MASTER THESIS

Thesis submitted in fulfillment of the requirements for the degree of Master of Science in Engineering at the University of Applied Sciences Technikum Wien - Degree Program Mechatronics/Robotics

SAGE: Multi object semantic aware guided exploration with persistent memory

By: Kevin Eppacher, BSc

Student Number: 2310331013

Supervisor: Simon Schwaiger, MSc

Vienna, December 9, 2025

This work was conducted in the context of the project “Stadt Wien Kompetenzteam für Drohnentechnik in der Fachhochschulausbildung” (project number MA23 35-02, financed by the Department MA23 for Economic Affairs, Labour and Statistics of the City of Vienna).

Funded by

City of Vienna | Economic Affairs, Labour and Statistics

Declaration

“As author and creator of this work to hand, I confirm with my signature knowledge of the relevant copyright regulations governed by higher education acts (see Urheberrechtsgesetz /Austrian copyright law as amended as well as the Statute on Studies Act Provisions / Examination Regulations of the UAS Technikum Wien as amended).

I hereby declare that I completed the present work independently and that any ideas, whether written by others or by myself, have been fully sourced and referenced. I am aware of any consequences I may face on the part of the degree program director if there should be evidence of missing autonomy and independence or evidence of any intent to fraudulently achieve a pass mark for this work (see Statute on Studies Act Provisions / Examination Regulations of the UAS Technikum Wien as amended).

I further declare that up to this date I have not published the work to hand nor have I presented it to another examination board in the same or similar form. I affirm that the version submitted matches the version in the upload tool.“

Vienna, December 9, 2025 Signature

Kurzfassung

Hello, here is some text without a meaning. This text should show what a printed text will look like at this place. If you read this text, you will get no information. Really? Is there no information? Is there a difference between this text and some nonsense like “Huardest gefburn”? Kjift – not at all! A blind text like this gives you information about the selected font, how the letters are written and an impression of the look. This text should contain all letters of the alphabet and it should be written in of the original language. There is no need for special content, but the length of words should match the language.

Schlagworte: Keyword1, Keyword2, Keyword3, Keyword4

Abstract

Hello, here is some text without a meaning. This text should show what a printed text will look like at this place. If you read this text, you will get no information. Really? Is there no information? Is there a difference between this text and some nonsense like “Huardest gefburn”? Kjift – not at all! A blind text like this gives you information about the selected font, how the letters are written and an impression of the look. This text should contain all letters of the alphabet and it should be written in of the original language. There is no need for special content, but the length of words should match the language.

Keywords: Keyword1, Keyword2, Keyword3, Keyword4

Acknowledgements

Hello, here is some text without a meaning. This text should show what a printed text will look like at this place. If you read this text, you will get no information. Really? Is there no information? Is there a difference between this text and some nonsense like "Huardest gefburn"? Kjift – not at all! A blind text like this gives you information about the selected font, how the letters are written and an impression of the look. This text should contain all letters of the alphabet and it should be written in of the original language. There is no need for special content, but the length of words should match the language.

Contents

1 Introduction

Domain and Relevance:

- **Service Robotics:** Vacuum robot searching for a fridge in the kitchen using a semantic input prompt.
- **Search and Rescue (SAR):** Deployment of one or multiple robots to locate persons, e.g., with the hint that they may be in the bathroom.
- **Inspection:** Robots autonomously exploring areas not accessible via teleoperation, guided by semantic input.
- **Warehouse Automation:** Large warehouses requiring robots to locate items, pallets, or storage units that may not be consistently labeled or may be partially occluded.

1.1 Problem Statement

Problem: Classic geometric exploration methods often lead to inefficient search behavior, as they are primarily designed for mapping unexplored environments rather than for goal-directed semantic search tasks.

Solution: Incorporating semantic reasoning into exploration by using RGB camera input to associate geometric frontiers with language-based meaning and task relevance.

Current solutions:

Approach	**Training Required**	**Real-Time**	**Semantic Reasoning Model**
VLFM	✗ (zero-shot)	✓	BLIP-2 + GroundingDINO + SAM
SemUtil	✗ (training-free)	✓	Mask R-CNN + CLIP + BERT
ESC	✗ (zero-shot)	✓	GLIP + DeBERTa / ChatGPT reasoning
LGX	✗ (zero-shot)	✓	GPT-3 + GLIP + BLIP
CoW	✗ (zero-shot)	✓	CLIP similarity scoring
ZSON	✓ (RL pretraining)	✗	CLIP-based RL policy
PONI	✓ (supervised)	✗	Learned potential-field network
PIRLNav	✓ (BC + RL)	✗	DINO-based CNN-RNN policy

Table 1: Overview of Semantic Exploration Approaches without Persistent Memory

None of the above methods incorporate a persistent semantic memory for efficient multi-object navigation and are often computationally demanding.

Problem:

- Lack of persistent memory in semantic exploration frameworks limits long-term efficiency in multi-object search tasks.
- Existing methods often require extensive offline training, reducing adaptability to new environments.
- Deep Reinforcement Learning (DRL) approaches are computationally expensive and lack persistent memory for effective multi-object goal search.

Current solutions:

Approach	**Training Required**	**Real-Time**	**Semantic Reasoning Model**
VLFM	✗ (zero-shot)	✓	BLIP-2 + GroundingDINO + SAM
SemUtil	✗ (training-free)	✓	Mask R-CNN + CLIP + BERT
ESC	✗ (zero-shot)	✓	GLIP + DeBERTa / ChatGPT reasoning
LGX	✗ (zero-shot)	✓	GPT-3 + GLIP + BLIP
CoW	✗ (zero-shot)	✓	CLIP similarity scoring
ZSON	✓ (RL pretraining)	✗	CLIP-based RL policy
PONI	✓ (supervised)	✗	Learned potential-field network
PIRLNav	✓ (BC + RL)	✗	DINO-based CNN-RNN policy

Table 2: Overview of Semantic Zero-Shot and Trained Exploration Approaches

Despite notable progress, these methods often suffer from heavy GPU requirements, offline training procedures, or noisy semantic maps that reduce navigation reliability. Some further depend heavily on object detectors, which can introduce false positives in open-vocabulary settings.

Core Gaps in Existing Work:

Limitation	Example Works	Implication
No persistent memory	VLFM, CoW, LGX, ESC	No long-term fusion or recall; repeated exploration of known areas.
Offline training required	ZSON, PONI, PIRLNav, SemExp	Heavy RL/supervised training; poor adaptability to new scenes.
No balance between exploration and memory	OneMap, RayFronts, VLMaps	Either passive mapping or short-term exploration; inefficient search.
No zero-shot exploration	VLFM, CoW, LGX	Detect novel objects but fail to explore unseen regions strategically.
Premapping needed	ConceptGraphs, VLMaps, GeFF	Depend on pre-recorded data; not suited for online autonomy.
Limited robustness	PONI, SemExp, PIRLNav	Closed-set categories; fragile under real-world variation.
Low real-world applicability	ConceptGraphs, VLMaps	High GPU cost or simulation-only; not deployable on mobile robots.

Table 3: Identified Gaps in Existing Semantic Exploration Frameworks

Derived Requirements from the State of the Art:

- Zero-shot frontier exploration framework.
- Persistent 3D semantic mapping with confidence-based fusion.
- Use of pretrained models only (no additional training required).
- Hybrid fusion strategy to balance exploration and memory.
- Multi-source detection fusion for robust object identification.
- GPU-efficient design for real-time deployment on mobile robots.
- Modular architecture for easy adaptation and extension.
- Independence from fixed, pre-trained object category sets.

Scientific Contribution

This work contributes to the state of the art by introducing a hybrid semantic exploration framework that integrates zero-shot semantic frontier scoring with persistent 3D scene representation, enabling autonomous search guided by open-vocabulary text queries. The system combines real-time semantic reasoning during exploration with a long-term spatial memory, allowing the robot to dynamically balance between discovering new information and exploiting previously acquired knowledge.

Unlike previous approaches that focus exclusively on either geometric frontiers or static semantic maps, the proposed framework continuously fuses information from multiple semantic sources to maintain a unified, confidence-based world model. Adaptive weighting enables the robot to adjust its behavior between exploration and exploitation according to the reliability of recent observations and the stability of stored semantic memory.

The framework further investigates how the quality and granularity of the underlying semantic information influence task success, navigation efficiency, and robustness. By systematically varying the trust between exploration and memory components, this work provides new insights into how semantic reasoning and persistent mapping can be effectively combined for open-vocabulary, multi-object search in dynamic environments.

To evaluate the contribution of the proposed system, the following research questions are formulated:

1. **How does integrating zero-shot semantic exploration and persistent 3D semantic mapping affect multi-object search performance and navigation efficiency compared to existing methods?**
   Metrics: Performance is quantified in terms of task success and path efficiency, measured through Success Rate (SR), Success per Path Length (SPL), and Multi-Object Success Rate (MSR) relative to representative state-of-the-art systems such as OneMap, VLFM, and Pigeon.

2. **How does the interaction between live exploration and accumulated semantic memory influence overall system performance?**
   Metrics: The weighting factor between exploration and memory is varied during graph node fusion to assess impacts on SR and SPL, identifying optimal trade-offs between reactivity and exploitation.

3. **How does the granularity of semantic map retrieval affect map quality, and can dynamic weighting between exploration and memory compensate for potential noise?**
   Metrics: The semantic granularity in the 3D semantic mapper is varied while adjusting exploration weight to evaluate effects on SR and SPL.

4. **How does multi-source fusion of detection confidence, semantic similarity, and memory confidence impact detection robustness and false-positive suppression during exploration?**
   Metrics: Precision, Recall, F1-Score, Confusion Matrix, and SR under different fusion weight configurations across COCO, open-vocabulary, and zero-shot classes.

5. **What is the computational footprint and real-world robustness of the hybrid framework?**

Metrics: Frames per second (FPS), GPU/CPU usage, inference latency, and detection stability under sensor noise during physical deployment on a mobile robot.

These research questions guide the design of the experimental evaluation, where each question is systematically addressed through targeted ablation studies, comparative benchmarks, and real-world validation presented in Chapter **??**.

1.2 Thesis Structure

2 State of the Art

In this chapter, the current state of research in semantic multi-object search, map reconstruction, and object detection is reviewed. The goal is to identify strengths and limitations of existing methods and establish the technological context for the proposed hybrid approach. The chapter is divided into three key areas: approaches for searching multiple objects semantically, techniques for building and maintaining persistent semantic maps, and recent advances in object detection and promptable models for open-vocabulary tasks.

2.1 Geometric Exploration

- Geometric exploration aims to explore unknown environments by navigating toward frontiers — the boundaries between known and unknown space — typically within a SLAM-based mapping framework.
- Selecting the frontier closest to the robot minimizes path cost (*greedy strategy*) [1]. *Advantage:* simple and efficient. *Limitation:* ignores potential information gain.
- Prioritizing frontiers based on maximum expected information gain (e.g., entropy, mutual information) enables more informative exploration [2, 3].
- Selecting the largest frontier clusters improves stability and continuity in exploration [4].
- These approaches are purely geometric and lack semantic awareness. They are not suitable for open-vocabulary object queries or multi-object search tasks requiring semantic reasoning.

2.2 Vision-Language-Guided Exploration

- Review of methods targeting simultaneous or sequential search for multiple objects in unknown environments.
- Analysis of ...
  - VLFM [5]
  - SemUtil [**chen2023semutil**]
  - ESC [**liu2023esc**]
  - LGX [**shah2024lgx**]
  - CoW [**qi2023cow**]
  - ZSON [**shah2023zson**]
  - PONI [**qi2024poni**]
  - PIRLNav [**ma2024pirlnav**]

  regarding aspects such as:
  - Training Required (Pretrained vs. Fine-tuned)
  - Real-Time Capability
  - VRAM Requirements
  - Sensor Modalities (RGB, RGB-D, LiDAR)
  - Semantic reasoning (BLIP2, CLIP, GPT-4V, etc.)
  - Persistent Memory (None, 2D Map, 3D Map, Scene Graph)
- Evaluation of performance metrics used in multi-object search:
  - Success Rate (SR)
  - Success weighted by Path Length (SPL)
  - MSR (Multi-object Success Rate)
- Discussion of semantic exploration frameworks combining language models with spatial reasoning.
- Challenges of maintaining semantic context across multiple targets.

2.3 Map Reconstruction and Persistent Semantic Mapping

- Overview of approaches to build and update semantic maps during exploration:
  - 2D grid maps
  - Pointclouds
  - Voxel grids

- Octomaps
- Scene graphs
- Neural Radiance Fields (NeRFs)
- Feature Fields

- Techniques for fusing sensor data into persistent 2D/3D representations:
  - Storing Visual Embeddings (e.g., CLIP features) in 3D maps for semantic querying.
  - Incremental updating of semantic labels based on new observations.
  - Handling uncertainty and conflicting detections over time.
- Comparison of representations (Octomaps, point clouds, voxel grids) in terms of:
  - Memory efficiency.
  - Ability to store semantic labels persistently.
- Discussion of ...
  - ConceptGraphs
  - ConceptGraph-Online
  - OpenFusion
  - Clio
  - OpenScene
  - GeFF
  - CLIP-Fields
  - ConceptFusion
  - VLMaps
  - LERF

  as examples of global 3D semantic maps.
- Limitations in updating or correcting the map after wrong detections.

2.4 Object Detection and Promptable Models

- Review of traditional and open-vocabulary object detection methods.
- Analysis of grounding-capable detectors and segmentation models for zero-shot tasks.
- Specific evaluation of the following models for their suitability in semantic multi-object search:
  - YOLOv7
  - GroundingDINO

- MobileSAM
  - GroundedSAM
  - SEEM
  - OWL-ViT
  - MaskDINO

- Discussion of promptable vision-language models supporting multi-modal queries (text, image, audio).
- Challenges with false positives in zero-shot settings and their implications for reliable multi-object detection.

3 Methods

This chapter details the methods developed for semantic exploration, persistent 3D mapping, promptable object detection, and robust fusion strategies for multi-object search.

3.1 System Overview

- Presentation of the overall architecture of the exploration (3.2), detection (3.4), mapping (3.3), and fusion (3.5) pipeline.
- Description of data flow between exploration (frontier evaluation), detection (promptable models), and exploitation (persistent semantic mapping), as shown in Figure 4.
- Explanation of how exploration and mapping components interact to progressively build a semantic understanding of the environment.

3.2 Semantic Frontier Exploration

Exploration 2D Occupancy Map

- The SLAM map is used for navigation.
- Generating frontiers for each prompt would require deleting and rebuilding the SLAM map.
- This approach is inefficient and impractical for navigation.

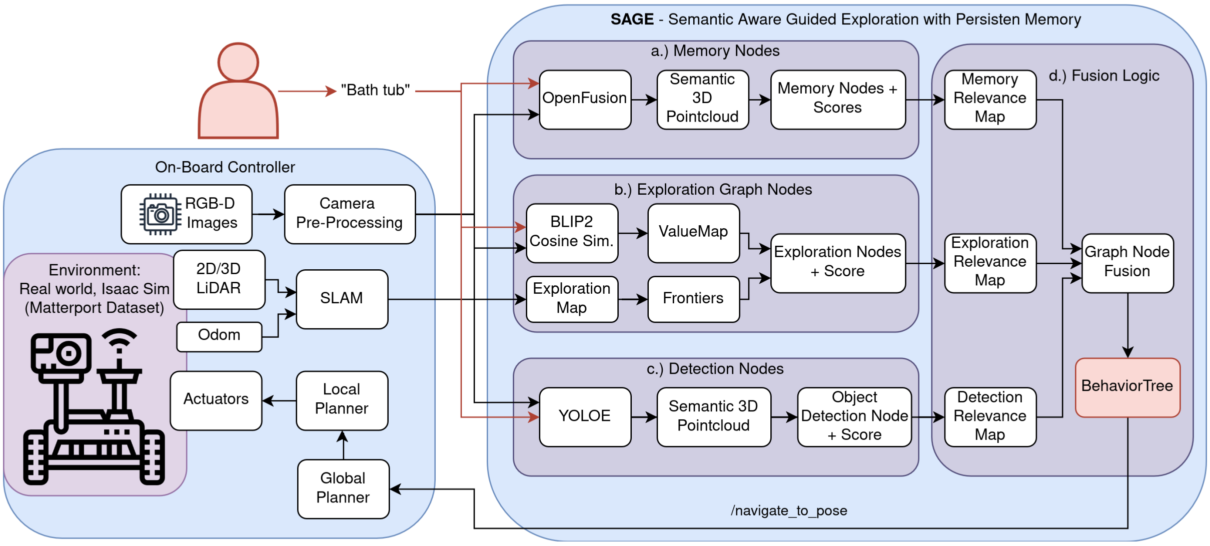

Figure 1: System architecture

- Therefore, a separate 2D occupancy grid is created exclusively for exploration.
- The exploration map is constructed by:
  - collecting robot poses, and
  - performing raytracing against the parent SLAM map.
- For each new semantic prompt:
  - all stored poses are cleared, and
  - the exploration occupancy grid is rebuilt from scratch.

Frontier Detection and Calculation

- Detection of frontiers on a 2D occupancy grid to identify candidate regions for exploration.
- Application of classical frontier-based exploration algorithms extended with semantic information.

Value Map Generation using Vision-Language Models

- Computation of value maps by evaluating cosine similarity between text queries and scene observations.
- Dynamic update of value maps as new observations are integrated.

3.3 Persistent Semantic 3D Mapping

Global Map Construction with Open-Fusion

- Incremental creation of a global semantic point cloud map integrating RGB-D observations over time.

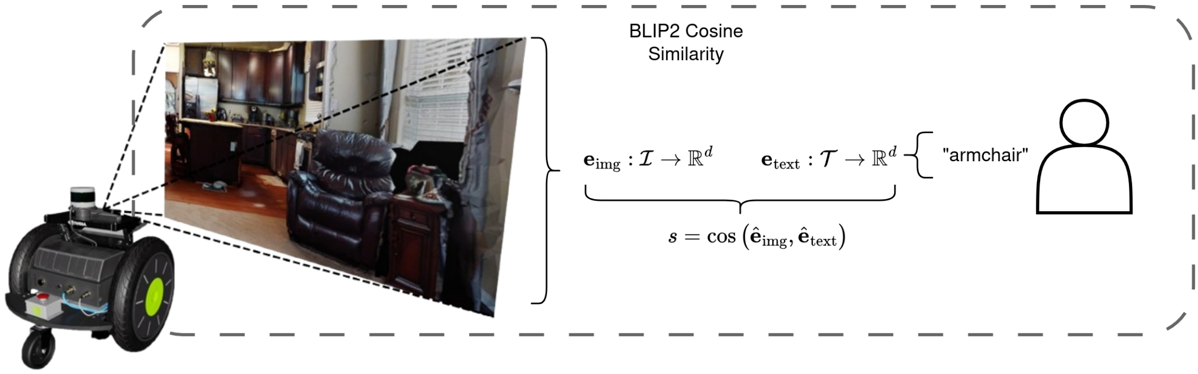

Figure 2: System architecture

- Registration of observations using robot poses to maintain a consistent world representation.
- Association of semantic labels with 3D points based on query relevance scores.

Semantic Clustering and Graph Node Generation

- Clustering of points with similar semantic labels to form object-level hypotheses.
- Construction of semantic graph nodes representing detected object instances with aggregated confidence scores.
- Maintenance of the semantic graph as a persistent memory for multi-object search tasks.

3.4 Promptable Zero-Shot Detection

- In this work YOLO-E [**yolo_e**] is used as the promptable zero-shot detection model.
- YOLO-E has the following advantages:
  - High inference speed suitable for real-time applications.
  - Ability to handle open-vocabulary object detection based on text prompts.
  - Integration of both visual and textual information for robust detection.
  - Pre-trained on large-scale datasets, enabling zero-shot generalization to unseen object categories.

Open-Vocabulary Object Detection with YOLO-E

- Utilization of the YOLO-E model for open-vocabulary object detection based on text prompts.
- Extraction of 2D bounding boxes and associated confidence scores for detected objects.
- Segmentation of detected objects to isolate relevant pixels for 3D localization.

Depth-Based 3D Localization

- With camera intrinsics and depth information, the 2D bounding boxes are projected into 3D space.
- Calculation of 3D coordinates for each detected object using depth values within the bounding box.
- Semantic detection pointclouds are passed are then clustered and the centroid of each cluster is computed to obtain robust 3D object locations.
- For each cluster, the mean of the confidence scores of the associated 2D detections is calculated to assign a confidence score to the 3D localization.

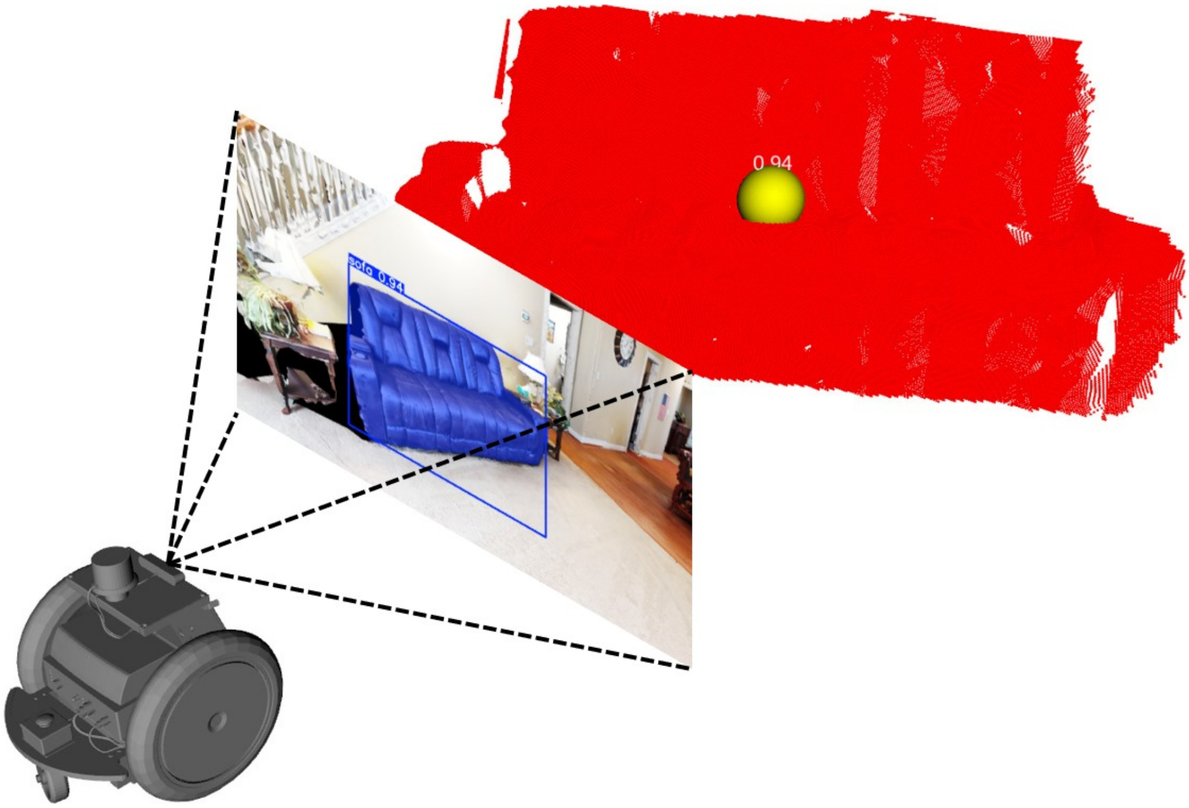

Figure 3: YOLO-E detection to graph node 3D localization

3.5 Fusion Strategy

Exploration–Memory Weighting

- Exploration and memory graph nodes are fused and weighted as follows:
  - Proximity weighting: Nodes closer to the robot's current position are given higher weights.
  - Exploration vs Memory: Nodes from the exploration source are prioritized over memory nodes to encourage discovery of new information.

  - Costmap weighting: Nodes located in areas with lower navigation costs are favored to optimize path planning and navigation efficiency.

Multi-Source Detection Fusion

- Detection graph nodes are weighted based on:
  - YOLO-E confidence scores: Higher confidence detections are given more weight.
  - BLIP-2 value map: Detections with higher semantic relevance to the text prompt are prioritized.
  - The nearer detection graph nodes are to memory graph nodes, the higher their weight.

Relevance Filtering and Node Suppression

- Each source's graph nodes are filtered based on a relevance threshold to eliminate graph nodes within the fov map.
- Relevance map is build over time
- If a graph node is located in an area that has already been explored and found to be irrelevant to the prompt, it is suppressed.

3.6 Behavior Tree for Semantic-Guided Exploration

High-Level Task Structure

- The behavior tree (BT) is designed to manage the high-level task structure for semantic-guided exploration.
- The BT consists of the following main components:
  - Initialization: Clearing Maps, Publishing Prompts
  - Detection Branch: If object is detected over a threshold, navigate to it, realign to object take picture
  - Exploration Branch: While object not detected, perform semantic frontier exploration navigating to highest valued frontiers or memory nodes
  - Termination: If object found, end mission; If time limit reached, end mission
  - Behavior tree is called with a ros2 action server, which returns on termination success or failure, and actual path taken

Integration with Navigation Stack

- Navigation stack used for low-level path planning and obstacle avoidance.
- Action used: navigate_to_pose, Spin

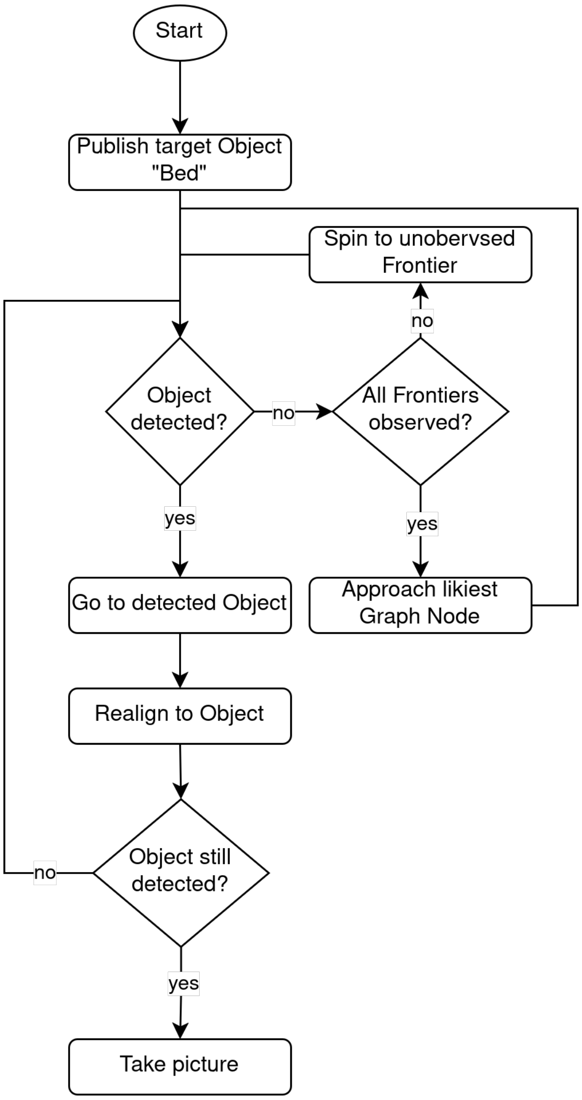

Figure 4: System architecture

4 Implementation

This section details the practical implementation of the proposed approach, covering the simulation and real-world setup, datasets, software stack, and hardware configuration.

4.1 Simulation Environment

- Evaluation of simulation frameworks for indoor semantic navigation:
  - HabitatSim: Realistic Matterport3D-based environments with semantic annotations.

- Isaac Sim / Isaac Lab: GPU-accelerated simulation, advanced physics, support for RTX ray tracing.
- MuJoCo: High-speed physics engine, limited support for complex indoor scenes.
- Ignition Gazebo: Modular simulator, ROS2 integration, good for real-robot transfer.

- ...

4.2 Dataset

- Use of **Matterport3D** scenes for realistic indoor environments with ground truth 3D reconstruction and semantic annotations.
- While the Habitat Navigation Challenge 2023 defines Success Rate (SR) and Success weighted by Path Length (SPL) as standard evaluation metrics within the Habitat-Sim environment, this work extends their application to Isaac Sim. Using Isaac Sim allows for a more physically accurate and sensor-consistent setup, incorporating realistic depth noise, lighting variation, and robot dynamics. To ensure comparability, SR and SPL are calculated following the official Habitat definitions, maintaining consistency with prior benchmarks while improving the realism of scene interaction and perception.

4.3 Used Software

- ROS2-based implementation (Humble Hawksbill) as middleware.
- Navigation stack: Navigation2 (Nav2) for frontier-based exploration and path planning.
- DDS communication layer for distributed communication between detection, mapping, and control nodes.
- Integration of promptable models (OpenFusion, BLIP-2, YOLO-E) for real-time zero-shot detection during exploration and exploitation.

4.4 Used Hardware

- **PC:**
  - CPU: AMD Ryzen 9 5950X 16-Core Processor
  - Motherboard: B550 Gaming X V2
  - GPU: ASUS TUF Gaming RTX 4090 24GB OC Edition
  - RAM: 64GB Corsair Vengeance LPX DDR4
- **Real Robot:** Configuration and components to be determined (TurtleBot Waffle).

4.5 Evaluation Metrics

This section defines the evaluation metrics used throughout the experiments and assigns them to each corresponding experiment.

Evaluation Pipeline Overview

- **Semantic Map Generation:** OpenFusion performs semantic segmentation of RGB-D input using the Matterport3D class list. Each segment is assigned its most likely class label from the detection model.
- **Manual Correction:** Incorrectly labeled segments can be manually relabeled within a dedicated ROS 2 node for semantic correction.
- **Data Storage:** OpenFusion saves both the 3D semantic pointcloud and the corresponding 2D SLAM map for each episode. All experiment data follows the `sage_datasets/matterport_isaac` directory structure.
- **Evaluation Initialization:** During evaluation, the saved maps and class definitions are loaded together with a list of target objects (e.g., “bed”, “toilet”, “chair”).
- **Class Filtering and Centroid Extraction:** The evaluator node filters the semantic pointcloud according to the target classes and extracts the 3D centroids of matching clusters.
- **Path Planning:** The shortest-path planner computes the geodesic-optimal path from the robot’s current pose to the nearest centroid of the selected target class, with the Global Path Planner from ROS2 Navigation2.
- **Metric Computation:** The evaluator node compares the planned and executed trajectories to compute Success Rate (SR), Success weighted by Path Length (SPL), and Multi-Object Success Rate (MSR).
- **Result Storage:** Evaluation metrics, trajectories, and intermediate results are stored per episode for analysis and benchmarking.

- **Experiment 1 – Success Rate (SR):** Measures the proportion of tasks in which the robot successfully reaches the queried single goal object. This metric reflects the system’s ability to semantically ground a user-specified object and to navigate toward it reliably. It serves as a fundamental indicator of task success and is essential for evaluating overall system effectiveness in basic search scenarios. *Evaluation against:* Vision-Language Frontier Maps for Zero-Shot Semantic Navigation (VLFM), Visual Language Maps for Robot Navigation (VLMaps), One Map to Find Them All (OneMap), Learning Generalizable Feature Fields for Mobile Manipulation (GeFF)

$$\text{SR} = \frac{1}{N}\sum_{i=1}^{N} S_i$$

  where $S_i = 1$ if the goal was reached in episode i, and 0 otherwise; N is the total number of episodes.

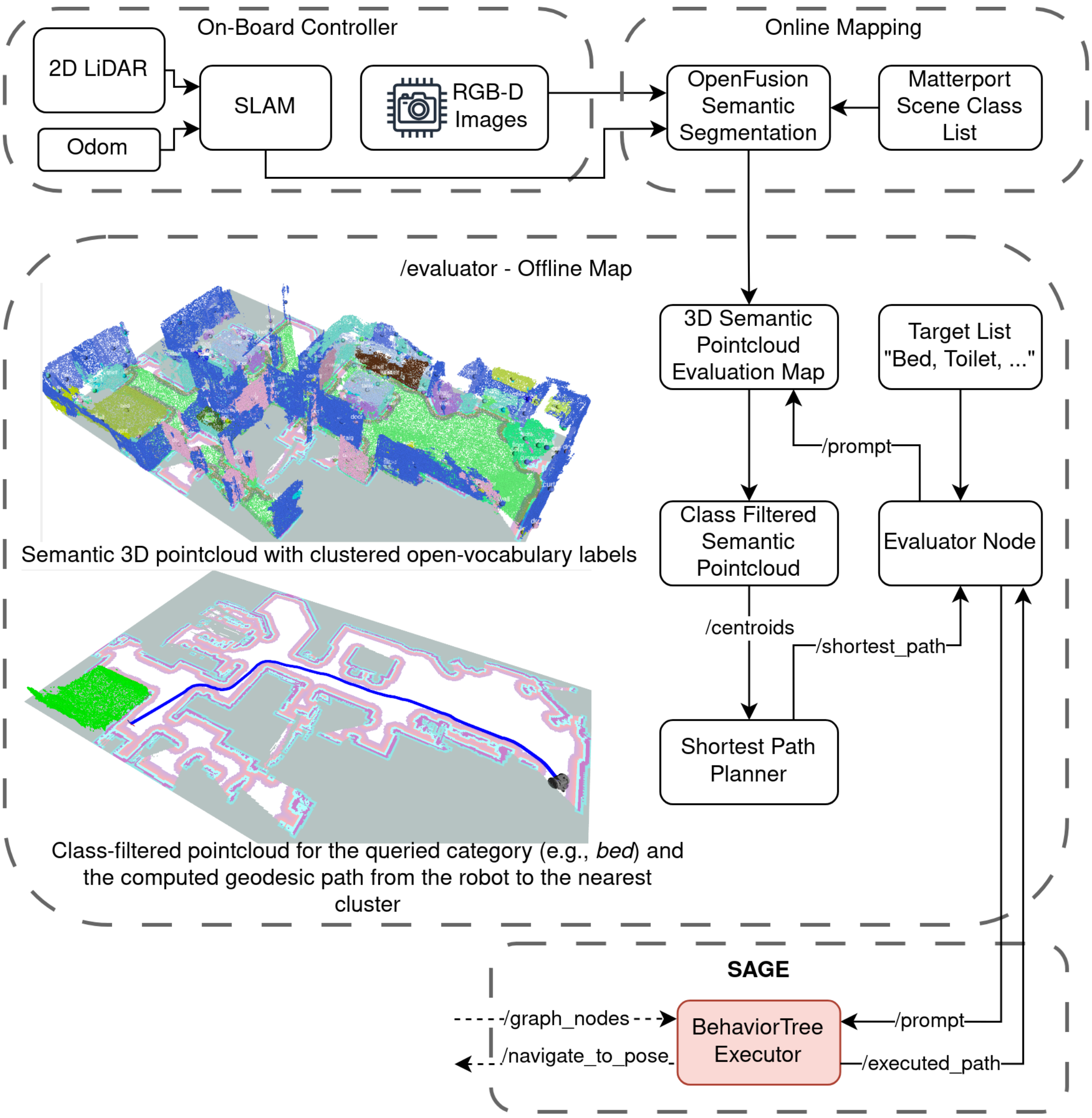

Figure 5: Evaluation pipeline for benchmarking SR, SPL, and MSR in the semantic exploration framework.

- **Experiment 2 – Path Efficiency (SPL):** SPL measures the efficiency of successful navigation by comparing the shortest possible path to the actual path taken. It is defined only for successful runs and penalizes overly long trajectories. In the context of semantic exploration, SPL provides insight into how effectively the system prioritizes relevant regions and minimizes detours when searching for target objects.

$$\text{SPL} = \frac{1}{N}\sum_{i=1}^{N} S_i \cdot \frac{l_i}{\max(p_i, l_i)}$$

where S_i is the success indicator for episode i, l_i is the shortest path length to the goal, p_i is the actual path length taken, and N is the total number of episodes.

- **Experiment 3 – Multi-Object Success Rate (MSR):** The average number of successfully found objects per episode (*Progress, PR*) captures partial success in multi-goal navigation. SPL is computed separately for each object in sequence, conditioned on the success of the previous one. This highlights the system's ability to reuse semantic map information and improve efficiency across successive targets.

$$\text{PR} = \frac{1}{N}\sum_{i=1}^{N} C_i$$

where C_i is the number of successfully found objects in episode i, and N is the total number of episodes.

- **Experiment 4 – Ablation: Memory Component (OpenFusion):** Comparison of multi-object progress and SPL with and without the semantic 3D memory component. Highlights the contribution of global semantic mapping to task success and efficiency in the hybrid system.

- **Experiment 5 – Robustness to False Positives (Fusion Strategy): Metrics:** Evaluated using semantic precision and false positive rate. Assesses the effect of the fusion strategy on filtering erroneous detections to improve overall task success.

$$\text{Precision} = \frac{TP}{TP + FP} \qquad \text{FPR} = \frac{FP}{FP + TN}$$

where TP and FP are the numbers of true and false positive semantic detections, and TN is the number of true negatives (i.e., correctly rejected background regions or non-target classes).

- **Experiment 6 – Real-World System Performance: Metrics:**
  - SR, MSR, SPL – for search performance under real-world conditions.
  - System metrics – CPU/GPU usage, FPS, inference latency.

  Objective: Assess robustness, efficiency, and deployability in physical environments.

5 Discussion and Results

This chapter presents the experimental evaluation of the proposed hybrid semantic exploration system. Each experiment targets a specific research question and is evaluated using quantitative performance metrics.

5.1 Experiment 1: Benchmarking on Matterport Scenes

Evaluates baseline performance in multi-object search compared to state-of-the-art frameworks (OneMap, VLFM, Pigeon) using SR, SPL, and MSR.

5.2 Experiment 2: Impact of Exploration–Memory Weighting

Analyzes how varying the balance between live exploration and persistent memory influences navigation efficiency and task success.

5.3 Experiment 3: Sensitivity to Semantic Map Granularity

Investigates how varying the semantic retrieval depth affects mapping robustness and overall navigation stability.

5.4 Experiment 4: Effect of Multi-Source Semantic Fusion

Examines how combining detection confidence, semantic similarity, and memory reliability improves detection robustness and reduces false positives.

5.5 Experiment 5: System Efficiency and Real-World Validation

Assesses runtime performance, resource utilization, and stability under real-world sensor noise during physical deployment.

6 Summary and Outlook

Hello, here is some text without a meaning. This text should show what a printed text will look like at this place. If you read this text, you will get no information. Really? Is there no information? Is there a difference between this text and some nonsense like "Huardest gefburn"? Kjift – not at all! A blind text like this gives you information about the selected font, how the letters are written and an impression of the look. This text should contain all letters of the alphabet and it should be written in of the original language. There is no need for special content, but the length of words should match the language.

Bibliography

[1] A. Topiwala, P. Inani, and A. Kathpal, Frontier based exploration for autonomous robot, 2018. arXiv: 1806.03581 [cs.RO]. [Online]. Available: https://arxiv.org/abs/1806.03581.

[2] F. Bourgault, A. Makarenko, S. Williams, B. Grocholsky, and H. Durrant-Whyte, "Information based adaptive robotic exploration," in IEEE/RSJ International Conference on Intelligent Robots and Systems, vol. 1, 2002, 540–545 vol.1. DOI: 10.1109/IRDS.2002.1041446.

[3] A. Suresh, C. Nieto-Granda, and S. Martínez, "Robotic exploration using generalized behavioral entropy," IEEE Robotics and Automation Letters, vol. 9, no. 9, pp. 8011–8018, Sep. 2024, ISSN: 2377-3774. DOI: 10.1109/lra.2024.3433207. [Online]. Available: http://dx.doi.org/10.1109/LRA.2024.3433207.

[4] C. Stachniss, O. Mozos, and W. Burgard, "Efficient exploration of unknown indoor environments using a team of mobile robots," Annals of Mathematics and Artificial Intelligence, vol. 52, 205ff, Jan. 2009. DOI: 10.1007/s10472-009-9123-z.

[5] N. Yokoyama, S. Ha, D. Batra, J. Wang, and B. Bucher, "Vlfm: Vision-language frontier maps for zero-shot semantic navigation," in International Conference on Robotics and Automation (ICRA), 2024.

List of Figures

List of Tables

List of source codes

A Appendix A

B Appendix B