Scene Graphs as Relational State Abstractions for Belief-Space Planning in Semi-Static Environments

Phan Le*, Kavindie Katuwandeniya†, Hanna Kurniawati‡, Dana Kulić*†
*Monash University, Australia
†CSIRO Robotics, Australia
‡Australian National University, Australia

***Abstract*—Robots searching for objects in human environments must operate under partial observability while adapting to semi-static environmental changes introduced by human activity over extended operation. Existing object-search POMDP formulations provide principled planning under uncertainty, but typically assume static environments and metric state representations. We propose a Scene Graph (SG)-based POMDP formulation for object search in semi-static environments, representing uncertainty directly over relational object states and modelling semi-static environmental dynamics through stochastic relational transitions. To instantiate this formulation, we integrate the long-horizon online planner with an SG-adapted belief update mechanism, forming a belief-space planning framework operating directly over relational scene representations. Simulations comparing to a conventional metric-based formulation demonstrate improved planning efficiency under identical planning conditions, providing empirical validation of the proposed framework and supporting the use of Scene Graphs as an effective structured state abstraction for object search under uncertainty in evolving environments.**

I. Introduction

Objects search is a prerequisite for most domestic service tasks and a core capability for Long-Term Autonomy (LTA) in human-shared environments [3, 4]. Such environments impose two compounding challenges. The first is *partial observability*: the robot perceives only a local region and must infer target locations from its observation history. The second is *environmental variability*: human activity relocates objects at unpredictable times, producing what Bore et al. [2] term as *semi-static* changes. These challenges force the robot to navigate under information that is both incomplete and potentially outdated.

The POMDP framework addresses partial observability by having the agent maintain a *belief* – a distribution over states consistent with its observation history – and select actions by reasoning over this belief [6]. Previous works define the state over the robot's configuration and target object locations, so that belief captures uncertainty over the full search context [11, 1, 14]. Whereas most formulations ground the state space in dense metric grids, an alternative line of work represents the environment as a *Scene Graph* (SG), where nodes denote objects and edges encode spatial or semantic relations between them [9, 5]. This abstraction scales with the number of objects while providing contextual reasoning over object locations. Despite this, existing object-search POMDP formulations assume a *static environment* and address only *single-target* search.

To address the environmental variability, complementary SG-based approaches have explored directions for modelling evolving object relationships over time: incrementally updated scene representations and explicit probabilistic modelling of object-level uncertainty. Updatable SG frameworks [17, 13, 10] maintain a structural memory revised incrementally as observations arrive, maintaining topological associations between semi-static and static objects; this memory then guides downstream navigation. In parallel, explicit probabilistic frameworks model object-location likelihoods through time-varying distributions over SG nodes' attributes [8] and edges [5], modelling semi-static changes as probabilistic estimates without being explicitly integrated into action planning. Together, these directions indicate that robust object search in semi-static environments depends on maintaining a updated structural memory of previously observed object relationships while reasoning over evolving relational uncertainty. However, neither category embeds these representations within a principled belief-space planner: inferred object-location likelihoods guide reactive action selection, but are not propagated through a lookahead that explicitly weighs information gathering against goal-directed navigation.

These two research directions remain largely disconnected: POMDP frameworks explicitly plan under partial observability but assume the environments have limited dynamics, whereas SG-based approaches model semi-static dynamics but reduce action selection to reactive execution rather than principled lookahead. We therefore propose a unified formulation that integrates semi-static scene modelling with belief-space planning, enabling long-horizon object search in evolving environments.

II. Proposed approach

Our formulation models long-horizon object search in semi-static environments as a POMDP. Unlike existing object-search formulations, our approach models (1) the joint robot-object state directly over a Scene Graph representation, and (2) semi-static environmental dynamics through a stochastic transition function defined over inter-object relations.

A. Preliminaries

a) Task definition: At arbitrary points during operation, the robot may receive a semi-static object query as the current

target, whose location is not directly observable and must be inferred from observation history. Once the target is found, subsequent search queries may be issued; Thus, requiring the robot to preserve accumulated belief over previously observed objects to continue the task.

b) Scene graph structure: We represent the underlying environment state using a closed-set scene graph $\mathcal{G}_t = (V, E_t)$, which serves as an abstracted representation that preserves object-level semantic and relational structure while omitting low-level metric spatial information. The graph consists of a fixed set of object nodes V, and the time-varying edge set E_t. Specifically, each node $v_t^i \in V$ is represented as a tuple $v_t^i = (class, label)$, where $class \in \{\textit{static}, \textit{semi-static}\}$ denotes whether an object's state is fixed or may change over time due to external factors, and $label$ uniquely identifies the object. Each edge encoding a spatial relationship is represented as $e_t^{i,j} = (v_t^i, v_t^j, label)$, where v_t^i and v_t^j denote the connected node pair, and $label$ specifies the relation type. We consider two edge labels: *traversable* between static nodes, defining possible traversal movement, and *in* between semi-static and static nodes, denoting the spatial relation between a semi-static object and its supporting static entity.

This SG formulation is motivated by two key design objectives. First, by abstracting away metric spatial coordinates, the representation captures object-centric relational structure rather than low-level geometric information. This enables reasoning over contextual dependencies between objects, allowing object locations and semantic attributes to be inferred under uncertainty. Second, the proposed graph structure provides a scalable representation for semi-static environments. Since semi-static changes correspond to relocations between existing objects rather than the creation or removal of objects, they are modelled by maintaining a fixed node set V while allowing only the edge set E_t to vary over time. Consequently, environmental changes are incorporated through local edge updates, requiring modification only to affected object-place relations rather than reconstruction of the entire graph.

B. Problem Formulation

We formulate the problem as a POMDP $\langle S, A, T, \mathcal{O}, Z, R, \gamma, b_0 \rangle$ defined directly over the SG representation. At time step t, the joint-state $s_t = (\mathcal{G}_t, p_t)$ where $\mathcal{G}_t$ defines the object states, and $p_t \in V$ defines the robot state as which object node the robot is currently at. As a result, the action space can be parameterised using the node set V_t, which consists of transition actions between object nodes ($Goto(v^j)$) and remaining at the current node ($Wait()$).

To accommodate the joint state s_t of the robot and objects, the transition function decomposes into robot and environment dynamics. Robot motion T^{robot} is deterministic and constrained by graph connectivity, while environment dynamics T^{env} estimates semi-static changes using a relocation probability $\hat{p}_{relocate}$ over a frequency $\hat{f}_{relocate}$. Since semi-static object relocations are driven by external human activity, the robot cannot observe the true environment dynamics directly. Therefore, $\hat{p}_{relocate}$ and $\hat{f}_{relocate}$ represent prior heuristic estimates used by the robot to predict how the environment may evolve. Since the node set encoding V is fixed, the semi-static change is represented solely by changes in the edge set E_t. Assuming that robot actions do not directly affect object states, the transition function factorises as

$$\begin{aligned} T(s_{t+1} \mid s_t, a_t) = T^{robot}(p_{t+1} \mid p_t, a_t) \\ \cdot\ T^{env}(\mathcal{G}_{t+1} \mid \mathcal{G}_t, \hat{p}_{relocate}, \hat{f}_{relocate}) \end{aligned} \tag{1}$$

After executing an action, the robot receives a local SG observation $o_t = \mathcal{G}_t^{obs} \subseteq \mathcal{G}_t$. The reward function penalises movement by distance travelled and provides a reward R_{found} when the current target object is located.

Under partial observability of object states, the robot maintains a belief SG over the time-varying edge sets, $b_t(\mathcal{G}_t) = (V, b_t(E_t))$, where each edge belief denotes the probability that the corresponding spatial relation exists, i.e., $b_t(e_t^{i,j}) \in [0, 1]$. Assuming independent semi-static changes of the object in the SG, the joint belief factorises over object state. The belief can be initialised from prior knowledge and updated through Bayesian inference:

$$\begin{aligned} b_{t+1}(\mathcal{G}_{t+1}) = \eta Z(\mathcal{G}_t^{obs} \mid \mathcal{G}_{t+1}) \\ \sum T^{env}(\mathcal{G}_{t+1} \mid \mathcal{G}_t)\, b_t(\mathcal{G}_t) \end{aligned} \tag{2}$$

where Z is the observation model (assumed perfect) and η is a normalisation constant. Although observations are local, new observations may indirectly affect belief estimates elsewhere in the graph through shared object associations. For example, observing a relation involving one object can strengthen or weaken confidence in alternative spatial relations associated with that same object.

C. Method Instantiation

The proposed SG-based POMDP supports long-horizon operation in evolving environments by continuously maintaining belief over the Scene Graph and performing target-directed planning whenever object search is required. In this work, we instantiate this planning process by integrating existing planning and belief-estimation components adapted to operate directly on SG relational states rather than metric coordinates.

For planning, we employ ROP-RAS3 [7] and adapt its macro-action parameterisation so that planning operates on object nodes in the Scene Graph rather than on metric-coordinate goals. For belief maintenance, we adapt the object-factorised SIR pipeline from OO-POMDP [11] by redefining particles over SG edges instead of coordinate-based object states. These components form an online planning loop that iterates between planning, execution, observation, and belief updating, as summarised in Algorithm 1. The resulting routine operates on the currently active target while preserving accumulated belief that can be reused for future search tasks.

Algorithm 1 Online Planning under SG-based POMDP

Require: Prior belief $b_0(\mathcal{G}_0)$, queried target node v_t^{target}

Initialise particle set $\mathcal{P}_0$ from prior $b_0(\mathcal{G}_0)$
Condition belief on queried target node v_t^{target}
while v_t^{target} not found **do**
    Select action a^* using ROP-RAS3 [7] planning
    Execute a^* and receive observation $\mathcal{G}_{t+1}^{obs}$
    $b_{t+1}(\mathcal{G}_{t+1}) \leftarrow$ BELIEFUPDATE$(b_t(\mathcal{G}_t), \mathcal{G}_{t+1}^{obs}, T^{env}, Z)$
    if $v_t^{\text{target}} \in mathcalG_{t+1}^{obs}$ **then**
        Terminate current search episode
    end if
end while

D. Experimental setup

a) Simulated environment: We evaluate the proposed framework by adapting the 2D gridworld Multi-Object Search (MOS) environment from the `pomdp-py` library [15], augmented with stochastic semi-static environmental dynamics. As illustrated in Fig. 1, the workspace is discretised into a 28×20 grid world, where each cell represents a spatial location indicating the presence of permanent obstacles, static objects, semi-static objects, and the robot. Permanent obstacles and static objects occupy fixed cells that define the environment layout, whereas semi-static objects and the robot may occupy different cells over time as they relocate or navigate. Static objects occupy fixed 3×3 regions, where a semi-static object may occupy the same cell as its associated static object, while the robot perceives its surroundings through a local 3×3 observation window. The environment contains 8 static objects (labelled s_1 to s_8) and 2 semi-static objects, both serving as potential targets.

During execution, the environment evolves according to two processes. First, to model environmental variability, each semi-static object has a relocation probability $p_{relocate} = 0.05$ of transitioning among predefined candidate static objects after every 200 executed cell-to-cell movement actions. Second, to approximate long-horizon object search under partial observability, we adopt a simplified MultiON-style evaluation protocol [12], where only one semi-static object is designated as the active target at any time. Once the active target is located, the remaining semi-static object becomes the next active target. Since the environment contains only two semi-static objects, the target assignment sequence alternates between the two throughout the episode.

b) SG-based POMDP: The scene graph abstraction is constructed over the simulated environment and used to initialise the prior belief. Each static object occupying a fixed 3×3 grid region is abstracted as a single static node, while each semi-static object occupying a single grid cell is mapped to a corresponding semi-static node. The interpretation of the $label$ attribute depends on the node class. For static nodes, $label$ identifies the corresponding static object in the underlying gridworld (e.g., s_1 to s_8), whereas for semi-static nodes, it encodes the object's current target status (active target or non-target). For planning over the scene graph, each static node maintains *traversable* edges with every other static node, allowing the planner heuristic to sample navigation actions toward all possible candidate object locations.

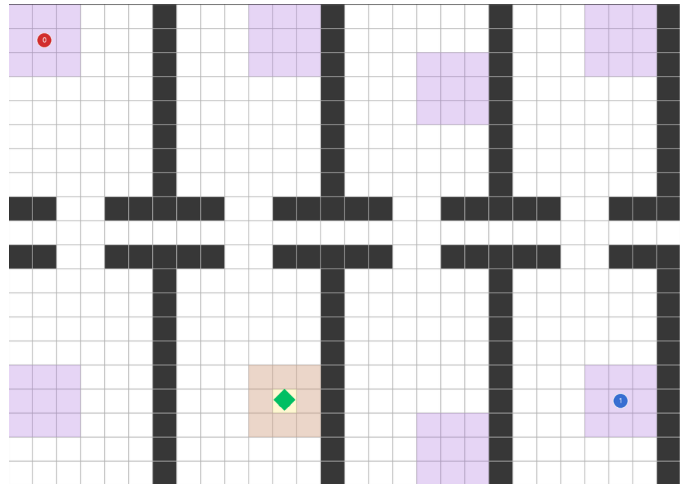

Fig. 1. Example gridworld environment for evaluation. The robot (green diamond) searches for semi-static objects, including the current target (red circle) and possible future targets (blue cicle). Each semi-static object is located *in* a static object, represented by a cluster of occupancy cells (purple). During execution, the robot perceives the environment through a local observation window (yellow) and must avoid permanent obstacles (black).

To validate the proposed formulation under stochastic environmental changes, the belief is defined only over edges with label *in*, representing the spatial relations between semi-static and static objects, while *traversable* edges are given as prior knowledge, assumed fully observable, and remain fixed throughout execution. An example belief scene graph $b_t(\mathcal{G}_t)$ is illustrated in Fig. 2. Following the modelling assumptions in Section II-B, the planner constructs the environment transition model using estimated semi-static dynamics $\hat{p}_{relocate} = p_{relocate}/10$ and $\hat{f}_{relocate} = 1$ planned action (while the true simulated environment evolves according to $p_{relocate}$ and $f_{relocate}$). The robot action transition model is assumed perfect. Furthermore, whenever a new target object is assigned, the belief is updated by modifying the corresponding node $label$, ensuring that the robot knows the currently active target while preserving previously accumulated beliefs over non-target objects.

c) Planner and Execution interface: The ROP-RAS3 solver [7] is configured with 3000 belief particles, 2000 Monte Carlo tree simulations per decision, and a lookahead depth of 50 cell-to-cell movement steps. The reference policy is to sample the action $Goto()$ toward all candidate object nodes. The SG-based planner operates at the graph abstraction level, while a shared low-level controller handles action execution and observation acquisition in the underlying grid world. The planner outputs macro-actions for navigation toward destination nodes, which are executed as shortest cell-to-cell paths computed using breadth-first search. The robot perceives its surroundings through a local 3×3 observation window. For SG-based planning, this observation is converted into a partial scene graph consisting of the occupied static node and any

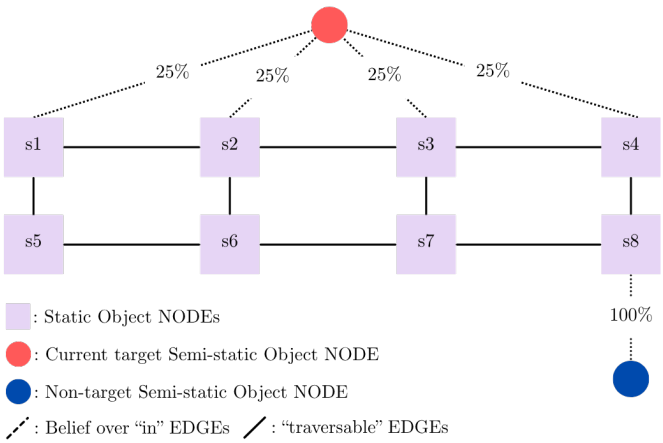

Fig. 2. Example belief scene graph $b_t(\mathcal{G}_t)$ over only *in* edges (dashed) linking semi-static object node to candidate static object nodes. The current target semi-static object (red) has location belief distributed across candidate static objects (purple) $\{s_1, s_2, s_3, s_4\}$, whereas the non-target object (blue) remains localised *in* static object $\{s_8\}$ with 100% belief confidence. For visual clarity, only a subset of *traversable* edges linking static object nodes is illustrated.

observable semi-static object relations. The observation model is assumed perfect. The reward model assigns a cost of -1 for every cell-to-cell movement and a reward of $+30$ when the active target object is successfully located.

E. Evaluation Setup

For empirical comparison, we adopt a conventional coordinate-based OO-POMDP baseline [11] and extend its transition function with the same stochastic semi-static relocation dynamics (T^{env}) of the SG-based formulation. This isolates the effect of state abstraction by keeping the simulation environment, reward structure, and planning configuration identical across both formulations. The formulation differ only in how the state, belief, and action spaces are parameterised: whereas the proposed formulation defines state and belief over an SG, the baseline represents both robot and object states explicitly as metric-based grid coordinates. For both formulations, ROP-RAS3 uses an equivalent reference policy that samples the sub-goal toward all available static objects, represented as SG nodes or coordinate-based navigation goals, respectively.

Both formulations are evaluated across 20 independent trials, each executing 100 macro-actions under identical environmental conditions. Performance is reported as mean $\pm$ 95% confidence intervals across all trials using three metrics: **Avg Planning Time** (sec), measuring average planning time per action; **Total Targets Found**, measuring the number of targets located per trial; and **Per-Milestone (M) Path Length** ($cells$), measuring the average path travelled before locating the active target. A milestone is defined as the interval between successfully locating the current target and the subsequent target assignment.

Experiment were conducted on a workstation equipped with an Intel Core Ultra 7 165H CPU, 30 GB RAM, and an NVIDIA RTX 1000 Ada GPU, with all planning and simulation executed on CPU.

III. Results and Discussion

We compare the metric-based and SG-based POMDP formulations to evaluate the search efficiency and task performance quantitatively.

TABLE I
Quantitative Comparison of Long-Horizon Object Search Performance in Semi-Static Environments. (M) denotes per-milestone metrics

Metric	Metric-based POMDP	SG-based POMDP
Avg Planning Time	9.384 $\pm$ 0.124	0.704 $\pm$ 0.002
Total target found	52.500 $\pm$ 3.356	76.500 $\pm$ 1.508
(M) Path length	46.557 $\pm$ 2.405	33.126 $\pm$ 0.993

Table I shows that the SG-based formulation outperforms the metric-based baseline across all metrics. The SG-based formulation achieves faster planning, discovers more target objects within the action budget, and reaches targets with shorter path lengths per search milestone. This corresponds to a 45.7% (52.500 to 76.500) increase in targets found and a 92.5% (9.384 to 0.704) reduction in average planning time. Both formulations share identical planning budgets, solver configurations, and action-sampling heuristics, suggesting these gains arise from the underlying state representation rather than planning strategy.

The performance gap stems from how uncertainty is represented. Whereas metric-based planning distributes belief over a large coordinate space, the SG-based formulation concentrates uncertainty over a compact set of inter-object relations, resulting in more structured and informative belief updates. This reduces the effective belief space explored by Monte Carlo Tree Search, resulting in faster planning and more focused policy expansion. As a result, the planner avoids exploring irrelevant search branches, directly improving both path efficiency and task completion.

These results demonstrate that Scene Graphs provide an effective state abstraction for belief-space planning, enabling scalable reasoning over semi-static environmental uncertainty in evolving human-shared environments. More broadly, these findings support the central premise that integrating semi-static scene modelling with principled belief-space planning is a more effective foundation for long-horizon object search than conventional metric-based formulations.

A limitation of the formulation is the independence assumption imposed over object relations, which enables tractable belief maintenance but restricts modelling of correlated environmental changes. Future work will investigate relaxing this assumption so that observations of one semi-static object can inform beliefs about other related objects. In addition, richer semantic relationships encoded within the Scene Graph could be incorporated into the reference policy, allowing semantic priors between related objects to guide exploration beyond the current explicit semi-static and static object relations. Similar to [14, 16], when searching for a *mouse*, the planner may first visit a semantically related static object *desk*, or even another semi-static object *computer*.

References

[1] Soroush Amiri, Donghyeon Choi, Jiajun Wu, Yuke Wang, and Dieter Fox. Reasoning with scene graphs for robot planning under partial observability. In *Proceedings of Robotics: Science and Systems (RSS)*, 2022.

[2] Nils Bore, Johan Ekekrantz, Patric Jensfelt, and John Folkesson. Detection and tracking of general movable objects in large 3d maps. *CoRR*, abs/1712.08409, 2017. URL http://arxiv.org/abs/1712.08409.

[3] Jiafei Duan, Samson Yu, Hui Li Tan, Hongyuan Zhu, and Cheston Tan. A survey of embodied ai: From simulators to research tasks. *IEEE Transactions on Emerging Topics in Computational Intelligence*, 6(2):230–244, 2022. doi: 10.1109/TETCI.2022.3141105.

[4] Lars Kunze, Nick Hawes, Tom Duckett, Marc Hanheide, and Tomáš Krajník. Artificial intelligence for long-term robot autonomy: A survey. *IEEE Robotics and Automation Letters*, 3(4):4023–4030, 2018. doi: 10.1109/LRA.2018.2860628.

[5] Andrey Kurenkov, Michael Lingelbach, Tanmay Agarwal, Emily Jin, Chengshu Li, Ruohan Zhang, Li Fei-Fei, Jiajun Wu, Silvio Savarese, and Roberto Martín-Martín. Modeling dynamic environments with scene graph memory, 2023. URL https://arxiv.org/abs/2305.17537.

[6] Hanna Kurniawati. Partially observable markov decision processes and robotics. *Annual Review of Control, Robotics, and Autonomous Systems*, 5(1): 253–277, May 2022. ISSN 2573-5144. doi: 10.1146/annurev-control-042920-092451. URL http://dx.doi.org/10.1146/annurev-control-042920-092451.

[7] Yuanchu Liang, Edward Kim, J. Arden Knoll, Wil Thomason, Zachary Kingston, Lydia E. Kavraki, and Hanna Kurniawati. Think fast and far: Long-horizon online pomdp planning via rapid state sampling, 2026. URL https://arxiv.org/abs/2606.04355.

[8] 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. doi: 10.1109/ICRA48891.2023.10161212.

[9] Isaac Remy, Abhishek Gupta, and Karen Leung. Semantically-driven object search using partially observed 3d scene graphs. In *NeurIPS 2023 Foundation Models for Decision Making Workshop*, 2023. URL https://openreview.net/forum?id=YVZ03XsMfg.

[10] Yujie Tang, Meiling Wang, Yinan Deng, Zibo Zheng, Jingchuan Deng, Sibo Zuo, and Yufeng Yue. Openin: Open-vocabulary instance-oriented navigation in dynamic domestic environments. *IEEE Robotics and Automation Letters*, 10(9):9256–9263, 2025. doi: 10.1109/LRA.2025.3592071.

[11] A. Wandzel, Y. Oh, M. Fishman, N. Kumar, L. L. Wong, and S. Tellex. Multi-object search in object-oriented pomdps. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 7194–7200. IEEE, 2019.

[12] Saim Wani, Shivansh Patel, Unnat Jain, Angel X. Chang, and Manolis Savva. Multion: Benchmarking semantic map memory using multi-object navigation. *arXiv preprint arXiv:2012.03912*, 2020. URL https://arxiv.org/abs/2012.03912.

[13] Zhijie Yan, Shufei Li, Zuoxu Wang, Lixiu Wu, Han Wang, Jun Zhu, Lijiang Chen, and Jihong Liu. Dynamic open-vocabulary 3d scene graphs for long-term language-guided mobile manipulation, 2024. URL https://arxiv.org/abs/2410.11989.

[14] Zhen Zeng, Adrian Röfer, and Odest Chadwicke Jenkins. Semantic linking maps for active visual object search. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1984–1990, 2020. doi: 10.1109/ICRA40945.2020.9196830.

[15] Kaiyu Zheng and Stefanie Tellex. pomdp_py: A framework to build and solve pomdp problems. In *ICAPS 2020 Workshop on Planning and Robotics (PlanRob)*, 2020. URL https://icaps20subpages.icaps-conference.org/wp-content/uploads/2020/10/14-PlanRob_2020_paper_3.pdf. Arxiv link: "https://arxiv.org/pdf/2004.10099.pdf".

[16] Kaiyu Zheng, Rohan Chitnis, Yoonchang Sung, George Konidaris, and Stefanie Tellex. Towards optimal correlational object search. In *2022 International Conference on Robotics and Automation (ICRA)*, pages 7313–7319, 2022. doi: 10.1109/ICRA46639.2022.9812252.

[17] Fangbo Zhou, Huaping Liu, Huailin Zhao, and Lanjun Liang. Long-term object search using incremental scene graph updating. *Robotica*, 41(3):962–975, 2023. doi: 10.1017/S0263574722001205.