LET THE SCENE DECIDE: RETRIEVAL-AUGMENTED 3D REASONING SEGMENTATION

Anonymous authors
Paper under double-blind review

ABSTRACT

3D reasoning segmentation predicts a mask in a scanned scene from a query that describes an object's purpose without naming it. Existing systems learn this mapping end to end from annotated scans with narrow indoor vocabularies, and struggle when the target lies outside them. The bottleneck is often grounding rather than language: an LLM can infer the intended category, but the model has not learned its 3D appearance. We propose **Retrieval-Augmented 3D Reasoning Segmentation (RA3D)**, which supplies this missing visual knowledge at inference time. An LLM proposes candidate categories, retrieves images for each, and lifts them into point clouds. Each exemplar is placed into the query scene before encoding, reducing the gap between isolated objects and scene geometry. RA3D matches exemplars through background-contrastive similarity maps, ranks candidates with parameter-free distribution-calibrated scores, and independently verifies shortlisted masks against the retrieved images, which also resolves category selection. RA3D requires no training and can be integrated with existing models. We further introduce NO-Reason3D, a benchmark whose targets lie outside the indoor vocabularies used for training. RA3D improves trained models on Reason3D without retraining and outperforms all evaluated methods on NO-Reason3D by a wide margin. The code will be released upon acceptance.

1 INTRODUCTION

Understanding 3D point cloud scenes is a fundamental capability for embodied AI, robotics, and other spatially aware systems that must interact with complex real-world environments. Recent progress has been driven by integrating large language models (LLMs) with 3D backbones, enabling language-driven tasks such as captioning, question answering, and referring grounding (Hong et al., 2023; Xu et al., 2024; Chen et al., 2024a; Lei et al., 2023). Among these language-driven 3D tasks, reasoning segmentation remains particularly challenging (Huang et al., 2025; He et al., 2024). Unlike conventional segmentation, reasoning segmentation requires identifying and segmenting objects from queries that describe their functions or contextual relationships without explicitly naming them.

Existing methods typically learn the mapping between language and 3D scenes through end-to-end training. They either integrate an LLM with a point cloud encoder (Huang et al., 2025) or adapt a multimodal LLM to process 3D inputs (Huang et al., 2026), followed by fine-tuning on annotated scenes. Although effective for objects covered by the training data, these methods struggle to generalize to unseen categories. This limitation does not necessarily arise from language reasoning, as the LLM can often identify the intended object correctly. Instead, the model lacks the visual knowledge needed to associate the predicted category with its 3D appearance (Fig. 1(a)). Under the conventional training paradigm, acquiring such knowledge requires annotated 3D scans.

To address this limitation, we explore retrieval as an alternative source of visual knowledge. Images of diverse objects are readily available on the web and can provide the appearance information that existing 3D models lack (Fig. 1(b)). Motivated by this observation, we propose **Retrieval-Augmented 3D Reasoning Segmentation (RA3D)**. Given a natural language query, RA3D uses an LLM to generate multiple candidate object categories ranked by likelihood, and retrieves corresponding images for each category. To enable this cross-modal matching, both the retrieved exemplars and the query scene must be represented in a common domain. Rather than rendering the scene

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.

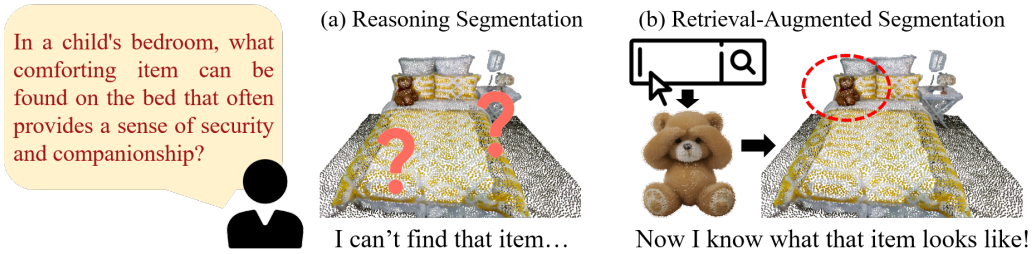

Figure 1: Motivation of our RA3D. (a) Previous methods struggle with unseen or long-tail objects. (b) Our RA3D retrieves images of the object from a search engine and finds the target.

into 2D views, we lift the retrieved images into 3D (Xiang et al., 2025; Li et al., 2025), allowing direct point cloud comparisons within the original scene representation.

Directly matching isolated exemplars with a complete scene introduces a representation gap. To address this, we place each exemplar into the query scene before encoding, guided by a coarse surface hint from the LLM. We then perform background-contrastive matching, identifying target regions by contrasting exemplar similarity with similarity to a scene-level background prototype. Candidate masks are obtained by thresholding each contrast map and clustering the remaining points with DBSCAN (Ester et al., 1996). Since these maps have different score distributions, we introduce distribution-calibrated scoring, which standardizes candidate scores using the mean and standard deviation of each map for consistent comparison across exemplars.

However, candidate generation and scoring rely on the same similarity map, allowing false candidates to receive high scores. To address this issue, we introduce an independent verification stage. Each shortlisted candidate is isolated from the 3D scene and rendered into 2D views, which are compared with the original retrieved images in a shared visual feature space. This provides additional evidence for verifying candidate masks independently of the scene-level matching scores. The verification scores also guide category selection, allowing a lower-ranked category to be selected when its candidate receives stronger verification support than those from higher-ranked categories.

The benefits of retrieval are most evident when the target object is unfamiliar to the model. However, existing 3D reasoning segmentation benchmarks contain objects from the indoor categories represented in their training datasets, making it difficult to evaluate generalization to unseen objects. To address this gap, we introduce Novel Object Reasoning 3D Segmentation (NO-Reason3D), a benchmark that inserts unseen ShapeNet objects (Chang et al., 2015) into real ScanNet (Dai et al., 2017) and Matterport3D (Chang et al., 2017) scenes and poses indirect language queries about them. Objects are placed in physically plausible configurations and processed to match the point-sampling characteristics of the surrounding scans, preventing models from identifying targets through local point statistics alone. We employ an LLM-assisted annotation pipeline (Anthropic, 2026) to generate indirect queries that describe target objects without explicitly naming their categories. Each sample undergoes human review to ensure that it uniquely identifies the intended target, while disjoint generation pools and ShapeNet instances prevent overlap between training and validation splits.

To the best of our knowledge, RA3D is the first retrieval-augmented framework for 3D reasoning segmentation. It operates entirely without training and can be integrated with existing methods. On Reason3D (Huang et al., 2025), routing RA3D with existing baseline models yields consistent improvements. On the proposed NO-Reason3D benchmark, it outperforms all evaluated methods by a wide margin. Extensive experiments further validate the contribution of each component, examine key design choices, and demonstrate consistent improvements using retrieved exemplars.

2 Related Work

2.1 3D Scene Understanding with LLM

3D scene understanding increasingly builds on large language models, raising a common question: how should a 3D scene be represented for an LLM? One line of work relies on 2D representations and uses camera geometry to connect them back to 3D. Some methods lift per-view features before language reasoning (Hong et al., 2023), while others lift predictions from a 2D model (Huang et al.,

2026; Chen et al., 2024b). Others retain 2D patch tokens and augment them with 3D positions derived from depth and camera parameters (Zhu et al., 2024; Zheng et al., 2025; Wang et al., 2025). A second line directly encodes 3D geometry, projecting point- or superpoint-level features from a 3D backbone into the LLM token space before alignment and instruction tuning (Xu et al., 2024; Chen et al., 2024a; Deng et al., 2025; He et al., 2024; Huang et al., 2023; 2024; Yu et al., 2025). Reasoning segmentation is especially challenging because its queries describe an object's purpose without naming it explicitly (Huang et al., 2025). Such query-mask pairs are expensive to annotate, leaving existing datasets small and confined to a limited set of indoor scans; recent work instead derives pseudo masks from 2D models (Huang et al., 2026). In either case, grounding remains tied to concepts represented in the training scans, leaving long-tail and unseen objects largely unexplored.

2.2 Retrieval-Augmented Segmentation

Retrieval lets models access query-relevant knowledge from non-parametric memory rather than storing it entirely in their weights (Lewis et al., 2020; Guu et al., 2020; Karpukhin et al., 2020; Izacard & Grave, 2021; Borgeaud et al., 2022; Asai et al., 2024). Vision models use the same idea (Long et al., 2022; Liu et al., 2023; Iscen et al., 2023; Udandarao et al., 2023; Kim et al., 2024). In segmentation, ReCo retrieves unlabelled images by concept and co-segments them for training (Shin et al., 2022). Other methods use retrieval only at inference. FOSSIL and FreeDA retrieve from offline banks built with diffusion-generated images (Barsellotti et al., 2024a;b). kNN-CLIP expands a segmenter's vocabulary with retrieved instance embeddings (Gui et al., 2024), while SegRAG turns feature-bank matches into point prompts (Boudiaf et al., 2026). These methods remain limited to images, leaving retrieval largely unexplored in 3D. We apply retrieval to 3D reasoning segmentation, where inference-time exemplars provide object knowledge absent from the trained model.

3 Method

Given a scene point cloud $\mathcal{P}$ and a natural-language query q that refers to a target object indirectly, 3D reasoning segmentation predicts a binary mask over $\mathcal{P}$. Following existing work (Huang et al., 2025), we over-segment the scene into superpoints $\{S_i\}_{i=1}^{N}$ with sizes $w_i = |S_i|$ and represent the predicted mask as a union of these superpoints. RA3D (Fig. 2) is training-free, with no task-specific fitting or weight updates. An LLM maps q to candidate categories, for which 3D exemplars are obtained on demand from web images lifted into point clouds (Sec. 3.1). Each exemplar is placed in the query scene (Sec. 3.2). It is then encoded with its surroundings, while the query scene is encoded separately using the same 3D encoder. The resulting exemplar and scene features are then matched and scored relative to the scene to produce candidate masks and a shortlist (Sec. 3.3). The shortlisted candidates are independently verified outside the scene to select the final mask (Sec. 3.4). RA3D can be used alone or routed with an existing model, as evaluated in Sec. 5.

3.1 Exemplar Acquisition

We prompt an LLM (Qwen et al., 2025) using only the query q and ask it to predict the three most likely object categories $\hat{c}_1, \hat{c}_2, \hat{c}_3$. Web retrieval provides images, whereas the query scene is represented as a point cloud, so the two must first be brought into a common modality. One option is to render the scene from multiple viewpoints and perform matching in the image domain, but this requires a separate comparison for each view and introduces an additional choice of viewpoints that can materially affect the result. Instead, we lift each retrieved image once into 3D so that all subsequent matching is performed directly between point clouds. For each predicted category $\hat{c}$, we retrieve $n = 10$ web images using a fixed search template, denoted by $\mathcal{I}(\hat{c}) = \{r_1, \ldots, r_n\}$. An off-the-shelf 2D-to-3D generator (Xiang et al., 2025) converts each image into a single-object point cloud $\mathcal{O}_e$, while the original photographs are retained for verification in Sec. 3.4. Because point clouds reconstructed from single images lack metric scale, we ask the same LLM for the typical real-world extent of each category and rescale its exemplars accordingly, providing a category-level size prior without measuring the target object. When annotated 3D scans are available, we instead use object instances from training scenes as exemplars. Their point clouds already have metric scale and therefore require no rescaling. We refer to these two exemplar sources as the 2D bank and 3D bank, according to the modality of their source data, and detail their construction in the Appendix.

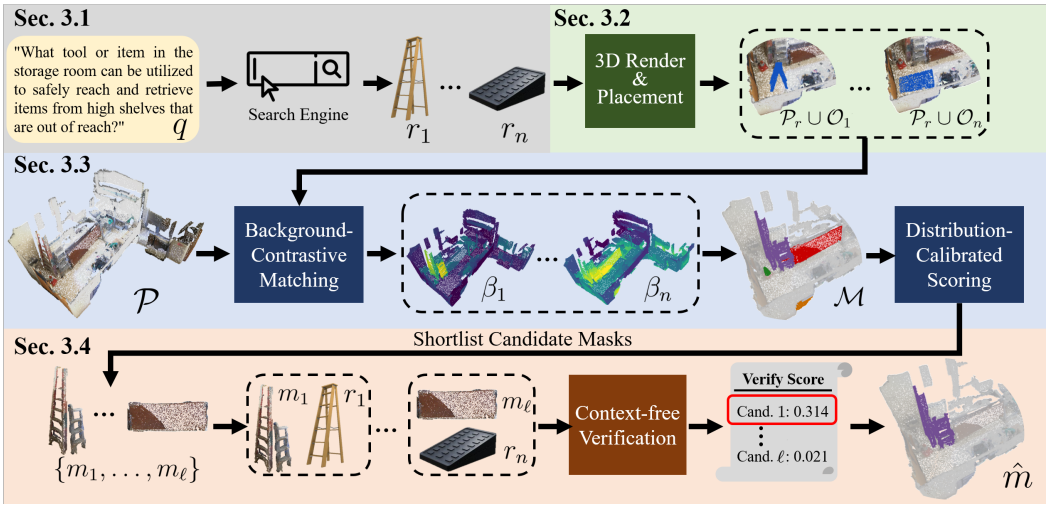

Figure 2: Overview of our RA3D. Images $r_1, \ldots, r_n$ retrieved for the categories an LLM predicts from the query q are lifted into point clouds by a 3D renderer and placed into the scene $\mathcal{P}_r$ as $\mathcal{O}_e$. Each exemplar yields a background-contrastive similarity map β_e, clustered into candidate masks $\mathcal{M}$ and ranked by the distribution-calibrated score. The shortlist $m_1, \ldots, m_\ell$ is verified outside the scene against the retrieved photographs, and the best match becomes the prediction $\hat{m}$.

3.2 In-Scene Placement and Contextual Encoding

Representing both sides as point clouds does not by itself make them directly comparable. A 3D encoder produces context-dependent features, so a scene object is represented together with its surroundings, whereas a retrieved exemplar is an isolated object in empty space. When matching the isolated exemplar to the scene, similarity can reflect this context mismatch as much as object identity. We reduce the mismatch by placing each exemplar into a scene before encoding it.

We obtain a surface hint from the same LLM call, asking whether the object typically rests on the floor, against a wall, or below the ceiling. For categories without a typical indoor position, we include an *unsure* class that falls back to floor placement. We place each exemplar in the query scene at a location consistent with the hint, using an open collision-free floor region, a detected vertical structure at mid-height, or the ceiling band. The remaining horizontal position is sampled randomly. We denote the placed exemplar by $\mathcal{O}_e$, now expressed in the scene coordinate frame.

To provide sufficient context without encoding the full scene for every exemplar, we jointly encode $\mathcal{O}_e$ with the scene points $\mathcal{P}_r$ within 1.5m of the placement point. Their union is subsampled to at most 6,000 points while preserving the scene-to-object ratio. The exemplar representation is obtained by averaging the features of the inserted points, $g_e = \frac{1}{|\mathcal{O}_e|} \sum_{p \in \mathcal{O}_e} \Phi_p(\mathcal{P}_r \cup \mathcal{O}_e)$, where Φ_p denotes the feature assigned to point p by the 3D encoder. This local encoding keeps the cost manageable when processing multiple exemplars per category. The query scene is encoded separately with the same encoder, and each superpoint feature is obtained by mean pooling, $f_i = \frac{1}{w_i} \sum_{p \in S_i} \Phi_p(\mathcal{P})$.

For the 3D bank, explicit placement is unnecessary because each exemplar already comes from a scanned scene and is encoded together with its original context. Both banks therefore follow the same principle of encoding exemplars within scene context, differing only in whether that context is borrowed from the query scene or inherited from the source scan.

3.3 Scene-Relative Matching and Scoring

3.3.1 Background-Contrastive Matching

Placement makes the exemplar and scene features comparable, but their raw cosine similarity still lacks a scene-specific reference. Similarity distributions vary across scenes, making a fixed threshold unreliable. We therefore use the scene itself as the reference. Since most scene points belong to recurring background structures rather than the target, their average feature provides a natural background prototype. We compute this average over superpoints weighted by their sizes, avoiding

the bias of giving small and large regions equal weight. Writing $\hat{x} = x/\|x\|$ for unit normalization, the scene prototype p and background similarity b_i of superpoint i are $p = \sum_i w_i \hat{f}_i / \left\|\sum_i w_i \hat{f}_i\right\|$ and $b_i = \hat{f}_i^\top p$. We then measure how much each superpoint resembles the exemplar relative to the scene background. For exemplar e, this defines the background-contrastive similarity map β_e,

$$\beta_e(i) = \mathrm{minmax}_i\big(\hat{f}_i^\top \hat{g}_e\big) - \mathrm{minmax}_i(b_i), \tag{1}$$

where minmax rescales each similarity vector over the scene to $[0, 1]$ before subtraction. Each exemplar produces its own contrast map and candidate masks. We threshold each β_e at the knee of its sorted profile, cluster the surviving superpoints with DBSCAN (Ester et al., 1996), and retain every cluster. This yields a candidate pool $\mathcal{M} = \{m_j\}_{j=1}^{|\mathcal{M}|}$ for the query.

3.3.2 Distribution-Calibrated Scoring

Candidate masks from different exemplars are not directly comparable because their similarity maps have different score distributions. We therefore calibrate each against its source map. This preserves the ranking of candidates from the same exemplar while making scores comparable across exemplars. Writing $e(m)$ for the exemplar producing mask m, and μ_e and σ_e for the mean and standard deviation of β_e over the scene, we define the size-weighted mean contrast and standardized score as

$$\mu_w(m) = \frac{\sum_{i \in m} w_i \, \beta_{e(m)}(i)}{\sum_{i \in m} w_i}, \qquad z(m) = \frac{\mu_w(m) - \mu_{e(m)}}{\sigma_{e(m)}}. \tag{2}$$

Candidates are ranked by z, which measures how far each candidate stands above its own contrast map. The weighted mean avoids directly rewarding larger masks. The resulting scores determine the shortlist and its width in the next stage.

3.4 Context-Free Verification

The scene-relative score is computed from the same contrast map that generates the candidates. It can therefore rank candidates within that map, but cannot determine whether the map itself is wrong. We therefore introduce an independent verification stage that removes the scene context and compares each candidate directly with the references retrieved for its category. For the 2D bank, a candidate is rendered from two views along its principal axes and compared with the photographs from which the exemplars were lifted. For the 3D bank, the candidate is isolated from the scene and compared with bank exemplars that are likewise isolated and encoded in the same way.

Not every candidate needs verification, since those with low scene-relative scores are unlikely to be correct. We therefore verify only candidates within a shortlist margin of the highest score z_1, $\{m : z(m) \geq z_1 - \delta_z\}$. We use a narrower margin when z_1 is high and a wider one when z_1 is low, so confident queries verify fewer candidates and uncertain queries verify more. Details are provided in Appendix. This yields a shortlist of candidates to be checked independently. Within each category, the shortlisted candidate with the highest reference similarity becomes its verified prediction,

$$\hat{m} = \arg \max_{m \in \text{shortlist}} \; \max_{r \in \mathcal{R}(\hat{c})} \cos\big(\Psi(m), \Psi(r)\big), \tag{3}$$

where $\mathcal{R}(\hat{c})$ contains the references for category $\hat{c}$. These are the retrieved photographs $\mathcal{I}(\hat{c})$ for the 2D bank and isolated encodings of $\{\mathcal{O}_e\}$ for the 3D bank. Both sides are embedded in the modality they share, so Ψ is a 2D image encoder (Siméoni et al., 2025) for the 2D bank and the same 3D encoder used in Sec. 3.2 for the 3D bank.

The verification scores also provide evidence about whether the predicted category itself is correct. We preserve the order proposed by the LLM and use its first category by default. For each considered category $\hat{c}_k$, let v_k denote the reference similarity of its best verified candidate. A lower-ranked category replaces the current choice only when it exceeds the best score by a category margin, $v_k > v_{\text{best}} + \delta_v$. This margin is important because each v_k is a maximum over a candidate pool, giving categories with more candidates more opportunities to obtain a high score by chance. The final mask is then selected from the winning category.

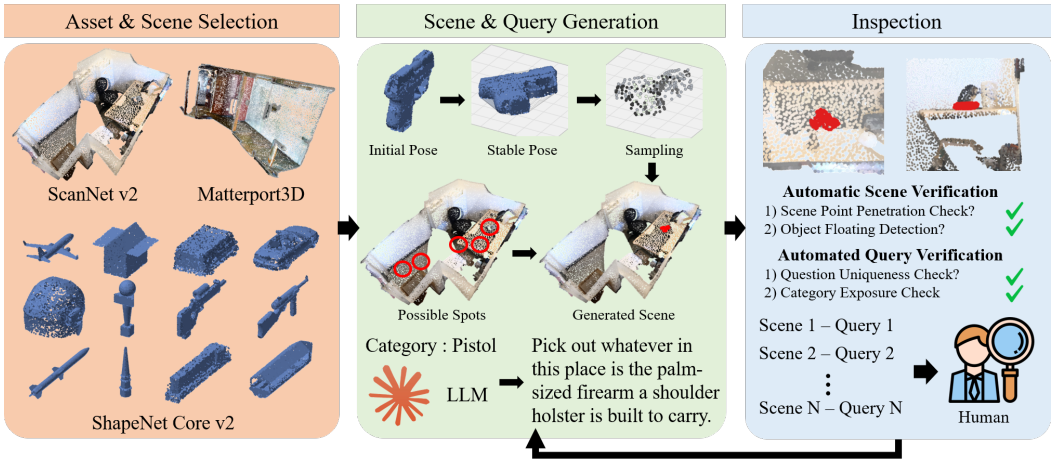

Figure 3: Generation process of NO-Reason3D. A CAD model from ShapeNet is brought to its most stable pose and sampled at the target scan's point density, then placed into a ScanNet or Matterport3D scene at a geometrically plausible position. A language model writes an indirect query for the category. Automatic checks cover scene penetration, floating objects, question uniqueness and category exposure, and a human verifies every query and scene before they are kept.

4 A Benchmark for Novel Object Reasoning 3D Segmentation

Retrieval is worth its cost only when the target is something the segmentation model has not learned. Existing 3D reasoning segmentation benchmarks do not test this setting. Their target objects are drawn from the same indoor vocabulary used for training, so a model that has learned this vocabulary can perform well without truly reasoning about the query. This also makes it difficult to distinguish the benefit of retrieval from that of a strong closed-set model. We therefore build NO-Reason3D, a benchmark for novel object reasoning segmentation whose targets lie outside the training vocabulary of existing indoor segmentation models. We use twelve ShapeNet categories as novel targets: airplane, birdhouse, bus, car, helmet, microphone, pistol, rifle, rocket, tower, train, and watercraft. We construct the benchmark by inserting CAD models from these categories (Chang et al., 2015) into real scans from ScanNet (Dai et al., 2017) and Matterport3D (Chang et al., 2017), and pair them with indirect queries that describe the target without naming it (Fig. 3).

Each CAD model is brought to a physically plausible state. We compute its stable resting poses and use the most stable one, avoiding unnatural configurations such as balancing on a handle or barrel, and rescale it to a measured typical size for its category. We then convert the mesh into points to match the sampling characteristics of the scene. The mesh is decimated to the scene's point spacing, vertices are used as points, and the scene dataset's own segmentation routine is applied to obtain superpoints. Matching the surrounding scan prevents the inserted object from being distinguished by local point statistics, helping ensure that it must instead be identified from the query.

Placement is constrained by scene geometry. We reject positions where scene points intersect the object volume, using a contact band scaled by the object's height so that flat and tall objects are tested consistently. We also require the support beneath the object to be complete, evaluated at each object point rather than on a coarse grid, which rules out placements over holes and scan boundaries. The object footprint must be sufficiently supported without overhanging an edge, and the dominant semantics beneath it must correspond to the floor or another plausible support surface. Each scene contains up to three novel categories, with one to three instances per category placed at distinct locations. For each query, the union of all instances of its target category defines the ground-truth mask, following the convention of the real benchmarks used for comparison.

Queries are generated with an LLM (Claude (Anthropic, 2026)), which is given a category and asked to describe it indirectly through its function, typical use, or appearance without naming it. To prevent the query text itself from becoming an identifier of the target, the clue and frame pools used for generation are disjoint between the training and validation splits. This ensures that no question is repeated across splits and that no category name appears in any query. Each query is then reviewed by a human annotator, who verifies that it uniquely identifies the intended object and discards it otherwise. NO-Reason3D contains 3,322 training and 980 validation queries, with CAD instances also disjoint between the splits.

Table 1: Comparison between RA3D with 2D bank and prior work on Reason3D and NO-Reason3D. RA3D with 2D bank evaluates the retrieval branch on its own, while a row of the form baseline plus RA3D reports the routed setting that combines that baseline with ours.

Method	Reason3D: ScanNet			Reason3D: Matterport3D		
	Acc@0.25	Acc@0.50	mIoU	Acc@0.25	Acc@0.50	mIoU
3D-STMN (Wu et al., 2024)	25.43	17.78	18.23	20.68	10.81	13.47
Reason3D (Huang et al., 2025)	43.21	32.10	31.20	31.22	17.43	19.54
MLLM-For3D (Huang et al., 2026)	44.18	34.80	32.90	33.41	27.99	22.50
RA3D with 2D bank	30.63±1.00	20.89±1.10	20.93±0.60	24.49±0.29	14.54±0.31	15.77±0.02
3D-STMN + RA3D (2D)	36.90±0.67	26.62±0.53	26.81±0.44	30.19±0.46	17.44±0.10	19.43±0.18
Reason3D + RA3D (2D)	47.08±0.27	34.52±0.15	34.12±0.16	34.93±0.11	22.62±0.06	23.22±0.11

Method	NO-Reason3D: ScanNet			NO-Reason3D: Matterport3D		
	Acc@0.25	Acc@0.50	mIoU	Acc@0.25	Acc@0.50	mIoU
3D-STMN (Wu et al., 2024)	1.77	0.16	1.17	3.06	1.11	1.71
Reason3D (Huang et al., 2025)	6.76	2.74	4.19	8.36	4.46	5.21
RA3D with 2D bank	19.97±0.92	12.56±0.73	14.50±0.69	28.88±1.74	15.60±0.99	20.08±1.24

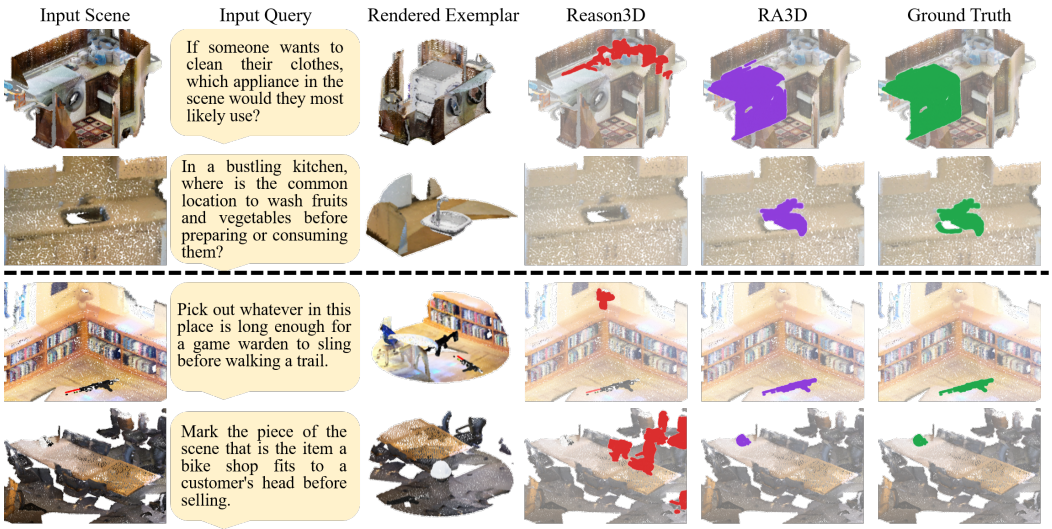

Figure 4: Qualitative results. Each row shows the input scene, the query, selected exemplar rendered from the 2D bank, the prediction of the base model, our prediction and the ground truth. Rows above the dashed line come from Reason3D and rows below from NO-Reason3D.

5 Experiments

5.1 Settings

Datasets and Metrics. We evaluate RA3D on Reason3D (Huang et al., 2025) and our proposed NO-Reason3D benchmark, both built from ScanNet v2 (Dai et al., 2017) and Matterport3D (Chang et al., 2017) scenes. We report mIoU and accuracy at IoU thresholds of 0.25 and 0.5.

Implementation Details. Qwen2.5-7B (Qwen et al., 2025) provides categories and surface hints. The 2D bank is collected from Google Image Search and lifted to point cloud with TRELLIS (Xiang et al., 2025), while the 3D bank uses the training split of each benchmark. Scenes and exemplars are encoded by Concerto (Zhang et al., 2026a). Each placed exemplar is encoded with scene points within 1.5m and subsampled to 6,000 points. Contrast maps are thresholded at the knee and clustered with DBSCAN (Ester et al., 1996) using $\epsilon = 0.10$m. We set $\delta_z = 0.05$ when $z_1 \geq 4$ and 0.75 otherwise, and set $\delta_v = 0.15$. Routing uses Reason3D (Huang et al., 2025) and 3D-STMN (Wu et al., 2024). Further details are provided in the appendix.

5.2 Performance Evaluation

Experiments with 2D bank. Table 1 reports results using a 2D bank built entirely from web images, with no exemplars taken from the target scans. Retrieval alone outperforms the benchmark-trained 3D-STMN on both ScanNet and Matterport3D. Routing retrieval with trained models further im-

Table 2: Comparison between RA3D with 2D and 3D banks on Reason3D and NO-Reason3D.

Method	Reason3D: ScanNet			Reason3D: Matterport3D		
	Acc@0.25	Acc@0.50	mIoU	Acc@0.25	Acc@0.50	mIoU
RA3D with 2D bank	30.63±1.00	20.89±1.10	20.93±0.60	24.49±0.29	14.54±0.31	15.77±0.02
3D-STMN + RA3D (2D)	36.90±0.67	26.62±0.53	26.81±0.44	30.19±0.46	17.44±0.10	19.43±0.18
Reason3D + RA3D (2D)	47.08±0.27	34.52±0.15	34.12±0.16	34.93±0.11	22.62±0.06	23.22±0.11
RA3D with 3D bank	45.89±1.31	29.65±1.14	30.96±0.65	35.84±0.12	19.04±0.50	21.38±0.05
3D-STMN + RA3D (3D)	45.45±0.70	32.25±0.15	32.57±0.29	33.81±0.10	18.92±0.15	21.37±0.03
Reason3D + RA3D (3D)	51.19±0.40	37.34±0.27	36.92±0.17	38.47±0.10	24.49±0.10	25.16±0.05
Method	NO-Reason3D: ScanNet			NO-Reason3D: Matterport3D		
	Acc@0.25	Acc@0.50	mIoU	Acc@0.25	Acc@0.50	mIoU
RA3D with 2D bank	19.97±0.92	12.56±0.73	14.50±0.69	28.88±1.74	15.60±0.99	20.08±1.24
RA3D with 3D bank	28.72±0.27	17.07±0.35	20.24±0.31	39.37±1.03	23.31±0.73	29.15±0.76

Table 3: Contribution of each component on Reason3D and NO-Reason3D datasets. All experiments use the retrieval branch of RA3D with the 2D bank.

Exp.	Placement	Scene-rel.	Verification	Reason3D			NO-Reason3D		
				Acc@0.25	Acc@0.50	mIoU	Acc@0.25	Acc@0.50	mIoU
A				7.51	3.93	5.08	4.80	2.14	3.32
B	✓			17.55	9.78	10.98	12.76	6.33	9.57
C		✓		13.19	7.95	8.58	16.22	8.27	11.26
D	✓	✓		22.01	14.24	14.59	23.88	13.98	17.02
E	✓	✓	✓	26.29	16.77	17.29	24.80	14.69	17.67

proves performance, giving 3D-STMN gains of 8.58 and 5.96 mIoU and Reason3D gains of 2.92 and 3.68. These gains suggest that retrieval resolves many queries missed by the trained models. MLLM-For3D (Huang et al., 2026) also uses the corresponding images and thus falls outside our point-cloud-only setting. Despite this, Reason3D with RA3D remains competitive on both ScanNet and Matterport3D. Since its checkpoint is unavailable, we cannot evaluate routing with RA3D, which may provide additional gains. The advantage of retrieval becomes much larger on NO-Reason3D, where target categories are absent from the training set. Both baselines drop sharply, with Reason3D around 4–5 mIoU and 3D-STMN below 2, while the 2D bank reaches 14.50 and 20.08. The pipeline is unchanged across benchmarks, with only the categories retrieved varying.

Qualitative Results. Figure 4 shows four queries together with the exemplar used for prediction. On the two Reason3D examples, the base model either spreads over unrelated surfaces or returns no mask, while RA3D correctly localizes the appliance and sink. The NO-Reason3D examples expose the limitation more clearly. The base model falls back on familiar furniture, whereas the target is an inserted object outside its training vocabulary. RA3D identifies the correct target in all four cases, with the remaining errors confined mainly to object boundaries.

Experiments with 3D bank. Table 2 compares the 2D and 3D banks. The 3D bank consistently outperforms the 2D bank across NO-Reason3D and Reason3D, bringing retrieval alone to the level of Reason3D on ScanNet and above it on Matterport3D. Since only the exemplar source changes, this gain mainly reflects the mismatch between generated and scanned geometry rather than differences in matching or selection. The 3D bank therefore provides an idealized case of retrieval with scan-matched geometry. Improvements in single-image reconstruction may transfer directly to the 2D bank, since the pipeline is not tied to a particular generator.

5.3 Ablation Study

Component Analysis. Table 3 evaluates the contribution of each component in the 2D bank pipeline. Placement and scene-relative reading each substantially improve performance. Placement contributes more on Reason3D, where targets often appear among geometrically similar indoor objects, while scene-relative reading contributes more on NO-Reason3D, where unfamiliar targets make raw similarity less reliably calibrated. Their gains are roughly additive, suggesting that they address complementary problems rather than correcting the same failure. Verification provides a further improvement by refining candidate selection without changing the candidate pool.

2D-to-3D Generator Analysis. Table 4 compares TRELLIS (Xiang et al., 2025), TRELLIS2 (Xiang et al., 2026), and LaRI (Li et al., 2025) as the 3D renderer for the 2D bank while keeping the rest

Table 4: Ablation study of the 2D-to-3D generator method. (Acc@0.25/Acc@0.5/mIoU)

3D Generator	Reason3D	NO-Reason3D
TRELLIS	26.29/16.77/17.29	24.80/14.69/17.67
TRELLIS2	23.32/14.50/15.28	23.06/12.45/16.06
LaRI	20.00/11.97/13.14	20.10/11.12/14.13

Table 5: Ablation study of the 3D encoder architecture. (Acc@0.25/Acc@0.5/mIoU)

3D Encoder	Reason3D	NO-Reason3D
Concerto large	26.29/16.77/17.29	24.80/14.69/17.67
Concerto base	22.88/13.28/14.94	24.59/14.39/17.63
Utonia	26.29/16.24/16.73	19.80/10.82/13.48

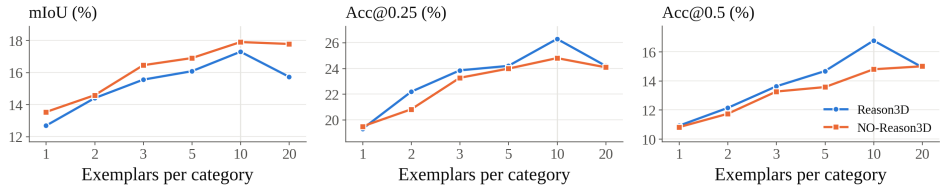

Figure 5: Effect of the number of retrieved exemplars per category. All metrics improve up to ten exemplars and flatten or fall back beyond it. We choose ten exemplars as default setting.

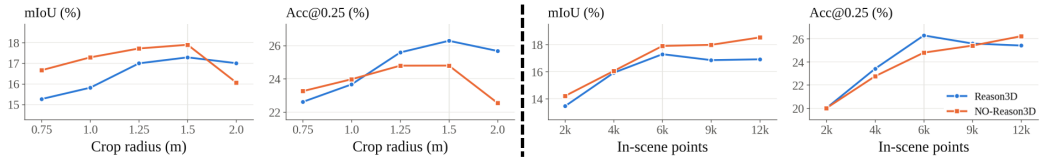

Figure 6: Analysis on the in-scene encoding budget of a placed exemplar. The left varies the crop radius and the right varies the number of sampling points.

of the pipeline fixed. Performance varies moderately across the three choices rather than changing dramatically, indicating that RA3D is not strongly tied to a particular renderer. TRELLIS performs best overall on both benchmarks, so we use it as the default renderer in all other experiments.

3D Encoder Analysis. Table 5 compares several encoder choices. Concerto large consistently outperforms Concerto base, with a larger gain on Reason3D than on NO-Reason3D. A different self-supervised point transformer (Zhang et al., 2026b) approaches the largest Concerto on Reason3D but falls well behind on NO-Reason3D. Encoders that perform similarly on familiar indoor objects therefore diverge once the target is novel, precisely the setting where retrieval is most needed.

5.4 Analysis on Exemplar Budget

Number of Exemplars per Category. Figure 5 varies the number of exemplars per category. Performance improves sharply from one to three exemplars and continues increasing up to ten, after which it saturates or declines. More exemplars improve coverage of category shape variation but also introduce additional candidates that match the category without matching the target. Beyond ten, this growing ambiguity offsets the benefit of broader coverage.

In-Scene Encoding Budget. Figure 6 varies the crop radius around a placed exemplar and the number of points used for encoding. Increasing the radius improves performance up to 1.5m on both benchmarks, but expanding it to 2.0m hurts most metrics, especially on NO-Reason3D. This suggests that the exemplar benefits from nearby scene context, while too much surrounding geometry dilutes its representation. We subsample the crop and exemplar together at a fixed ratio, so the point budget controls the resolution of both. Performance improves substantially from 2,000 to 6,000 points. Beyond 6,000, performance either slightly declines or improves only modestly. We therefore use a 1.5m crop with 6,000 points as a balance between performance and encoding cost.

6 Conclusion

RA3D is a training-free pipeline for 3D reasoning segmentation that retrieves exemplars for categories inferred from the query and matches them against the scene. Its design addresses the representation gap between isolated exemplars and objects embedded in scene context. RA3D places each exemplar into the query scene before encoding, performs scene-relative matching and scoring, and verifies shortlisted candidates independently outside the scene. We also introduce NO-Reason3D to evaluate segmentation of unfamiliar objects outside the training vocabulary of existing indoor models. Together, these results show that retrieving external object knowledge at inference time offers a practical alternative to acquiring it through additional annotated 3D data.

AI USE STATEMENT

In this work, we used generative AI tools to generate synthetic data sets, namely the natural language queries of NO-Reason3D described in Section 4. We have not used generative AI tools for hypothesis refinement, methodology design, implementation, data cleaning or the interpretation of results, and the development of theoretical models or conceptual frameworks, the formulation of mathematical claims and their proofs, and translation are not applicable to this work. Additionally, we used generative AI tools for drafting and editing the text of this paper. We have reviewed all AI-assisted work. Every generated query was read by an author who confirmed that it identifies the intended object and admits no other reading, and discarded it otherwise, and every claim, number and citation in the text was checked by the authors against our own results and the cited sources. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

Section 3 gives every stage of the method together with the quantities it computes, Section 5.1 gives the models, the composition of each bank and the values we use, and the appendix gives the remaining settings together with the margin rule and the derivation of its threshold. Section 4 describes how NO-Reason3D is built, from the pose and sampling of each inserted object to the geometric checks and the query generation. Reported numbers are averaged over repeated inference runs whose only variation is exemplar placement, subsampling and encoder nondeterminism, and the spread of those runs is given alongside each number. Upon acceptance we will release the code of the pipeline, the code that generates the benchmark, and the annotations of NO-Reason3D.

REFERENCES

Anthropic. Claude opus 5 system card. Technical report, Anthropic PBC, 2026. https://www.anthropic.com/claude-opus-5-system-card.

Akari Asai, Zeqiu Wu, Yizhong Wang, Avi Sil, and Hannaneh Hajishirzi. Self-rag: Learning to retrieve, generate, and critique through self-reflection. In *International conference on learning representations*, volume 2024, pp. 9112–9141, 2024.

Luca Barsellotti, Roberto Amoroso, Lorenzo Baraldi, and Rita Cucchiara. Fossil: Free open-vocabulary semantic segmentation through synthetic references retrieval. In *2024 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pp. 1453–1462. IEEE, 2024a.

Luca Barsellotti, Roberto Amoroso, Marcella Cornia, Lorenzo Baraldi, and Rita Cucchiara. Training-free open-vocabulary segmentation with offline diffusion-augmented prototype generation. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3689–3698. IEEE, 2024b.

Sebastian Borgeaud, Arthur Mensch, Jordan Hoffmann, Trevor Cai, Eliza Rutherford, Katie Millican, George Bm Van Den Driessche, Jean-Baptiste Lespiau, Bogdan Damoc, Aidan Clark, et al. Improving language models by retrieving from trillions of tokens. In *International conference on machine learning*, pp. 2206–2240. PMLR, 2022.

Abderrahmene Boudiaf, Irfan Hussain, and Sajid Javed. Segrag: Training-free retrieval-augmented semantic segmentation. *arXiv preprint arXiv:2605.17630*, 2026.

Angel Chang, Angela Dai, Thomas Funkhouser, Maciej Halber, Matthias Niessner, Manolis Savva, Shuran Song, Andy Zeng, and Yinda Zhang. Matterport3d: Learning from rgb-d data in indoor environments. *arXiv preprint arXiv:1709.06158*, 2017.

Angel X Chang, Thomas Funkhouser, Leonidas Guibas, Pat Hanrahan, Qixing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, et al. Shapenet: An information-rich 3d model repository. *arXiv preprint arXiv:1512.03012*, 2015.

Sijin Chen, Xin Chen, Chi Zhang, Mingsheng Li, Gang Yu, Hao Fei, Hongyuan Zhu, Jiayuan Fan, and Tao Chen. Ll3da: Visual interactive instruction tuning for omni-3d understanding, reasoning, and planning. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 26418–26428. IEEE, 2024a.

Tianrun Chen, Chunan Yu, Jing Li, Jianqi Zhang, Lanyun Zhu, Deyi Ji, Yong Zhang, Ying Zang, Zejian Li, and Lingyun Sun. Reasoning3d–grounding and reasoning in 3d: Fine-grained zero-shot open-vocabulary 3d reasoning part segmentation via large vision-language models. *arXiv preprint arXiv:2405.19326*, 2024b.

Angela Dai, Angel X Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *2017 IEEE conference on computer vision and pattern recognition (CVPR)*, pp. 2432–2443. IEEE, 2017.

Jiajun Deng, Tianyu He, Li Jiang, Tianyu Wang, Feras Dayoub, and Ian Reid. 3d-llava: Towards generalist 3d lmms with omni superpoint transformer. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3772–3782. IEEE, 2025.

Martin Ester, Hans-Peter Kriegel, Jörg Sander, Xiaowei Xu, et al. A density-based algorithm for discovering clusters in large spatial databases with noise. In *kdd*, volume 96, pp. 226–231, 1996.

Zhongrui Gui, Shuyang Sun, Runjia Li, Jianhao Yuan, Zhaochong An, Karsten Roth, Ameya Prabhu, and Philip Torr. knn-clip: Retrieval enables training-free segmentation on continually expanding large vocabularies. *arXiv preprint arXiv:2404.09447*, 2024.

Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Mingwei Chang. Retrieval augmented language model pre-training. In *International conference on machine learning*, pp. 3929–3938. PMLR, 2020.

Shuting He, Henghui Ding, Xudong Jiang, and Bihan Wen. Segpoint: Segment any point cloud via large language model. In *European Conference on Computer Vision*, pp. 349–367. Springer, 2024.

Yining Hong, Haoyu Zhen, Peihao Chen, Shuhong Zheng, Yilun Du, Zhenfang Chen, and Chuang Gan. 3d-llm: Injecting the 3d world into large language models. *Advances in Neural Information Processing Systems*, 36:20482–20494, 2023.

Haifeng Huang, Yilun Chen, Zehan Wang, Rongjie Huang, Runsen Xu, Tai Wang, Luping Liu, Xize Cheng, Yang Zhao, Jiangmiao Pang, et al. Chat-scene: Bridging 3d scene and large language models with object identifiers. *Advances in Neural Information Processing Systems*, 37:113991–114017, 2024.

Jiangyong Huang, Silong Yong, Xiaojian Ma, Xiongkun Linghu, Puhao Li, Yan Wang, Qing Li, Song-Chun Zhu, Baoxiong Jia, and Siyuan Huang. An embodied generalist agent in 3d world. *arXiv preprint arXiv:2311.12871*, 2023.

Jiaxin Huang, Runnan Chen, Ziwen Li, Zhengqing Gao, Xiao He, Yandong Guo, Mingming Gong, and Tongliang Liu. Mllm-for3d: Adapting multimodal large language model for 3d reasoning segmentation. *Advances in Neural Information Processing Systems*, 38:54961–54984, 2026.

Kuan-Chih Huang, Xiangtai Li, Lu Qi, Shuicheng Yan, and Ming-Hsuan Yang. Reason3d: Searching and reasoning 3d segmentation via large language model. In *2025 International Conference on 3D Vision (3DV)*, pp. 1177–1186. IEEE, 2025.

Ahmet Iscen, Alireza Fathi, and Cordelia Schmid. Improving image recognition by retrieving from web-scale image-text data. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 19295–19304. IEEE, 2023.

Gautier Izacard and Edouard Grave. Leveraging passage retrieval with generative models for open domain question answering. In *Proceedings of the 16th conference of the european chapter of the association for computational linguistics: main volume*, pp. 874–880, 2021.

Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. Dense passage retrieval for open-domain question answering. In *Proceedings of the 2020 conference on empirical methods in natural language processing (EMNLP)*, pp. 6769–6781, 2020.

Jooyeon Kim, Eulrang Cho, Sehyung Kim, and Hyunwoo J Kim. Retrieval-augmented open-vocabulary object detection. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 17427–17436. IEEE, 2024.

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 *2023 IEEE/CVF international conference on computer vision (ICCV)*, pp. 3992–4003. IEEE, 2023.

Yinjie Lei, Zixuan Wang, Feng Chen, Guoqing Wang, Peng Wang, and Yang Yang. Recent advances in multi-modal 3d intelligence: A comprehensive survey and evaluation. *arXiv preprint arXiv:2310.15676*, 2023.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in neural information processing systems*, 33: 9459–9474, 2020.

Rui Li, Biao Zhang, Zhenyu Li, Federico Tombari, and Peter Wonka. Lari: Layered ray intersections for single-view 3d geometric reasoning. *arXiv preprint arXiv:2504.18424*, 2025.

Haotian Liu, Kilho Son, Jianwei Yang, Ce Liu, Jianfeng Gao, Yong Jae Lee, and Chunyuan Li. Learning customized visual models with retrieval-augmented knowledge. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 15148–15158. IEEE, 2023.

Alexander Long, Wei Yin, Thalaiyasingam Ajanthan, Vu Nguyen, Pulak Purkait, Ravi Garg, Alan Blair, Chunhua Shen, and Anton Van den Hengel. Retrieval augmented classification for long-tail visual recognition. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6949–6959. IEEE, 2022.

Qwen, :, An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, Keqin Bao, Kexin Yang, Le Yu, Mei Li, Mingfeng Xue, Pei Zhang, Qin Zhu, Rui Men, Runji Lin, Tianhao Li, Tianyi Tang, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yu Wan, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zihan Qiu. Qwen2.5 technical report, 2025. URL `https://arxiv.org/abs/2412.15115`.

Gyungin Shin, Weidi Xie, and Samuel Albanie. Reco: Retrieve and co-segment for zero-shot transfer. *Advances in Neural Information Processing Systems*, 35:33754–33767, 2022.

Oriane Siméoni, Huy V Vo, Maximilian Seitzer, Federico Baldassarre, Maxime Oquab, Cijo Jose, Vasil Khalidov, Marc Szafraniec, Seungeun Yi, Michaël Ramamonjisoa, et al. Dinov3. *arXiv preprint arXiv:2508.10104*, 2025.

Vishaal Udandarao, Ankush Gupta, and Samuel Albanie. Sus-x: Training-free name-only transfer of vision-language models. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 2725–2736. IEEE, 2023.

Haochen Wang, Yucheng Zhao, Tiancai Wang, Haoqiang Fan, Xiangyu Zhang, and Zhaoxiang Zhang. Ross3d: Reconstructive visual instruction tuning with 3d-awareness. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 9275–9286. IEEE, 2025.

Changli Wu, Yiwei Ma, Qi Chen, Haowei Wang, Gen Luo, Jiayi Ji, and Xiaoshuai Sun. 3d-stmn: Dependency-driven superpoint-text matching network for end-to-end 3d referring expression segmentation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 5940–5948, 2024.

Jianfeng Xiang, Zelong Lv, Sicheng Xu, Yu Deng, Ruicheng Wang, Bowen Zhang, Dong Chen, Xin Tong, and Jiaolong Yang. Structured 3d latents for scalable and versatile 3d generation. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 21469–21480. IEEE, 2025.

Jianfeng Xiang, Xiaoxue Chen, Sicheng Xu, Ruicheng Wang, Zelong Lv, Yu Deng, Hongyuan Zhu, Yue Dong, Hao Zhao, Nicholas Jing Yuan, et al. Native and compact structured latents for 3d generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 14419–14429, 2026.

Runsen Xu, Xiaolong Wang, Tai Wang, Yilun Chen, Jiangmiao Pang, and Dahua Lin. Pointllm: Empowering large language models to understand point clouds. In *European Conference on Computer Vision*, pp. 131–147. Springer, 2024.

Chandan Yeshwanth, Yueh-Cheng Liu, Matthias Nießner, and Angela Dai. Scannet++: A high-fidelity dataset of 3d indoor scenes. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 12–22. IEEE, 2023.

Hanxun Yu, Wentong Li, Song Wang, Junbo Chen, and Jianke Zhu. Inst3d-lmm: Instance-aware 3d scene understanding with multi-modal instruction tuning. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 14147–14157. IEEE, 2025.

Yujia Zhang, Xiaoyang Wu, Yixing Lao, Chengyao Wang, Zhuotao Tian, Naiyan Wang, and Hengshuang Zhao. Concerto: Joint 2d-3d self-supervised learning emerges spatial representations. *Advances in Neural Information Processing Systems*, 38:69498–69522, 2026a.

Yujia Zhang, Xiaoyang Wu, Yunhan Yang, Xianzhe Fan, Han Li, Yuechen Zhang, Zehao Huang, Naiyan Wang, and Hengshuang Zhao. Utonia: Toward one encoder for all point clouds. *arXiv preprint arXiv:2603.03283*, 2026b.

Duo Zheng, Shijia Huang, and Liwei Wang. Video-3d llm: Learning position-aware video representation for 3d scene understanding. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8995–9006. IEEE, 2025.

Chenming Zhu, Tai Wang, Wenwei Zhang, Jiangmiao Pang, and Xihui Liu. Llava-3d: A simple yet effective pathway to empowering lmms with 3d-awareness. *arXiv preprint arXiv:2409.18125*, 2024.

A Implementation Details

A.1 Models and exemplar banks

Categories and surface hints are predicted by Qwen2.5-7B (Qwen et al., 2025) using greedy decoding, making the output deterministic for a given query. A single call jointly returns a ranked list of candidate categories and a surface hint. The template for web search is "photo of an indoor {category} isolated on white background, entire object fully visible, single object". We retain the three highest-ranked categories. For each category, a second call estimates its typical real-world extent, which is used to rescale lifted exemplars. This call is made once per category rather than once per query.

The 2D bank is constructed by issuing one image search for each predicted category through a commercial Google Image Search API. We retain ten images per category and reconstruct each image with TRELLIS (Xiang et al., 2025) using twelve sampling steps. The bank vocabulary is the union of all categories predicted from the queries of the benchmark under evaluation. Thus, bank construction has access to the categories inferred from the queries, but never to the ground-truth target object or mask for any query. No ground-truth annotation is used at any stage of bank construction. For Reason3D (Huang et al., 2025), the resulting bank contains $1{,}250$ categories and $12{,}494$ exemplars. NO-Reason3D introduces an additional 602 categories beyond this vocabulary.

The 3D bank is constructed from the training split of the benchmark under evaluation. For Reason3D, we use annotated object instances from the training scenes. Each instance is encoded in its original scene context and mean-pooled to form an exemplar representation, yielding 944 categories and $1{,}334$ exemplars. For NO-Reason3D, the bank consists of the objects inserted into the training split, yielding 12 categories and $3{,}322$ exemplars.

A.2 Encoding and Verification

Scenes and placed exemplars are encoded with Concerto (Zhang et al., 2026a) using the $0.02\,\mathrm{m}$ voxel size adopted during pretraining. For each placed exemplar, we include all scene points within a $1.5\mathrm{m}$ radius and subsample the resulting scene–exemplar union to $6{,}000$ points. The scene-to-object sampling ratio is preserved so that the exemplar occupies a consistent fraction of the point budget regardless of the density or openness of its surroundings. We obtain candidate regions by thresholding each contrast map at the knee of its sorted score profile and clustering the surviving points with DBSCAN (Ester et al., 1996) using $\epsilon = 0.10\,\mathrm{m}$. All resulting clusters are retained as candidates.

Verification depends on the bank modality. For the 2D bank, each shortlisted candidate is rendered from two views aligned with its principal axes at a resolution of 448 pixels. The rendered views and the retrieved reference photographs are embedded with DINOv3 (Siméoni et al., 2025) in full precision and compared in the shared feature space. For the 3D bank, each shortlisted candidate is extracted from the scene, capped at $3{,}000$ points, and re-encoded in isolation with Concerto at the same $0.02\,\mathrm{m}$ voxel size. The bank exemplars are encoded in the same manner and serve as references. We use a category margin of $\delta_v = 0.15$, while the shortlist margin δ_z follows Appendix C.

For routing, we use the released reasoning checkpoint of Reason3D (Huang et al., 2025) and 3D-STMN (Wu et al., 2024) fine-tuned for 40 epochs on the same training data. A query is routed to retrieval when the maximum mask logit produced by the base model falls below a fixed threshold. The same threshold is used across all base models, banks, and benchmarks.

A.3 Hardware

All experiments are conducted on a machine equipped with two NVIDIA H200 141GB GPUs, an Intel Xeon Platinum 8462Y+ CPU, and 2TB of system memory. The Reason3D checkpoint reproduction is additionally verified on a separate server equipped with four NVIDIA RTX A6000 GPUs.

Table 6: Comparison with 2D projection baselines on Reason3D. Each entry reports Acc@0.25 / Acc@0.5 / mIoU (%). "Matching only" denotes our 3D matching pipeline before verification.

Method	Overall	ScanNet	Matterport3D
P-A: 2D patch voting	10.66 / 4.37 / 6.67	18.51 / 8.12 / 10.42	7.77 / 2.99 / 5.29
P-B: 2D + SAM	9.00 / 3.49 / 5.95	13.64 / 6.49 / 8.47	7.29 / 2.39 / 5.02
P-C: Coarse-to-fine	15.37 / 7.77 / 9.56	19.81 / 10.39 / 11.97	13.74 / 6.81 / 8.68
RA3D, Matching only	22.01 / 14.24 / 14.59	27.27 / 20.45 / 19.33	20.07 / 11.95 / 12.85
RA3D, Full	26.29 / 16.77 / 17.29	30.63 / 20.89 / 20.93	24.49 / 14.54 / 15.77

B 2D Projection Baselines

A natural alternative to our 3D matching stage is to keep the retrieved exemplars in 2D, render the query scene from multiple views, and lift the resulting 2D evidence back to the point cloud. We evaluate three such baselines on Reason3D (Huang et al., 2025), while keeping the LLM category prediction, ten reference images per category, scene superpoints, and evaluation protocol fixed. Each scene is rendered from nine views, and DINOv3 (Siméoni et al., 2025) features are used for image-level matching. **P-A** directly matches rendered scene patches against the retrieved photographs and projects the resulting similarity maps back to superpoints. **P-B** uses the same similarity maps to prompt SAM (Kirillov et al., 2023) and lifts the resulting masks back to 3D. **P-C** strengthens the 2D route in two ways. It renders the reconstructed exemplars as references to reduce the photo-to-render domain gap and performs a second, zoomed matching pass around the initially selected region.

Table 6 shows that all three 2D alternatives remain below direct 3D matching. P-C performs best among them after reducing the appearance gap between references and scene renders and adding a coarse-to-fine pass, but it still trails our matching-only pipeline on both ScanNet and Matterport3D. Adding SAM does not improve the result, indicating that replacing the 2D selection stage with a stronger segmenter does not close the gap. These results motivate performing retrieval matching directly in 3D.

C Margins

C.1 The Shortlist Margin

The shortlist in Sec. 3.4 is controlled at two levels. For each query, the adaptive margin depends only on z_1. We set the margin to 0.05 when $z_1 \geq 4$ and to 0.75 otherwise. This decision is made before verification, so no verifier output affects the shortlist. At the dataset level, we decide whether to use this adaptive rule at all. When the verifier does not separate categories reliably enough, we instead use a fixed margin of 0.10 for every query. The adaptive rule follows the interpretation of z_1. A large z_1 indicates that matching has already isolated a strong candidate, so verification only needs to resolve nearby alternatives and the shortlist can remain narrow. A small z_1 indicates that no candidate clearly stands out, so a wider shortlist passes more of the decision to verification.

We set the boundary at $z_1 = 4$ from the scale on which z is defined. The score is measured in units of the standard deviation of β_e over the scene, so testing whether a candidate stands out from its contrast map can be viewed as testing against the null that it follows the same score distribution as the rest of the scene. Because this comparison is made over all superpoints, a threshold appropriate for a single test would produce chance exceedances somewhere in the scene. Using a Bonferroni correction for a family-wise error rate of 0.05 over approximately $N = 1000$ superpoints gives a per-test level of 5×10^{-5}. The corresponding one-sided Gaussian quantile is 3.89, which we round to 4. We further vary both margin values around the defaults of 0.05 and 0.75 and observe only minor changes in performance, indicating that the method is not sensitive to their exact settings. We use the same adaptive rule on both Reason3D (Huang et al., 2025) and NO-Reason3D. Since the margin depends only on z_1 for each query, it requires no ground-truth labels or dataset-specific calibration and applies unchanged across benchmarks.

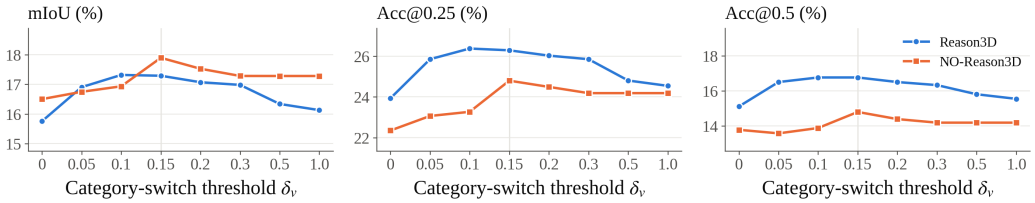

Figure 7: Effect of the category margin δ_v. A margin of zero lets any later category take over and is the worst setting on both benchmarks, while δ_v=1.0 removes the fallback altogether.

Table 7: Results on Instruct3D. Both base models are trained on Reason3D and evaluated here without further training, so every row is out of domain. RA3D rows use the 2D bank.

Method	Acc@0.25	Acc@0.50	mIoU
3D-STMN (Wu et al., 2024)	13.84	8.58	9.62
Reason3D	18.52	10.72	11.70
RA3D with 2D bank	15.66±0.49	11.18±0.66	10.79±0.36
3D-STMN + RA3D (2D)	16.57±0.32	10.72±0.32	11.56±0.18
Reason3D + RA3D (2D)	20.53±0.40	12.61±0.46	13.14±0.23

C.2 The category margin

The category margin δ_v controls how much better a later category must verify before replacing the first. Figure 7 sweeps this margin. With $\delta_v = 0$, any improvement is sufficient for replacement, and mIoU drops to 15.76 on Reason3D (Huang et al., 2025) and 16.50 on NO-Reason3D. Each category score v_k is the maximum similarity over its candidate pool, so categories with more candidates have more opportunities to attain a high score. A positive margin prevents small differences caused by this maximum operation from triggering a replacement. Both benchmarks perform best around $\delta_v = 0.15$, which we use throughout.

At $\delta_v = 1.0$, a later category would need to exceed the first by a full unit of cosine similarity. Since cosine similarity is at most 1, this cannot occur when the first category verifies above zero. This setting therefore effectively disables category fallback and provides a control for measuring its contribution. Relative to this control, category fallback improves mIoU by 1.16 on Reason3D and 0.62 on NO-Reason3D.

D Results on Instruct3D

Instruct3D (He et al., 2024) is built on ScanNet++ (Yeshwanth et al., 2023), a dataset unseen by either base model during training. The evaluation is therefore out of domain for both models. However, the public Instruct3D annotations refer to objects using identifiers that no longer align with the current ScanNet++ release. Using these identifiers directly assigns many validation queries to incorrect objects. We therefore reconstruct the ground-truth masks by matching each annotated object name to the corresponding ScanNet++ labels and taking all matching instances. This procedure recovers 512 of the 513 validation queries and agrees with the public identifiers for 93.2% of them in terms of mask size. As a sanity check, the released Reason3D (Huang et al., 2025) checkpoint obtains 11.70 mIoU under our reconstruction, closely matching the 11.64 reported by its authors. This indicates that the reconstruction corrects the identifier mismatch without materially changing the published evaluation.

Table 7 reports the results. The two base models obtain 11.70 and 9.62 mIoU. Retrieval alone reaches 10.79 mIoU, trailing Reason3D in mIoU and accuracy at 0.25 but exceeding it at 0.5, with 11.18 versus 10.72. Routing retrieval with the base model improves all three metrics, reaching 13.14 mIoU, 20.53 accuracy at 0.25, and 12.61 at 0.5. The resulting 1.44 mIoU gain is smaller than those observed in Sec. 5, consistent with the premise of our method. Instruct3D largely contains the familiar indoor object vocabulary represented during base-model training, whereas retrieval is most useful when the target falls outside that vocabulary.

Table 8: Effect of the scene used to encode each exemplar. The first row is the default setting used throughout the paper. The second places each exemplar in a scene sampled at random from the same dataset, while the third encodes it in isolation. All rows use the 2D bank.

Exemplar encoded in	Reason3D			NO-Reason3D		
	Acc@0.25	Acc@0.50	mIoU	Acc@0.25	Acc@0.50	mIoU
Query scene	26.29	16.77	17.29	24.80	14.69	17.67
Random scene	24.37	14.93	15.80	24.80	14.59	17.48
No scene	19.21	11.00	12.39	16.12	8.37	11.18

Table 9: Ablation of scene-relative matching and distribution-calibrated scoring. Each entry reports Acc@0.25 / Acc@0.5 / mIoU (%).

Method	Reason3D	NO-Reason3D
Raw cosine	18.69 / 9.00 / 11.34	12.86 / 6.53 / 9.67
Uncalibrated scoring	25.15 / 14.76 / 16.31	20.92 / 11.63 / 14.74
RA3D	**26.29 / 16.77 / 17.29**	**24.80 / 14.69 / 17.67**

E Placement in a Random Scene

Placement can help for two different reasons. It can provide an exemplar with generic scene context, or with the specific context of the query scene. Table 8 separates these effects by encoding each exemplar in a randomly sampled scene from the same dataset rather than in the query scene, while leaving the rest of the pipeline unchanged.

Most of the benefit comes from providing scene context at all rather than from using the specific query scene. On Reason3D (Huang et al., 2025), random-scene placement recovers 3.41 of the 4.90 mIoU gained by query-scene placement. On NO-Reason3D, it recovers 6.30 of the 6.49 mIoU gain. Replacing the random scene with the query scene adds another 1.49 mIoU on Reason3D but only 0.19 on NO-Reason3D. The accuracy metrics show the same pattern. On NO-Reason3D, the two placements achieve identical Acc@0.25 and differ by only 0.10 at Acc@0.50.

The difference between the two benchmarks reflects the nature of their targets. In Reason3D, the target is typically a familiar indoor object among other furniture, so the surrounding scene provides useful cues about how such objects are usually placed and what tends to surround them. In NO-Reason3D, the target lies outside the indoor vocabulary, so no particular room provides a naturally privileged context. This explains why random-scene placement captures nearly all of the placement benefit on NO-Reason3D, while the query scene provides a larger additional gain on Reason3D. Overall, the method benefits primarily from encoding the exemplar within a plausible scene rather than from using the exact query scene. We use the query scene in practice because it is naturally available at inference time.

F Ablation of Scene-Relative Matching and Scoring

We further ablate the two components of our scene-relative matching stage. Raw cosine replaces the background-contrastive similarity map with direct cosine similarity between the exemplar and scene features, while keeping the remaining pipeline unchanged. Uncalibrated scoring retains our scene-relative contrast map but ranks candidates using their raw map scores without distribution calibration. Table 9 shows that both modifications reduce performance on Reason3D and NO-Reason3D. Using raw cosine similarity causes the larger degradation, showing the importance of measuring exemplar similarity relative to the scene background. Removing score calibration also consistently hurts performance, particularly on NO-Reason3D, confirming that calibration is useful when comparing candidates produced by different exemplar-specific maps.

G Number of Views at Verification

For the 2D bank, each candidate is compared with the retrieved photographs through rendered views of its point cloud. Figure 8 varies the number of views used for verification. On Reason3D (Huang

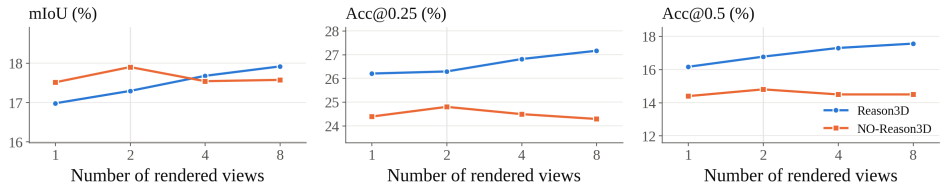

Figure 8: Effect of the number of rendered views at verification. More views keep helping on Reason3D, while two views are best on NO-Reason3D. The vertical line marks the default of two.

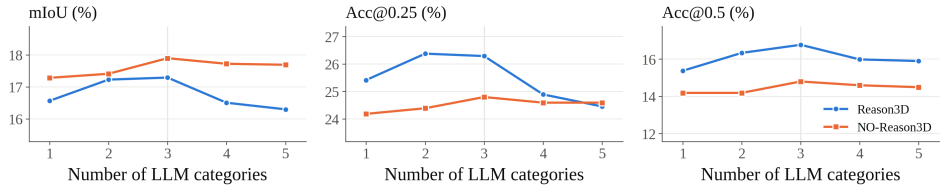

Figure 9: Effect of the number of categories predicted per query. Both benchmarks peak at three, and Reason3D falls more steeply beyond it. The vertical line marks the default of three.

et al., 2025), performance generally improves with additional views, whereas on NO-Reason3D it peaks early and shows little benefit from more views.

This difference reflects how candidate geometry is formed in the two benchmarks. In Reason3D, a candidate comes from a partial scan and contains only the surfaces observed by the sensor, so additional views increase the chance of matching the viewpoint of a retrieved photograph. In NO-Reason3D, inserted objects are sampled from complete meshes, so two principal views already capture most of their shape. We therefore use two views throughout, which performs best on NO-Reason3D while remaining competitive on Reason3D at substantially lower rendering cost.

H Number of Predicted Categories

Figure 9 varies the number of categories predicted by the language model. Both benchmarks perform best with three categories. Increasing the number from one to three helps recover cases where the top prediction is incorrect but a later prediction identifies the target category. Beyond three, performance on NO-Reason3D remains relatively stable, while Reason3D degrades more noticeably.

Lower-ranked predictions are typically semantically close to the first. On Reason3D, these neighboring categories are often other indoor objects that may occur in the scene, allowing an incorrect later prediction to find a plausible match and displace the correct one. On NO-Reason3D, neighboring categories of a novel target are less likely to appear in the scene, so incorrect later predictions receive less supporting evidence. We therefore use the top three predicted categories throughout.

I Inference Time

Table 10 reports the per-query inference time of RA3D on Reason3D (Huang et al., 2025) using three predicted categories and ten exemplars per category. We report two settings. The uncached setting assumes that neither the predicted categories nor the scene has been encountered before, so all thirty exemplars must be reconstructed for the query. The cached setting is measured over a single pass through all 1,145 validation queries, starting from an empty cache. A category or scene is processed when it is first encountered and reused by subsequent queries. Thus, nothing is precomputed before evaluation, and each reported time is the total time spent on that operation divided by the number of queries.

With reconstruction distributed across 15 parallel workers, RA3D takes about 45 seconds per query without caching and 14.5 seconds per query over the full validation pass. Exemplar reconstruction is the dominant cost in both settings, requiring 39 seconds per uncached query and an amortized 8.7 seconds per query over the pass. Its amortized cost would decrease further over longer query streams as more category banks are reused. The largest cost incurred independently for every query is placement, in-scene encoding, and candidate generation, which together take 3.7 seconds. When

Table 10: Time per query on Reason3D, in seconds, with three categories and ten exemplars per category. No cache assumes every exemplar and scene is computed for the query alone. Single pass runs once through the $1{,}145$ validation queries from an empty cache, computing each category and scene the first time it is needed, and the quantity in parentheses is the work done in that pass divided by the number of queries.

Stage	No cache		Single pass	
	Sequential	multi-workers	Sequential	multi-workers
Lifting photographs by TRELLIS	59	39	13.3	8.7
Scene encoding by Concerto	0.85	0.85	0.25	0.25
Loading exemplars	0.6	0.6	0.15	0.15
Placement, in-scene encoding and candidate generation	3.7	3.7	3.7	3.7
Reference encoding by DINOv3	0.2	0.2	0.05	0.05
Rendering and verification by DINOv3	0.9	0.9	0.9	0.9
Category and surface prediction by Qwen2.5	0.05	0.05	0.05	0.05
Total, three categories	≈ 65	≈ 45	≈ 19	≈ 14.5
Total, one category	≈ 23	≈ 16	≈ 7.4	≈ 5.7

using a single predicted category without verification, the amortized inference time decreases to about 5.7 seconds per query.

J Limitations

RA3D has several limitations. First, inference remains computationally expensive, taking about 14.5 seconds per query with caching and up to 45 seconds without it, largely due to exemplar reconstruction and repeated 3D encoding. Second, the retrieval pipeline depends on the categories proposed by the LLM, so an incorrect category set can prevent the correct object from being retrieved. The 2D bank also inherits noise from web images, and imperfect image-to-3D reconstruction can introduce a geometry mismatch between retrieved exemplars and scanned objects. Finally, NO-Reason3D currently covers twelve novel object categories. While these categories provide a controlled test of reasoning segmentation beyond the indoor training vocabulary, broader categories and scene domains are needed to assess how well the approach scales to more diverse open-world settings.

K Additional Qualitative Results

Figure 10 presents additional examples, with the first four from Reason3D (Huang et al., 2025) and the last four from NO-Reason3D. On Reason3D, the base model either returns no mask or selects unrelated regions, while RA3D identifies the intended door, drying rack, lamp, and ceiling. The difference is more pronounced on NO-Reason3D, where the targets lie outside the training vocabulary. The base model again fails to localize the intended objects, whereas RA3D recovers the bus, rifle, pistol, and airplane. These examples further show that retrieval helps both when the base model misses familiar indoor targets and when the query refers to novel objects absent from training.

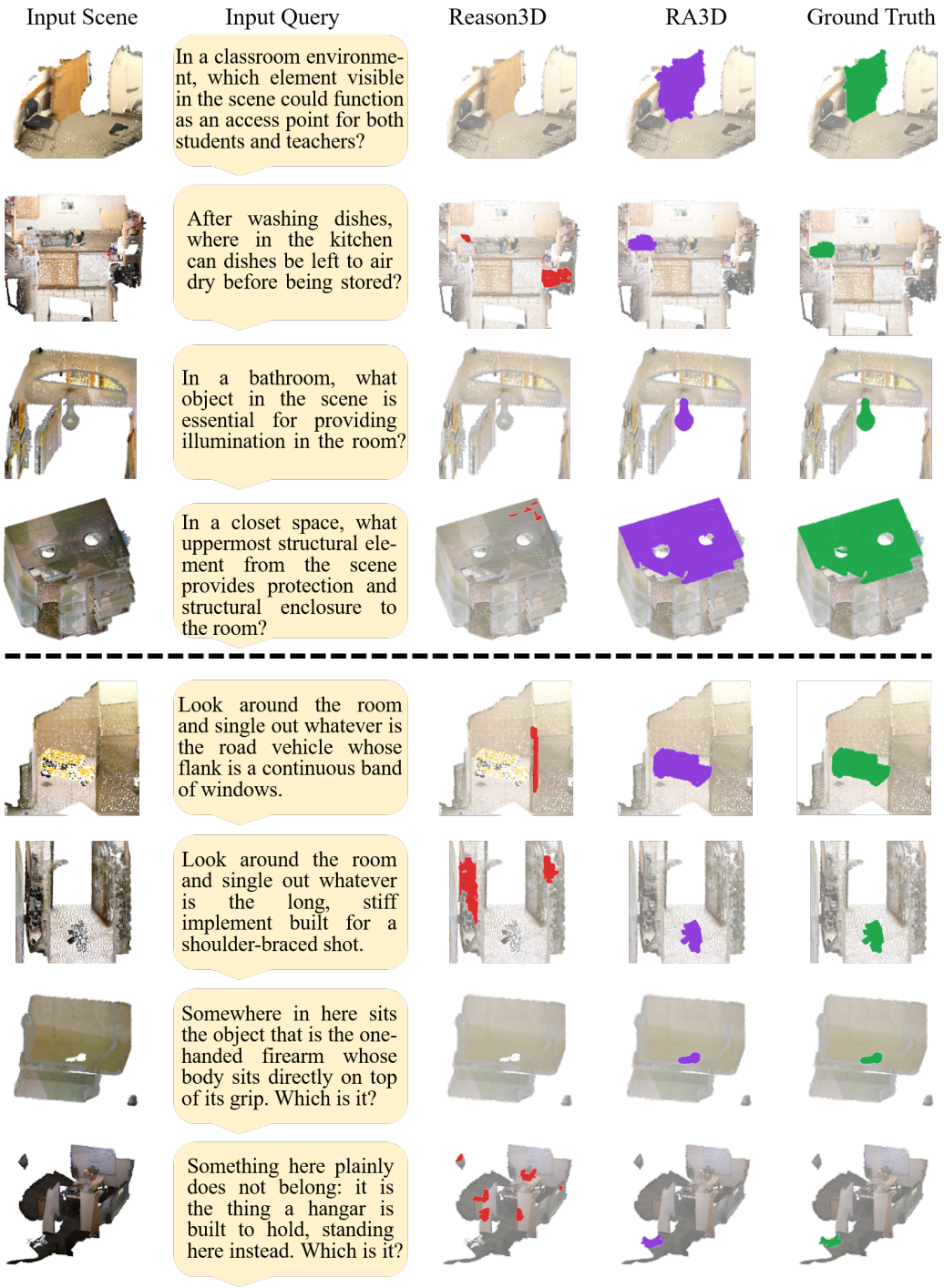

Figure 10: Additional qualitative results. Each row shows the input scene, the query, the prediction of the base model in red, our prediction in purple and the ground truth in green. The first four rows come from Reason3D and the last four from NO-Reason3D.