R3D: QUANTITATIVE 3D SPATIAL REASONING FOR EGOCENTRIC WEARABLES

Anonymous authors
Paper under double-blind review

ABSTRACT

Quantitative 3D spatial reasoning from egocentric RGB-D video is an important capability for next-generation wearable assistants. Yet existing benchmarks do not reflect the challenges of handling (1) natural egocentric video, (2) posed RGB-D video inputs, and (3) challenging quantitative 3D spatial reasoning Q&A. To fill this gap, we introduce **R3D-Bench** (**R**easoning in **3D**), a benchmark of 3,033 quantitative spatial reasoning questions across 15 types—spanning multiple-choice, distance-based, and volumetric reasoning questions—built on top of 57 egocentric video sequences from Aria Digital Twin (Pan et al., 2023). To set a strong baseline on this dataset, we introduce **R3D**, a model-agnostic spatial tool-calling framework. In contrast to existing approaches that directly embed 3D information into the model's input representation, R3D constructs a 3D scene from video using segmentation and depth-lifted object representations. It provides this information to an LLM through eight composable spatial tools. On R3D-Bench, R3D with Qwen3-VL 235B achieves 73.5% mean relative accuracy—substantially outperforming the best depth-enabled baseline (CuTR+Tools, 61.9%) and the best RGB-only baseline (Gemini 3 Flash, 46.5%). We publicly release both R3D-Bench and our full implementation; anonymized code is included in the supplementary material (public URLs withheld for review).

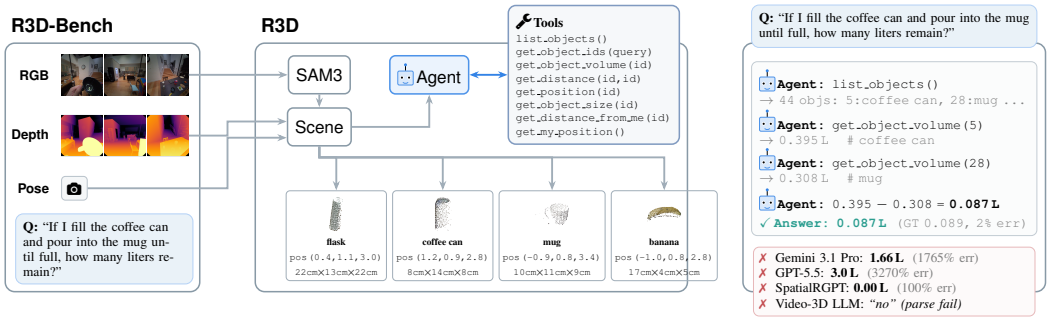


Figure 1: **Left: R3D-Bench** presents egocentric RGB-D frames, camera pose, and quantitative spatial reasoning questions. **Middle: R3D** uses SAM3 to segment objects in a video, then lifts them into a 3D scene. An LLM answers the question by calling spatial tools over the scene. **Right:** the LLM's tool-calling trace — it identifies the coffee can and mug, queries their volumes, and computes the answer (2% error), while frontier baselines hallucinate or fail to parse.

1 INTRODUCTION

Wearable compute devices equipped with RGB-D images and real-time SLAM are rapidly maturing. Depth sensing and 6-DoF pose tracking are already standard on shipped augmented reality and mixed reality devices such as HoloLens 2 (Microsoft, 2019), Quest 3 (Meta, 2023), and Apple Vision Pro (Apple, 2024). Research platforms like Project Aria (Engel et al., 2023) are bringing SLAM-enabled wearables to a lightweight glasses form factor. A natural application of these sensors is equipping wearable AI systems with quantitative 3D spatial reasoning. Users wearing such devices will ask questions like "If I fill my mug, how much liquid is left in my coffee can?" or "How far is the

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.

couch from the TV?" Answering these questions requires quantitative spatial reasoning—producing outputs in absolute units (e.g. meters, liters) and applying reasoning on top of the outputs—rather than only evaluating qualitative relationships (e.g. "above", "below").

Existing 3D spatial reasoning benchmarks do not provide a realistic evaluation of the quantitative reasoning performance of wearable AI assistants. Many of them are purely image-based (Chen et al., 2024; Cheng et al., 2024), or operate on pre-scanned 3D scenes with complete reconstructions (Azuma et al., 2022; Ma et al., 2023), or use scanning walkthroughs with unnatural motion (Yang et al., 2025; Wang et al., 2024), or focus only on qualitative rather than quantitative queries (Majumdar et al., 2024). None combine the three properties required to evaluate a next-generation wearable spatial AI assistant: (1) **natural egocentric video**, (2) **calibrated depth and camera pose**, and (3) **challenging quantitative 3D spatial reasoning Q&A**.

To address this gap, we introduce **R3D-Bench** (Figure 1), a quantitative 3D spatial reasoning benchmark built on top of 57 video sequences from the Aria Digital Twin (ADT) dataset (Pan et al., 2023). ADT contains videos captured with Project Aria glasses during natural activities (cleaning, cooking, meal preparation, working, and decoration). Our Q&A benchmark annotates these videos with 3,033 annotations across 15 question types in three categories: multiple choice (comparing two or more metric quantities), distance-based, and volume-based. Our dataset features accurate ground truth derived from detailed object annotations from ADT.

We find that existing multimodal large language models (LLMs) perform poorly on R3D-Bench. Recent methods that incorporate depth information in a learned latent representation (Zheng et al., 2025; Cheng et al., 2024) provide strong results on qualitative questions (e.g. "the cup is to the left of the plate"), but do not generalize well to providing quantitative measurements (Yang et al., 2025).

To provide a strong baseline on R3D-Bench, we develop **R3D** (Figure 1), a model-agnostic spatial tool-calling framework. Given egocentric video, depth, and pose, R3D segments the scene with SAM3 (Carion et al., 2025) and lifts objects to 3D via depth back-projection. R3D then combines partial observations using multi-view consistency and KNN filtering to produce a point cloud. This point cloud is used to generate bounding boxes suitable for distance-based Q&A. Additionally, meshes are generated with SAM3D (Chen et al., 2025) and resized into metric space for answering volumetric questions. These bounding boxes and meshes are then queried by an LLM through eight spatial tools. Our method can be applied zero-shot to any LLM with tool calling capabilities.

On R3D-Bench, R3D with Qwen3-VL 235B-A22 achieves 73.5% mean relative accuracy—substantially outperforming the best RGB-only baseline (Gemini 3 Flash, 46.5%) and other 3D methods (CuTR (Daxberger et al., 2025) at 61.9%, Video-3D LLM (Zheng et al., 2025) at 25.3% and SpatialRGPT (Cheng et al., 2024) at 27.7%). We provide analysis of the failure modes and opportunities for further improving performance.

In summary, our contributions are:

- **R3D-Bench**: a challenging quantitative 3D spatial reasoning benchmark for egocentric wearables with 57 videos and 3,033 metric-space Q&A annotations across 15 types.
- **R3D**: a model-agnostic spatial tool-calling framework that constructs a 3D scene from egocentric video, producing state-of-the-art accuracy on R3D-Bench while requiring zero training FLOPs.

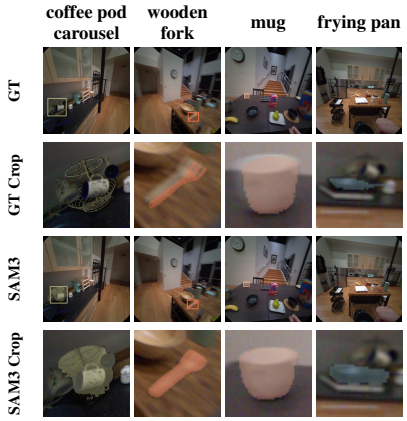
The rest of our paper is organized as follows: we discuss related works in Section 2. We describe construction of R3D-Bench in Section 3. We describe R3D in Section 4. Our main results appear in Section 5. We discuss limitations in Section 6. We conclude in Section 7.

2 Related Work

3D Q&A Datasets Many prior works explore Q&A over 3D datasets. The primary difference with our work is that we explore the egocentric wearables use case. Several datasets pose semantic and relational questions assuming the availability of room scans rather than streaming egocentric video (Azuma et al., 2022; Ma et al., 2023; Hong et al., 2023; Huang et al., 2024; Wang et al., 2024). A few others introduce quantitative questions Yang et al. (2025); Cheng et al. (2024); Daxberger et al. (2025), but not in the RGB-D egocentric video setting. OSI-Bench (Wu et al., 2026) has accurate ground truth but captures with a LiDAR rig and does not provide aligned depth as an input.

	Nat. Egocentric	Video	Depth+Pose	Quantitative Q&A
ScanQA	–	–	–	–
SQA3D	–	–	–	–
EmbodiedScan	–	✓	✓	–
SpatialVLM	–	–	–	✓
SpatialRGPT-Bench	–	–	–	✓
CA-VQA	–	–	✓	✓
OpenEQA	✓	✓	✓	–
VSI-Bench	–	✓	–	✓
OSI-Bench	✓	✓	–	✓
R3D-Bench	✓	✓	✓	✓

(a) Comparison of R3D-Bench and existing benchmarks.



(b) Example visual inputs. Rows 1–2: ground truth mask with zoomed crop. Rows 3–4: SAM3 mask with zoomed crop.

Figure 2: Benchmark comparison and example visual inputs. The examples in (b) illustrate challenges in R3D-Bench: distant objects with thin elements (coffee pod carousel), motion blur (wooden fork), and distant objects on dark surfaces (frying pan).

OpenEQA (Majumdar et al., 2024) is complementary: an egocentric video QA benchmark focused on episodic memory rather than metric measurement. Figure 2a summarizes these differences.

3D Q&A Methods Many previous works explore *qualitative* 3D spatial reasoning. Video-3D LLM (Zheng et al., 2025) encodes 3D position information directly into video token embeddings. Other works solve 3D Q&A by building 3D scenes (Jatavallabhula et al., 2023; Gu et al., 2024; Kerr et al., 2023; Peng et al., 2023) or through inference techniques leveraging pre-scanned 3D scenes (Azuma et al., 2022; Ma et al., 2023; Hong et al., 2023; Huang et al., 2024). These works do not handle partially-observed objects from egocentric video or quantitative reasoning Q&A.

Among works capable of quantitative spatial reasoning Q&A, SpatialVLM (Chen et al., 2024) and SpatialRGPT (Cheng et al., 2024) address image-based Q&A. MM-Spatial (Daxberger et al., 2025) incorporates depth and pose into spatial understanding, but evaluates on multi-view images.

A few works explore tool calling (Schick et al., 2023; Yao et al., 2023; Patil et al., 2024) for spatial reasoning (Chen et al., 2026; Zhang et al., 2026; Luo et al., 2026; Marsili et al., 2025), but do not leverage calibrated depth or operate in the egocentric wearables setting. Concurrent to our work, RieMind (Ropero et al., 2026) combines an LLM with a 3D scene graph and geometric tools similar to our method. However, it operates on ground truth 3D annotations rather than running perception itself, representing an upper bound rather than a deployable system. A few other concurrent works (Cho et al., 2026; Dai et al., 2026) investigate agentic spatial tool use over existing benchmarks. We differ in that our focus is on evaluation and deployment of a realistic wearable spatial AI benchmark.

3 R3D-Bench

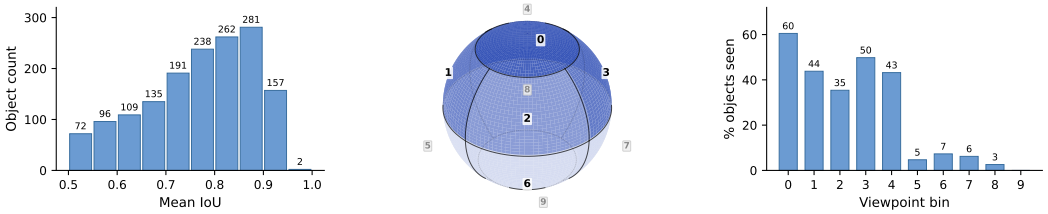
Here we discuss the construction of R3D-Bench. In Section 3.1, we discuss our choice of building annotations on top of Aria Digital Twin (ADT). In Section 3.2, we discuss the construction of Q&A pairs. In Section 3.3, we discuss characteristics of our challenging benchmark.

3.1 Aria Digital Twin

Our dataset is built on 57 sequences from Aria Digital Twin (Pan et al., 2023), captured with Project Aria glasses (Engel et al., 2023). This dataset contains 90–100 second RGB-D videos of everyday activities (cleaning, cooking, meal preparation, working, and decoration). Each frame has pose annotations and detailed object annotations. Our main motivations for using ADT are (1) natural egocentric video, (2) the presence of depth and pose information, and (3) rich annotations for quantitative Q&A, including object meshes for volumetric estimation ground truth generation. In

Table 1: The 15 question types in R3D-Bench with example questions, answer format, and question count. Each question has an average of 280 input frames (93.3 seconds at 3 FPS). Bool = yes/no, Str = object name, Float = number in meters or liters.

Category	Type	Fmt	N	Example question
Multiple Choice	gap_fit	Bool	62	"Could I place the red armchair in the gap between the white chair and the kitchen island?"
	nearest_from_set	Str	228	"Which is closest to me: the coffee can, the wooden fork, or the large frame?"
	top_higher	Bool	224	"Is the top of the portable cooktop above the top of the black round table?"
	which_taller	Str	227	"Between the green pear and the coffee maker, which one is taller?"
	which_longer	Str	228	"Between the red armchair and the curtain, which one is bigger in its longest dimension?"
Distance	how_far	Float	228	"How many meters apart are the refrigerator and the double door?"
	far_from_me	Float	228	"How far away from me is the black bar stool, in meters?"
	how_long	Float	228	"How big is the picture ledge along its longest axis, in meters?"
	much_taller	Float	226	"By how many meters is the bookcase taller than the red armchair?"
	much_longer_dim	Float	228	"How much longer is the carrot than the green pear in their longest dimensions?"
	fly_distance	Float	228	"What is the total straight-line distance from me to A, then from A to B?"
	walk_distance	Float	228	"How far would I walk horizontally if I go to A first, then to B?"
Volume	volume	Float	168	"How much liquid can the large pot hold, in liters?"
	pour_leftover	Float	151	"Fill the large pot with water, pour into the mug until full. How many liters remain?"
	pour_room_left	Float	151	"Pour the mug of water into the bowl. How much room is left in the bowl, in liters?"



(a) SAM3 IoU distribution. (b) Viewpoint binning scheme. (c) % of objects seen from each bin.

Figure 3: Dataset characteristics. (a) SAM3 segmentation quality. (b) Viewpoint binning scheme used in (c). (c) Percentage of objects seen from each binned viewpoint.

Figure 2b, we visualize a few images from Aria Digital Twin alongside segmentation masks produced by a state-of-the-art object detector, SAM3 (Carion et al., 2025). These examples highlight challenges with thin objects, small objects, and motion blur.

3.2 R3D-BENCH CONSTRUCTION

R3D-Bench includes frame timestamps, questions, answers, and pointing annotations. We discuss generation of these elements here.

Frame Timestamps: An always-on AI assistant typically streams frames to cloud-based systems for analysis. Such systems cannot provide high-FPS image streams due to power and latency considerations. Thus, for consistent and fair evaluation, we first generate a list of valid frame timestamps at 3FPS. We use this list of timestamps to extract RGB-D frames (with fisheye distortion removed) and camera poses from ADT. We downsample images to 512x512, to simulate a low-power device streaming frames for cloud processing at acceptable resolution.

Questions and Answers: Next, we produce quantitative 3D spatial reasoning question and answer annotations for each sequence. First, we produce natural names for objects, mapping ADT names (e.g. "AirPurifier_1") to natural names (e.g. "air purifier"). We then define 15 question types organized into three categories (Table 1):

- **Multiple Choice:** e.g. "Does the couch fit in the gap between the TV and the plant?".
- **Distance:** e.g. "How much longer is the couch than the chair?".
- **Volume:** e.g. "If I pour the flask into the mug, how much water is left?"

Each category contains 3–7 subtypes (Table 1). Each subtype has 2–4 natural-language templates for linguistic variety. Templates reference specific objects using their natural names and are populated from the ADT ground truth 3D annotations. Ground truth answers are computed from ADT bounding boxes for distance-based questions, and functional mesh volumes for volume-based questions.

Pointing Annotations: To disambiguate semantically similar objects (e.g. "cup" and "glass"), we provide a "pointing reference" to each referred object in each question. This reference is a ground truth mask for the object the first time it appears in the video. We do not provide ground truth segmentations for any other timestamp.

Filtering: To ensure our benchmark focuses mainly on metric spatial Q&A without contamination from object ambiguity or extremely heavily occluded objects, we filter objects by name uniqueness and by whether it can be identified by a state-of-the-art object tracker at some point in the video. We also filter out questions that are too obvious, such as "Is the carrot taller than the refrigerator?" See Appendix A for details.

3.3 Dataset Characteristics

Our final dataset contains (1) natural egocentric video, (2) calibrated depth and camera pose, and (3) challenging quantitative 3D spatial reasoning Q&A to provide a challenging benchmark for depth-enabled wearables. We highlight some of the challenges here:

Visual Inputs: As shown in Figure 2b, our dataset contains motion blur, small objects viewed from a distance, and thin objects. These all contribute to challenges in visually parsing objects. We show the distribution of SAM3 tracking IoU for each object in Figure 3a. Note that objects with mean IOU below 0.5 are filtered out of the dataset (see Appendix A).

Partial Observations: Our final dataset exhibits challenging scenarios with partially-observed objects. In Figure 3b and Figure 3c, we visualize the distribution of viewpoints using an equal-area spherical binning scheme with 10 bins of equal solid angle. We note that 60.5% of objects are seen from above, but few objects are seen from below (3–7% per bin, with 0% of objects seen from directly below). This contrasts sharply with scan-based datasets (Dai et al., 2017; Yeshwanth et al., 2023) where deliberate scanning motions include low viewpoints with the camera pointing upwards.

4 R3D: 3D Spatial Reasoning with Tools

To address the challenges posed by R3D-Bench, we develop R3D, a method for quantitative 3D spatial reasoning with tools. Motivated by recent works in agentic AI (Yao et al., 2023; Schick et al., 2023), we use tools to inject 3D information in LLMs. The core of our method involves aggregating 3D information from RGB-D images and pose in order to expose this information to an LLM capable of reasoning. We note that even 3D LLMs that operate in latent space still typically use object detectors to provide input signals to the LLMs (Zheng et al., 2025; Daxberger et al., 2025). Our approach pushes this concept further, providing rich 3D signals to the model in the form of tools.

Our overall approach is shown in Figure 1. Our method takes in RGB-D video, camera pose, and questions, and outputs answers through scene construction (Section 4.1) and multi-step tool calling (Section 4.2). Our method is model-agnostic, and can be integrated with any tool-calling-enabled reasoning model with no training required.

4.1 Scene Construction

For each object in the scene, we build tracks using SAM3 (Carion et al., 2025), which propagates object tracks forward using visual feature embeddings. We then lift these points to 3D from all views of the objects to obtain a 3D point cloud. Obtaining precise segmentations from our 3FPS video stream at 512x512 resolution poses a challenge (Figure 2b), leading to spurious background points that massively inflate the object's point cloud.

To address this, we implement point cloud filtering in two steps. First, we discard points that are seen in fewer than 50% of image segmentations (without penalizing points that are missing due to occlusion). Next, we use outlier removal to drop points far away from the central mass of the object. See Appendix B for details.

Table 2: Main results on R3D-Bench across 15 spatial question types. **PF%**: parse failure rate (failures excluded from accuracy). Best per column is **bold**, computed separately above (>100B param) and below (<100B param) the dividing line. Ours in gray.

			Multiple Choice (Acc ↑)						Distance (MRA ↑)								Volume (MRA ↑)				
	Depth	*PF% ↓*	*Gap Fit*	*Nearest*	*Top Higher*	*Wh. Taller*	*Wh. Longer*	*MC Avg*	*How Far*	*Far From Me*	*How Long*	*Much Taller*	*Much Longer*	*Fly Dist.*	*Walk Dist.*	*Dist Avg*	*Volume*	*Pour Left.*	*Pour Room*	*Vol Avg*	*Avg*
Gemini 2.5 Pro	✗	7.5	64.5	42.5	66.2	74.0	74.6	64.4	27.5	26.8	49.7	31.3	31.5	37.1	27.7	33.1	20.4	5.4	12.8	12.9	39.5
Gemini 3 Flash	✗	0.1	80.6	39.5	63.4	82.8	81.1	69.5	34.6	24.6	63.0	46.7	40.7	47.1	38.9	42.2	33.0	6.2	16.0	18.4	46.5
Gemini 3.1 Pro	✗	0.3	83.9	38.6	66.5	85.5	80.7	71.0	29.9	23.7	57.1	43.1	37.3	39.2	34.3	37.8	27.3	10.1	11.6	16.3	44.6
GPT-5.5	✗	0.0	70.5	40.3	61.9	85.0	77.2	67.0	27.9	31.4	63.1	48.1	48.1	31.2	26.6	39.5	25.7	14.3	19.1	19.7	44.7
Qwen3-VL 235B-A22 Think RGB	✗	0.7	74.2	42.5	60.3	80.6	79.4	67.4	28.2	27.2	42.2	30.2	31.2	33.0	21.2	30.5	14.1	1.5	3.7	6.4	38.0
CuTR + Tools (Q3-VL 235B-A22 Think)	✓	**0.0**	71.0	92.1	85.7	88.1	91.7	85.7	82.9	88.6	43.7	23.3	28.2	**80.9**	**58.5**	58.0	43.9	24.4	25.8	31.4	61.9
R3D (Qwen3-VL 235B-A22 Think)	✓	**0.0**	**95.2**	**95.6**	**90.2**	**92.5**	**95.2**	**93.7**	**89.6**	**95.0**	**82.2**	**66.0**	**58.5**	75.2	55.3	**74.6**	**46.0**	**32.2**	**33.9**	**37.3**	**73.5**
Qwen3 8B text-only	✗	0.1	56.5	35.5	54.0	56.4	54.4	51.4	17.9	24.0	44.4	23.9	31.3	5.1	4.1	21.5	31.7	0.0	0.6	10.8	29.3
Video-3D LLM	✓	54.0	46.8	55.2	51.0	49.7	48.3	50.2	40.3	40.0	10.3	14.3	0.0	7.5	9.9	17.5	1.1	4.6	0.0	1.9	25.3
SpatialRGPT + Median	✓	9.1	50.0	24.0	53.1	70.6	68.8	53.3	16.1	29.9	44.6	7.3	11.3	0.0	0.0	15.6	38.6	0.0	1.1	13.2	27.7
CuTR + Tools (Qwen3-VL 8B Instruct)	✓	1.3	70.2	83.4	**88.4**	61.4	89.3	78.5	81.0	78.0	41.8	15.5	30.4	76.0	**42.6**	52.2	43.9	24.7	26.9	31.8	56.9
CuTR + Tools (Qwen3.6 35B-A3B)	✓	0.3	71.0	91.7	81.7	89.7	91.1	85.0	82.9	88.6	43.9	24.2	28.1	**77.6**	40.5	55.1	43.9	24.8	25.8	31.5	60.4
R3D (Qwen3-VL 8B Instruct)	✓	1.3	88.7	82.3	71.4	89.8	**94.7**	85.4	89.1	64.8	79.3	66.2	50.7	32.9	14.1	56.7	46.4	**33.3**	**35.5**	**38.4**	62.6
R3D (Qwen3.6 35B-A3B)	✓	**0.1**	**98.4**	**95.6**	80.4	**92.0**	**94.7**	**92.2**	**89.5**	**95.1**	**82.4**	**68.4**	**56.6**	69.4	37.0	**71.2**	**47.3**	32.6	34.5	38.1	**71.6**

4.2 Multi-Step Tool Calling

After constructing our scene, we perform inference using multi-step tool calling. Our tools utilize computations performed on the scene's 3D object point clouds. Our tools are built on top of (1) a bounding box representation for reasoning over position and object extent, and (2) a mesh-based representation used for volumetric reasoning. Both representations are computed directly from the point cloud.

Bounding Box Representation: For distance-based Q&A, we compute gravity-aligned bounding boxes over the point clouds. The vertical extent is the vertical range of the point cloud. The horizontal orientation is determined by 2D PCA on the XZ-plane projection of the point cloud.

Mesh-Based Representation: Estimating the object's volume presents a challenge, as many objects are only seen from a few angles (Figure 3). This makes volumetric estimation from a point cloud challenging, as many objects will have massive holes. To address this, we use SAM3D (Chen et al., 2025) to produce an estimated mesh for objects using the first SAM3 segmentation of the object as input. Because SAM3D does not provide meshes in metric space, we need to rescale it. To do this, we scale the mesh in each dimension uniformly by the factor

$$s = \left(\frac{V_{\text{bbox}}}{V_{\text{mesh}}} \right)^{1/3}, \tag{1}$$

where V_{bbox} is the volume of the object's gravity-aligned bounding box (computed from the point cloud, discussed above) and V_{mesh} is the volume of the SAM3D mesh's bounding box. We then compute the functional volume of the scaled mesh using voxel-based cavity detection.

Using the above building blocks, R3D provides eight composable spatial tools to the LLM as function definitions:

1. `list_objects()` — discover available objects and their IDs
2. `get_object_ids(query)` — resolve a text description to object ID(s). This connects a SAM3 query description to an object ID.
3. `get_distance(id1, id2)` — Euclidean distance between the closest points on bounding boxes.
4. `get_position(id)` — 3D position of an object's bounding-box center.
5. `get_object_size(id)` — gravity-aligned bounding box dimensions.
6. `get_object_volume(id)` — computes the functional volume of the scaled mesh.
7. `get_distance_from_me(id)` — distance from the current camera position to the closest point on an object's bounding box.
8. `get_my_position()` — current camera position in the scene.

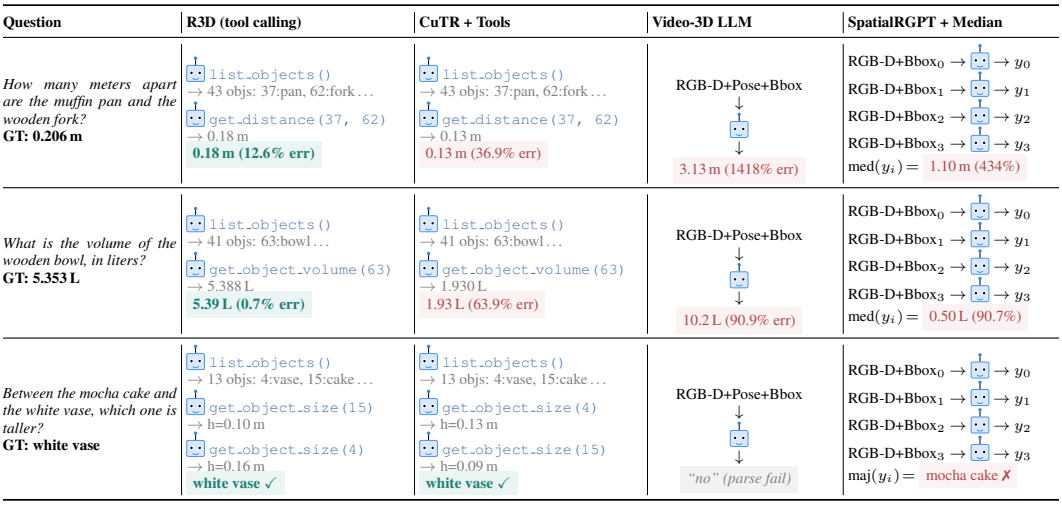


Figure 4: Qualitative examples across three question types. The robot icon denotes an LLM call. R3D and CuTR answer via multi-step tool calling; Video-3D LLM is an end-to-end model; SpatialRGPT+Median runs per-frame inference and aggregates by median/majority vote. **Teal**: correct or $\leq$25% error. Red: incorrect or >25% error.

Note that the design follows a resolve-first pattern: the LLM must identify objects by numeric ID (via `list_objects` or `get_object_ids`) before querying their spatial properties.

5 Experiments

In Section 5.1, we compare R3D against other methods on R3D-Bench, to (1) demonstrate the effectiveness of R3D, and (2) demonstrate the challenges presented by R3D-Bench. In Section 5.2, we provide error analysis. In Section 5.3, we discuss the particularly challenging case of volumetric estimation. We show additional analysis of runtime considerations in Appendix E.

5.1 Main Results

Our main results appear in Table 2. We experiment with using R3D with a Qwen (Bai et al., 2025) model at three different sizes. For efficiency, we evaluate R3D without image inputs, relying on tool use for the visual signal. In ablations, we found no difference in performance with and without image inputs on R3D-Bench. Thus our method is compatible with image inputs, but we omit them in R3D evaluations for computational efficiency.

We divide our results into two sections, for models with $\geq 100B$ parameters and $\leq 100B$ parameters. For multiple-choice questions, we report accuracy. For numerical questions, we report Mean Relative Accuracy (MRA) (Yang et al., 2025), which averages accuracy over 10 thresholds:

$$\text{MRA}(\hat{y}, y) = \frac{1}{10}\sum_{k=0}^{9} \mathbf{1}\Big[\frac{|\hat{y}-y|}{|y|} < 1-(0.50+0.05k)\Big]. \tag{2}$$

R3D Outperforms State-of-the-Art LLMs: We compare R3D with large variants of Gemini (Gemini Team, Google, 2023), GPT (OpenAI, 2023), and Qwen (Bai et al., 2025) models in Table 2. These RGB models do not have a provision for accepting depth, so we do not provide it. For fair comparison, we overlay RGB images with SAM3 masks and bounding box labels for the queried objects, to assist these LLMs in the identification of small objects.

We find that RGB-only models perform reasonably well on multiple-choice Q&A, but still trail R3D by a significant margin, with higher accuracy gaps in measurement-based and volumetric questions. State-of-the-art models still struggle with precise 3D localization, emphasizing the need for depth inputs. We visualize a few example questions and outputs in Figure 4.

R3D Outperforms Alternative 3D Methods: We compare with other depth-based methods in Table 2. We compare with Video-3D LLM, which finetunes a multimodal LLM on RGBD+Pose

Table 3: Error analysis across three R3D model sizes. N is the number of incorrect answers in each category and % is the share of that model's incorrect answers. The shaded *Meas. total* columns are the sum of the volume, length, and distance columns.

		Volume		Length		Distance		*Meas. total*		Tool calling		Reasoning		Parse	
Model	Wrong	N	%	N	%	N	%	N	%	N	%	N	%	N	%
R3D (235B)	729	293	40.2	153	21.0	67	9.2	513	70.4	1	0.1	215	29.5	0	0.0
R3D (35B)	797	287	36.0	147	18.4	75	9.4	509	63.9	14	1.8	270	33.9	4	0.5
R3D (8B)	1137	283	24.9	166	14.6	171	15.0	620	54.5	3	0.3	476	41.9	38	3.3

videos for spatial understanding tasks. We evaluate it with ground truth ADT scene boxes to rule out poor box quality explaining its poor performance (see Appendix C for details). Unfortunately, we find that this model exhibits parse failures on 54.0% of examples (see Figure 4), likely because the model was primarily trained on qualitative (not quantitative) relationships. On the questions that don't exhibit parse failures, it still frequently underperform the Qwen3 8B text-only baseline. This underscores the need for more quantitative reasoning datasets for evaluating 3D methods.

We also construct a strong "CuTR+Tools" baseline, in which CuTR acts as the perception engine in place of our depth-lifted point cloud and SAM3D representation, with the tool interface held fixed.[1] We find that our method outperforms "CuTR+Tools". The biggest performance gap appears on object length (How Long, Much Taller, Much Longer). Visualization inspection suggests that CuTR typically predicts boxes that are too small compared to ground truth.

We additionally adapt SpatialRGPT (Cheng et al., 2024), an image-based method, to video by aggregating its per-frame predictions (Appendix C). This method fails to outperform the Qwen3 8B text-only baseline. Performance is particularly low on questions that require seeing multiple objects (How Long, Much Taller, Fly Dist) as the model may not see these objects simultaneously.

R3D-Bench is Challenging: In spite of the strong performance of R3D, we note that overall performance peaks at 73.5%, even when using MRA as the primary metric (which gives partial credit for measurement errors up to 50%, see Equation 2). Multiple choice questions have accuracies that are nearly saturated (at most 93.7%), but still have room for improvement. Volumetric reasoning peaks at 37.3%, indicating the particular challenge of these questions and opportunities for algorithmic improvement. Text-only Qwen3 8B can achieve up to 31.7% on volume estimation by guessing reasonable values, but achieves less than 1% accuracy on volumetric pouring questions.

5.2 Error Analysis

We perform error analysis of R3D by analyzing tool calls and reasoning traces on quantitative answers with error above 25% as well as incorrect multiple choice questions. For each tool call made by the model, we compare the tool's numeric output to the ground truth value. If the tool output is off by over 25%, we attribute the failure to measurement error. For remaining incorrect answers, we then observe whether the model made the tool call(s) necessary to answer the question. If not, we attribute the error to tool calling. Remaining errors are attributed to reasoning. The results appear in Table 3.

We find that measurement-based errors account for 64–70% of errors for the 235B and 35B models, and 54.5% for the 8B model. Volume is the largest source of measurement error across all models ($\sim$36–40% for 235B/35B), followed by bounding box dimensions. Each model leverages the same underlying perception system, so improving the reconstruction pipeline would directly reduce error rates for all models.

Qwen3-VL 8B is reasoning-bottlenecked: 41.9% of its errors are reasoning failures. It makes 2.2$\times$ more reasoning errors than 235B (479 vs 216). The 235B and 35B models are pipeline-bottlenecked — when they answer wrong, it is predominantly because the tool returned inaccurate data.

[1]We sought to compare directly with MM-Spatial (Daxberger et al., 2025), which performs multi-view spatial Q&A, but at the time of writing, its code release omits model weights. Therefore, we compare against CuTR, which MM-Spatial builds upon. See Appendix C for more details.

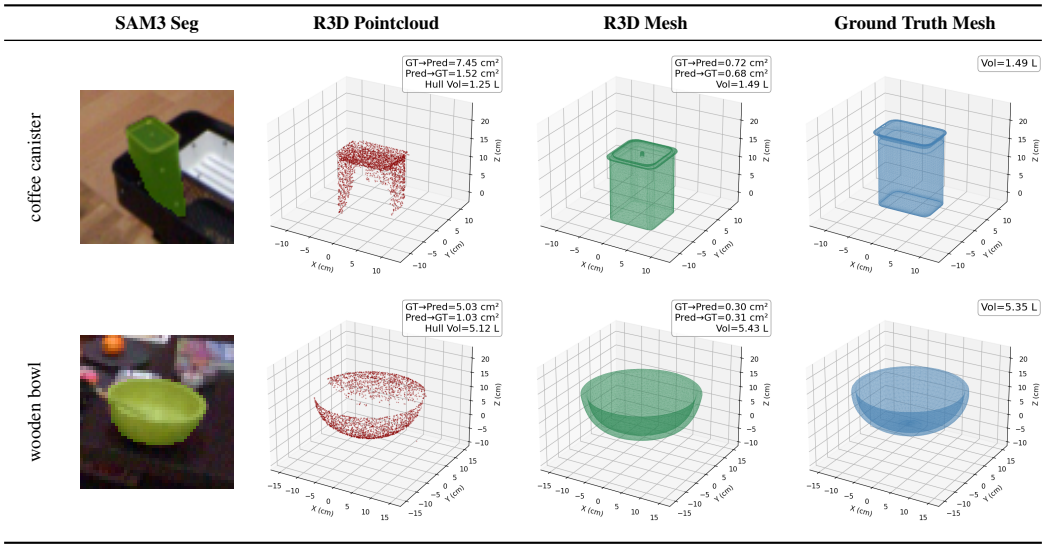


Figure 5: 3D reconstruction comparison. SAM3 Seg: the SAM3 segmentation crop passed to SAM3D. Pointcloud: depth-lifted point cloud aggregated from multiple views. R3D Mesh: SAM3D mesh rescaled to match the pointcloud bounding box. GT: ground-truth mesh from ADT. We show Chamfer Distance, volumes, and the convex hull volume of the point cloud.

5.3 Volumetric Reasoning Performance

Volumetric questions present the greatest challenge and the lowest accuracy numbers for all models (Table 2) due to the sparse viewpoints present for objects. In Figure 5, we show two examples of the inputs used to create our 3D representation: (1) the SAM3 mask used by SAM3D, and (2) the point cloud derived from multiple views used for scaling via Equation 1. We also show the mesh produced by R3D (using SAM3D and scaling), compared to the ground truth. SAM3D helps improve the representation by filling in missing details, but still struggles to precisely reconstruct objects from the single-view segmentation alone. Additional analysis measures the reconstruction quality against the ground truth meshes using Chamfer Distance in Appendix D.

6 Limitations

R3D requires calibrated depth maps and 6-DoF camera pose as input, which are currently available only on devices with active depth sensors and onboard SLAM (e.g. augmented reality or mixed reality headsets). This is by design: our focus is on developing evaluations and methods for next-generation wearable devices where calibrated depth and pose are available.

7 Conclusion

We present **R3D-Bench**, a benchmark for quantitative spatial reasoning from egocentric video. Our benchmark provides (1) natural egocentric video, (2) calibrated depth and camera pose, and (3) challenging quantitative 3D spatial reasoning Q&A to provide a realistic benchmark for depth-enabled wearables. R3D-Bench contains 3,033 3D quantitative spatial reasoning questions across 15 types over 57 egocentric video sequences. Our dataset provides challenges such as small objects, sparse viewpoints, low resolution, and motion blur.

We also present **R3D**, a tool calling framework that can be integrated into any tool-enabled LLM without requiring training. R3D constructs a 3D scene from video and exposes it to an unmodified LLM through eight composable spatial tools, achieving 73.5% MRA, a strong improvement compared to the best depth-enabled baseline (CuTR+Tools, 61.9%) and the strongest RGB-only baseline (Gemini 3 Flash, 46.5%).

AI USE STATEMENT

We used generative AI tools throughout this work. Following the ICLR policy, we address every category on the required-disclosure list below, including the categories that do not apply, and then describe our uses from the recommended-disclosure list.

Required disclosures.

- **Generating synthetic data sets**: R3D-Bench question–answer pairs are produced programmatically from Aria Digital Twin ground-truth annotations; no generative model is invoked at generation time, and no model authors an answer or a 3D annotation. However, the natural-language question templates behind the 15 question types, and the mapping from raw ADT object names to natural names, were drafted with generative AI rather than written by hand. Separately, a text-only Qwen3 8B model is used as a *filter* to remove questions answerable without visual input (Appendix A), which shapes the benchmark's final composition. All resulting annotations were validated by the authors through repeated visualization and inspection.
- **Implementing methods**: AI coding assistants were used while developing the R3D implementation, the R3D-Bench generation pipeline, and the tooling that manages jobs on our compute cluster.
- **Cleaning and reformatting the dataset**: the frame extraction, undistortion, downsampling, and object-filtering steps described in Section 3.2 and Appendix A are performed by pipeline code written with AI assistance.
- **Proposing or refining hypotheses**: we used LLM assistants for open-ended brainstorming. The research idea, and the development and iteration that shaped it, were carried out by the authors.
- **Designing or providing feedback on research methodology or experiments**: we solicited feedback from LLM assistants on our methodology and experimental design, and LLM assistants suggested a small number of hyperparameters. The authors made all final design decisions.
- **Supporting qualitative and thematic data analysis, and interpreting results**: generative AI was used to support the error analysis in Section 5.2, which classifies incorrect answers by inspecting tool calls and reasoning traces. The error taxonomy and the conclusions drawn from it were settled by the authors. Scoring itself is fully deterministic: responses are parsed and compared numerically against ground truth, with no LLM judge anywhere in the evaluation.

Required disclosures that do not apply.

- **Developing theoretical models or conceptual frameworks**: not applicable; this work contains no theoretical results.
- **Formulating mathematical claims**: not applicable, for the same reason.
- **Providing critical ingredients for proving mathematical claims**: not applicable, for the same reason.
- **Assisting in the writing of proofs**: not applicable, for the same reason.
- **Assisting with translation**: not applicable; the paper was written in English by the authors.

Recommended disclosures.

- **Creating or editing software code, and creating artifacts**: beyond implementing methods, AI coding assistants were used to create and edit software code and the released artifacts generally.
- **Creating or modifying scientific figures or images**: AI assistance was used to generate the visualizations the authors used to verify the code, the model, and the benchmark.
- **Suggesting experimental parameters**: LLM assistants suggested a small number of hyperparameters, as noted above.
- **Brainstorming**: we used LLM assistants for open-ended discussion of the research direction.
- **Sourcing and searching for information**: we used LLM assistants to look up and gather background material.
- **Drafting parts of a research paper**: LLM assistance was used to prepare initial drafts of some sections.

- **Editing a research paper to improve readability**: LLM assistance was used to edit text for clarity and readability.
- **Suggesting a structure for a research paper**: LLM assistance was used to suggest document structure.
- **Formatting references**: LLM assistance was used to format the bibliography.

The framing of the paper, its organization, and the flow of its argument are the authors' own, and the sections were primarily written by the authors.

Verification and responsibility. Critical components of the code were reviewed by hand by the authors, and the released benchmark was developed iteratively through repeated visualization and inspection of candidate annotations. We have reviewed all AI-assisted work. We take responsibility for the final content of this paper, including all text, claims, and released artifacts.

ETHICS STATEMENT

R3D-Bench introduces new annotations on top of an existing data collection. It is built entirely on top of the publicly released Aria Digital Twin dataset (Pan et al., 2023), and we rely on the consent, capture, and release procedures established by that dataset's authors. Our contribution over ADT is derived annotation: question and answer pairs computed programmatically from ADT's existing object annotations and meshes, together with pointing masks for referred objects. All sequences are staged indoor captures rather than recordings of unaware bystanders.

We also note that R3D composes third-party LLMs that carry their own documented biases and failure modes. Because our tools return measurements rather than free-form descriptions, errors tend to surface as wrong numbers instead of confident prose, but we do not claim this completely eliminates the underlying risk associated with incorrect outputs.

REPRODUCIBILITY STATEMENT

We release both the R3D-Bench benchmark and the full R3D implementation (`<redacted for review>` and `<redacted for review>`; withheld here to preserve anonymity during review). An anonymized copy of the full implementation, including the benchmark generation pipeline, the R3D scene construction and tool-calling code, and the evaluation and scoring scripts, is provided as supplementary material with this submission.

REFERENCES

Apple. Apple Vision Pro. `https://www.apple.com/apple-vision-pro/`, 2024.

Daichi Azuma, Taiki Miyanishi, Shogo Kurita, and Motoaki Kawanabe. ScanQA: 3d question answering for spatial scene understanding. In *CVPR*, 2022.

Shuai Bai et al. Qwen2.5-VL technical report. *arXiv preprint arXiv:2502.13923*, 2025.

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

Boyuan Chen, Zhuo Xu, Sean Kirmani, Brian Ichter, Danny Driess, Pete Florence, Dorsa Sadigh, Leonidas Guibas, and Fei Xia. SpatialVLM: Endowing vision-language models with spatial reasoning capabilities. In *CVPR*, 2024.

Siyi Chen, Mikaela Angelina Uy, Chan Hee Song, Faisal Ladhak, Adithyavairavan Murali, Qing Qu, Stan Birchfield, Valts Blukis, and Jonathan Tremblay. SpaceTools: Tool-augmented spatial reasoning via double interactive RL. In *CVPR*, 2026.

Xingyu Chen, Fu-Jen Chu, Pierre Gleize, Kevin J Liang, Alexander Sax, Hao Tang, Weiyao Wang, Michelle Guo, Thibaut Hardin, Xiang Li, Aohan Lin, Jiawei Liu, Ziqi Ma, Anushka Sagar, Bowen Song, Xiaodong

Wang, Jianing Yang, Bowen Zhang, Piotr Dollár, Georgia Gkioxari, Matt Feiszli, and Jitendra Malik. SAM 3D: 3dfy anything in images. *arXiv preprint arXiv:2511.16624*, 2025.

An-Chieh Cheng, Hongxu Yin, Yang Fu, Qiushan Guo, Ruihan Yang, Jan Kautz, Xiaolong Wang, and Sifei Liu. SpatialRGPT: Grounded spatial reasoning in vision language models. In *NeurIPS*, 2024.

Seokju Cho, Ryo Hachiuma, Abhishek Badki, Hang Su, Byung-Kwan Lee, Chan Hee Song, Sifei Liu, Subhashree Radhakrishnan, Seungryong Kim, Yu-Chiang Frank Wang, and Min-Hung Chen. SpatialClaw: Rethinking action interface for agentic spatial reasoning. *arXiv preprint arXiv:2606.13673*, 2026.

Angela Dai, Angel X. Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. ScanNet: Richly-annotated 3d reconstructions of indoor scenes. In *CVPR*, 2017.

Yalun Dai, Hao Li, Shulin Tian, Runmao Yao, Yuhao Dong, Fangzhou Hong, Zhaoxi Chen, Fangfu Liu, Baoliang Tian, Dingwen Zhang, Tao Wang, Kim-Hui Yap, and Ziwei Liu. S-Agent: Spatial tool-use elicits reasoning for spatial intelligence. *arXiv preprint arXiv:2606.20515*, 2026.

Erik Daxberger, Nina Wenzel, David Griffiths, Haiming Gang, Justin Lazarow, Gefen Kohavi, Kai Kang, Marcin Eichner, Yinfei Yang, Afshin Dehghan, and Peter Grasch. MM-Spatial: Exploring 3d spatial understanding in multimodal LLMs. In *ICCV*, 2025.

Jakob Engel, Vladlen Koltun, and Daniel Cremers. Project Aria: A new tool for egocentric multi-modal AI research. *arXiv preprint arXiv:2308.13561*, 2023.

Gemini Team, Google. Gemini: A family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*, 2023.

Qiao Gu, Alihusein Kuwajerwala, Sacha Morin, Krishna Murthy Jatavallabhula, Bipasha Sen, Aditya Agarwal, Corban Rivera, William Paul, Kirsty Ellis, Rama Cherian, et al. ConceptGraphs: Open-vocabulary 3d scene graphs for perception and planning. In *ICRA*, 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. In *NeurIPS*, 2023.

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. In *ICML*, 2024.

Krishna Murthy Jatavallabhula, Alihusein Kuwajerwala, Qiao Gu, Mohd Omama, Tao Chen, Alaa Maalouf, Shuang Li, Ganesh Iber, Riccardo Sober, Mustafa Elshaer, et al. ConceptFusion: Open-set multimodal 3d mapping. In *RSS*, 2023.

Justin Kerr, Chung Min Kim, Ken Goldberg, Angjoo Kanazawa, and Matthew Tancik. LERF: Language embedded radiance fields. In *ICCV*, 2023.

Zhanpeng Luo, Ce Zhang, Silong Yong, Cunxi Dai, Qianwei Wang, Haoxi Ran, Guanya Shi, Katia Sycara, and Yaqi Xie. pySpatial: Generating 3d visual programs for zero-shot spatial reasoning. In *ICLR*, 2026.

Xiaojian Ma, Silong Yong, Zilong Zheng, Qing Li, Yitao Liang, Song-Chun Zhu, and Siyuan Huang. SQA3D: Situated question answering in 3d scenes. In *ICLR*, 2023.

Arjun Majumdar, Anurag Ajay, Xiaohan Zhang, Pranav Putta, Sriram Yenamandra, Grady Henaff, Sumith Silber, Tiffany Olkin, Chris Paxton, Oleksandr Maksymets, et al. OpenEQA: Embodied question answering in the era of foundation models. In *CVPR*, 2024.

Damiano Marsili, Rohun Agrawal, Yisong Yue, and Georgia Gkioxari. Visual agentic AI for spatial reasoning with a dynamic API. *arXiv preprint arXiv:2502.06787*, 2025.

Meta. Meta Quest 3. `https://www.meta.com/quest/quest-3/`, 2023.

Microsoft. Microsoft HoloLens 2. `https://www.microsoft.com/hololens`, 2019.

OpenAI. GPT-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

Xiaqing Pan, Nicholas Charron, Yongqian Yang, Scott Peters, Thomas Whelan, Chen Kong, Omkar Parkhi, Richard Newcombe, and Yuheng Carl Ren. Aria digital twin: A new benchmark dataset for egocentric 3d machine perception. In *ICCV*, 2023.

Shishir G. Patil, Tianjun Zhang, Xin Wang, and Joseph E. Gonzalez. Gorilla: Large language model connected with massive APIs. In *NeurIPS*, 2024.

Songyou Peng, Kyle Genova, Chiyu Jiang, Andrea Tagliasacchi, Marc Pollefeys, and Thomas Funkhouser. OpenScene: 3d scene understanding with open vocabularies. In *CVPR*, 2023.

Fernando Ropero, Erkin Turkoz, Daniel Matos, Junqing Du, Antonio Ruiz, Yanfeng Zhang, Lu Liu, Mingwei Sun, and Yongliang Wang. RieMind: Geometry-grounded spatial agent for scene understanding. *arXiv preprint arXiv:2603.15386*, 2026.

Timo Schick, Jane Dwivedi-Yu, Roberto Dessì, Roberta Raileanu, Maria Lomeli, Eric Hambro, Luke Zettlemoyer, Nicola Cancedda, and Thomas Scialom. Toolformer: Language models can teach themselves to use tools. In *NeurIPS*, 2023.

Tai Wang, Xiaohan Mao, Chenming Zhu, Runsen Xu, Ruiyuan Lyu, Peisen Li, Xiao Chen, Wenwei Zhang, Kai Chen, Tianfan Xue, Xihui Liu, Cewu Lu, Dahua Lin, and Jiangmiao Pang. EmbodiedScan: A holistic multi-modal 3d perception suite towards embodied AI. In *CVPR*, 2024.

Mingrui Wu, Xuweiyi Chen, Yifan Jiang, and Zezhou Cheng. From indoor to open world: Revealing the spatial reasoning gap in MLLMs. In *CVPR*, 2026.

Jihan Yang, Shusheng Yang, Anjali W. Gupta, Rilyn Han, Li Fei-Fei, and Saining Xie. Thinking in space: How multimodal large language models see, remember, and recall spaces. In *CVPR*, 2025.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. ReAct: Synergizing reasoning and acting in language models. In *ICLR*, 2023.

Chandan Yeshwanth, Yueh-Cheng Liu, Matthias Nießner, and Angela Dai. ScanNet++: A high-fidelity dataset of 3d indoor scenes. In *ICCV*, 2023.

Zaibin Zhang, Yuhan Wu, Lianjie Jia, Yifan Wang, Zhongbo Zhang, Yijiang Li, Binghao Ran, Fuxi Zhang, Zhuohan Sun, Zhenfei Yin, Lijun Wang, and Huchuan Lu. Think3D: Thinking with space for spatial reasoning. *arXiv preprint arXiv:2601.13029*, 2026.

Duo Zheng, Shijia Huang, and Liwei Wang. Video-3D LLM: Learning position-aware video representation for 3d scene understanding. In *CVPR*, 2025.

A Object and Question Filtering

We filter objects within each sequence according to the following criteria:

- **Unique names:** Object names within each sequence must be unique (no ambiguous duplicates).
- **Minimum visibility:** in at least 5 frames of the sequence, the object should occupy at least 6° of the camera's field of view, and should be at least 50% visible.
- **Trackability:** To avoid imperceptibly small objects at our 512x512 resolution, objects must be trackable by a state-of-the-art object tracker. We run SAM3 on the sequence, discarding objects with mean IoU ≤ 0.50 across the sequence. To discard cases where the wrong object is tracked, we also discard cases where fewer than 20% of SAM3 annotations have IoU ≤ 0.05 with the ground truth.

For the final filtering pass over question obviousness, we evaluate the multiple-choice questions with Qwen3 8B (text-only), and filter the semantic questions to achieve near-random accuracy using weighted sampling. Our final dataset contains 3033 questions across the 15 types (see Table 1).

B Point Cloud Filtering

Here we give the details of the two filtering steps summarized in Section 4.1. First, we perform voting. For each point $\mathbf{p}_i$, we count the fraction of frames whose segmentation mask contains $\mathbf{p}_i$ when projected into the frame. We discard points seen in fewer than 50% of segmentations:

$$\mathcal{P}_{\text{vote}} = \left\{ \mathbf{p}_i \;\middle|\; \frac{1}{|\mathcal{F}|} \sum_{f \in \mathcal{F}} \mathbf{1}[\mathbf{p}_i \in M_f] \geq 0.5 \right\}, \tag{3}$$

where $\mathcal{F}$ is the set of frames and M_f is the segmentation mask for frame f. Next, we perform KNN outlier removal. For each point in $\mathcal{P}_{\text{vote}}$, we compute $\bar{d}_i$, its mean Euclidean distance to its k=6 nearest neighbors. We then retain points with

$$\mathcal{P}_{\text{final}} = \left\{ \mathbf{p}_i \in \mathcal{P}_{\text{vote}} \;\middle|\; \bar{d}_i \leq 3\,\tilde{d} \right\}, \tag{4}$$

where $\tilde{d}$ is the median of $\{\bar{d}_i\}$ over all points. This removes isolated outliers from repeated oversegmentations while preserving the main object surface.

C Baseline Details

Video-3D LLM. We use the spatial grounding inference protocol described in Zheng et al. (2025), which accepts scene bounding boxes. To rule out box quality as a source of the observed low performance, we provide ground truth ADT bounding boxes as the scene bounding boxes.

CuTR+Tools. We attempted to compare with MM-Spatial (Daxberger et al., 2025), which uses RGB-D and pose with multiview images as input for spatial question answering. As of the time of experimenting, their code release does not include model weights. We therefore compare with CuTR, which MM-Spatial is built on top of, and construct our "CuTR+Tools" baseline by combining CuTR with our tool calling framework. In this setup, CuTR acts as the perception engine, replacing our depth-lifted point cloud and SAM3D representation. We run CuTR on every video frame at 3FPS, choosing the highest-confidence bounding box that overlaps with a SAM3 segmentation as the object representation. This box is used to compute length, height, position, and volume. This experiment provides a direct comparison of our R3D perception system with a state-of-the-art depth-based detection system, while keeping the tool interface fixed.

SpatialRGPT. Because few works study reasoning on RGB-D video, we adapt this image-based spatial understanding work to operate on videos. We run SpatialRGPT (Cheng et al., 2024) on every frame, then choose the median (or mode for multiple-choice questions) of the output distribution as the response. For fair comparison, we feed SpatialRGPT ground truth depth maps instead of using monocular depth estimation.

D Reconstruction Quality

Table 4: Chamfer distance (cm^2) between 3D representations and ground truth meshes, reported as 5th, 50th, and 95th percentiles over all objects used in the "volume" question type. Gray row : oracle using ground-truth segmentation masks.

	CD: Pred→GT			CD: GT→Pred			CD (Full)		
	P5	P50	P95	P5	P50	P95	P5	P50	P95
R3D Pointcloud	0.14	1.53	16.1	0.37	2.93	28.5	0.58	4.66	42.9
R3D Aniso	0.17	4.46	175.6	0.19	2.17	24.8	0.37	7.59	198.5
R3D Iso	**0.12**	**0.89**	**14.5**	**0.15**	**0.88**	**18.2**	**0.28**	**1.86**	**30.6**
R3D Iso (GT Seg)	0.12	0.60	7.0	0.15	0.68	6.5	0.29	1.33	13.0

Here we examine the quality of the 3D representations described in Section 5.3 using Chamfer distance (CD). Given a predicted point set $\mathcal{P}$ and a ground truth point set $\mathcal{G}$, we compute two directional distances:

$$\text{CD}_{\text{P}\to\text{G}} = \frac{1}{|\mathcal{P}|} \sum_{\mathbf{p}\in\mathcal{P}} \min_{\mathbf{g}\in\mathcal{G}} \|\mathbf{p}-\mathbf{g}\|^2, \tag{5}$$

$$\text{CD}_{\text{G}\to\text{P}} = \frac{1}{|\mathcal{G}|} \sum_{\mathbf{g}\in\mathcal{G}} \min_{\mathbf{p}\in\mathcal{P}} \|\mathbf{g}-\mathbf{p}\|^2. \tag{6}$$

$\text{CD}_{\text{P}\to\text{G}}$ measures *precision* (how close predicted points are to the true surface), while $\text{CD}_{\text{G}\to\text{P}}$ measures *completeness* (how much of the true surface is covered). The full CD is their sum. We

compute CD between each representation and the ground truth mesh for all objects used in the "volume" question type by sampling 5000 points from ground truth and predicted meshes. Table 4 shows the results. "R3D Iso" refers to our method, and "R3D Aniso" refers to an ablation in which each bounding box axis is scaled independently:

$$s_i = \frac{d_i^{\text{bbox}}}{d_i^{\text{mesh}}}, \quad i \in \{1, 2, 3\}, \tag{7}$$

where d_i^{bbox} and d_i^{mesh} are the extents of the predicted bounding box and mesh bounding box along each principal axis, respectively. R3D Iso (GT Seg) refers to an ablation in which ground truth segmentations are used as the input to SAM3D instead of SAM3 masks.

The R3D Iso mesh achieves the strongest results among methods that don't use ground truth masks. Compared to the point cloud, R3D Iso achieves much stronger completeness ($\text{CD}_{\text{G}\to\text{P}}$), filling in surface regions unseen by the point cloud. R3D Iso (GT Seg) obtains the strongest results, but for fair comparison, we do not allow R3D to ingest ground truth masks in our main results in Table 2.

E Inference Time

Table 5: Per-question runtime comparison, averaged over R3D-Bench queries. Benchmarked on a single H100 GPU and an AMD EPYC 9004 CPU.

	Avg Preproc. Time (s)				Model	Total
	SAM3	CuTR	Lift	SAM3D	Latency (s)	(s)
Video-3D LLM	52.2	22.0	—	—	0.8	75.0
SpatialRGPT	52.2	—	—	—	3.1	55.3
CuTR+Tools (8B)	52.2	22.0	—	—	2.0	76.2
CuTR+Tools (35B)	52.2	22.0	—	—	5.7	79.9
R3D (8B)	52.2	—	1.1	4.8	2.0	60.1
R3D (35B)	52.2	—	1.1	4.8	5.7	63.8

We analyze the runtime performance of our method and baselines in Table 5. Benchmarking is done on a single H100 GPU and an AMD EPYC 9004 CPU. The average processing times for each query from R3D-Bench are shown. All methods rely on SAM3 to provide object tracks. Video-3D LLM and CuTR+Tools additionally use 3D bounding box candidates (which are then matched with SAM3 object identities). For fair comparison, we include the CuTR box generation time as part of Video-3D LLM inference time even though our evaluations in Table 2 used ground truth 3D ADT boxes in an effort to improve accuracy. Each system component is benchmarked separately, resulting in identical values in all columns except Model Latency.

The preprocessing times shown represent latency that can be hidden by running these systems on the scene's objects before the user's query arrives. Model latency represents actual perceived user latency. We find that preprocessing times are dominated by SAM3. Our method can achieve a competitive model latency with Video-3D LLM and SpatialRGPT by using a similar sized model (Qwen3 8B), but latency is still higher due to multiple rounds of tool calling rather than one-shot Q&A. For our larger and more accurate variants, accuracy increases at the expense of latency.