2026 IEEE International Conference on Robotics and Automation (ICRA 2026)
June 1-5, 2026. Vienna, Austria

DSSM-SG: Dynamic 3D Scene Graphs with Spatio-Semantic Memory for Long-Term Indoor Navigation Tasks

Yi Ruan[1], Yaowen Zhang[1], Miaoxin Pan[1], Yi Yang[1]*, Mengyin Fu[1,2]

***Abstract*— Dynamic indoor environments pose significant challenges for autonomous robots, as objects frequently move and scenes continuously change, requiring robust scene representation and adaptive navigation strategies. In this work, we introduce DSSM-SG, a dynamic open-vocabulary 3D scene graph framework enhanced with spatial-semantic memory, to support complex language instruction parsing and goal navigation in dynamic environments. First, we construct a multi-layered scene graph by combining waypoint topology with semantic object information, and propose a viewpoint-based mechanism to model object dynamics and detect scene changes, enabling more precise semantic-geometric representation. Second, we design an efficient incremental graph update strategy that adapts to object-level dynamics and navigation-observed obstacles, thereby maintaining graph consistency and alleviating mismatch during re-navigation. Finally, we introduce a subgraph generation and matching approach driven by large language models, significantly improving the system's ability to interpret and ground ambiguous goal descriptions. Experimental results demonstrate that DSSM-SG achieves superior performance in scene graph accuracy, update efficiency, and language goal navigation success compared to existing baselines in dynamic indoor environments.**

I. INTRODUCTION

Autonomous robots operating in real-world environments face multiple challenges: they must interpret complex and often ambiguous human instructions, maintain long-term task execution, and adapt to dynamically changing surroundings. To meet these challenges, they require robust multimodal language understanding, hierarchical task planning with long-term memory, and dynamic perception grounding combined with adaptive reasoning.

3D scene graphs provide a structured representation that enables multi-level semantic understanding and relational reasoning among objects and regions, making them a natural tool for addressing such tasks. However, most existing scene graph methods assume static environments and fixed object configurations, limiting their ability to handle dynamic scene changes and leading to discrepancies between the graph and the actual environment. Moreover, current object navigation methods often rely on pre-defined or single-target sequences, failing to capture implicit relationships among objects and the inherent ambiguity present in natural language instructions. These limitations hinder robust long-term navigation and complex goal-directed behavior in dynamic real-world scenarios.

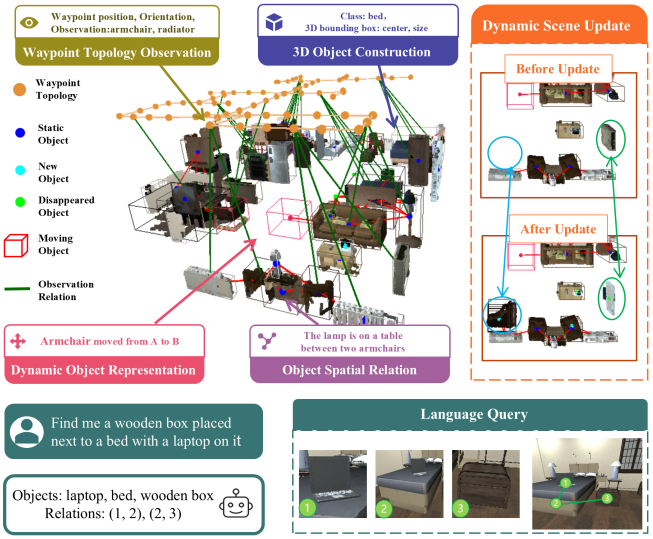

Fig. 1: We propose DSSM-SG, a framework for dynamic indoor environments representation that models object semantics, waypoint topology, and dynamics to support scene updates and language-guided navigation.

To address these challenges, we propose DSSM-SG, a dynamic 3D scene graph construction and updating framework with spatio-semantic memory. We extract object-level semantic features using an open-vocabulary perception model and fuse them with geometry to build a dynamic spatio-semantic representation, enhancing 3D scene graph expressiveness in changing environments. Furthermore, based on viewpoint matching, the framework supports scene graph updates across different levels of object dynamics, ensuring adaptability to changing environments. Finally, we introduce a subgraph-based language grounding and goal navigation method that improves the agent's understanding of inter-object relationships and ambiguous language queries, thereby enabling robust task execution in dynamic settings. An illustrative result of DSSM-SG is shown in Fig.1. Our main contributions are summarized as follows:

- We propose DSSM-SG, a novel framework for constructing dynamic open-vocabulary 3D scene graphs, which incorporates dynamic spatio-semantic memory and object-level dynamic attributes to enable comprehensive representation of dynamic environments.
- We design an incremental scene graph updating mechanism that leverages object dynamics to effectively adapt and correct the scene representation as the environment

This work was supported by National Natural Science Foundation of China (Grant No. NSFC 62233002, 92370203) and National Key RD Program of China (2022YFC2603600).

[1]School of Automation, Beijing Institute of Technology, Beijing, China.
[2]School of Automation, Nanjing University of Science and Technology, Nanjing, China.
*Corresponding author: Yi Yang (yang_yi@bit.edu.cn).

changes.

- We develop a subgraph-based language grounding and goal navigation method, which utilizes a large language model (LLM) to interpret ambiguous queries and matches local subgraphs to identify the target's global position within the scene graph, enabling accurate and context-aware goal retrieval.

This paper is organized as follows. Section II discusses related work. Section III describes the proposed method DSSM-SG. Section IV details the experimental setup and results, and Section V concludes the paper.

II. RELATED WORK

A. Open-Vocabulary 3D Scene Graph

3D scene graphs [1]–[3] aim to represent objects and their relationships in a structured form, enabling higher-level scene understanding and reasoning for robotic tasks. With the rapid development of Vision-Language Models (VLMs), 3D scene graphs have adopted open-vocabulary mechanisms to address the zero-shot problem in unknown environments [4]–[7]. ConceptGraphs [8] integrates 3D geometric cues with LLMs and VLMs, assigning semantic labels to objects and supporting free-form language queries with rich semantic descriptions. HOV-SG [9] extends this approach to multi-floor indoor environments, enabling cross-level accessibility analysis and object navigation. In outdoor scenarios, CURB-SG [10] introduces a collaborative urban scene graph for modeling traffic environments. However, most of these methods rely on static assumptions and struggle to adapt to dynamic real-world changes, limiting their robustness in long-term tasks.

B. Dynamic Environments Mapping

Mapping in dynamic environments remains a fundamental challenge in navigation [11], [12]. Khronos [13] introduced a spatio-temporal semantic perception framework that integrates short-term dynamics with long-term global change detection to construct dense spatio-temporal maps. DynaMem [14] employs dynamic spatio-semantic memory to represent the environment as a voxel map, enabling real-time updates and dynamic target tracking in mobile manipulation tasks. DovSG [15] builds a dynamic open-vocabulary 3D scene graph with a local update mechanism, while OpenIN [16] utilizes a Carrier-Relationship Scene Graph for dynamic scene graph updates. However, existing approaches often overlook the influence of dynamic changes on object attributes and inter-object relationships, making it difficult to model temporal regularities and constraining long-term task performance.

C. Language Goal Navigation

As one of the most significant foundation tasks for intelligent agent, goal navigation can be classified to object goal navigation [17]–[20] and language goal navigation [21], [22]. In contrast, language goal navigation raises a higher request, not only specifying object category, but also identifying both the attributes and spatial relationships in natural language. VLMaps [23] built semantic maps through LSeg [24] Model encoding images and aligning pixel embedding with 3D map location. LM-Nav [25] adapted a strategy, leveraging GPT-3 to transfer text queries to landmarks and combining CLIP [26] feature to optimize navigation target. In addition, VLN-Game [27] proposed a game-theoretic based vision-language goal navigation method, identifying the most promising areas through 3D object-centric spatial map.

To address these challenges, we propose DSSM-SG, a dynamic open-vocabulary 3D scene graph framework with spatio-semantic memory. It combines the advantages of scene graph construction and dynamic mapping, enabling the generation of semantically rich and logically structured maps that adapt to environmental changes through continuous updates. To support language-goal navigation, we further represent language queries as graph structures, allowing accurate interpretation and efficient retrieval in dynamic indoor environments.

III. METHOD

We propose DSSM-SG, a dynamic open-vocabulary 3D scene graph construction and updating framework capable of building stable and semantically enriched 3D representations in long-term dynamic indoor environments, supporting subgraph-based language goal navigation. An overview of the introduced framework is illustrated in Fig.2.

A. Dynamic Scene Graph Construction

Given an RGB-D sequence $I = \{I_1, I_2, ..., I_t\}$, we construct a 3D scene graph $M = \{N, E\}$, where the node set $N = N_O \cup N_W$ includes object and waypoint nodes, and the edge set $E = E_O \cup E_W \cup E_{WO}$ captures spatial relations among them.

Object-centric map construction: Each frame I_t is processed through open-vocabulary detection and segmentation [28]–[30] to obtain object masks $\{m_{t,i}\}$, bounding boxes $\{b_{t,i}\}$, and semantic labels $\{c_{t,i}\}$, from which we extract CLIP-based visual and semantic features $\{f_{t,i}\}$. These 2D masks are projected into 3D point clouds $\{p_{t,i}\}$ using the depth image I_t^{depth} and camera pose θ_t.

To associate new observations with existing objects $o_{t-1,j} = \langle p_{o_j}, f_{o_j} \rangle$, we compute geometric similarity $s_{geo}(i,j)$ based on neighborhood overlap, and semantic similarity $s_{sem}(i,j)$ via normalized cosine distance. The fusion score $s(i,j)$ is a sum of both. if $s(i,j) < \delta_{sim}$, a new object node is created, otherwise the point cloud and features are merged.

Object nodes with confidence above δ_{conf} are retained, each storing a 3D bounding box B_{o_j}, observation index $I_{o_j}^{idx}$, and textual feature $f_{o_j}^{text}$ extracted using the CLIP text encoder. Edges E_O are created between nearby objects based on the minimum bounding box distance.

Waypoint topology construction: In our framework, we adopt a viewpoint fusion strategy to construct the waypoint topology. Specifically, given a sequence of camera poses, a new viewpoint is added and a topological edge is created when the angular or spatial difference between adjacent poses exceeds predefined thresholds. Viewpoints with identical positions are merged into a single waypoint

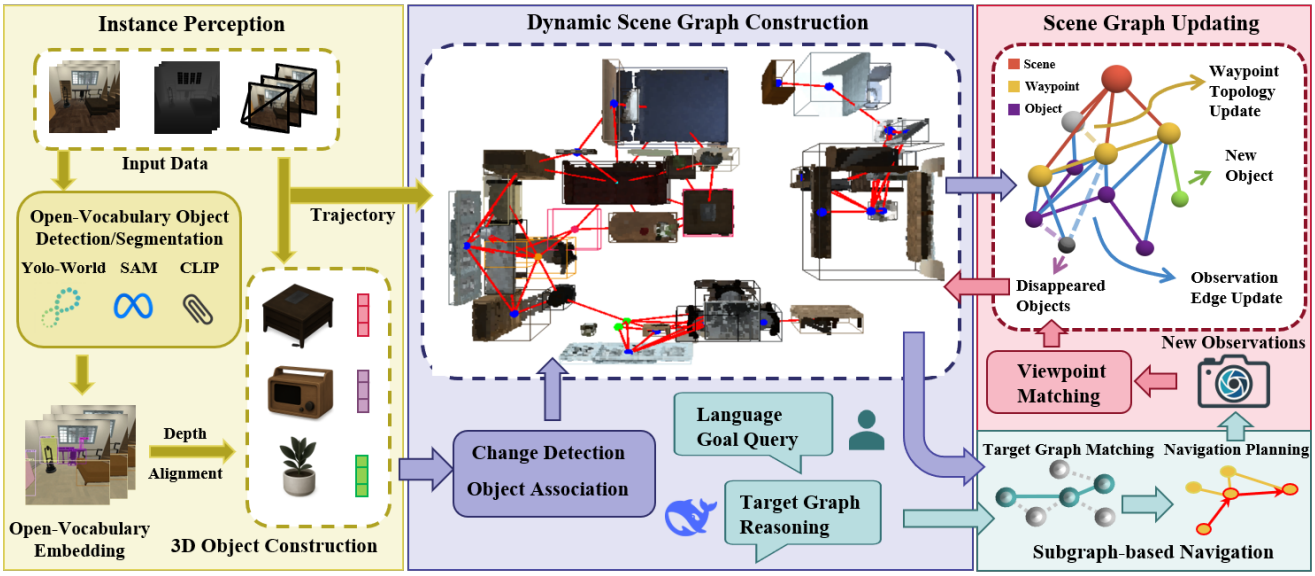

Fig. 2: The DSSM-SG framework consists of open-vocabulary static instance construction, dynamic scene representation, target graph-based language query, and scene graph updating.

node, which aggregates multi-directional observations. For repeated orientations across different timestamps, only the most recent observation is retained to ensure consistency. Observation edges E_{WO} connect waypoints to visible objects, constrained by a range threshold to discard unreachable targets. This forms a sparse graph where each waypoint corresponds to a subgraph of locally visible objects.

Dynamic spatio-semantic memory: To detect scene changes, we group observations by viewpoint and apply a sliding window strategy. For disappearing objects, we define the final consistent frames as the reference set O_{base}, and the remaining as comparison set O_{comp}. Objects missing in O_{base} but present in O_{comp} are labeled as disappeared. For new object detection, the initial frames are used as the reference set O_{base}, while the later frames serve as the comparison set O_{comp}. Objects appearing in O_{comp} but not in O_{base} are identified as newly appeared.

We treat object movement as the association between newly appeared and recently disappeared objects, based on fused RGB, semantic, and volumetric similarity. For each candidate pair (dis, app), we compute cosine similarity of RGB features s_{rgb}, semantic embeddings s_{sem}, and volume similarity s_V. The final motion score is:

$$\Delta s_{sem}(dis, app) = s_{sem}(dis, app) - \delta_{sem} \tag{1}$$

$$s_{move} = \frac{1}{2}(s_{rgb}(dis, app) + \delta_{sem} s_V(dis, app) + \Delta s_{sem}) \tag{2}$$

If s_{move} exceeds a threshold, the pair is considered a moving object and labeled with dynamic state $d^s_{o_j}$.

Historical object locations are preserved by maintaining disappeared or displaced entities as historical nodes in the scene graph, establishing a dynamic spatio-semantic memory that supports temporal reasoning. Additionally, to estimate the likelihood of object movement due to human interaction or environmental influence, we provide the object's category

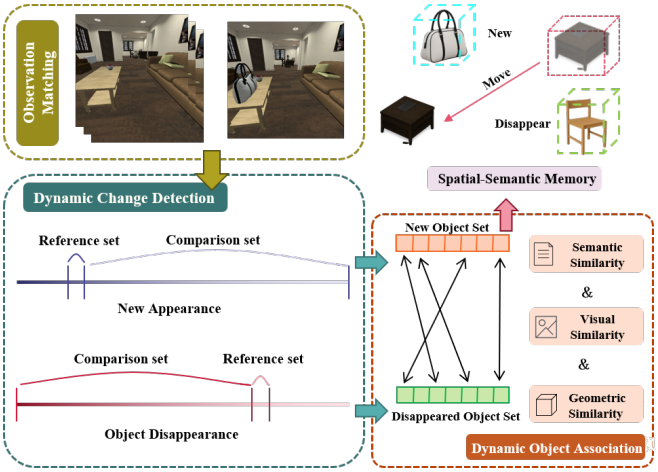

Fig. 3: Dynamic Spatio-Semantic Memory. Perceptual changes are detected via window-based comparison; dynamic objects are associated through fused semantic, visual, and geometric similarity.

label c_{o_j}, bounding box B_{o_j}, and surrounding semantic context to a large language model. The resulting score is assigned as the dynamic attribute $d^a_{o_j}$. Fig.3 illustrates the overall process of constructing a dynamic 3D scene graph with spatio-semantic memory.

B. Dynamic Scene Graph Updating

To adapt to structural changes during repeated exploration of dynamic scenes, it is necessary to update the scene graph in a timely manner.DSSM-SG utilizes RGB-D observations I^{re}_t collected at waypoints during re-exploration to dynamically update both object nodes and the waypoint topology.

Dynamic object update: The current observation is first matched with existing viewpoints in the 3D scene graph. For successfully matched frames, we use an open-vocabulary vision-language model to extract the observed object set

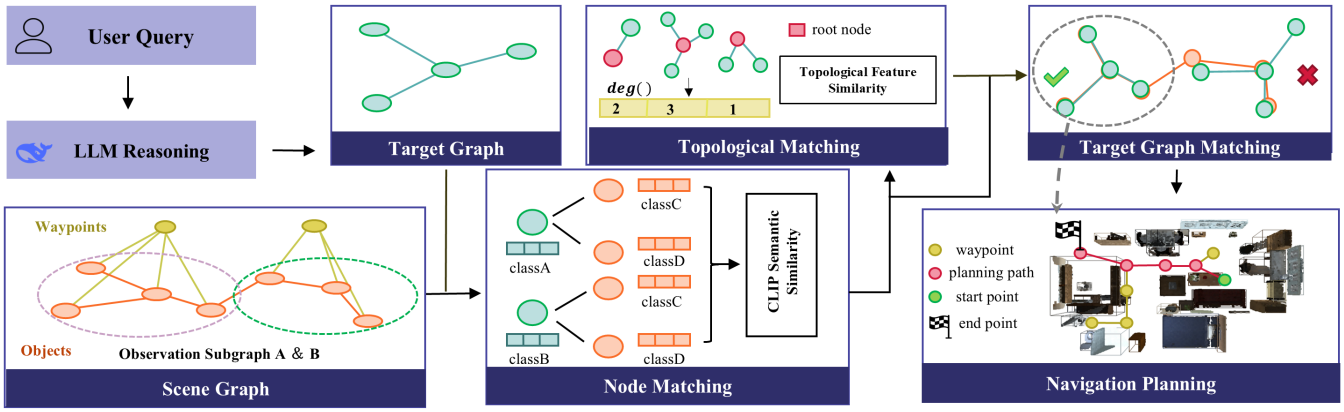

Fig. 4: Subgraph-based Language Goal Navigation.A target graph is generated via LLM-based query reasoning. Candidate subgraphs are matched using CLIP-based semantic and topological similarity. Navigation is planned based on the best match.

O_{now}, and associate it with the corresponding historical objects O_{ref}. Based on dynamic attributes and their potential impact on scene structure, objects are categorized into three levels: O_{L1}: highly dynamic objects, potentially exhibiting large-scale appearance/disappearance or intra-frame displacement;O_{L2}: objects with moderate dynamics, considered for large-scale changes only;O_{L3}: objects with low dynamic scores, assumed to be static.

For $O_{L1} \cup O_{L2} \subseteq O_{ref}$, we compute semantic similarity with O_{now}. If $s'_{sem} > \delta'_{sem}$, objects in O_{L2} are considered unchanged. For O_{L1}, additional geometric consistency checks are applied: objects are marked unchanged only if their 3D Intersection over Union(IoU) exceeds δ_{iou} and center distance is below δ_c. After matching, unmatched objects in O_{ref} are considered unobserved. Their observation edges e^{re}_{WO} are removed, and if no viewpoints observe them, they are marked as disappeared nodes N^{dis}_O. Newly observed objects are added as new dynamic nodes N^{app}_O, with relational edges e^{re}_O to nearby objects and observation edges e^{re}_{WO} from the current waypoint.

Waypoint topology maintenance: During navigation, if an obstacle is detected along the path from the current waypoint w_k to the next waypoint w_{k+1}, the connecting edge $e_{w_k,w_{k+1}}$ is marked as non-traversable and removed from the edge set E_W. If a waypoint w_i loses all its connections, it is considered isolated and removed from the node set N_W, along with its observation edges E_{WO}. To improve adaptability to environmental changes, deleted edges are retained as candidates, and active sensing is continuously performed. If a previously blocked path is verified to be clear over multiple consecutive frames, the corresponding edge is reinserted into the topology, enabling dynamic path recovery.

C. Subgraph-based Language Goal Navigation

In real-world environments, semantically similar objects are often described using contextual and relational cues. To support such queries, we propose a graph-matching-based language navigation framework that grounds descriptive language using spatial and semantic information from the scene graph, as shown in Fig.4.

Target graph generation: We utilize a LLM to transform natural language queries into structured target graphs. Queries are categorized into two types:

- **Explicit queries** specify target objects and spatial relations (e.g.,"a book on the table"). For these, the LLM extracts object categories and relational terms to construct a graph with semantic nodes and spatial edges.
- **Implicit queries** refer to functional or abstract goals (e.g., "a place suitable for reading"). The LLM infers likely object compositions and relationships through contextual reasoning.

The final target graph includes semantic nodes N_{tar}, relational edges E_{tar}, and a designated core target. Each node is encoded with CLIP embeddings to enable semantic-level matching.

Target graph matching: Each local observation subgraph associated with a waypoint is treated as a candidate. A candidate subgraph is denoted as $(N_{\text{can}}, E_{\text{can}})$, where each node $n \in N_{\text{can}}$ is equipped with a semantic embedding $f^{\text{sem}}_{\text{can}}$ and a dynamic attribute.

We perform node-level matching between the target graph and each candidate subgraph. For each node $n_{\text{tar},i}$ in the target graph $(i = 1, \ldots, N_{\text{tar}})$, we compute its semantic similarity with all nodes in the candidate subgraph and retain those exceeding a predefined threshold, forming a matching set $N^{(i)}_{\text{match}} \subseteq N_{\text{can}}$. The node-level similarity score between the target and candidate graphs is then computed as:

$$s_{\text{node}} = \frac{1}{N_{\text{tar}}} \sum_{i=1}^{N_{\text{tar}}} \max_{n \in N_{\text{can}}} S_{\text{sem}}(n_{\text{tar},i}, n) \tag{3}$$

where $S_{\text{sem}}(\cdot,\cdot)$ denotes the cosine similarity between semantic embeddings. This score reflects the overall semantic alignment between the target and candidate subgraphs at the node level.

We then compute the topological similarity between each matching node pair based on local graph structure. For each node in the semantic matching set N_{match}, we construct a topological feature vector $\vec{v}$ of length $|N_{\text{match}}|$, where each element is the degree $\deg(n)$ of a neighboring node.

The vectors are normalized, and topological similarity is computed as:

$$\vec{v} = [deg(n_1), deg(n_2), ..., deg(n_{N_{match}})] \tag{4}$$

$$s_{top} = \frac{\vec{v}_{tar} \cdot \vec{v}_{can}}{\|\vec{v}_{tar}\| \cdot \|\vec{v}_{can}\|} \tag{5}$$

The final graph similarity s_{graph} combines node and topology similarity s_{graph}=s_{node}+s_{top}. We enumerate all permutations of the matching nodes and compute s_{graph} for each, selecting the candidate subgraph with the highest similarity as the best match. The corresponding viewpoint of this subgraph is selected as the navigation goal.

Navigation Planning: Once the goal waypoint is determined, we perform path planning using Dijkstra's algorithm over the current waypoint topology. The agent follows the planned route, adjusting orientation based on angular differences between consecutive waypoints. If a path is blocked, the system prunes the invalid edge and re-plans from the previous waypoint, ensuring robust and adaptive navigation.

IV. Experimental Results

In this section, we conduct experiments in the iGibson simulation environment [31] using two representative indoor scenes. Both scenes contain a diverse set of static and dynamic objects with rich dynamic properties, providing a realistic and comprehensive simulation of real-world environments.

These objects vary in size, attributes, and dynamic behaviors. A LoCoBot mobile robot is used as the navigation agent, collecting RGB-D sequences throughout the simulation. The evaluation focuses on three core modules: dynamic scene graph construction, scene graph updating, and language goal navigation. The experiments aim to address the following two key research questions:

1) How accurately can our framework construct open-vocabulary scene graphs in dynamic indoor environments, capturing both static and dynamic objects along with their semantic and geometric properties?
2) How robustly can the constructed scene graphs be incrementally updated to adapt to environmental changes and recover from early perception errors?
3) How reliably can our framework perform language-based navigation queries, retrieving target locations in evolving scenes?

A. Dynamic Scene Adaption and Scene Graph Construction

We first assess the framework's performance in dynamic scene graph construction. For static objects, each detected instance is matched to a ground truth object as a positive sample. For dynamic objects-categorized as appearing, disappearing, or moving-only detections with the correct dynamic state are treated as true positives. Using two representative indoor scenes, we compare our method with ConceptGraphs and DovSG, evaluating geometric consistency, semantic accuracy, and dynamic state classification. Precision, recall, and F1 scores for both static and dynamic object construction are reported in Table I.

TABLE I: Performance Comparison on Dynamic Scene Graph Construction

Scene	Method	Static			Dynamic		
		Pre(%)	Rec(%)	F1(%)	Pre(%)	Rec(%)	F1(%)
Ihlen_1_int	ConceptGraphs [8]	90.2	90.2	90.2	40.0	40.0	40.0
	DovSG [15]	90.2	**91.5**	90.8	75.0	75.0	75.0
	DSSM-SG	**94.2**	89.0	**91.5**	**77.8**	**87.5**	**82.4**
Rs_int	ConceptGraphs [8]	82.1	62.2	70.8	50.0	50.0	50.0
	DovSG [15]	81.8	75.0	74.2	75.0	50.0	60.0
	DSSM-SG	**83.3**	**75.8**	**79.4**	**83.3**	**83.3**	**83.3**

Our method achieves superior performance in both static and dynamic object construction tasks. For static objects, instead of relying solely on large language models to describe the central object in an image, we aggregate semantic information through confidence-based voting across multiple detections. This strategy ensures higher accuracy and recall by preserving more reliable and complete semantic cues. In terms of dynamic objects, ConceptGraphs lacks the capability to handle dynamic environments and can only incrementally add newly detected objects. DovSG employs voxel-based back-projection to detect scene changes by comparing current observations with the voxel map, but its update mechanism is limited to instance-level additions and deletions, without modeling actual object dynamics. In contrast, our method performs dynamic change detection and object association to classify and track different dynamic states, enabling more accurate perception and construction of dynamic scene graphs.

We further evaluate the construction performance across different dynamic object states. Dynamic objects from both scenes are unified for analysis, reported in Table II. The consistently high recall indicates comprehensive detection across all dynamic types. Notably, our object association module achieves 100% precision in identifying moving objects. In contrast, precision for newly appeared and disappeared objects is comparatively lower. Although techniques such as windowed observation and confidence-based filtering improve robustness, detection instability still affects change recognition. Nonetheless, our approach demonstrates strong capability in representing and distinguishing dynamic objects.

B. Re-navigation Graph Update

Re-navigation graph update is a key capability for adapting to dynamic environments. Based on the constructed dynamic open-vocabulary 3D scene graphs, we collect two observation sequences of similar length from scenes undergoing further dynamic changes. We assess the scene graph accuracy before and after the changes, as well as after each valid update triggered by object-level matching. To better reflect

TABLE II: Performance of Dynamic State Classification for Different Object Changes

Dynamic State	Pre (%)	Rec (%)	F1 (%)
Appear	57.1	80.0	66.6
Disappear	50.0	75.0	60.0
Relocation	100.0	80.0	88.9
Total	64.7	78.5	70.9

the impact of re-navigation updates and ensure comparability across scenes, we introduce a relative accuracy metric by normalizing each update's accuracy gain with respect to the total accuracy gap between pre- and post-change graphs:

$$\Delta Acc = Acc_{\text{prev}} - Acc_{\text{new}} \tag{6}$$

$$Acc_{\text{rel}} = \frac{Acc_i - Acc_{\text{new}}}{\Delta Acc} \tag{7}$$

Let Acc_{prev} and Acc_{new} denote the scene graph accuracy before and after scene changes, and Acc_i the accuracy after the $i-th$ valid update. Acc_{rel} is the relative accuracy after normalizing. Results are shown in Fig.5.

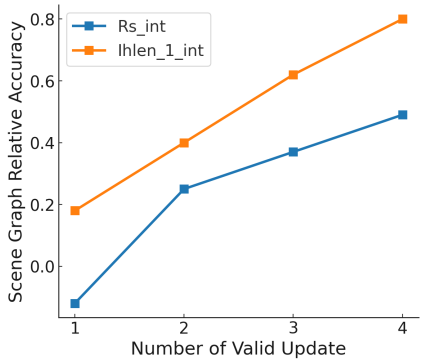

Fig. 5: Relative Scene Graph Accuracy via Re-navigation Updates. Each valid update incrementally improves the normalized accuracy, highlighting the effectiveness of the proposed re-navigation update mechanism in recovering dynamic scene graphs.

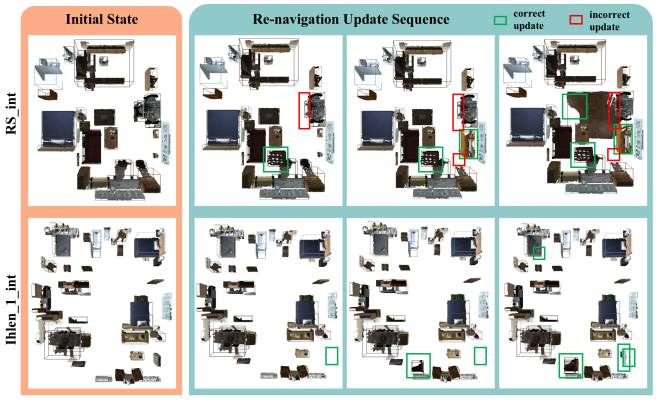

Fig. 6: Visualization of Re-navigation Dynamic Scene Graph Update

As shown in Fig.5 and 6, the system's ability to adapt to scene changes improves steadily with each update iteration. This trend indicates that the proposed incremental update mechanism effectively supports error correction and structural recovery. Through continuous observations, it incrementally mitigates early construction errors and progressively reconstructs a graph structure closer to the real environment.

Moreover, we evaluate three key metrics during re-navigation in the Rs_int and Ihlen_1_int scenes: the dynamic update success rate, the static retention success rate, and the scene graph accuracy gain. As shown in Table III, our method achieves near-complete reconstruction for all changed dynamic objects, demonstrating robust responsiveness and adaptation. Meanwhile, static objects are preserved with high consistency, avoiding erroneous modifications and ensuring the structural stability and semantic continuity of the scene graph across sequential tasks.

TABLE III: Performance of re-navigation update metrics across scenes.

Metric	Rs_int	Ihlen_1_int
Dynamic Update Success Rate	90%	100.0%
Static Retention Success Rate	93.3%	100.0%
Scene Graph Accuracy Gain	8.9%	5.0%

C. Open-vocabulary Object Retrieval

To demonstrate the advantages of our graph-based language goal matching framework, we conduct retrieval experiments on two types of textual queries:

- **Explicit Queries**: Sentences that specify a target object and its spatial relations with surrounding objects (e.g.,"a laptop on the table between a sofa and a stool").
- **Implicit Queries**: Vague descriptions that reflect scene functionality or characteristics without specifying concrete objects (e.g.,"a bright place suitable for working").

In the two indoor scenes (Ihlen_1_int and Rs_int), we construct 20 queries per type, each associated with one or more ground-truth viewpoints. The queries are input into three methods-CLIP image features, ConceptGraphs, and our approach-to retrieve the top-1, top-2, and top-3 most relevant views. Table IV summarizes the results in terms of whether the target viewpoints are included in the top-ranked outputs.

TABLE IV: Retrieval performance (R@K) for two query types in scenes Rs_int and Ihlen_1_int

Scene	Query Type	Method	R@1	R@2	R@3
Rs_int / Ihlen_1_int	Explicit Query	CLIP-based [26]	**0.55**	0.60	0.65
		ConceptGraphs [8]	0.45	0.65	0.70
		Ours	**0.55**	**0.80**	**0.85**
	Implicit Query	CLIP-based [26]	0.45	0.50	0.65
		ConceptGraphs [8]	0.30	0.35	0.45
		Ours	**0.65**	**0.70**	**0.70**

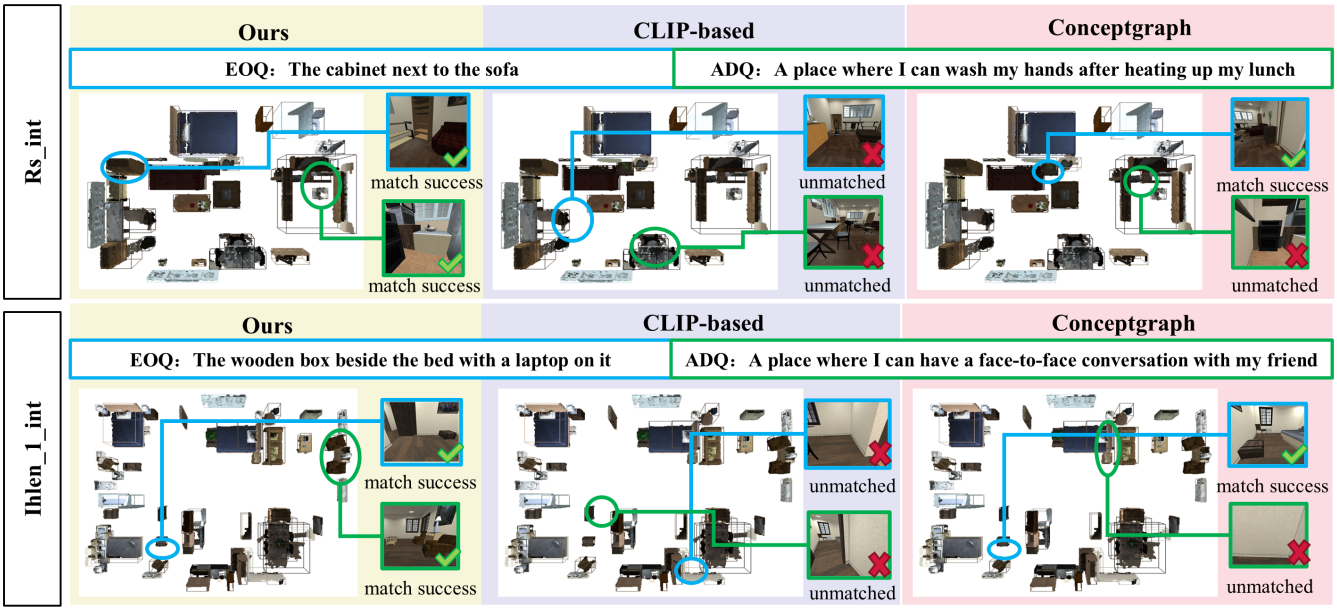

Fig. 7: Visualization of retrieval performance for two query types in dynamic scenes

For explicit queries, CLIP underperforms due to its lack of spatial relational reasoning, and ConceptGraphs yields better results. Our method further improves upon this by dynamically associating objects and tracking their positions before and after scene changes, enabling precise identification of moved objects and distinction of similar objects in different contexts.

For implicit queries, ConceptGraphs performs poorly due to its object-centric representation and lack of compositional scene understanding, while CLIP fails to capture semantic subtleties. In contrast, our method leverages large language model-based associative reasoning to infer potential object compositions, leading to significantly improved matching accuracy.

Thanks to the dynamic spatio-semantic memory, our method remains robust in dynamic environments. When queries refer to vanished objects, the system recalls their pre-change positions; when target objects have moved, the similarity score derived from matched subgraphs is transferred to the updated observation. This allows the system to localize targets despite positional shifts, demonstrating superior reasoning and matching capability under language-based queries in evolving scenes.

D. Language Goal Navigation Application

To assess our framework's adaptability to dynamic tasks, we conduct a language-guided navigation experiment in the dynamic iGibson scene Ihlen_1_int. The system initializes by constructing a dynamic open-vocabulary scene graph and receives the ambiguous instruction: "Find me a place suitable for washing up." A large language model infers relevant concepts (e.g., sink, mirror), guiding the generation of a goal subgraph. Through joint semantic-topological matching, the system localizes the restroom and plans a path accordingly. During navigation, a large piano appears, blocking

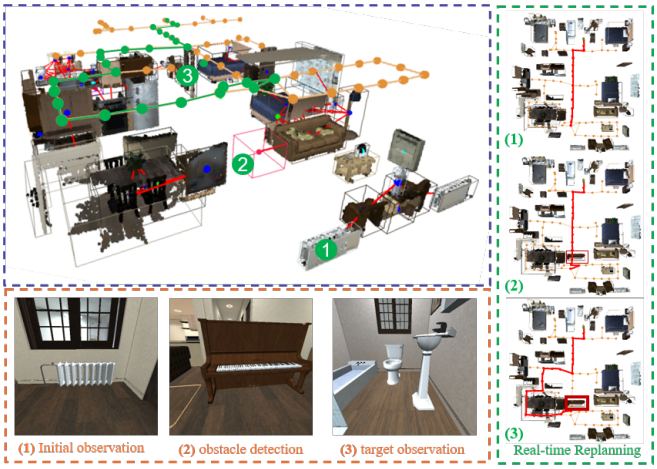

Fig. 8: Visualization of Dynamic Language Goal Navigation Process.

the original route. The system detects the obstacle, updates the object and topological layers, and re-plans an alternative path when local rerouting fails. Throughout execution, the graph is incrementally updated, and the agent adapts its route to reach the target. This experiment demonstrates the framework's robustness in grounding ambiguous language, tracking scene changes, and maintaining reliable navigation through dynamic environments. The full process is illustrated in Fig.8.

V. CONCLUSIONS

We propose DSSM-SG, a dynamic open-vocabulary 3D scene graph system enhanced with spatial-semantic memory, enabling continuous updating of multi-layered scene representations in dynamic indoor environments. Our framework supports robust parsing of ambiguous language goals and facilitates reliable navigation under changing conditions. We introduce a viewpoint-based dynamic change detection and

object state modeling mechanism, along with a spatial-semantic memory module that captures both semantic and geometric evolution. To ensure consistency and responsiveness, we design a hierarchical graph update strategy synchronizing object-level, relational, and topological layers. Furthermore, we develop a graph-based language navigation approach leveraging large language model-driven reasoning to ground complex queries and plan paths under dynamic topology. Experiments show DSSM-SG significantly outperforms prior methods in accuracy, update efficiency, and language-driven navigation.

REFERENCES

[1] I. Armeni, Z.-Y. He, J. Gwak, A. R. Zamir, M. Fischer, J. Malik, and S. Savarese, "3d scene graph: A structure for unified semantics, 3d space, and camera," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2019, pp. 5664–5673.

[2] N. Hughes, Y. Chang, S. Hu, *et al.*, "Foundations of spatial perception for robotics: hierarchical representations and real-time systems.(2023)," *arXiv preprint arXiv:2305.07154*.

[3] S.-C. Wu, K. Tateno, N. Navab, and F. Tombari, "Incremental 3d semantic scene graph prediction from rgb sequences," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2023, pp. 5064–5074.

[4] S. Koch, N. Vaskevicius, M. Colosi, P. Hermosilla, and T. Ropinski, "Open3dsg: Open-vocabulary 3d scene graphs from point clouds with queryable objects and open-set relationships," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 14 183–14 193.

[5] Y. Mehan, K. Gupta, R. Jayanti, A. Govil, S. Garg, and M. Krishna, "Questmaps: Queryable semantic topological maps for 3d scene understanding," in *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2024, pp. 13 311–13 317.

[6] J. Strader, N. Hughes, W. Chen, A. Speranzon, and L. Carlone, "Indoor and outdoor 3d scene graph generation via language-enabled spatial ontologies," *IEEE Robotics and Automation Letters*, vol. 9, no. 6, pp. 4886–4893, 2024.

[7] H. Chang, K. Boyalakuntla, S. Lu, S. Cai, E. Jing, S. Keskar, S. Geng, A. Abbas, L. Zhou, K. Bekris, *et al.*, "Context-aware entity grounding with open-vocabulary 3d scene graphs," *arXiv preprint arXiv:2309.15940*, 2023.

[8] Q. Gu, A. Kuwajerwala, S. Morin, K. M. Jatavallabhula, B. Sen, A. Agarwal, C. Rivera, W. Paul, K. Ellis, R. Chellappa, *et al.*, "Conceptgraphs: Open-vocabulary 3d scene graphs for perception and planning," in *2024 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2024, pp. 5021–5028.

[9] A. Werby, C. Huang, M. Büchner, A. Valada, and W. Burgard, "Hierarchical open-vocabulary 3d scene graphs for language-grounded robot navigation," in *First Workshop on Vision-Language Models for Navigation and Manipulation at ICRA 2024*, 2024.

[10] T. Steinke, M. Büchner, N. Vödisch, and A. Valada, "Collaborative dynamic 3d scene graphs for open-vocabulary urban scene understanding," *arXiv preprint arXiv:2503.08474*, 2025.

[11] F. Pomerleau, P. Krüsi, F. Colas, P. Furgale, and R. Siegwart, "Long-term 3d map maintenance in dynamic environments," in *2014 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2014, pp. 3712–3719.

[12] G. Kim and A. Kim, "Lt-mapper: A modular framework for lidar-based lifelong mapping," in *2022 International Conference on Robotics and Automation (ICRA)*. IEEE, 2022, pp. 7995–8002.

[13] L. Schmid, M. Abate, Y. Chang, and L. Carlone, "Khronos: A unified approach for spatio-temporal metric-semantic slam in dynamic environments," *arXiv preprint arXiv:2402.13817*, 2024.

[14] P. Liu, Z. Guo, M. Warke, S. Chintala, C. Paxton, N. M. M. Shafiullah, and L. Pinto, "Dynamem: Online dynamic spatio-semantic memory for open world mobile manipulation," *arXiv preprint arXiv:2411.04999*, 2024.

[15] Z. Yan, S. Li, Z. Wang, L. Wu, H. Wang, J. Zhu, L. Chen, and J. Liu, "Dynamic open-vocabulary 3d scene graphs for long-term language-guided mobile manipulation," *IEEE Robotics and Automation Letters*, 2025.

[16] Y. Tang, M. Wang, Y. Deng, Z. Zheng, J. Deng, S. Zuo, and Y. Yue, "Openin: Open-vocabulary instance-oriented navigation in dynamic domestic environments," *IEEE Robotics and Automation Letters*, no. 99, pp. 1–8, 2025.

[17] E. Wijmans, A. Kadian, A. Morcos, S. Lee, I. Essa, D. Parikh, M. Savva, and D. Batra, "Dd-ppo: Learning near-perfect pointgoal navigators from 2.5 billion frames," *arXiv preprint arXiv:1911.00357*, 2019.

[18] R. Ramrakhya, E. Undersander, D. Batra, and A. Das, "Habitat-web: Learning embodied object-search strategies from human demonstrations at scale," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022, pp. 5173–5183.

[19] D. S. Chaplot, D. P. Gandhi, A. Gupta, and R. R. Salakhutdinov, "Object goal navigation using goal-oriented semantic exploration," *Advances in Neural Information Processing Systems*, vol. 33, pp. 4247–4258, 2020.

[20] A. Majumdar, G. Aggarwal, B. Devnani, J. Hoffman, and D. Batra, "Zson: Zero-shot object-goal navigation using multimodal goal embeddings," *Advances in Neural Information Processing Systems*, vol. 35, pp. 32 340–32 352, 2022.

[21] M. Khanna, R. Ramrakhya, G. Chhablani, S. Yenamandra, T. Gervet, M. Chang, Z. Kira, D. S. Chaplot, D. Batra, and R. Mottaghi, "Goat-bench: A benchmark for multi-modal lifelong navigation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 16 373–16 383.

[22] P. Anderson, Q. Wu, D. Teney, J. Bruce, M. Johnson, N. Sünderhauf, I. Reid, S. Gould, and A. Van Den Hengel, "Vision-and-language navigation: Interpreting visually-grounded navigation instructions in real environments," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 3674–3683.

[23] C. Huang, O. Mees, A. Zeng, and W. Burgard, "Visual language maps for robot navigation," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2023, pp. 10 608–10 615.

[24] B. Li, K. Q. Weinberger, S. Belongie, V. Koltun, and R. Ranftl, "Language-driven semantic segmentation," *arXiv preprint arXiv:2201.03546*, 2022.

[25] D. Shah, B. Osiński, S. Levine, *et al.*, "Lm-nav: Robotic navigation with large pre-trained models of language, vision, and action," in *Conference on robot learning*. PMLR, 2023, pp. 492–504.

[26] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, *et al.*, "Learning transferable visual models from natural language supervision," in *International conference on machine learning*. PmLR, 2021, pp. 8748–8763.

[27] B. Yu, Y. Liu, L. Han, H. Kasaei, T. Li, and M. Cao, "Vln-game: Vision-language equilibrium search for zero-shot semantic navigation," *arXiv preprint arXiv:2411.11609*, 2024.

[28] Y. Zhang, X. Huang, J. Ma, Z. Li, Z. Luo, Y. Xie, Y. Qin, T. Luo, Y. Li, S. Liu, *et al.*, "Recognize anything: A strong image tagging model," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 1724–1732.

[29] T. Cheng, L. Song, Y. Ge, W. Liu, X. Wang, and Y. Shan, "Yolo-world: Real-time open-vocabulary object detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 16 901–16 911.

[30] A. Kirillov, E. Mintun, N. Ravi, H. Mao, C. Rolland, L. Gustafson, T. Xiao, S. Whitehead, A. C. Berg, W.-Y. Lo, *et al.*, "Segment anything," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2023, pp. 4015–4026.

[31] C. Li, F. Xia, R. Martín-Martín, M. Lingelbach, S. Srivastava, B. Shen, K. E. Vainio, C. Gokmen, G. Dharan, T. Jain, A. Kurenkov, K. Liu, H. Gweon, J. Wu, L. Fei-Fei, and S. Savarese, "igibson 2.0: Object-centric simulation for robot learning of everyday household tasks," in *Proceedings of the 5th Conference on Robot Learning*, ser. Proceedings of Machine Learning Research, A. Faust, D. Hsu, and G. Neumann, Eds., vol. 164, 2022, pp. 455–465.