queries.

Verification. We employ a two-stage quality verification process, combining automatic filtering and manual refinement. First, for objects with the same label, we use multiple VLMs to vote on object-query pairs (via selected perspectives and object IDs), retaining only majority-approved pairs to manual review. Subsequently, human annotators verify these pairs, with the ability to mark them as "unidentifiable" or revise queries, ensuring high dataset quality.

Statistics. OpenTarget contains 7,724 object-query pairs spanning 50 object classes and 70 part classes, built on ScanNet++ scenes with part-level objects from Articulate3D. Compared with object-level 3DVG benchmarks, OpenTarget poses a challenging open-world setting, since targets are fine-grained parts that

are absent from the $\mathcal{OLT}$ and the training datasets of 3D segmentor, and appear higher ambiguity. As shown in Fig. 2, OpenTarget exhibits a larger number of same-label instances per scene (average **8.35** vs. **4.13** on ScanRefer and **3.03** on Nr3D), which substantially increases distractors for grounding. Meanwhile, OpenTarget queries are longer (median length **31** words vs. **19** on ScanRefer and **9** on Nr3D), reflecting possible multi-step reasoning requirement. These statistics highlight that OpenTarget evaluates open-world grounding under both higher visual ambiguity and more complex queries.

4 Preliminary

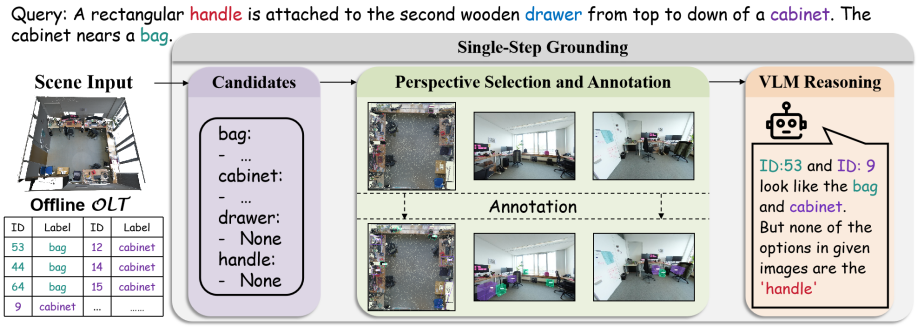


Fig. 3: Single-Step Grounding. Recent zero-shot methods [25,26,34,46,62,71] follow the paradigm: decompose the query into objects and retrieve candidates from the predefined $\mathcal{OLT}$, then select informative views to observe and annotate candidates, finally find target ID through VLM reasoning and obtain 3D bounding box from $\mathcal{OLT}$.

OpenGround is built upon the OLT-based Single-Step Grounding paradigm which is widely adopted in many recent zero-shot 3DVG methods, and extends it to handle open-world targets and complex queries. Given a 3D scene S (e.g., point cloud) and a natural-language query $\mathcal{Q}$, they first construct an object lookup table ($\mathcal{OLT}$) via existing 3D scene segmentation model, denoted as

$$\mathcal{OLT} = \{(\mathbf{ID}_i, \mathbf{label}_i, \mathbf{bbox}_i)\}_{i=1}^{N}, \tag{1}$$

which maps object IDs to their semantic labels and 3D bounding boxes.

As illustrated in Fig. 3, the paradigm consists of three stages.

Candidate Retrieval. The paradigm first extracts objects mentioned in $\mathcal{Q}$ and decomposes them into anchors $\mathcal{A}$ and targets $\mathcal{T}$, where anchors provide references and targets denote the objects to be grounded (e.g., in "find the *laptop* in front of the *chair*", *chair* $\in \mathcal{A}$ and *laptop* $\in \mathcal{T}$). Then, it retrieves candidates of these objects from $\mathcal{OLT}$ and filters out impossible candidates.

Perspective Selection and Annotation. Given retrieved candidates, the paradigm selects informative views that reveal these candidates, so that VLM

can distinguish them. The selected views are rendered from the 3D scene, and each candidate's 3D bounding box is projected onto the corresponding images using camera poses, producing 2D visual annotations (e.g., boxes with instance IDs). The annotated multi-view images explicitly indicate the coordinates of each candidate, serving as the visual evidence for the subsequent VLM reasoning.

VLM Reasoning. Finally, the paradigm feeds $\mathcal{Q}$ and the annotated multi-view images into a VLM, which performs single-step grounding by selecting the most plausible instance ID among the annotated candidates according to the semantics and spatial relations. The predicted ID is then mapped back to its corresponding 3D bounding box in $\mathcal{OLT}$, yielding the target in the 3D scene.

5 Methodology

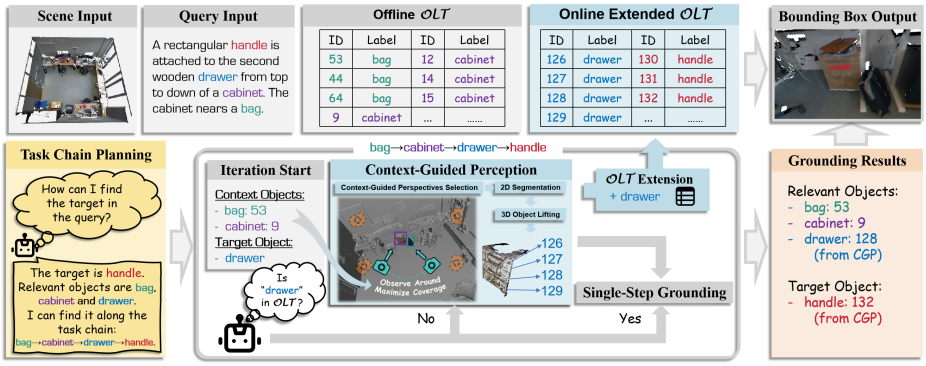
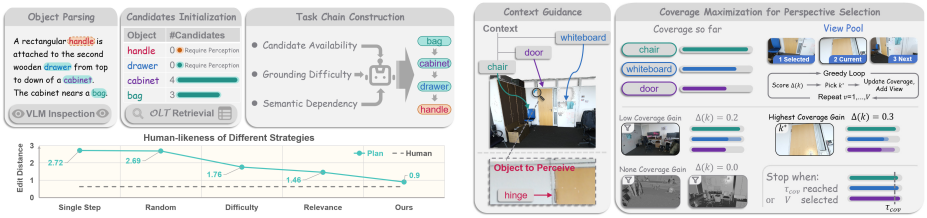


Fig. 4: Overview of OpenGround. OpenGround keeps the OLT-based single-step grounding backbone and augments it with Task-Chain Planning and Context-Guided Perception (CGP). It first plans an ordered sequence of sub-goals, then iterates over the sequence. If a sub-goal is missing from the current $\mathcal{OLT}$, CGP discovers new objects around grounded objects and extends the $\mathcal{OLT}$ online, after which single-step grounding proceeds on the extended $\mathcal{OLT}$ to retrieve the 3D bounding box.

Overview. OpenGround builds on the OLT-based single-step grounding in Sec. 4 and extends it to open-world 3DVG with two components in Fig. 4: **Task-Chain Planning** plans an ordered sequence of sub-goals over context objects and the final target, so grounding can proceed step-by-step and each sub-goal continuously narrows the remaining search space for subsequent steps; **Context-Guided Perception (CGP)** discovers novel objects under context guidance from task chain, which moves the offline OLT to an online OLT. When the current sub-goal does not exist in $\mathcal{OLT}$, CGP leverages already grounded context objects to filter out irrelevant regions and performs perception to discover missing objects and extend the $\mathcal{OLT}$ on-the-fly. Subsequent single-step grounding then continues on the extended online $\mathcal{OLT}$, enabling grounding of both predefined and previously unseen objects under complex open-world queries.



(a) Task-Chain Planning. **(b)** Context-Guided Perspectives Selection.

Fig. 5: Illustrations of key components in OpenGround. (a): Task-Chain Planning decomposes a query into an ordered sequence of sub-goals. (b): Context-Guided Perspective Selection finds views greedily maximizing context objects coverage.

5.1 Task-Chain Planning

When grounding fine-grained objects or in crowded scenarios, queries naturally become complex. In such cases, humans rarely make a one-shot decision; instead, they leverage context and make intermediate confirmations. For example, given query "find a *handle* of the top *drawer* of the *kitchen cabinet*", humans would locate the *cabinet* in the *kitchen*, then identify *drawer* and *handle*, instead of selecting the handle from tens of candidates across drawers and rooms. Even when human grounding appears parallel, it can be viewed as sequential atomic steps, necessary for deterministic program. This motivates grounding strategy: decomposing a query into a sequence of sub-goals, where grounded objects narrow the search space for next steps. However, obtaining a useful strategy is non-trivial. Simple strategies, e.g., ground target directly once an anchor confirmed, using a random order, sorting by difficulty (proxy by the candidate set size), or sorting by semantic relevance, misalign with the dependency structure in complex queries (e.g., attempting a missing target too early or ignoring spatial priors). We validate this via a user study (settings in **Appendix**) and find that they produce less human-like (larger edit distance) plans, as illustrated in Fig. 5a. Thus, we introduce a novel task chain strategy to obtain human-like plans.

Object Parsing and Candidates Initialization. We use a VLM to extract object mentioned in $\mathcal{Q}$, including one target label L^{tgt} and a set of relevant context labels $\{L_j^{ctx}\}_{j=1}^n$. For each label in $\mathcal{L} = \{L^{tgt}\} \cup \{L_j^{ctx}\}_{j=1}^n$, we initialize its candidate set from the predefined offline $\mathcal{OLT}$ using the same candidate retrieval described in Sec. 4. The candidate set size $|\mathcal{C}_j|$ serves as a proxy for grounding difficulty, following the difficulty partitioning in Nr3D [1].

Task-Chain Construction. The core challenge lies in determining an optimal grounding order that respects the implicit context-dependency structure within compositional query. We identify three key factors that influence the ordering:

1. **Candidate Availability**: Objects absent ($|\mathcal{C}_j| = 0$) require context-guided perception and must be grounded *after* their context-dependency objects.
2. **Grounding Difficulty**: Objects with larger candidate sets have broader search spaces and thus higher grounding difficulty. Grounding them later allows previously grounded context objects to narrow the search space.
3. **Semantic Dependency**: Objects linked by spatial or possessive relations (e.g., "handle *of* the drawer") should be grounded in dependency order.

To capture these factors, we design a structured prompt that inputs: (1) the original query $\mathcal{Q}$; (2) the extracted labels $\mathcal{L}$ with their candidate counts; (3) special markers for objects without any candidates, which require Context-Guided Perception. The VLM is instructed to reason over semantic dependencies based on linguistic cues and commonsense, then output an ordered index sequence:

$$\mathcal{T} = \mathbf{VLM}\left(\mathcal{Q}, \mathcal{L}, \{\|\mathcal{C}_j\| \mid j \in [1, n+1]\}\right). \tag{2}$$

Notably, we adopt a soft constraint to encourage grounding the target object last through a prompt, but permits early target grounding when the target possesses strong unique identifiers (e.g., much less candidates, or salient visual features). This flexibility allows adaptively prioritizing the target when necessary.

Validation via User Study. To assess alignment with human planning, we compare with the baselines above. Our method achieves an edit distance of 0.90 ± 0.49 to human annotations, outperforming alternative strategiesand approaches the inter-participant agreement level of 0.63 ± 0.29, indicating human-like strategies. Details about the user study can be found in the **Appendix**.

5.2 Context-Guided Perception

Since task-chain planned and previously grounded context objects provide spatial priors, we introduce Context-Guided Perception (CGP) to leverage the context to filter irrelevant views and discover missing objects to extend $\mathcal{OLT}$ online. As illustrated in Fig. 4, CGP selects informative views under context guidance, applies 2D foundation models to perceive missing objects, lifts them to 3D, and inserts discovered objects into $\mathcal{OLT}$. This converts the offline OLT into an online OLT while preserving the same downstream interface for single-step grounding.

Context-Guided Perspective Selection. Selecting observation views is crucial for efficient and accurate perception. Existing 3DVG methods rely on heuristics that either underuse contextual cues or waste views: SeeGround [25] uses a single geometry-derived view that often misses task-relevant regions or yields distant views with insufficient detail; SeqVLM [26] favors zoomed-in views where each candidate dominates the image, producing redundant views and losing context; Struct2D [71] uses bird's-eye views that suffer from occlusion and miss side details. These approaches share a common limitation: they lack awareness of complex task structure and cannot leverage the progressively grounded context from task-chain to focus on where the missing object is most likely to appear.

Our key insight is that grounded context objects $[O_1, \cdots, O_{t-1}]$ serve as spatial anchors for task-relevant regions implied by the query's compositional relations. We thus select views that maximize coverage of these context objects, which exposes nearby regions where the missing object is likely to appear.

Specifically, for the current task T_t with missing candidates, we select views $\mathcal{I}^\star \subseteq \mathcal{I}$ around $[O_1, \cdots, O_{t-1}]$ as context guidance. We prefer views that increase coverage of these context objects, as illustrated in Fig. 5b. We adopt a greedy strategy to select at most V perspectives (global optimization is NP-hard), inspired by Video-LLM [67] which uses greedy strategy to sample views to obtain

maximum coverage of full scene. For each object O_i, initialize $\mathcal{R}_{O_i}^0 = \emptyset$ and iteratively add views for $v = 1, \cdots, V$. For each view k, coverage gain is:

$$\Delta(k) = \sum_{O_i} \frac{|\mathcal{P}_{O_i} \cap \mathcal{P}_k \backslash \mathcal{R}_{O_i}^{v-1}|}{|\mathcal{P}_{O_i}|}, \tag{3}$$

where $\mathcal{P}_{O_i}$ is the point set of O_i, $\mathcal{P}_k$ denotes the 3D points visible from view k, and $\mathcal{R}_{O_i}^{v-1}$ denotes the observed region after $v-1$ selected perspectives. At step v, we select $k^\star = \arg\max_k \Delta(k)$, add $\mathbf{I}_{k^\star}$ into $\mathcal{I}^\star$, and update:

$$\mathcal{R}_{O_i}^{v} = \mathcal{R}_{O_i}^{v-1} \cup \left(\mathcal{P}_{O_i} \cap \mathcal{P}_{k^\star}\right), \forall O_i. \tag{4}$$

The summation encourages balanced coverage across all context objects. The process stops when coverage reaches τ_{cov} or V views are selected. With context-guided views $\mathcal{I}^\star$ ready, we proceed to detect missing objects via following.

2D Segmentation and Lifting. Given $\mathcal{I}^\star$, we detect missing objects in each image with an open-world 2D segmentor and lift the masks to 3D to extend the $\mathcal{OLT}$. For each view k, we obtain $\mathcal{M}_k = \mathbf{SEG}(\mathbf{I}_k, L_{T_t})$, and lift them to 3D using the corresponding point cloud and camera parameters. We aggregate lifted masks across views and iteratively merge those with high overlap (3D IoU $\geq \tau_{iou}$) to obtain distinct 3D instances $\{\mathcal{P}_i\}_{i=1}^M$, following existing work [36]. Finally, we covert each object to a 3D bounding box and insert to $\mathcal{OLT}$:

$$\mathcal{OLT} \leftarrow \mathcal{OLT} \cup \Big\{ \big(\mathbf{ID}(i),\, L_{T_t},\, \mathbf{bbox}(\mathcal{P}_i)\big) \;\big|\; i \in [1, M] \Big\}, \tag{5}$$

where $\mathbf{ID}(i)$ denotes the next unique id and $\mathbf{bbox}(\mathcal{P}_i)$ is the bounding box.

In summary, Task-Chain Planning orchestrates the pipeline by processing $\mathcal{T} = [T_1, \cdots, T_{n+1}]$ sequentially: for each T_t, if candidates exist in $\mathcal{OLT}$, directly apply single-step grounding; otherwise, CGP uses previously grounded objects to discover missing objects, extends $\mathcal{OLT}$ online, and then applies single-step grounding. Each resolved sub-goal progressively narrows the search space, enabling OpenGround to handle complex queries and novel objects in open-world.

6 Experiments

6.1 Experimental Settings

Datasets. We evaluate OpenGround on three datasets to assess both open-world and OLT-dependent grounding performance. For open-world evaluation, we conduct experiments on all 7,724 queries of OpenTarget (details in Sec. 3), which contains part-level objects beyond usual categories to simulate the open-world scenarios, to test the ability of grounding open-world objects. For OLT-dependent evaluation, we use two popular benchmarks: ScanRefer [6] and Nr3D [1]. ScanRefer provides 9,508 queries in validation set, on ScanNet [10] scenes. Nr3D, part of the ReferIt3D benchmark [1], includes 7,805 queries in validation set and offers ground-truth 3D bounding boxes for objects.

Implementation Details. Our experiments utilize the GLM-4.5V [50] as the VLM for OpenTarget evaluation, CLIP-ViT-Base-Patch16 text encoder [41] for text similarity matching, GroundedSAM [43] as **SEG** and set V=3, τ_{cov}=0.95, τ_{iou}=0.5 where τ_{cov} and τ_{iou} follow existing works [36, 67]. For the $\mathcal{OLT}$ on ScanRefer and OpenTarget benchmarks, we follow the object detection procedure outlined in ZSVG3D [61] for fair comparison, which utilizes Mask3D [44].

6.2 Comparative Study

We conduct experiments across benchmarks, including Nr3D [1], ScanRefer [6], and OpenTarget. We present detailed performance on Nr3D and ScanRefer, with category-wise breakdowns in Tabs. 2 and 3, and report OpenTarget comparisons in Tab. 1. Due to time constraints, we only evaluate the performance of recently proposed top-performing and open-source zero-shot methods on OpenTarget.

Table 1: Performance on **OpenTarget**. * denotes results on randomly selected 300 samples due to low efficiency. Methods fail due to missing objects in the Mask3D $\mathcal{OLT}$. **Easy** contains queries with hierarchy length ≤ 2 (5,737 samples), while **Hard** contains queries with hierarchy length > 2 (1,987 samples).

Method	$\mathcal{OLT}$	Easy		Hard		Overall	
		Acc@0.25	Acc@0.50	Acc@0.25	Acc@0.50	Acc@0.25	Acc@0.50
SeeGround [25]	Mask3D [44]	13.3	11.2	1.2	1.1	10.2	8.6
VLM-Grounder* [58]	-	14.5	12.1	6.1	2.3	12.3	9.6
VLM-Grounder* [58]	Mask3D [44]	16.4	12.0	10.2	4.8	14.8	10.1
SeqVLM [26]	Mask3D [44]	13.6	11.2	1.4	1.1	10.5	8.6
GPT4Scene [39]	Mask3D [44]	10.1	7.7	0.9	0.7	7.7	5.9
ZSVG3D [61]	Mask3D [44]	8.9	6.9	0.8	0.7	6.8	5.3
Ours	Mask3D [44]	**51.8**	**38.9**	**30.2**	**20.6**	**46.2**	**34.2**
SeeGround [25]	GT	20.2	19.8	11.3	10.4	17.9	17.4
VLM-Grounder* [58]	GT	31.4	21.3	20.6	17.8	28.6	20.4
SeqVLM [26]	GT	21.5	21.3	13.4	13.2	19.4	19.2
GPT4Scene [39]	GT	13.6	13.4	7.9	7.3	12.1	11.8
ZSVG3D [61]	GT	12.2	12.0	7.6	6.8	11.0	10.7
Ours	GT	**57.9**	**57.4**	**45.7**	**45.3**	**54.8**	**54.3**

OpenTarget. As illustrated in Tab. 1, on the open-world benchmark, existing methods are constrained by predefined offline OLT which inevitably lacks the target objects, resulting in severely degraded performance (visualized in Fig. 6(a)). To fairly assess their upper bound, according to the conclusion in relevant works that better $\mathcal{OLT}$ results better performance, we further evaluate them with ground-truth (GT) $\mathcal{OLT}$. However, they are still confounded by the larger candidate pool, resulting in erroneous predictions visualized in Fig. 6(b) and reported in Tab. 1. In contrast, our method far surpasses baselines although they are given GT $\mathcal{OLT}$. It shows that OpenGround effectively extends the $\mathcal{OLT}$ with novel objects and resolves complex queries, enabling open-world grounding.

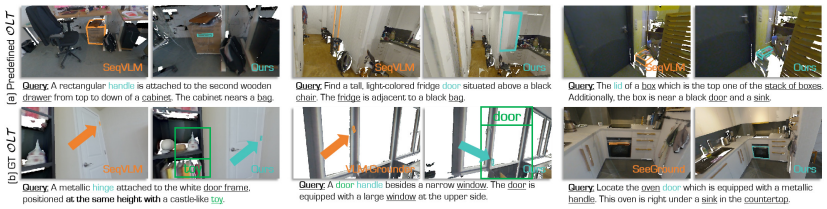


Fig. 6: Qualitative Comparisons on **OpenTarget**. (a) compares our method with previous open-source SoTA method, SeqVLM [26] with predefined $\mathcal{OLT}$. They fail to ground objects out-of-$\mathcal{OLT}$. (b) compares our method with previous methods equipped with ground-truth $\mathcal{OLT}$. these baselines either overlook key objects (e.g., toy, door) or are distracted by numerous relevant objects, leading to failures.

Table 2: Quantitative Comparisons on **ScanRefer** [6]. Results are reported for "Unique" (scenes with a single target object) and "Multiple" (scenes with distractors of the same class) subsets. * denotes results on selected 250 samples.

Method	Supervision	VLM	Unique		Multiple		Overall	
			Acc@0.25	Acc@0.50	Acc@0.25	Acc@0.50	Acc@0.25	Acc@0.50
ViewSRD [14]	Supervised	-	82.1	68.2	37.4	29.0	45.4	36.0
3D-R1 [16]	Supervised	-	-	-	-	-	**65.8**	**59.2**
GPT4Scene [39]	Supervised	-	90.3	83.7	56.4	50.9	62.6	57.0
TSP3D [11]	Supervised	-	-	-	-	-	56.4	46.7
SeeGround [25]	Zero-Shot	Qwen2-VL-72b [54]	75.7	68.9	34.0	30.0	44.1	39.4
Ours	Zero-Shot	Qwen2-VL-72b [54]	**76.6**	**70.3**	**48.1**	**40.4**	**53.7**	**46.3**
SeqVLM [26]	Zero-Shot	Doubao-1.5-pro [5]	77.3	72.7	47.8	41.3	55.6	49.6
Ours	Zero-Shot	Doubao-1.5-pro [5]	**78.3**	**75.1**	**59.2**	**49.4**	**63.0**	**54.4**
VLM-Grounder* [58]	Zero-Shot	GPT-4o [37]	51.6	32.8	**66.0**	29.8	48.3	33.5
SPAZER [21]	Zero-Shot	GPT-4o [37]	80.9	72.3	51.7	43.4	57.2	48.8
Ours	Zero-Shot	GPT-4o [37]	**83.5**	**75.2**	59.9	**51.2**	**64.5**	**55.9**
ZSVG3D [61]	Zero-Shot	GPT-4 turbo [37]	63.8	58.4	27.7	24.6	36.4	32.7
Ours	Zero-Shot	GPT-4 turbo [37]	**78.0**	**75.8**	**58.5**	**48.2**	**62.3**	**53.6**

Nr3D & ScanRefer. OpenGround achieves competitive performance across both benchmarks. On Nr3D [1], we achieve competitive accuracy to the zero-shot SoTA SPAZER [21]. On ScanRefer [6], we outperform all zero-shot baselines and narrow the gap to supervised methods. These improvements are mainly incremental rather than dramatic, because the evaluation settings are simpler: queries are shorter, and scenes contain fewer candidates, which reduces the need for task-chain planning, shown in Sec. 3. Moreover, targets in Nr3D and ScanRefer are largely covered by the predefined OLT, so our context-guided perception is rarely used and thus contributes little under these close-world conditions.

6.3 Ablation Study

To validate each module in OpenGround, we conduct ablation studies on the OpenTarget. All experiments use the same settings as the main results.

Table 3: Detailed Performance on **Nr3D** [1]. Queries are categorized as "Easy" (with one distractor) or "Hard" (with distractors), and as "Dep." (View-Dependent) or "Indep." (View-Independent) based on viewpoint requirements for grounding. "**Ours**+SPAZER [21]" seamlessly integrates its advanced module into our framework.

Method	Supervision	VLM	Easy	Hard	Dep.	Indep.	Overall
3D-R1 [16]	Supervised	-	-	-	-	-	68.8
TSP3D [11]	Supervised	-	-	-	-	-	48.7
ViewSRD [14]	Supervised	-	75.3	64.8	68.6	70.6	**69.9**
SeeGround [25]	Zero-Shot	Qwen2-VL-72b [54]	54.5	38.3	42.3	48.2	46.1
Ours	Zero-Shot	Qwen2-VL-72b [54]	**54.9**	**44.6**	**46.4**	**52.9**	**50.6**
SeqVLM [26]	Zero-Shot	Doubao-1.5-pro [5]	58.1	47.4	51.0	54.5	53.2
Ours	Zero-Shot	Doubao-1.5-pro [5]	**63.8**	**57.4**	**58.9**	**61.4**	**60.5**
VLM-Grounder [58]	Zero-Shot	GPT-4o [37]	55.2	39.5	45.8	49.4	48.0
SPAZER [21]	Zero-Shot	GPT-4o [37]	**68.0**	58.8	**59.9**	**66.2**	**63.8**
Ours	Zero-Shot	GPT-4o [37]	64.3	**59.3**	59.2	63.1	61.7
Ours+SPAZER [21]	Zero-Shot	GPT-4o [37]	**70.1**	**59.8**	**60.4**	**67.2**	**64.8**
ZSVG3D [61]	Zero-Shot	GPT-4 turbo [37]	46.5	31.7	36.8	40.0	39.0
Ours	Zero-Shot	GPT-4 turbo [37]	**59.3**	**55.1**	**55.1**	**58.2**	**57.1**

Effect of Initial $\mathcal{OLT}$. As shown in Tab. 4, OpenGround remains effective even without an initial $\mathcal{OLT}$, outperforming baselines with GT $\mathcal{OLT}$. Nevertheless, incorporating an initial $\mathcal{OLT}$ offers priors that further improves performance.

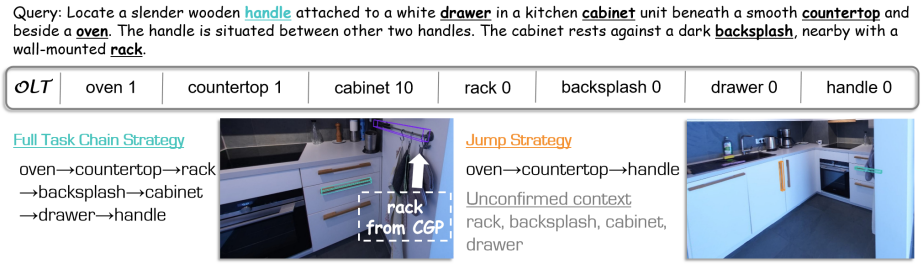
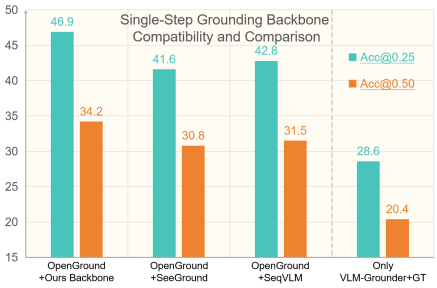


Fig. 7: Grounding Strategy Results Comparison. Jump strategy incorrectly identifies the object (orange) because it skips the key object, rack. **Task Chain** strategy correctly identifies the object (teal), considering the rack (purple).

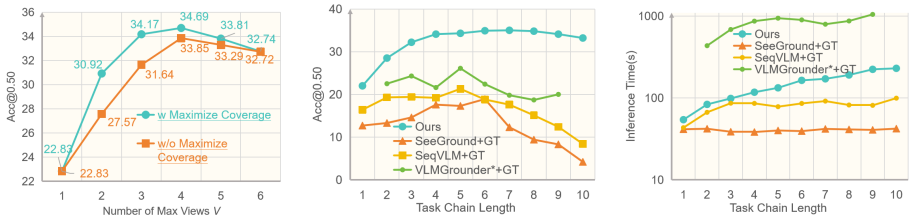
Task Chain Construction Strategy. We compare several grounding plan construction strategies (Tab. 4, rows 1, 3–6): **Random** uses a random order; **Jump** grounds target directly once an object confirmed; **Difficulty** prioritizes easier objects by candidate set size; **Semantic** orders by relevance to the target; **Task Chain** is our strategy and achieves the best performance. Fig. 7 further compares with **Jump**, where ours yields more coherent intermediate grounding and more accurate grounding via progressive planning. Together with Fig. 5a, these results demonstrate that human-like planning improves open-world grounding.

Table 4: Ablation study of different components on OpenTarget Acc@0.50. **Initial $\mathcal{OLT}$**: whether using initial $\mathcal{OLT}$ from Mask3D [44]; **Grounding Strategy**: strategy for constructing grounding sequence; **VLM**: the vision-language model used. **Right:** swap only the Single-Step Grounding backbone within OpenGround (others fixed).

#	Initial $\mathcal{OLT}$	Grounding Strategy	VLM	Acc
(1)	Yes	Task Chain	GLM-4.5V [50]	34.2
(2)	*No*	Task Chain	GLM-4.5V [50]	27.1
(3)	Yes	*Jump*	GLM-4.5V [50]	29.8
(4)	Yes	*Relevance*	GLM-4.5V [50]	32.6
(5)	Yes	*Difficulty*	GLM-4.5V [50]	31.5
(6)	Yes	*Random*	GLM-4.5V [50]	29.2
(7)	Yes	Task Chain	*Qwen3-VL-32B* [48]	30.4
(8)	Yes	Task Chain	*Qwen3-VL-235B* [48]	32.8
(9)	Yes	Task Chain	*Step3* [49]	33.4



Influence of VLM. To verify VLM dependence and adaptability, we replace the default GLM-4.5V [50] with VLMs of varying scales. As shown in Tabs. 2 to 4, OpenGround's advantage stems from our contributions rather than large-scale VLM capabilities. Notably, smaller VLMs induces minimal performance degradation, even that the Qwen3-VL-32B [48] only drops 3.8%, still outperforms baselines that leverage GT $\mathcal{OLT}$ (e.g., 10% higher than VLM-Grounder).



(a) Perspective selection vs. V. **(b)** Accuracy vs. chain length. **(c)** Time vs. chain length.

Fig. 8: Ablation studies. (a): impact of perspective selection strategies with varying max views V. (b) and (c): performance and efficiency across query complexity which is measured by task chain length our method planned. "VLMGrounder*+GT" is evaluated only on a subset of 300 samples due to its extremely high computational cost, and lacks values at $L = 1$ and $L = 10$ since these lengths do not appear in the sampled subset.

Single-Step Grounding Backbones. We evaluate the compatibility of our framework with existing methods. As shown in Tab. 4, the single-step grounding module can be seamlessly replaced with prior methods such as SeeGround [25] and SeqVLM [26]. Results show that CGP enables their open-world grounding ability and Task-Chain Planning boosts performance, confirming that OpenGround is a general and flexible framework rather than a standalone method.

Perspectives Selection Strategy. As illustrated in Fig. 8a, we further study the impact of the perspectives selection strategy:

- **Number of views V:** When $V = 1$, both strategies start at the same low baseline. As V increases to 4, performance of both strategies rises significantly and peaks at 34.7 and 33.8. Exceeding $V = 4$ leads to performance drops, indicating that excessive views introduce redundancy and confusion. Though $V = 4$ is most accurate, it only brings a 0.52% gain but increases input views by 33.3%. We adopt $V = 3$, balancing performance and efficiency.
- **Maximizing Coverage Strategy:** The 'w Maximize Coverage' outperforms across all V values. This demonstrates that maximizing coverage in view selection avoids CGP missing critical cues, resulting in higher accuracy.

Query Complexity Robustness. As illustrated in Figs. 8b and 8c, our method is robust when query complexity grows, only costing expected linear time.

Limitation. Our method is designed for static scenes and assumes spatially proximal relevant objects, which are common in current 3DVG. Additionally, CGP's performance depends on the segmentation model, which lies beyond the scope of our research which focuses on open-world 3DVG. These aspects are left for future exploration. More details about segmentation model influence and possible additional mechanism and error propagation discussion about task chain can be found in **Appendix**.

7 Conclusion

This paper presents OpenGround, a zero-shot framework for open-world 3D visual grounding. Unlike prior methods that either exhaustively complete the object lookup table ($\mathcal{OLT}$) or abandon it entirely, OpenGround introduces two modules: Task-Chain Planning that decomposes complex queries into ordered sub-goals for progressive grounding, and Context-Guided Perception that extends the $\mathcal{OLT}$ online to discover novel objects on demand. We also contribute the OpenTarget benchmark with 7,724 object-query pairs for open-world evaluation. Experiments show that OpenGround remains competitive on standard datasets (Nr3D, SceneRefer) and achieves state-of-the-art performance on OpenTarget, demonstrating its flexibility and capability in open-world scenarios.

Acknowledgments

This work was supported by NSFC under Grant No. 62376121, Basic Research Program of Jiangsu under Grant No. BK20251999, Gusu Innovation Leading Talent Program under Grant No. ZXL2025319, Jiangsu Provincial Science & Technology Major Project under Grant No. BG2024042, and funded by Nanjing University-China Mobile Communications Group Co.,Ltd. Joint Institute.

References

1. Achlioptas, P., Abdelreheem, A., Xia, F., Elhoseiny, M., Guibas, L.: Referit3d: Neural listeners for fine-grained 3d object identification in real-world scenes. In: European conference on computer vision. pp. 422–440. Springer (2020)
2. Bai, J., Bai, S., Yang, S., Wang, S., Tan, S., Wang, P., Lin, J., Zhou, C., Zhou, J.: Qwen-vl: A versatile vision-language model for understanding, localization, text reading, and beyond. arXiv preprint arXiv:2308.12966 (2023)
3. Bai, S., Chen, K., Liu, X., Wang, J., Ge, W., Song, S., Dang, K., Wang, P., Wang, S., Tang, J., Zhong, H., Zhu, Y., Yang, M., Li, Z., Wan, J., Wang, P., Ding, W., Fu, Z., Xu, Y., Ye, J., Zhang, X., Xie, T., Cheng, Z., Zhang, H., Yang, Z., Xu, H., Lin, J.: Qwen2.5-vl technical report. arXiv preprint arXiv:2502.13923 (2025)
4. Baruch, G., Chen, Z., Dehghan, A., Dimry, T., Feigin, Y., Fu, P., Gebauer, T., Joffe, B., Kurz, D., Schwartz, A., et al.: Arkitscenes: A diverse real-world dataset for 3d indoor scene understanding using mobile rgb-d data. arXiv preprint arXiv:2111.08897 (2021)
5. ByteDance: Volcengine (2025), `https://console.volcengine.com/`, accessed: 2025-02-18
6. Chen, D.Z., Chang, A.X., Nießner, M.: Scanrefer: 3d object localization in rgb-d scans using natural language. In: European conference on computer vision. pp. 202–221. Springer (2020)
7. Chen, R., Liu, Y., Kong, L., Zhu, X., Ma, Y., Li, Y., Hou, Y., Qiao, Y., Wang, W.: Clip2scene: Towards label-efficient 3d scene understanding by clip. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 7020–7030 (2023)
8. Chen, Y., Qi, Z., Zhang, W., Jin, X., Zhang, L., Liu, P.: Reasoning in space via grounding in the world (2025), `https://arxiv.org/abs/2510.13800`, accessed: 2026-06-26
9. Chen, Z., Wu, J., Wang, W., Su, W., Chen, G., Xing, S., Zhong, M., Zhang, Q., Zhu, X., Lu, L., et al.: Internvl: Scaling up vision foundation models and aligning for generic visual-linguistic tasks. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 24185–24198 (2024)
10. Dai, A., Chang, A.X., Savva, M., Halber, M., Funkhouser, T., Nießner, M.: Scannet: Richly-annotated 3d reconstructions of indoor scenes. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 5828–5839 (2017)
11. Guo, W., Xu, X., Wang, Z., Feng, J., Zhou, J., Lu, J.: Text-guided sparse voxel pruning for efficient 3d visual grounding. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 3666–3675 (2025)
12. Halacheva, A.M., Miao, Y., Zaech, J.N., Wang, X., Gool, L.V., Paudel, D.P.: Holistic understanding of 3d scenes as universal scene description. arXiv preprint arXiv:2412.01398 (2024)
13. Huang, C., Mees, O., Zeng, A., Burgard, W.: Visual language maps for robot navigation. arXiv preprint arXiv:2210.05714 (2022)
14. Huang, R., Yang, H., Cai, Y., Xu, X., Zhang, H., He, S.: Viewsrd: 3d visual grounding via structured multi-view decomposition. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 9726–9736 (2025)
15. Huang, S., Hou, Y., Liu, L., Yu, X., Shen, X.: Real-time object detection meets dinov3. arXiv (2025)
16. Huang, T., Zhang, Z., Tang, H.: 3d-r1: Enhancing reasoning in 3d vlms for unified scene understanding. arXiv preprint arXiv:2507.23478 (2025)

17. Huang, Z., Shangguan, Z., Zhang, J., Bar, G., Boyd, M., Ohn-Bar, E.: Assister: Assistive navigation via conditional instruction generation. In: European Conference on Computer Vision. pp. 271–289. Springer (2022)
18. Jain, A., Gkanatsios, N., Mediratta, I., Fragkiadaki, K.: Bottom up top down detection transformers for language grounding in images and point clouds. In: European Conference on Computer Vision. pp. 417–433. Springer (2022)
19. Jatavallabhula, K., Kuwajerwala, A., Gu, Q., Omama, M., Chen, T., Li, S., Iyer, G., Saryazdi, S., Keetha, N., Tewari, A., Tenenbaum, J., de Melo, C., Krishna, M., Paull, L., Shkurti, F., Torralba, A.: Conceptfusion: Open-set multimodal 3d mapping. Robotics: Science and Systems (RSS) (2023)
20. Jiang, L., Zhao, H., Shi, S., Liu, S., Fu, C.W., Jia, J.: Pointgroup: Dual-set point grouping for 3d instance segmentation. In: Proceedings of the IEEE/CVF conference on computer vision and Pattern recognition. pp. 4867–4876 (2020)
21. Jin, Z., Tu, R.C., Liao, J., Sun, W., Luo, X., Liu, S., Tao, D.: Spazer: Spatial-semantic progressive reasoning agent for zero-shot 3d visual grounding (2025), https://arxiv.org/abs/2506.21924, accessed: 2026-06-26
22. Kirillov, A., Mintun, E., Ravi, N., Mao, H., Rolland, C., Gustafson, L., Xiao, T., Whitehead, S., Berg, A.C., Lo, W.Y., Dollár, P., Girshick, R.: Segment anything. arXiv:2304.02643 (2023)
23. Kong, L., Liu, Y., Li, X., Chen, R., Zhang, W., Ren, J., Pan, L., Chen, K., Liu, Z.: Robo3d: Towards robust and reliable 3d perception against corruptions. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 19994–20006 (2023)
24. Li, J., Wang, H., Liu, Y., Dou, Z., Ma, Y., Yang, S., Li, Y., Wang, W., Dong, Z., Yang, B., et al.: Cityanchor: City-scale 3d visual grounding with multi-modality llms. In: The Thirteenth International Conference on Learning Representations
25. Li, R., Li, S., Kong, L., Yang, X., Liang, J.: Seeground: See and ground for zero-shot open-vocabulary 3d visual grounding. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2025)
26. Lin, J., Bian, S., Zhu, Y., Tan, W., Zhang, Y., Xie, Y., Qu, Y.: Seqvlm: Proposal-guided multi-view sequences reasoning via vlm for zero-shot 3d visual grounding. In: Proceedings of the 33rd ACM International Conference on Multimedia. p. 3094–3103. MM '25, Association for Computing Machinery, New York, NY, USA (2025). https://doi.org/10.1145/3746027.3754985, https://doi.org/10.1145/3746027.3754985, accessed: 2026-06-26
27. Lin, Z., He, S., Tan, C., Wen, B.: Groundflow: A plug-in module for temporal reasoning on 3d point cloud sequential grounding. arXiv preprint arXiv:2506.21188 (2025)
28. Liu, H., Li, C., Li, Y., Lee, Y.J.: Improved baselines with visual instruction tuning (2023)
29. Liu, H., Li, C., Li, Y., Li, B., Zhang, Y., Shen, S., Lee, Y.J.: Llava-next: Improved reasoning, ocr, and world knowledge (January 2024), https://llava-vl.github.io/blog/2024-01-30-llava-next/, accessed: 2026-06-26
30. Liu, H., Li, C., Wu, Q., Lee, Y.J.: Visual instruction tuning (2023)
31. Liu, S., Zeng, Z., Ren, T., Li, F., Zhang, H., Yang, J., Li, C., Yang, J., Su, H., Zhu, J., et al.: Grounding dino: Marrying dino with grounded pre-training for open-set object detection. arXiv preprint arXiv:2303.05499 (2023)
32. Liu, Y., Chen, W., Bai, Y., Liang, X., Li, G., Gao, W., Lin, L.: Aligning cyber space with physical world: A comprehensive survey on embodied ai. IEEE/ASME Transactions on Mechatronics (2025)

33. Liu, Z., Zhang, Z., Cao, Y., Hu, H., Tong, X.: Group-free 3d object detection via transformers. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 2949–2958 (2021)
34. Liu, Z., Wang, Y., Zheng, S., Pan, T., Liang, L., Fu, Y., Xue, X.: Reasongrounder: Lvlm-guided hierarchical feature splatting for open-vocabulary 3d visual grounding and reasoning. In: CVPR. pp. 3718–3727. Computer Vision Foundation / IEEE (2025), http://dblp.uni-trier.de/db/conf/cvpr/cvpr2025.html#Liu0ZPL0025, accessed: 2026-06-26
35. Ma, T., Li, R., Liang, J.: An examination of the compositionality of large generative vision-language models. arXiv preprint arXiv:2308.10509 (2023)
36. Nguyen, P., Ngo, T.D., Kalogerakis, E., Gan, C., Tran, A., Pham, C., Nguyen, K.: Open3dis: Open-vocabulary 3d instance segmentation with 2d mask guidance. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 4018–4028 (2024)
37. OpenAI: Gpt-4 technical report (2024), https://arxiv.org/abs/2303.08774, accessed: 2026-06-26
38. Peng, S., Genova, K., Jiang, C.M., Tagliasacchi, A., Pollefeys, M., Funkhouser, T.: Openscene: 3d scene understanding with open vocabularies. In: CVPR (2023)
39. Qi, Z., Zhang, Z., Fang, Y., Wang, J., Zhao, H.: Gpt4scene: Understand 3d scenes from videos with vision-language models. arXiv preprint arXiv:2501.01428 (2025)
40. Qian, Z., Ma, Y., Lin, Z., Ji, J., Zheng, X., Sun, X., Ji, R.: Multi-branch collaborative learning network for 3d visual grounding. In: European Conference on Computer Vision. pp. 381–398. Springer (2024)
41. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., Krueger, G., Sutskever, I.: Learning transferable visual models from natural language supervision (2021), https://arxiv.org/abs/2103.00020, accessed: 2026-06-26
42. Ravi, N., Gabeur, V., Hu, Y.T., Hu, R., Ryali, C., Ma, T., Khedr, H., Rädle, R., Rolland, C., Gustafson, L., Mintun, E., Pan, J., Alwala, K.V., Carion, N., Wu, C.Y., Girshick, R., Dollár, P., Feichtenhofer, C.: Sam 2: Segment anything in images and videos. arXiv preprint arXiv:2408.00714 (2024), https://arxiv.org/abs/2408.00714, accessed: 2026-06-26
43. Ren, T., Liu, S., Zeng, A., Lin, J., Li, K., Cao, H., Chen, J., Huang, X., Chen, Y., Yan, F., Zeng, Z., Zhang, H., Li, F., Yang, J., Li, H., Jiang, Q., Zhang, L.: Grounded sam: Assembling open-world models for diverse visual tasks (2024)
44. Schult, J., Engelmann, F., Hermans, A., Litany, O., Tang, S., Leibe, B.: Mask3d: Mask transformer for 3d semantic instance segmentation. arXiv preprint arXiv:2210.03105 (2022)
45. Shao, Z., Wang, P., Zhu, Q., Xu, R., Song, J., Zhang, M., Li, Y., Wu, Y., Guo, D.: Deepseekmath: Pushing the limits of mathematical reasoning in open language models (2024), https://arxiv.org/abs/2402.03300, accessed: 2026-06-26
46. Shi, J., Xiang, M., Sun, H., Huang, Y., Weng, Z.: Chain of semantics programming in 3d gaussian splatting representation for 3d vision grounding. In: CVPR. pp. 24560–24569. Computer Vision Foundation / IEEE (2025), http://dblp.uni-trier.de/db/conf/cvpr/cvpr2025.html#ShiXSHW25, accessed: 2026-06-26
47. Solmaz, S., Jorissen, L., Zogopoulos, V.: Scanverse: An extended reality authoring platform for industrial digital twins. Procedia CIRP **136**, 624–629 (2025)
48. Team, Q.: Qwen3 technical report (2025), https://arxiv.org/abs/2505.09388, accessed: 2026-06-26
49. Team, S.: Step-3 is large yet affordable: Model-system co-design for cost-effective decoding (2025), https://arxiv.org/abs/2507.19427, accessed: 2026-06-26

50. Team, V., Hong, W., Yu, W., Gu, X., Wang, G., Gan, G., Tang, H., Cheng, J., Qi, J., Ji, J., Pan, L., Duan, S., Wang, W., Wang, Y., Cheng, Y., He, Z., Su, Z., Yang, Z., Pan, Z., Zeng, A., Wang, B., Chen, B., Shi, B., Pang, C., Zhang, C., Yin, D., Yang, F., Chen, G., Xu, J., Zhu, J., Chen, J., Chen, J., Chen, J., Lin, J., Wang, J., Chen, J., Lei, L., Gong, L., Pan, L., Liu, M., Xu, M., Zhang, M., Zheng, Q., Yang, S., Zhong, S., Huang, S., Zhao, S., Xue, S., Tu, S., Meng, S., Zhang, T., Luo, T., Hao, T., Tong, T., Li, W., Jia, W., Liu, X., Zhang, X., Lyu, X., Fan, X., Huang, X., Wang, Y., Xue, Y., Wang, Y., Wang, Y., An, Y., Du, Y., Shi, Y., Huang, Y., Niu, Y., Wang, Y., Yue, Y., Li, Y., Zhang, Y., Wang, Y., Wang, Y., Zhang, Y., Xue, Z., Hou, Z., Du, Z., Wang, Z., Zhang, P., Liu, D., Xu, B., Li, J., Huang, M., Dong, Y., Tang, J.: Glm-4.5v and glm-4.1v-thinking: Towards versatile multimodal reasoning with scalable reinforcement learning (2025), `https://arxiv.org/abs/2507.01006`, accessed: 2026-06-26
51. Unal, O., Sakaridis, C., Saha, S., Van Gool, L.: Four ways to improve verbo-visual fusion for dense 3d visual grounding. In: European Conference on Computer Vision (ECCV) (October 2024)
52. Unal, O., Sakaridis, C., Saha, S., Van Gool, L.: Four ways to improve verbo-visual fusion for dense 3d visual grounding. In: European Conference on Computer Vision. pp. 196–213. Springer (2024)
53. Vu, T., Kim, K., Luu, T.M., Nguyen, T., Yoo, C.D.: Softgroup for 3d instance segmentation on point clouds. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 2708–2717 (2022)
54. Wang, P., Bai, S., Tan, S., Wang, S., Fan, Z., Bai, J., Chen, K., Liu, X., Wang, J., Ge, W., Fan, Y., Dang, K., Du, M., Ren, X., Men, R., Liu, D., Zhou, C., Zhou, J., Lin, J.: Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. arXiv preprint arXiv:2409.12191 (2024)
55. Wang, W., Gao, Z., Gu, L., Pu, H., Cui, L., Wei, X., Liu, Z., Jing, L., Ye, S., Shao, J., et al.: Internvl3.5: Advancing open-source multimodal models in versatility, reasoning, and efficiency. arXiv preprint arXiv:2508.18265 (2025)
56. Wang, X., Yang, X., Xu, Y., Wu, Y., Li, Z., Zhao, N.: Affordbot: 3d fine-grained embodied reasoning via multimodal large language models (2025), `https://arxiv.org/abs/2511.10017`, accessed: 2026-06-26
57. Wang, Y., Li, Y., Wang, S.: Gˆ 3-lq: Marrying hyperbolic alignment with explicit semantic-geometric modeling for 3d visual grounding. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 13917–13926 (2024)
58. Xu, R., Huang, Z., Wang, T., Chen, Y., Pang, J., Lin, D.: Vlm-grounder: A vlm agent for zero-shot 3d visual grounding. In: CoRL (2024)
59. Xu, R., Wang, X., Wang, T., Chen, Y., Pang, J., Lin, D.: Pointllm: Empowering large language models to understand point clouds. In: European Conference on Computer Vision. pp. 131–147. Springer (2024)
60. Yeshwanth, C., Liu, Y.C., Nießner, M., Dai, A.: Scannet++: A high-fidelity dataset of 3d indoor scenes. In: Proceedings of the International Conference on Computer Vision (ICCV) (2023)
61. Yuan, Z., Ren, J., Feng, C.M., Zhao, H., Cui, S., Li, Z.: Visual programming for zero-shot open-vocabulary 3d visual grounding. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 20623–20633 (2024)
62. Zantout, N., Zhang, H., Kachana, P., Qiu, J., Zhang, J., Wang, W.: Sort3d: Spatial object-centric reasoning toolbox for zero-shot 3d grounding using large language models. CoRR **abs/2504.18684** (April 2025), `http://dblp.uni-trier.de/db/journals/corr/corr2504.html#abs-2504-18684`, accessed: 2026-06-26

63. Zhang, Y., Gong, Z., Chang, A.X.: Multi3drefer: Grounding text description to multiple 3d objects. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 15225–15236 (2023)
64. Zhang, Y., Ma, Z., Li, J., Qiao, Y., Wang, Z., Chai, J., Wu, Q., Bansal, M., Kordjamshidi, P.: Vision-and-language navigation today and tomorrow: A survey in the era of foundation models. arXiv preprint arXiv:2407.07035 (2024)
65. Zhang, Y., Li, H., Gao, Y., Duan, H., Huang, Y., Zheng, Y.: Prototype correlation matching and class- relation reasoning for few-shot medical image segmentation. IEEE Transactions on Medical Imaging **43**(11), 4041–4054 (2024). https://doi.org/10.1109/TMI.2024.3412420
66. Zhao, L., Cai, D., Sheng, L., Xu, D.: 3dvg-transformer: Relation modeling for visual grounding on point clouds. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 2928–2937 (2021)
67. Zheng, D., Huang, S., Wang, L.: Video-3d llm: Learning position-aware video representation for 3d scene understanding. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 8995–9006 (2025)
68. Zheng, H., Shi, H., Peng, Q., Chng, Y.X., Huang, R., Weng, Y., Shi, Z., Huang, G.: Densegrounding: Improving dense language-vision semantics for ego-centric 3d visual grounding. arXiv preprint arXiv:2505.04965 (2025)
69. Zhou, Y., Gu, J., Chiang, T.Y., Xiang, F., Su, H.: Point-SAM: Promptable 3d segmentation model for point clouds. In: The Thirteenth International Conference on Learning Representations (2025), https://openreview.net/forum?id=yXCTDhZDh6, accessed: 2026-06-26
70. Zhu, C., Wang, T., Zhang, W., Pang, J., Liu, X.: Llava-3d: A simple yet effective pathway to empowering lmms with 3d-awareness. arXiv preprint arXiv:2409.18125 (2024)
71. Zhu, F., Wang, H., Xie, Y., Gu, J., Ding, T., Yang, J., Jiang, H.: Struct2d: A perception-guided framework for spatial reasoning in large multimodal models. CoRR **abs/2506.04220** (June 2025), http://dblp.uni-trier.de/db/journals/corr/corr2506.html#abs-2506-04220, accessed: 2026-06-26
72. Zhu, Z., Ma, X., Chen, Y., Deng, Z., Huang, S., Li, Q.: 3d-vista: Pre-trained transformer for 3d vision and text alignment. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 2911–2921 (2023)
73. Zhu, Z., Wang, X., Li, Y., Zhang, Z., Ma, X., Chen, Y., Jia, B., Liang, W., Yu, Q., Deng, Z., et al.: Move to understand a 3d scene: Bridging visual grounding and exploration for efficient and versatile embodied navigation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 8120–8132 (2025)