Describe Anything Anywhere At Any Moment

Nicolas Gorlo[1] Lukas Schmid[1,2] Luca Carlone[1]
[1]Massachusetts Institute of Technology, [2]University of Technology Nuremberg
ngorlo@mit.edu, lukas.schmid@utn.de, lcarlone@mit.edu

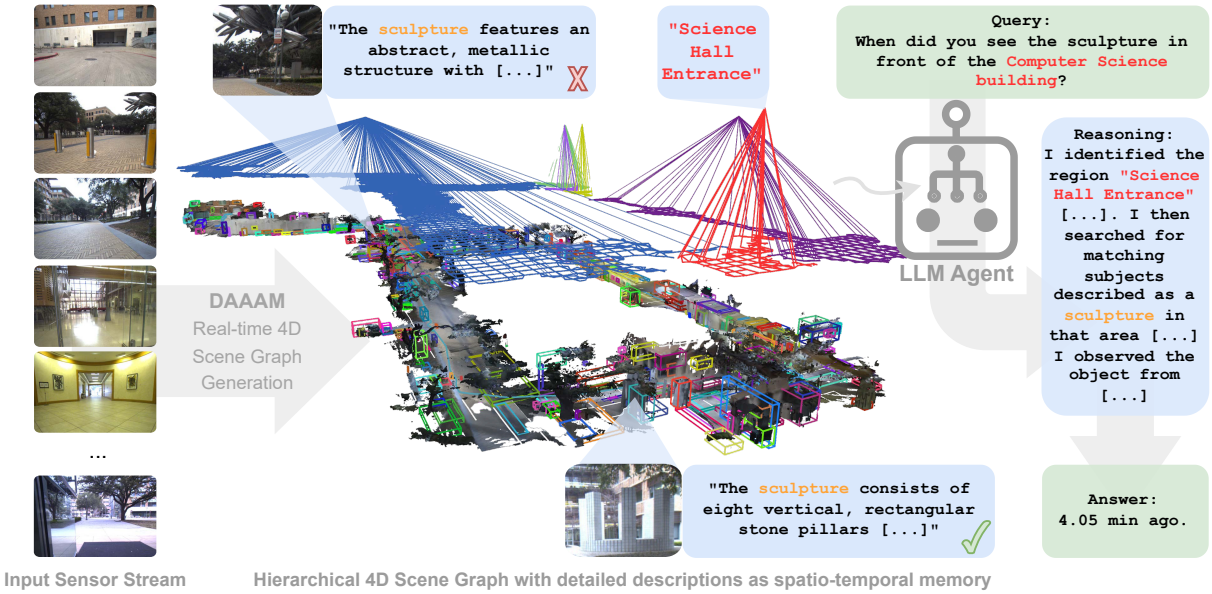

Figure 1. We present ***D**escribe **A**nything, **A**nywhere, at **A**ny **M**oment (DAAAM)*, a real-time, large-scale, spatio-temporal memory for embodied question answering and 4D reasoning. Given RGB-D sensor input *DAAAM* incrementally constructs a hierarchical 4D scene graph with highly detailed annotations that acts as an effective and scalable spatio-temporal memory representation for LLM Agents.

Abstract

Computer vision and robotics applications ranging from augmented reality to robot autonomy in large-scale environments require spatio-temporal memory frameworks that capture both geometric structure for accurate language-grounding and semantic detail. Existing methods face a tradeoff, where producing rich open-vocabulary descriptions comes at the expense of real-time performance when these descriptions have to be grounded in 3D. To address these challenges, we propose Describe Anything, Anywhere, at Any Moment (DAAAM)*, a novel spatio-temporal memory framework for large-scale and real-time 4D scene understanding.* DAAAM *introduces a novel optimization-based frontend to infer detailed semantic descriptions from localized captioning models, such as the Describe Anything Model (DAM), using batch processing to speed up inference by an order of magnitude for online deployment. It leverages such semantic understanding to build a hierarchical 4D scene graph (SG), which acts as an effective globally spatially and temporally consistent memory representation.* DAAAM *constructs 4D SGs with detailed, geometrically grounded descriptions while maintaining real-time performance. We show that* DAAAM*'s 4D SG interfaces well with a tool-calling agent for inference and reasoning. We thoroughly evaluate* DAAAM *in the complex task of spatio-temporal question answering (SQA) on the NaVQA benchmark and show its generalization capabilities for sequential task grounding on the SG3D benchmark. We further curate an extended OC-NaVQA benchmark for large-scale and long-time evaluations.* DAAAM *achieves state-of-the-art results in both tasks, improving OC-NaVQA question accuracy by 53.6%, reducing position errors by 21.9% and temporal errors by 21.6%, and improving SG3D task grounding accuracy by 27.8% over the most competitive baselines. We release our data and code open-source.*

1. Introduction

The ability to understand, reason about, and interact with complex, large-scale environments over long time horizons is a crucial challenge in computer vision and a prerequisite to a range of applications in robotics and augmented reality. In these applications, perception systems should be able to answer spatio-temporal queries; for instance, a robot operating on a factory floor should be able to answer questions like "where and when did you last see the red screwdriver?" or "can you go and grab the component we assembled last week?" However, this requires internal memory representations that i) support spatial reasoning and task planning, ii) extend over long time horizons and large environments, and iii) can be built in real-time with limited computation.

To this end, two main paradigms for spatio-temporal memory systems have emerged. First, metric-semantic maps ground objects geometrically in 3D reconstructions and semantically lift them [7, 11, 20, 23, 27, 43, 55, 63, 75, 76] to act as spatio-temporal memory. In particular, 3D scene graphs (SGs) [4, 22, 49, 50, 57] have found widespread interest due to their ability to capture semantic entities and their relationships in a topological graph, grounded in the 3D world. However, the need for as-expressive-as-possible scene descriptions is fundamentally at odds with the requirement for real-time, mobile computation. As a result, existing methods either lack semantic detail, using fast but closed vocabulary segmentation or embeddings [7, 20, 22, 43], or query large multimodal (MM) models on a per-object basis for highly detailed but very expensive open-vocabulary annotations [16, 29].

Alternatively, a second emerging paradigm leverages multimodal large language models (MM-LLMs) to generate detailed scene representations from natural language descriptions. Primarily, individual frames or video sequences are annotated by MM-LLMs [3, 61] and stored in a database for later retrieval [31]. This approach can lead to more expressive representations with notable performance for large scale visual question answering (VQA) [3, 61]. However, since annotations are stored by frame and not by content, they are often not sufficiently grounded in the 3D world and lack spatial and temporal consistency. For instance, since object observations are not associated and reconciled across frames, these models are typically unable to answer questions involving long-range spatial relationships or object quantities (*e.g.*, "count the number of chairs") [61].

To address these challenges, we propose *DAAAM: Describe Anything, Anywhere, At Any Moment*. Our approach, shown in Fig. 1, builds a hierarchical 4D scene graph (SG) as spatio-temporal memory representation by combining real-time 4D metric-semantic mapping with highly detailed natural-language descriptions for every observed entity. In particular, we introduce an optimization-based frontend to select key observations and infer detailed descriptions (*e.g.*, using DAM [32]) in batch, speeding up inference time by an order of magnitude for online deployment of large models. Our backend then globally optimizes and reconciles the object observations (or *fragments*) created in the frontend, yielding a spatially and temporally consistent 4D SG with description histories for each entity as memory representation. Finally, a tool-calling agent can efficiently leverage the proposed representation for queries. We demonstrate *DAAAM* in the complex setting of large-scale spatio-temporal question answering (SQA) on the NaVQA benchmark [3] and further show its generalization capabilities in sequential task grounding on the SG3D dataset [72], achieving state-of-the-art results in both cases. We make the following contributions:

- We introduce *DAAAM: Describe Anything, Anywhere, At Any Moment*, a novel real-time approach to create a 4D SG, an explicit large-scale spatio-temporal memory representation with highly detailed annotations.
- We formulate the task of selecting frames and masks to annotate with large localized captioning models in batch efficiently and online as an optimization problem, enabling real-time deployment of 3B-parameter models.
- We thoroughly evaluate *DAAAM*, achieving state-of-the-art results in spatio-temporal question answering (SQA) and sequential task grounding. We further curate an extended SQA benchmark and release our data and implementation open-source.

2. Related Works

Robotic agents need spatial memories that are simultaneously geometrically precise for manipulation, semantically rich for arbitrary natural language instructions, and computationally efficient for real-time operation. Existing approaches typically excel in only one of these dimensions.

Metric-Semantic Map-based Spatial Memory Systems. Various approaches embed foundation model features [42, 44, 69] into different 3D representations: point clouds [7, 11, 20, 23, 25, 43], Gaussian Splats [75, 76], radiance fields [27, 55] or TSDF volumes [63]. These approaches often suffer from limited expressiveness, large memory footprint, and inefficient queries (*e.g.*, searching unstructured feature fields in large-scale environments).

Alternatively, 3D SGs [4, 50, 51, 57] augment geometric mapping with semantic information and hierarchical structure. This was extended to real-time systems [22, 60] for 3D SG construction. While these achieve real-time performance, they often rely on closed-vocabulary semantics that limit expressiveness for complex robotic memory queries. Extensions include temporal changes [15, 37, 54], task-grounding [6, 39, 40], task-specific visual memory [52], and functional relationships [29, 71].

An adjacent line of research achieves richer semantic understanding by directly querying MM-LLMs per ob-

ject [16, 29], generating detailed descriptions. While semantically rich, these approaches require expensive per-object VLM queries, preventing real-time use.

Finally, capturing temporal dynamics is important, albeit very challenging for spatial representations. Some systems address this through episodic scene graphs linked across time [14], spatio-temporal metric-semantic SLAM handling short- and long-term changes [54], or dynamic point cloud memory with open-vocabulary features [34].

View-based Spatial Memory Systems. View-based representations directly annotate camera frames with MM-LLMs, sacrificing 3D structure for higher semantic detail. Retrieval-Augmented-Generation (RAG) [31] approaches store VLM-annotations of frames or short video segments in vector databases [3] or hierarchical semantic forests [61], enabling natural language queries over long horizons. 3D-Mem [64] retrieves observed frames directly, hoping multi-modal LLMs can infer 3D structure from individual views without explicit geometric grounding. Navigation systems ground instructions as textual landmarks in a sequence of per-frame observations [56], use tokenized frame observations [74] to encode observation history, or use segment-level representations [13] as a topo-semantic map. Hybrid approaches aim to combine view-based flexibility with 3D structure [19, 36] in indoor environments.

These methods offer detailed view-annotations, but their per-frame annotations are not sufficiently grounded in 3D space. Without metric grounding, they struggle with precise spatial reasoning, which is critical, e.g., for manipulation. Lightweight topological approaches [13, 67] trade semantic richness for efficiency while others require expensive annotations [36]. In contrast, our approach provides semantic detail alongside geometric grounding and computational efficiency.

VLMs for Spatial Understanding and Task Execution. Recent works expand spatial understanding capabilities of VLMs and LLMs [38]. Models can reason spatially by training on scene graphs [8, 10], by using 3D-aware architectures [18, 24, 62], or by training on large scale video data with spatial annotations [12]. While these models demonstrate spatial reasoning capabilities, reasoning over large-scale dynamic 3D environments and temporal changes remains challenging.

For task grounding and execution, systems use scene graphs for hierarchical planning [46], grounding in robotic affordances [2], structured LLM interfaces [48], embodied question answering [52], and navigation with chain-of-thought reasoning [65, 66]. To this end, our contribution provides an efficient, but detailed and general spatial memory that can be used by embodied agents for complex reasoning and task grounding.

3. Approach

In the following, we describe *DAAAM: Describe Anything, Anywhere, at Any Moment*. Given RGB-D images and poses as input, our method constructs a 4D scene graph with detailed open-vocabulary semantic annotations in real-time. This 4D scene graph acts as a spatio-temporal metric-semantic memory for embodied agents. An overview of our pipeline is given in Fig. 2. *DAAAM* is composed of several modules: an active window (A), a parallel thread for semantic lifting of segmentation fragments (B&C), and modules to construct, maintain and hierarchically structure the large-scale spatio-temporal memory (D&E).

A) Active Window and Real-time SG Construction. We follow the approach of Khronos [54] to extract temporally-consistent fragments from the sensor stream in dynamic scenes. In particular, we augment Khronos' active window by first splitting each input frame $I_t^{\text{RGB-D}}$ at time t into segments $s_j^t \in \mathbb{R}^{H \times W}$ using Fast-SAM [73] and track them over time using Bot-Sort [1]. Each track creates an object fragment $o_j^{0 \dots T_j} \in \mathbb{R}^{H \times W \times T_j}$ of T_j observations. We then use Khronos to lift each fragment to 3D and reconstruct their shape and position (through time for dynamic objects). Since geometric segmentation, tracking, and reconstruction is fast, it is run at the sensor rate of 10Hz.

B) Prompt Frame Selection. Since extraction of highly-detailed descriptions is an expensive operation, we propose to process only selected frames and annotate fragments in batch. Specifically, we accumulate fragment observations in consecutive time windows $w_t = [t_{start}, t_{start+m}]$. After each window, we propose to select frames where each fragment is visible and well-positioned for semantic grounding by the vision language model (VLM). We formulate the frame selection as a 2-step optimization problem.

Let $\mathcal{O} = \{o_1^w, \dots, o_m^w\}$ denote the set of tracked object fragments within window w, where each o_j^w represents a temporally consistent track across multiple frames. For each frame $f_i \in \mathcal{F}^w$ and $o_j^w \in \mathcal{O}$, we define a visibility indicator $v_{ij} \in \{0, 1\}$, $v_{ij} := 1$ iff object o_j^w is visible in frame f_i, and a view quality score $q_{ij} \in [0, 1]$ for object o_j^w in frame f_i.

Our selection problem has to balance two competing objectives: minimizing the number of frames sent to the VLM (for computational efficiency) while maximizing the quality of selected fragment-frame pairs (for annotation accuracy). To address this, we first solve the set cover problem to find the minimum number of frames $K^\star$ required to observe all object fragments o_j^w at least once:

$$\begin{aligned} K^\star = \min_{\mathcal{S} \subseteq \mathcal{F}^w} \quad & |\mathcal{S}| \\ \text{s.t.} \quad & \forall o_j^w \in \mathcal{O} : \exists f_i \in \mathcal{S} \text{ with } v_{ij} = 1 \end{aligned} \tag{1}$$

We solve (1) using a greedy algorithm. Given the minimum

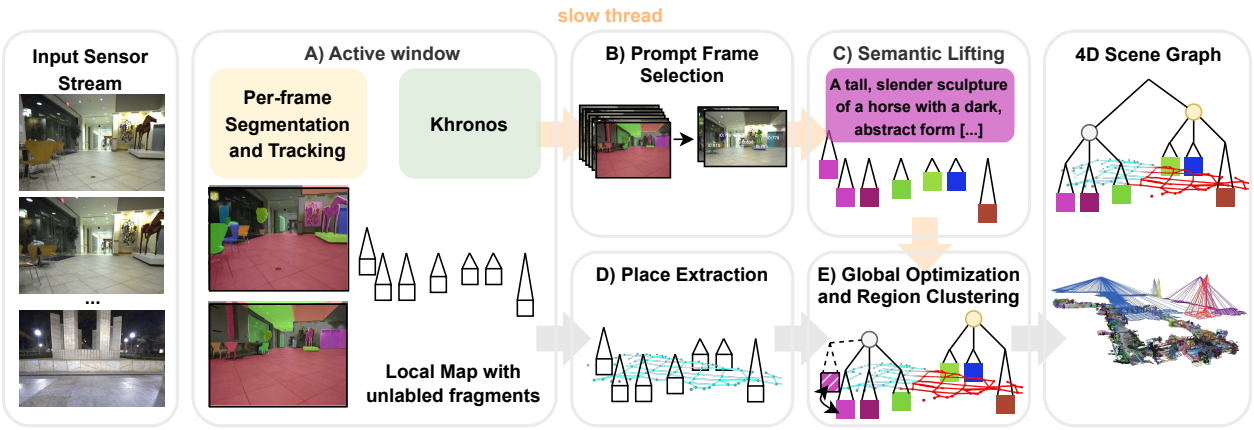

Figure 2. An overview of the proposed approach. Given an RGB-D video stream, we first segment the scene into fragments and track them over time in image space using a lightweight tracker [1, 5]. We perform metric-semantic mapping using Hydra [22] with the Khronos [54] frontend on the unlabeled segments to build a 4D map of the environment. To semantically lift the resulting map, we aggregate the tracked observations in parallel and select frames using an optimization-based frame selection algorithm. The selected frames and segments are batch-processed by the Describe Anything Model (DAM) [32] to generate detailed descriptions for each object. The generated descriptions are finally incorporated back into the map and a 4D scene graph is constructed and clustered into semantically informed regions.

frame count $K^\star$, we then solve the binary linear program:

$$\begin{aligned} \max_{x,y} \quad & \sum_{i=1}^{n}\sum_{j=1}^{m} q_{ij} \cdot y_{ij} \\ \text{s.t.} \quad & \sum_{i=1}^{n} x_i = K^\star + \epsilon, \quad \sum_{i=1}^{n} y_{ij} = 1, \\ & y_{ij} \le x_i,\ y_{ij} \le v_{ij},\ x_i \in \{0,1\},\ y_{ij} \in \{0,1\} \\ & \forall i \in [n],\ j \in [m], \end{aligned} \tag{2}$$

where $x_i \in \{0,1\}$ indicates selection of frame f_i, $y_{ij} \in \{0,1\}$ indicates assignment of fragment o_j^w to frame f_i, and ϵ is a slack parameter set to 1. The objective maximizes the quality score for each selected fragment, constrained by the conditions that i) the total number of selected frames is $(K^\star + \epsilon)$, ii) if a frame is not selected, no fragment is assigned to it, and iii) every fragment is assigned to exactly one frame where it is guaranteed to be visible.

While our formulation is applicable to any large model and suitable quality score q_{ij}, we design a heuristic q_{ij} that combines position and size components:

$$q_{ij} = \alpha \cdot q_{ij}^{\text{pos}} + (1-\alpha) \cdot q_{ij}^{\text{size}}. \tag{3}$$

The position score q_{ij}^{pos} is the entropy of normalized coordinates to favor centrally located objects, resulting in the maximum value when objects are centered and minimum at frame boundaries. The size score q_{ij}^{size} uses a hyperbolic tangent function that saturates for large objects while penalizing objects below a minimum area threshold A_{min}, ensuring that selected objects are sufficiently visible when being labeled. We use $\alpha = 0.5$.

C) Semantic Lifting. Given the image-fragment pairs from the frame selection algorithm, we batch selected images to annotate all fragments in a single pass of the Describe Anything Model (DAM) [32] to obtain a detailed natural-language description for each fragment. We extend DAM with a batch inference strategy that bundles multiple frames and masks into a single tensor, minimizing redundant computation and taking better advantage of parallelization. This allows *DAAAM* to run in real-time in real-world environments while still leveraging large, detailed models such as DAM. We further assign a CLIP [44] feature and a sentence embedding feature [41], which are similarly obtained in batch, to each fragment to aid semantic search, clustering, summarization, and reconciliation of repeatedly observed objects. Note that our frame selection algorithm naturally minimizes the number of frames passed to DAM while processing many masks per image, further improving the inference time of the batch-processing. In addition, each mask is situated as visible as possible in the image frame leading to high-quality labels.

D) Place Extraction. To capture detailed descriptions not only of objects but also of the background, inspired by [9, 17, 40], we further extract *place* nodes p_j in our SG. Following our approach for objects, we first extract geometric fragments in the active window. While other methods such as Voronoi diagrams [22] or sampling [53] are also admissible, we extract p_j based on ground *traversability* to capture relevant surfaces and topology. Our traversability-based places adapt the general hierarchical approach of [22, 47] to ground robots. Different traversability layers enable other embodiments and multi-floor environments. Volumetric 3D places would require a different strategy of semantic assignment (e.g., closest annotation). Traversability is estimated by convolving a robot bounding box with the local volumetric occupancy map maintained in Khronos' active window

and squashing along the Z-axis. We then tesselate this slice into places p_j by inscribing largest traversable rectangles. To ensure near uniform coverage, each rectangle is subject to a maximum size constraint of 2m.

For semantic lifting, each p_j is first projected to the groundand then projected to all frames that annotate the fragment covering it. The resulting descriptions and features are assigned by majority voting. While it may seem intuitive to use full-frame annotations instead of ground-fragment annotations to describe places p_j, we find that full-frame queries are often out-of-distribution (OOD) for DAM [32] and therefore use ground annotations. More details on place extraction in Appendix A.

E) Global Optimization and Region Clustering. To achieve a spatially globally consistent memory representation, we continuously optimize the positions of all nodes in our 4D SG in the backend using the factor graph formulation of [54]. For temporal consistency, object fragments and place nodes with similar geometry and descriptive features are merged in a reconciliation step. To retain temporal information, descriptions of merged objects are appended to form a history, where also the timestamps of the corresponding active window periods are retained.

Finally, we extract regions R_i as hierarchical abstractions by clustering the reconciled low-level SG. To this end, we first extract regions from the places graph by assigning edge-weights as the cosine distance of the respective semantic features and applying the most-stable-clique finding algorithm of Hydra [22]. Object nodes are assigned to the closest cluster. To obtain representative descriptions that summarize the content of regions, we use farthest point sampling of the features of all objects in a region, starting from the mean, and summarize them by prompting an LLM.

F) Retrieval-augmented Reasoning For inference, we use a tool-calling agent to answer natural language queries. The tool-calling agent has access to tools to a) retrieve objects based on semantic search b) retrieve information about the regions and c) retrieve information about the agent. Retrieved information includes spatial and temporal information about each retrieved 4D SG node. More detail about the LLM Agent as well as tool descriptions in Appendix B.

4. Experiments

We evaluate *DAAAM* on the challenging tasks of spatio-temporal question answering (SQA) as well as sequential task grounding to demonstrate its flexibility and generalization ability. We also ablate introduced components and provide run-time analysis for real-time use. In all evaluations, ground-truth poses are provided as input to all methods.

4.1. Spatio-Temporal Question Answering

Dataset. To assess *DAAAM*'s performance as an explicit large-scale spatio-temporal memory framework, we evaluate it on spatio-temporal question answering (SQA) in the CODa dataset [70], featuring large-scale indoor and outdoor scenes. To obtain depth images on the dataset, we use stereo depth estimation [58]. We adopt the QA-samples from the NaVQA [3] benchmark consisting of 210 samples of QA-pairs of different categories, including binary (yes/no) questions, spatial (position) questions, and temporal (time, duration) questions. The benchmark further distinguishes different sequence lengths of *Short*, *Medium*, and *Long*, corresponding to average memory durations of $1.2\,\mathrm{min}$, $4.4\,\mathrm{min}$, and $12.3\,\mathrm{min}$, respectively.

Baselines. We compare *DAAAM* against recent baselines, including several variations of the state-of-the-art spatio-temporal memory system *ReMEmbR* [3]. We further compare against a multi-frame VLM, representing a reasoning system without explicit memory, as well as ConceptGraphs [16], a recent open-set metric-semantic mapping-based method.

Results. We report SQA performance in Tab. 1. Our method shows strong performance even when compared to larger-sized models such as ReMEmbR+VILA1.5-13b [3, 33] and modern near-real-time methods such as ReMEmbR+NVILA-Lite-2B [35], especially for long sequences and temporal reasoning. This indicates that geometric structuring of spatio-temporal memory aids scene understanding and the 4D SG representation greatly improves temporal understanding. On the other hand, our method struggles with queries which are hard to infer from the limited set of tools (*e.g.*, "Was the robot driving on the left side of the sidewalk?"), albeit being encoded in the 4D SG, as well as queries that reference small objects. Note that the binary questions are skewed towards positive labels while LLMs generally predict conservatively, explaining the sub-0.5 question accuracy for some methods (as also observed in [3]).

Dataset Limitations. We find a number of limitations in the NaVQA benchmark that make comparisons tricky. First, we observed that for several methods in-context examples are provided to the LLM that also appear in the test set, confounding performance. These methods are marked † in Tab. 1, where we additionally evaluate them after removing the confounding examples. Second, ground truth annotations for spatial questions (e.g., "Where is the yellow police call pole?") in NaVQA are annotated as the position from which the object was observed instead of its actual 3D position, strongly favoring view-based memory systems like ReMEmbR. Although *DAAAM* is designed to predict actual object positions, we modified it to return the view position for the results in Tab. 1 to allow a fair comparison on the original NaVQA benchmark. Third, we find several annotations to be noisy or incorrect (in part leading to the weak performance for Medium spatial queries for *DAAAM*). For a dataset of 210 samples, this can greatly skew the re-

Table 1. Results on the original NaVQA benchmark [3]. Results of *ReMEmbR* [3] and multi-frame VLM are partially taken from [3], we further ran it with recent language models. Superscript † indicates confounding use of in-context examples in LLM prompts that also appear in the test set. Best (without in-context examples) is bold, best (with in-context examples) underlined if better than without. × indicates that inference was not possible, either due to insufficient context or due to not keeping track of observation times.

	Metric:	Descriptive Question Accuracy ↑				Positional Error [m] ↓				Temporal Error [min] ↓			
Method	Query Length	Short	Medium	Long	All	Short	Medium	Long	All	Short	Medium	Long	All
ReMEmbR [3]	NVILA-Lite 2B + GPT-5-mini	0.423	0.519	0.524	0.483	11.179	**23.877**	65.588	39.278	**0.313**	1.240	4.472	2.518
	NVILA-Lite-8B + GPT-5-mini	0.551	0.605	0.714	0.607	8.486	34.409	56.297	37.337	0.363	1.491	4.979	2.840
	NVILA-Lite 2B + GPT-5-mini †	0.577	0.556	0.643	0.582	8.535	25.030	49.259	31.536	0.272	1.248	4.975	2.743
	NVILA-Lite-8B + GPT-5-mini †	0.538	0.630	0.571	0.582	8.467	53.279	53.717	41.485	0.436	1.336	4.122	2.414
	VILA1.5-13b + GPT-4o †	0.62	0.58	0.65	0.61	5.1	27.5	46.25	29.96	0.3	1.8	3.6	2.3
	VILA1.5-13b + Llama3.1:8b †	0.31	0.33	0.21	0.30	159.9	151.2	165.3	159.9	9.5	7.9	18.7	13.3
Multi Frame VLM — GPT-4o †		0.55	×	×	×	7.5	×	×	×	0.5	×	×	×
Concept Graphs [16] — NVILA-Lite-8B + GPT-5-mini		0.385	0.296	0.143	0.299	85.67	126.35	158.67	130.03	×	×	×	×
DAAAM (Ours) — DAM-3B + GPT-5-mini		**0.654**	**0.630**	**0.786**	**0.672**	**7.282**	46.015	**42.116**	**33.89**	0.443	**1.030**	**2.538**	**1.591**

Table 2. Results on OC-NaVQA dataset. All models use GPT-5-mini to reason over the constructed memory. × indicates that method does not keep track of observation times.

Method	Question Accuracy ↑	Positional Error [m] ↓	Temporal Error [min] ↓
ReMEmbR - NVILA-Lite-2B [3, 35]	0.432	53.466	2.287
ReMEmbR - NVILA-Lite-8B [3, 35]	0.463	55.894	4.106
Concept-Graphs [16]	0.299	111.29	×
DAAAM (Ours)	**0.711**	**41.75**	**1.792**

sults. Further, we find that for 22 out of 210 samples the provided ground-truth observation time does not fall into the provided (*Short*/*Medium*/*Long*) context window. Finally, the ground truth time and position annotations are calculated from ReMEmbR's video query length, favoring the method, especially for short-horizon questions. To address these limitations, we re-annotate the spatial queries of the NaVQA dataset with the actual object positions and improve the accuracy of the labels with a custom 3D labeling tool. We further discard the observation windows and instead evaluate on the entire context of full sequences of the CODa dataset, reflecting more large-scale settings of up to 35.8 min. We call this the object-centric NaVQA (OC-NaVQA) dataset and release it publicly.

Results on OC-NaVQA. The results on OC-NaVQA are shown in Tab. 2. We observe that the 4D SG performance scales well to large-scale long-term settings of up to 35.8 min and 1.64 km of traveled distance, outperforming both metric-semantic map-based spatial memory systems [16] and view-based spatial memory systems [3]. At the scale of the benchmark, ConceptGraphs [16] struggles with memory limitations as it maintains a full point-cloud in memory. While ReMEmbR [3] scales better, we observe limitations in multi-view consistency and spatial reasoning in the large scale setting, leading to reduced performance.

4.2. Sequential Task Grounding

A vital but challenging task in robotics is grounding natural language instructions in the 3D environment. We evaluate our method in sequential task grounding on the SG3D [72] dataset. We follow the evaluation protocol of ASHiTA [6] (Table 2), evaluating on the HM3D [45] scenes and using the sequences provided in HOV-SG [59] to construct scene graphs. We compare *DAAAM* against the results reported by Chang *et al.* [6], showing sub-task and task accuracy (s-acc and t-acc, respectively) in Tab. 3. Task accuracy refers to a full task (composed of multiple subtasks) being correctly grounded in the environment.

We observe significant performance increases over Hydra [21], even when using ground-truth single-word semantic labels *Hydra(GT Seg)*, highlighting the importance of our highly detailed descriptions. DAAAM further achieves superior results to ASHiTA, a specialized method for hierarchical task analysis, highlighting the generalization ability and flexibility of our 4D SG memory: it represents semantic information in great detail while still providing accurate spatial grounding. It is worth pointing out that SG3D is a semi-synthetic dataset based on HM3D, leading to a real-to-sim gap resulting in a small loss of accuracy for the Describe Anything Model, which is only trained on real data.

4.3. Ablation Studies

Language-Augmented Localized Image-Text Retrieval. While our explicit descriptions generated using DAM aid in providing context and detail to reasoning LLMs as well as human interpretability, their use for pure retrieval tasks is less clear: One can use a sentence-embedding model to cre-

Table 3. Evaluation of Sequential Task Grounding on the SG3D [72] benchmark. Our method outperforms metric-semantic map-based spatial memory systems [22, 59] and performs competitively with specialized methods [6]. Results for top 4 methods are taken from Chang et al. [6].

Method	s-acc [%] ↑	t-acc [%] ↑
Hydra [22] + GPT	8.18	2.44
Hydra [22] (GT Seg) + GPT	14.2	6.34
HOV-SG [59]	8.98	1.95
ASHiTA [6]	21.7	8.78
DAAAM (Ours) + GPT	**22.16**	**11.22**

Table 4. Retrieval of localized object descriptions on refCOCOg [26, 68] and visual Genome [30] datasets.

Data	Method	Accuracy [%] ↑	Top-1	Top-5	Top-10
refCOCOg	CLIP ViT-L/14 [44]		19.59	41.12	52.51
	DAM-3B [32] w/ Sentence-T5-xl [41]		18.07	39.53	51.18
	CLIP + DAM w/ Sentence (Ours)		**25.11**	**50.10**	**62.96**
vis. Genome	CLIP ViT-L/14 [44]		21.64	40.04	48.4
	DAM-3B [32] w/ Sentence-T5-xl [41]		18.34	38.22	48.72
	CLIP + DAM w/ Sentence (Ours)		**24.92**	**47.72**	**57.76**

ate embedding vectors from the natural language descriptions. This invariably introduces an information bottleneck when comparing to jointly learned embedding spaces produced by VLMs like CLIP [44]. However, we find some of the retained information to be orthogonal to the information captured by contrastive VLMs and thus concatenate both features in our method.

To test this hypothesis, we measure the accuracy of retrieving a localized subject on an image based on the ground-truth description against random subsets of 2500 localized subjects. We take five subsets from the Visual Genome dataset [30] and use the entire validation set of the refCOCOg dataset [26, 68] as a 6th subset (2573 samples). Note that for DAM, we run SAM [28] to turn the bounding-box annotations into segments, potentially introducing additional error unrelated to the retrieval performance. The top-K retrieval accuracy is shown in Tab. 4.

As expected, the retrieval performance of DAM descriptions encoded into a sentence embedding falls short of using a CLIP model for retrieving localized image information. However, concatenating CLIP with sentence embeddings outperforms CLIP alone, indicating that the two methods capture complementary information. Consequently, explicit subject descriptions still aid retrieval tasks, despite the information bottleneck introduced by the text transformation.

A Case for Explicit Object Descriptions. An advantage of our method is the added interpretability, as all nodes of the hierarchical 4D SG are annotated with open-vocabulary natural-language labels. This not only aids in human understanding of the spatio-temporal memory, but also helps LLM agents. In Tab. 5, we compare against a baseline "`w/o DAM descriptions`" that, instead of DAM descriptions, only uses visual features and prompts image crops of observed fragments to the MM-LLM agent. The results suggest that explicit descriptions aid compositional reasoning of the retrieval-augmented agent, especially for positional and temporal queries. When answering binary questions, images appear to provide a better proxy for verification, resulting in superior performance. Binary questions benefit from direct visual verification via image crops prompted to the LLM agent. Spatial and temporal queries require reasoning over potentially many explicit descriptions and timestamps. Here, DAM helps with concise textual descriptions, explaining the 16.6% and 25.4% reductions in positional and temporal error. We note that our method was designed to run *any* comparatively large model for semantic lifting in parallel. Thus, if no explicit descriptions are necessary, a large image embedding model improving retrieval accuracy of downstream RAG systems can also be deployed.

Table 5. Ablation study of *DAAAM* on OC-NaVQA.

	Question Accuracy ↑	Positional Error [m] ↓	Temporal Error [min] ↓
DAAAM + GPT-5-mini	0.711	**41.75**	1.792
w/o DAM descriptions	**0.776**	50.05	2.396
w/o region clustering	0.707	48.93	3.576
w/o frame selection quality heuristic	0.627	49.92	**1.678**

Region Clustering. To analyze the benefit of our hierarchical region clustering in 4D SGs, Tab. 5 compares against a "`w/o region clustering`" version. We observe that *DAAAM* benefits from region clustering for all metrics. Especially for temporal queries, often focusing on large information context (e.g., "how much time did you spend inside"), the hierarchical structure of our 4D SG memory appears to facilitate accurate SQA.

Frame Selection Quality Score. "`w/o frame selection quality heuristic`" in Tab. 5 highlights the benefit of using the quality heuristic in our optimization-based frame-selection algorithm, improving spatial and QA accuracy and performing comparably in terms of temporal accuracy. Since temporal queries in the NaVQA benchmark often focus on larger subjects, the quality heuristic may be of lesser importance here.

4.4. Runtime Analysis

Since real-time computation is essential for robotics and VR, we evaluate the run-time of DAAAM as an overall system, inference speed-up of our batching strategy, as well as the resulting latency. All experiments are performed on

Table 6. Average frame rate of overall systems for spatio-temporal memory creation [Hz].

DAAAM (ours)	Concept-Graphs [16]	ReMEmbR [3] NVILA-Lite-2B [35]	NVILA-Lite-8B [35]
11.6	0.075	4.9	4.6

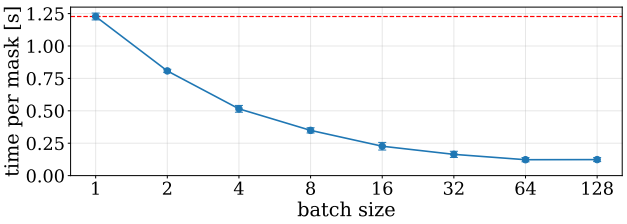

Figure 3. Speedup of DAM [32] inference via batching. Baseline (batch size = 1) dashed red, batch processing solid blue.

CODa [70] sequences on a single NVIDIA RTX 5090 GPU.

***DAAAM* is a Real-Time System.** The overall frame rate of our method is compared to several baselines in Tab. 6. We observe that, due to *DAAAM*'s efficient architecture and its threaded batch-inference of large models, it can run at the sensor rate of 10 Hz in CODa [70] even when using comparatively large models such as DAM. We find the main bottleneck of our method to be input segmentation and tracking, whereas the parallel thread for semantic annotation only becomes a bottleneck for very cluttered or fast-moving scenes with many fragments to annotate. Our mean annotation time per fragment is 0.18 ± 0.03s, thus 5 new fragments can be annotated per second by a single worker. In practice, we find that a single worker is sufficient for the real-time workload of a mobile ground robot in the wild.

In contrast, ConceptGraphs [16] is not real-time deployable at the scale of the dataset. ReMEmbR would require down-sampling to a lower frame rate (than the 10 Hz of the CODa dataset), potentially leading to a loss of accuracy.

Inference Speed Up through Batching. The inference time of DAM for different batch-sizes is shown in Fig. 3. Using our frame selection strategy, we observe notable inference speed-ups. In our experiments, we use a batch-size between 48-128 (depending on the frame selection outcome), leading to a speed-up by an order of magnitude.

Worker Latency. We time the frame selection latency at 1.2 ± 0.74s and semantic lifting latency at 9.2 ± 1.4s, both measuring a full batch. The increased latency is an inevitable side-effect of running large models, which delays detailed reasoning about observations by around 10s. Nonetheless, our experiments demonstrate the benefit of highly detailed memory for robotics and augmented reality, where for most large-scale long-horizon decision making the immediate past may be less important as long as all other observations are accurately summarized. This throughput-over-latency trade-off suits applications querying past observations (warehouse inventory, surveillance review). Real-time interaction (e.g., cooking assistance) might need lower-latency pipelines. Still, the geometric information about all fragments and the background is always maintained in real-time.

5. Limitations

By the standards of modern large VLMs, the training corpus used to train DAM is relatively modest (1.5M samples) [32]. Consequently, generated descriptions sometimes fail to capture out-of-distribution objects or uncommon visual features and can hallucinate towards the mean (*e.g.*, predicting elevator doors with handles). However, as multimodal LLMs are rapidly evolving, we expect that future detailed localized description models will be able to generate more accurate descriptions and integrate well into DAAAM.

Second, as shown in Sec. 4.4, using DAM an average of 5.2 new fragments can be annotated per second by a single worker on a Desktop GPU. While this suffices for a mobile ground robot, it may be too slow for a dynamic aerial robot or VR headset. We note that, since DAAAM aims to run "comparatively large models" on a thread with higher latency, smaller models can be run on smaller hardware (thus still comparatively large) or at higher throughput, enabling even aerial application.

Third, our tracking [1, 5] assumes object identity preservation. State transformations (e.g., cutting) break associations, creating new tracks without links to source objects.

Finally, although we generate a single spatially and temporally consistent 4D SG by merging nodes, a history of all descriptions is maintained in dynamic nodes which may not scale indefinitely. Future work should investigate summarization strategies to keep memory size bounded.

6. Conclusion

We presented *DAAAM*, a novel spatio-temporal memory framework that combines detailed semantic descriptions with geometric grounding for large-scale environments. By decoupling geometric tracking from semantic annotation through optimization-based frame selection and batch inference, *DAAAM* overcomes computational constraints of comparatively large vision annotation models. This enables the real-time construction of hierarchical 4D SGs with highly detailed natural language descriptions. Our approach demonstrates substantial improvements over recent methods, achieving state-of-the art results in spatio-temporal question answering and sequential task grounding. With real-time performance at 10Hz and scalability to sequences exceeding 35 min and 1.5 km distance, *DAAAM* provides a practical foundation for embodied agents to understand and interact with complex, large-scale, dynamic environments over an extended time horizon. We release our data and code open-source.

References

[1] Nir Aharon, Roy Orfaig, and Ben-Zion Bobrovsky. Bot-sort: Robust associations multi-pedestrian tracking. *arXiv preprint arXiv:2206.14651*, 2022. 3, 4, 8

[2] Michael Ahn, Anthony Brohan, Noah Brown, Yevgen Chebotar, Omar Cortes, Byron David, Chelsea Finn, Chuyuan Fu, Keerthana Gopalakrishnan, Karol Hausman, Alex Herzog, Daniel Ho, Jasmine Hsu, Julian Ibarz, Brian Ichter, Alex Irpan, Eric Jang, Rosario Jauregui Ruano, Kyle Jeffrey, Sally Jesmonth, Nikhil Joshi, Ryan Julian, Dmitry Kalashnikov, Yuheng Kuang, Kuang-Huei Lee, Sergey Levine, Yao Lu, Linda Luu, Carolina Parada, Peter Pastor, Jornell Quiambao, Kanishka Rao, Jarek Rettinghouse, Diego Reyes, Pierre Sermanet, Nicolas Sievers, Clayton Tan, Alexander Toshev, Vincent Vanhoucke, Fei Xia, Ted Xiao, Peng Xu, Sichun Xu, Mengyuan Yan, and Andy Zeng. Do as i can and not as i say: Grounding language in robotic affordances. In *arXiv preprint arXiv:2204.01691*, 2022. 3

[3] A. Anwar, J. Welsh, J. Biswas, S. Pouya, and Y. Chang. ReMEmbR: Building and reasoning over long-horizon spatio-temporal memory for robot navigation. In *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, 2025. 2, 3, 5, 6, 8

[4] I. Armeni, Z. He, J. Gwak, A. Zamir, M. Fischer, J. Malik, and S. Savarese. 3D scene graph: A structure for unified semantics, 3D space, and camera. In *Intl. Conf. on Computer Vision (ICCV)*, pages 5664–5673, 2019. 2

[5] Mikel Broström. Boxmot: pluggable state-of-the-art multi-object tracking modules, 2023. 4, 8

[6] Y. Chang, L. Fermoselle, D. Ta, B. Bucher, L. Carlone, and J. Wang. ASHiTA: Automatic scene-grounded hierarchical task analysis. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2025. 2, 6, 7

[7] Boyuan Chen, Fei Xia, Brian Ichter, Kanishka Rao, Keerthana Gopalakrishnan, Michael S Ryoo, Austin Stone, and Daniel Kappler. Open-vocabulary queryable scene representations for real world planning. In *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, pages 11509–11522. IEEE, 2023. 2

[8] B. Chen, Z. Xu, S. Kirmani, B. Ichter, D. Sadigh, L. Guibas, and F. Xia. SpatialVLM: Endowing vision-language models with spatial reasoning capabilities. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 14455–14465, 2024. 3

[9] W. Chen, S. Hu, R. Talak, and L. Carlone. Leveraging large (visual) language models for robot 3D scene understanding. *arXiv preprint: 2209.05629*, 2022. 4

[10] An-Chieh Cheng, Hongxu Yin, Yang Fu, Qiushan Guo, Ruihan Yang, Jan Kautz, Xiaolong Wang, and Sifei Liu. Spatial-rgpt: Grounded spatial reasoning in vision-language models. *Advances in Neural Information Processing Systems (NeurIPS)*, 37:135062–135093, 2024. 3

[11] Runyu Ding, Jihan Yang, Chuhui Xue, Wenqing Zhang, Song Bai, and Xiaojuan Qi. Pla: Language-driven open-vocabulary 3d scene understanding. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 7010–7019, 2023. 2

[12] Kaituo Feng, Kaixiong Gong, Bohao Li, Zonghao Guo, Yibing Wang, Tianshuo Peng, Junfei Wu, Xiaoying Zhang, Benyou Wang, and Xiangyu Yue. Video-r1: Reinforcing video reasoning in mllms. *arXiv preprint arXiv:2503.21776*, 2025. 3

[13] Sourav Garg, Krishan Rana, Mehdi Hosseinzadeh, Lachlan Mares, Niko Sünderhauf, Feras Dayoub, and Ian Reid. Robohop: Segment-based topological map representation for open-world visual navigation. In *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, pages 4090–4097, 2024. 3

[14] Muhammad Fadhil Ginting, Dong-Ki Kim, Xiangyun Meng, Andrzej Marek Reinke, Bandi Jai Krishna, Navid Kayhani, Oriana Peltzer, David Fan, Amirreza Shaban, Sung-Kyun Kim, Mykel Kochenderfer, Ali akbar Agha-mohammadi, and Shayegan Omidshafiei. Enter the mind palace: Reasoning and planning for long-term active embodied question answering. In *Conference on Robot Learning (CoRL)*, 2025. 3

[15] N. Gorlo, L. Schmid, and L. Carlone. Long-term human trajectory prediction using 3D dynamic scene graphs. *IEEE Robotics and Automation Letters (RA-L)*, 2024. 2

[16] Qiao Gu, Alihusein Kuwajerwala, Sacha Morin, Krishna Murthy Jatavallabhula, Bipasha Sen, Aditya Agarwal, Corban Rivera, William Paul, Kirsty Ellis, Rama Chellappa, Chuang Gan, Celso Miguel de Melo, Joshua B. Tenenbaum, Antonio Torralba, Florian Shkurti, and Liam Paull. Conceptgraphs: Open-vocabulary 3d scene graphs for perception and planning. In *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, 2024. 2, 3, 5, 6, 8

[17] Daniel Honerkamp, Martin Büchner, Fabien Despinoy, Tim Welschehold, and Abhinav Valada. Language-grounded dynamic scene graphs for interactive object search with mobile manipulation. *IEEE Robotics and Automation Letters*, 2024. 4

[18] 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 (NeurIPS)*, 2023. 3

[19] Wenbo Hu, Yining Hong, Yanjun Wang, Leison Gao, Zibu Wei, Xingcheng Yao, Nanyun Peng, Yonatan Bitton, Idan Szpektor, and Kai-Wei Chang. 3dllm-mem: Long-term spatial-temporal memory for embodied 3d large language model. *arXiv preprint arXiv:2505.22657*, 2025. 3

[20] Chenguang Huang, Oier Mees, Andy Zeng, and Wolfram Burgard. Multimodal spatial language maps for robot navigation and manipulation. *Intl. J. of Robotics Research*, 2025. 2

[21] N. Hughes, Y. Chang, and L. Carlone. Hydra: a real-time spatial perception engine for 3D scene graph construction and optimization. In *Robotics: Science and Systems (RSS)*, 2022. 6

[22] N. Hughes, Y. Chang, S. Hu, R. Talak, R. Abdulhai, J. Strader, and L. Carlone. Foundations of spatial perception for robotics: Hierarchical representations and real-time systems. *Intl. J. of Robotics Research*, 2024. 2, 4, 5, 7

[23] Krishna Murthy Jatavallabhula, Alihusein Kuwajerwala, Qiao Gu, Mohd Omama, Tao Chen, Shuang Li, Ganesh Iyer,

Soroush Saryazdi, Nikhil Keetha, Ayush Tewari, Joshua B. Tenenbaum, Celso Miguel de Melo, Madhava Krishna, Liam Paull, Florian Shkurti, and Antonio Torralba. Conceptfusion: Open-set multimodal 3d mapping. In *Robotics: Science and Systems (RSS)*, 2023. 2

[24] Baoxiong Jia, Yixin Chen, Huangyue Yu, Yan Wang, Xuesong Niu, Tengyu Liu, Qing Li, and Siyuan Huang. Sceneverse: Scaling 3d vision-language learning for grounded scene understanding. In *European Conf. on Computer Vision (ECCV)*, 2024. 3

[25] Christina Kassab, Matías Mattamala, Sacha Morin, Martin Büchner, Abhinav Valada, Liam Paull, and Maurice Fallon. The bare necessities: Designing simple, effective open-vocabulary scene graphs. *arXiv preprint arXiv:2412.01539*, 2024. 2

[26] Sahar Kazemzadeh, Vicente Ordonez, Mark Matten, and Tamara Berg. Referitgame: Referring to objects in photographs of natural scenes. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 787–798, 2014. 7

[27] J. Kerr, C.M. Kim, K. Goldberg, A. Kanazawa, and M. Tancik. LERF: Language embedded radiance fields. In *iccv*, 2023. 2

[28] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollar, and Ross Girshick. Segment anything. In *Intl. Conf. on Computer Vision (ICCV)*, pages 4015–4026, 2023. 7

[29] Sebastian Koch, Johanna Wald, Mirco Colosi, Narunas Vaskevicius, Pedro Hermosilla, Federico Tombari, and Timo Ropinski. Relationfield: Relate anything in radiance fields. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2025. 2, 3

[30] Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, et al. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *Intl. J. of Computer Vision*, 123(1):32–73, 2017. 7

[31] 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 (NeurIPS)*, 33:9459–9474, 2020. 2, 3

[32] Long Lian, Yifan Ding, Yunhao Ge, Sifei Liu, Hanzi Mao, Boyi Li, Marco Pavone, Ming-Yu Liu, Trevor Darrell, Adam Yala, and Yin Cui. Describe anything: Detailed localized image and video captioning. *arXiv preprint arXiv:2504.16072*, 2025. 2, 4, 5, 7, 8

[33] Ji Lin, Hongxu Yin, Wei Ping, Pavlo Molchanov, Mohammad Shoeybi, and Song Han. Vila: On pre-training for visual language models. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 26689–26699, 2024. 5

[34] Peiqi Liu, Zhanqiu Guo, Mohit Warke, Soumith Chintala, Chris Paxton, Nur Muhammad Mahi Shafiullah, and Lerrel Pinto. Dynamem: Online dynamic spatio-semantic memory for open world mobile manipulation. In *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, 2025. 3

[35] Zhijian Liu, Ligeng Zhu, Baifeng Shi, Zhuoyang Zhang, Yuming Lou, Shang Yang, Haocheng Xi, Shiyi Cao, Yuxian Gu, Dacheng Li, Xiuyu Li, Yunhao Fang, Yukang Chen, Cheng-Yu Hsieh, De-An Huang, An-Chieh Cheng, Vishwesh Nath, Jinyi Hu, Sifei Liu, Ranjay Krishna, Daguang Xu, Xiaolong Wang, Pavlo Molchanov, Jan Kautz, Hongxu Yin, Song Han, and Yao Lu. Nvila: Efficient frontier visual language models, 2024. 5, 6, 8

[36] Joel Loo, Zhanxin Wu, and David Hsu. Open scene graphs for open-world object-goal navigation. *Intl. J. of Robotics Research*, 2025. 3

[37] Samuel Looper, Javier Rodriguez-Puigvert, Roland Siegwart, Cesar Cadena, and Lukas Schmid. 3D VSG: Long-term semantic scene change prediction through 3D variable scene graphs. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 8179–8186, 2023. 2

[38] Xianzheng Ma, Yash Bhalgat, Brandon Smart, Shuai Chen, Xinghui Li, Jian Ding, Jindong Gu, Dave Zhenyu Chen, Songyou Peng, Jia-Wang Bian, et al. When llms step into the 3d world: A survey and meta-analysis of 3d tasks via multi-modal large language models. *arXiv preprint arXiv:2405.10255*, 2024. 3

[39] D. Maggio and L. Carlone. Bayesian Fields: Task-driven open-set semantic gaussian splatting. *arXiv preprint: 2503.05949*, 2025. 2

[40] D. Maggio, Y. Chang, N. Hughes, M. Trang, D. Griffith, C. Dougherty, E. Cristofalo, L. Schmid, and L. Carlone. Clio: Real-time task-driven open-set 3D scene graphs. *IEEE Robotics and Automation Letters (RA-L)*, 9(10):8921–8928, 2024. 2, 4

[41] Jianmo Ni, Gustavo Hernandez Abrego, Noah Constant, Ji Ma, Keith Hall, Daniel Cer, and Yinfei Yang. Sentence-t5: Scalable sentence encoders from pre-trained text-to-text models. In *Association for Computational Linguistics (ACL)*, pages 1864–1874, 2022. 4, 7

[42] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023. 2

[43] Songyou Peng, Kyle Genova, Chiyu "Max" Jiang, Andrea Tagliasacchi, Marc Pollefeys, and Thomas Funkhouser. Openscene: 3d scene understanding with open vocabularies. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2023. 2

[44] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *Intl. Conf. on Machine Learning (ICML)*, pages 8748–8763. PMLR, 2021. 2, 4, 7

[45] Santhosh Kumar Ramakrishnan, Aaron Gokaslan, Erik Wijmans, Oleksandr Maksymets, Alexander Clegg, John M Turner, Eric Undersander, Wojciech Galuba, Andrew Westbury, Angel X Chang, Manolis Savva, Yili Zhao, and Dhruv

Batra. Habitat-matterport 3d dataset (HM3d): 1000 large-scale 3d environments for embodied AI. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*, 2021. 6

[46] Krishan Rana, Jesse Haviland, Sourav Garg, Jad Abou-Chakra, Ian Reid, and Niko Suenderhauf. SayPlan: Grounding large language models using 3d scene graphs for scalable task planning. In *Conference on Robot Learning (CoRL)*, pages 23–72, 2023. 3

[47] A. Ray, C. Bradley, L. Carlone, and N. Roy. Task and motion planning in hierarchical 3D scene graphs. In *Proc. of the Intl. Symp. of Robotics Research (ISRR)*, 2024. 4

[48] A. Ray, J. Arkin, H. Biggie, C. Fan, L. Carlone, and N. Roy. Structured interfaces for automated reasoning with 3d scene graphs. *arXiv preprint arXiv:2510.16643*, 2025. 3

[49] A. Rosinol, M. Abate, Y. Chang, and L. Carlone. Kimera: an open-source library for real-time metric-semantic localization and mapping. In *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, pages 1689–1696, 2020. 2

[50] A. Rosinol, A. Gupta, M. Abate, J. Shi, and L. Carlone. 3D dynamic scene graphs: Actionable spatial perception with places, objects, and humans. In *Robotics: Science and Systems (RSS)*, 2020. 2

[51] A. Rosinol, A. Violette, M. Abate, N. Hughes, Y. Chang, J. Shi, A. Gupta, and L. Carlone. Kimera: from SLAM to spatial perception with 3D dynamic scene graphs. *Intl. J. of Robotics Research*, 40(12–14):1510–1546, 2021. 2

[52] Saumya Saxena, Blake Buchanan, Chris Paxton, Peiqi Liu, Bingqing Chen, Narunas Vaskevicius, Luigi Palmieri, Jonathan Francis, and Oliver Kroemer. Grapheqa: Using 3d semantic scene graphs for real-time embodied question answering. In *Conference on Robot Learning (CoRL)*, 2025. 2, 3

[53] L. Schmid, V. Reijgwart, L. Ott, J. Nieto, R. Siegwart, and C. Cadena. A unified approach for autonomous volumetric exploration of large scale environments under severe odometry drift. *IEEE Robotics and Automation Letters*, 6(3):4504–4511, 2021. 4

[54] L. Schmid, M. Abate, Y. Chang, and L. Carlone. Khronos: A unified approach for spatio-temporal metric-semantic SLAM in dynamic environments. In *Robotics: Science and Systems (RSS)*, 2024. 2, 3, 4, 5, 1

[55] Nur Muhammad Mahi Shafiullah, Chris Paxton, Lerrel Pinto, Soumith Chintala, and Arthur Szlam. Clip-fields: Weakly supervised semantic fields for robotic memory. *arXiv preprint arXiv:2210.05663*, 2022. 2

[56] Dhruv Shah, Błażej Osiński, Brian Ichter, and Sergey Levine. LM-nav: Robotic navigation with large pre-trained models of language, vision, and action. In *Conference on Robot Learning (CoRL)*, 2022. 3

[57] Johanna Wald, Helisa Dhamo, Nassir Navab, and Federico Tombari. Learning 3D semantic scene graphs from 3D indoor reconstructions. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 3961–3970, 2020. 2

[58] Bowen Wen, Matthew Trepte, Joseph Aribido, Jan Kautz, Orazio Gallo, and Stan Birchfield. Foundationstereo: Zero-shot stereo matching. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 5249–5260, 2025. 5

[59] Abdelrhman Werby, Chenguang Huang, Martin Büchner, Abhinav Valada, and Wolfram Burgard. Hierarchical open-vocabulary 3d scene graphs for language-grounded robot navigation. *Robotics: Science and Systems (RSS)*, 2024. 6, 7

[60] S. Wu, J. Wald, K. Tateno, N. Navab, and F. Tombari. SceneGraphFusion: Incremental 3D scene graph prediction from RGB-D sequences. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 7515–7525, 2021. 2

[61] Q. Xie, S.Y. Min, P. Ji, Y. Yang, T. Zhang, A. Bajaj, R. Salakhutdinov, M. Johnson-Roberson, and Y. Bisk. Embodied-RAG: General non-parametric embodied memory for retrieval and generation, 2024. 2, 3

[62] Zhuo Xu, Hao-Tien Lewis Chiang, Zipeng Fu, Mithun George Jacob, Tingnan Zhang, Tsang-Wei Edward Lee, Wenhao Yu, Connor Schenck, David Rendleman, Dhruv Shah, Fei Xia, Jasmine Hsu, Jonathan Hoech, Pete Florence, Sean Kirmani, Sumeet Singh, Vikas Sindhwani, Carolina Parada, Chelsea Finn, Peng Xu, Sergey Levine, and Jie Tan. Mobility vla: Multimodal instruction navigation with long-context vlms and topological graphs. In *Conference on Robot Learning (CoRL)*, pages 3866–3887, 2025. 3

[63] Kashu Yamazaki, Taisei Hanyu, Khoa Vo, Thang Pham, Minh Tran, Gianfranco Doretto, Anh Nguyen, and Ngan Le. Open-fusion: Real-time open-vocabulary 3d mapping and queryable scene representation. *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, 2024. 2

[64] Yuncong Yang, Han Yang, Jiachen Zhou, Peihao Chen, Hongxin Zhang, Yilun Du, and Chuang Gan. 3d-mem: 3d scene memory for embodied exploration and reasoning. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 17294–17303, 2025. 3

[65] Hang Yin, Xiuwei Xu, Zhenyu Wu, Jie Zhou, and Jiwen Lu. SG-nav: Online 3d scene graph prompting for LLM-based zero-shot object navigation. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. 3

[66] Hang Yin, Xiuwei Xu, Linqing Zhao, Ziwei Wang, Jie Zhou, and Jiwen Lu. Unigoal: Towards universal zero-shot goal-oriented navigation. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2025. 3

[67] N. Yokoyama, S. Ha, D. Batra, J. Wang, and B. Bucher. VLFM: Vision-language frontier maps for zero-shot semantic navigation. In *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, 2024. 3

[68] Licheng Yu, Patrick Poirson, Shan Yang, Alexander C Berg, and Tamara L Berg. Modeling context in referring expressions. In *European Conf. on Computer Vision (ECCV)*, pages 69–85. Springer, 2016. 7

[69] Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. Sigmoid loss for language image pre-training. In *Intl. Conf. on Computer Vision (ICCV)*, pages 11975–11986, 2023. 2

[70] Arthur Zhang, Chaitanya Eranki, Christina Zhang, Ji-Hwan Park, Raymond Hong, Pranav Kalyani, Lochana Kalyanaraman, Arsh Gamare, Arnav Bagad, Maria Esteva, et al. Towards robust robot 3d perception in urban environments: The

ut campus object dataset. *arXiv preprint arXiv:2309.13549*, 2023. 5, 8

[71] Chenyangguang Zhang, Alexandros Delitzas, Fangjinhua Wang, Ruida Zhang, Xiangyang Ji, Marc Pollefeys, and Francis Engelmann. Open-vocabulary functional 3d scene graphs for real-world indoor spaces. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 19401–19413, 2025. 2

[72] Z. Zhang, Z. Zhu, P. Li, T. Liu, X. Ma, Y. Chen, B. Jia, S. Huang, and Q. Li. Task-oriented sequential grounding in 3D scenes, 2024. 2, 6, 7

[73] Xu Zhao, Wenchao Ding, Yongqi An, Yinglong Du, Tao Yu, Min Li, Ming Tang, and Jinqiao Wang. Fast segment anything, 2023. 3

[74] Duo Zheng, Shijia Huang, Lin Zhao, Yiwu Zhong, and Liwei Wang. Towards learning a generalist model for embodied navigation. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2024. 3

[75] Hongyu Zhou, Jiahao Shao, Lu Xu, Dongfeng Bai, Weichao Qiu, Bingbing Liu, Yue Wang, Andreas Geiger, and Yiyi Liao. Hugs: Holistic urban 3d scene understanding via gaussian splatting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21336–21345, 2024. 2

[76] Xueyan Zou, Yuchen Song, Ri-Zhao Qiu, Xuanbin Peng, Jianglong Ye, Sifei Liu, and Xiaolong Wang. M3: 3d-spatial multimodal memory. In *Intl. Conf. on Learning Representations (ICLR)*, 2025. 2