Context-Aware Embodied AI Systems for Human-Centered Environments: From Assistive Guidance to Autonomous Robots

by

Shivendra Agrawal

B.S., M.S., Indian Institute of Technology Kharagpur, 2014

M.S., University of Colorado Boulder, 2019

A thesis submitted to the

Faculty of the Graduate School of the

University of Colorado in partial fulfillment

of the requirements for the degree of

Doctor of Philosophy

Department of Computer Science

2026

Committee Members:

Bradley Hayes, Chair

Alessandro Roncone

Nicolaus Correll

Christoffer Heckman

Jodi Forlizzi

Agrawal, Shivendra (Ph.D., Computer Science)

Context-Aware Embodied AI Systems for Human-Centered Environments: From Assistive Guidance to Autonomous Robots

Dissertation directed by Prof. Bradley Hayes

Real-world, human-centered environments are highly unstructured, governed not just by geometric constraints, but by semantic cues, social norms, dynamic conditions, and cluttered layouts. While existing robotic systems have achieved geometric competence, they lack the contextual awareness necessary to operate in public spaces such as stores, offices, libraries, and transit stations. This contextual gap limits the generalizability of autonomous agents and also poses a significant barrier for people with visual impairments, for whom much of this environmental information is only present visually and therefore hard to access directly.

To bridge this gap, this dissertation develops context-aware embodied AI methods that interpret implicit environmental context for robust guidance and spatial grounding. Using assistive technology as a testbed, this work first establishes foundational methods for context-aware guidance by modeling social dynamics and learning from human hand movement for grasp guidance.

The research then extracts spatial knowledge from multimodal data to equip broader autonomous systems with semantic understanding. It introduces novel semantic localization methods for quasi-static environments, initially through distributional semantic particle filtering, and then through vision-language model (VLM) enhanced localization to resolve geometric aliasing. Finally, it presents a semantic-topology representation to support intent-aware search, zone classification, one-shot semantic localization, and the generation of natural language route instructions.

Across these contributions, this thesis demonstrates that social, semantic, and spatial cues can improve robotic systems in environments where geometry alone is insufficient. Together, these systems lay the groundwork for deploying robust, socially intelligent systems capable of long-term, practical autonomy in human-centered spaces.

Acknowledgements

Growing up, I always wanted to be a scientist, not knowing what it entailed. I will admit that I lost my research zeal during my undergraduate years, but fate brought me to the foothills of the Flatirons, and I met some wonderful people along the way who helped me rediscover my passion for research and teaching. I am extremely grateful to my advisor, Prof. Bradley Hayes, for giving me a chance. I have not only learned how to conduct research from you, but I have also gained confidence in myself, rediscovered my passion for research and teaching, and gained more empathy towards everyone because of you and by imitating you.

I am also thankful to many other CU faculty for their support and advice. This list is not exhaustive and in no particular order, but I want to thank Prof. Roncone, Prof. Correll, Prof. Heckman, Prof. Forlizzi (my external thesis committee member), Prof. Grochow, Prof. Sankaranarayanan, Prof. Chen, Prof. Clauset, and Prof. Martin. To my students: helping you learn helped me learn so much. I will fondly recollect the memories and the conversations.

My Ph.D. journey was made much more enjoyable because I was lucky enough to do so with my wife. I will forever cherish that. I am incredibly thankful to my mom, dad, and sister, who tried their best to understand what I do and why it's difficult and important. Through it all, I know they would have been fine with it even if I had failed my Ph.D.

I am also grateful for my other family members, my labmates, my friends both in Boulder and away, and my tennis buddies who always supported and encouraged me and kept me going. I would not have crossed the line without all of you.

Contents

List of Tables

Table

List of Figures

Figure

List of Publications

This dissertation is based in part on the following peer-reviewed publications and preprints:

Publications Included in this Thesis

- **Agrawal, S.**, & Hayes, B. (2026). GIST: Multimodal Knowledge Extraction and Spatial Grounding via Intelligent Semantic Topology. **arXiv preprint arXiv:2604.15495**.
- **Agrawal, S.**, & Hayes, B. (2026). VLM-GLoc: Vision-Language Model Enhanced Monte Carlo Localization for Robust Semantic Global Localization in Cluttered Quasi-Static Environments. **arXiv preprint arXiv:2605.30506**. Under review.
- **Agrawal, S.**, Brawer, J., Naik, A., Roncone, A., & Hayes, B. (2026). ShelfAware: Real-Time Visual-Inertial Semantic Localization in Quasi-Static Environments with Low-Cost Sensors. **IEEE Robotics and Automation Letters**.
- **Agrawal, S.**, Nayak, S., Naik, A., & Hayes, B. (2023). ShelfHelp: Empowering Humans to Perform Vision-Independent Manipulation Tasks with a Socially Assistive Robotic Cane. In **Proceedings of the 2023 International Conference on Autonomous Agents and Multiagent Systems** (pp. 1514–1523).
- **Agrawal, S.**, West, M. E., & Hayes, B. (2022). A Novel Perceptive Robotic Cane with Haptic Navigation for Enabling Vision-Independent Participation in the Social Dynamics of Seat Choice. In **2022 IEEE/RSJ International Conference on Intelligent Robots and Systems** (pp. 9156–9163). IEEE.

Other Publications

- **Agrawal, S.**, Roncone, A., & Hayes, B. (2026). Distributional Semantics for Robust Global Localization in Cluttered, Geometrically Aliased Environments. In **ICRA 2026 Workshop on Semantics for Reliable Robot Autonomy**.
- **Agrawal, S.**, & Hayes, B. (2022). ShelfHelp: Empowering Humans to Perform Vision-Independent Manipulation Tasks with a Socially Assistive Robotic Cane. In **IROS 2022 SCIAR Workshop**.
- **Agrawal, S.** (2023). Assistive Robotics for Empowering Humans with Visual Impairments to Independently Perform Day-to-Day Tasks. In **Proceedings of the 2023 International Conference on Autonomous Agents and Multiagent Systems** (pp. 3023–3025).

Chapter 1

Introduction

1.1 Motivation

As AI agents move into complex human spaces such as hospitals, schools, grocery stores, offices, and campuses, they need social and semantic intelligence, not just navigational ability. They need to understand which seat is socially appropriate, which shelf likely holds pasta, or how to describe a path for a person to easily follow.

This problem is also prominent for people with visual impairments (PVI). Many everyday environments are less accessible, and many common tasks depend on visual context that is hard to access without sight. Finding tortilla chips in a store, finding a seat in a cafe, or reaching for the right product on a dense shelf are not only geometric tasks. They require understanding object meaning, social norms, spatial layout, and what information is useful to communicate. This makes assistive technology both a socially important domain and a rigorous testbed for context-aware embodied AI systems. If a system can interpret enough context to guide a human safely and usefully, it shows a deeper understanding of the environment.

Existing assistive robotics has made important progress, but much of the work is still focused on geometric competence. These systems can avoid obstacles, follow waypoints, or guide users down a lane. They are less able to interpret the human context of a place. Other systems work in complex environments by adding RFID tags, beacons, or other infrastructure. These approaches can help in controlled deployments, but they are hard to scale across real public spaces. There is also limited work on fine-grained manipulation/grasp guidance, where the user is already near the

goal but still needs help moving a hand to complete the task.

This thesis targets that gap. The work develops systems that interpret implicit context, operate without expensive pre-instrumentation, and use practical low-profile sensors. The goal is not only to make assistive devices better, but also to build techniques that generalize to autonomous robots in human-centered environments.

1.2 Thesis Statement

Recent advancements in embodied AI have established a strong foundation for geometric navigation and obstacle avoidance. However, these systems lack the ability to interpret implicit environmental context such as social norms, semantic object distributions, and spatial topologies. This contextual awareness is crucial for guiding robots through complex, unstructured, and human-centered spaces. Additionally, existing methods for spatial reasoning often rely on heavily instrumented environments. In this dissertation, I develop a suite of context-aware embodied AI approaches that leverage probabilistic semantic modeling, human behavioral priors, and foundational models to extract and utilize implicit environmental cues. I demonstrate this framework initially through assistive guidance applications for people with visual impairments, and extend it as an enabling technology for autonomous robotic navigation and spatial reasoning.

1.3 Overview

This dissertation is structured as follows.

Chapter 1 introduces the motivation, thesis statement, and chapter overview. It argues that many important public-space tasks require more than obstacle detection or geometric navigation. They require systems that can use implicit social, semantic, and spatial context.

Chapter 2 reviews assistive robotics for navigation and manipulation guidance, with focus on form factors, sensing, feedback, localization, and interaction. It also adds background on semantic spatial grounding, vision-language models, and semantic topology, since these areas become important in the later chapters.

Chapter 3 presents the first completed work in this series. It details a perceptive cane that identifies socially preferred seating in public spaces by integrating psychological models of privacy, intimacy, and convenience with environmental perception. The chapter evaluates haptic navigation guidance toward socially preferred goals in simulated scenarios with blindfolded participants.

Chapter 4 presents ShelfHelp, a robotic cane system for fine-grained manipulation/grasp guidance in grocery shelves. This work introduces a vision pipeline for product localization and a planner that generates verbal commands for product retrieval. The chapter shows that product retrieval is a complicated computer vision problem and human hand movement models can help form optimal policies to guide humans to acquire desired objects in cluttered scenes

Chapter 5 presents ShelfAware, the third paper in this thesis. ShelfAware targets robust global localization in quasi-static environments such as grocery stores. It models semantic observations as distributions over object categories and their spatial arrangement, then uses those observations inside a particle filter with inverse semantic proposals. This addresses a prerequisite for assistive guidance near a known shelf: the system must know that the user is in front of the correct shelf before fine-grained retrieval can begin.

Chapter 6 presents VLM-GLoc, which revisits localization with a richer semantic interface. Instead of relying on fixed object categories, VLM-GLoc uses vision-language models to create language-rich observations, which are fused within Monte Carlo Localization. This chapter expands the localization from retail-specific category semantics to broader indoor domains like offices.

Chapter 7 presents GIST, a semantic-topology framework for multimodal knowledge extraction and spatial grounding. GIST shows how rich environment representations can support several downstream tasks, including search, zone classification, one-shot semantic localization, and route instruction generation. This chapter acts as the broadest use case of semantic spatial representations in the thesis.

Chapter 8 concludes the dissertation. It summarizes the contributions, discusses limitations, and outlines future work for practical assistive systems and autonomous robots in human-centered environments.

Chapter 2

Related Work: Context-Aware Guidance, Navigation, and Interaction

2.1 Context-Aware Embodied AI in Human-Centered Environments

It is estimated that 295 million people have some form of vision impairment, of whom 43.3 million are blind [22]. White canes and guide dogs are the most common assistive aids used by blind and visually impaired individuals. Canes are useful for tracing curbs, walls, doors, stairs, and nearby obstacles, but they have limited utility for finding a socially appropriate seat, locating a specific grocery shelf, or selecting the correct object from a visually dense scene. Guide dogs are also cost prohibitive, costing approximately $50,000 to train and $1,200 on average annually [93]. Reliance on sighted guides can limit independence and can create privacy concerns, especially for tasks involving sensitive products or personal choices [9].

This dissertation treats assistive guidance as both a socially important application area and a rigorous testbed for context-aware embodied AI. A system that can guide a person through a human-centered environment must parse more than geometry. It must understand what objects and regions mean, which cues are stable enough to trust, how humans behave in shared spaces, and how spatial information should be communicated to a user. In a cafe, context includes empty chairs, other people, walls, bags, and the social meaning of privacy and interpersonal distance. In a grocery store, context includes product categories, brands, shelf layout, long-tail visual semantics, and the fact that stock moves over time. In larger indoor spaces, context also includes topological structure, zone boundaries, landmarks, and route descriptions that can be followed without visual confirmation.

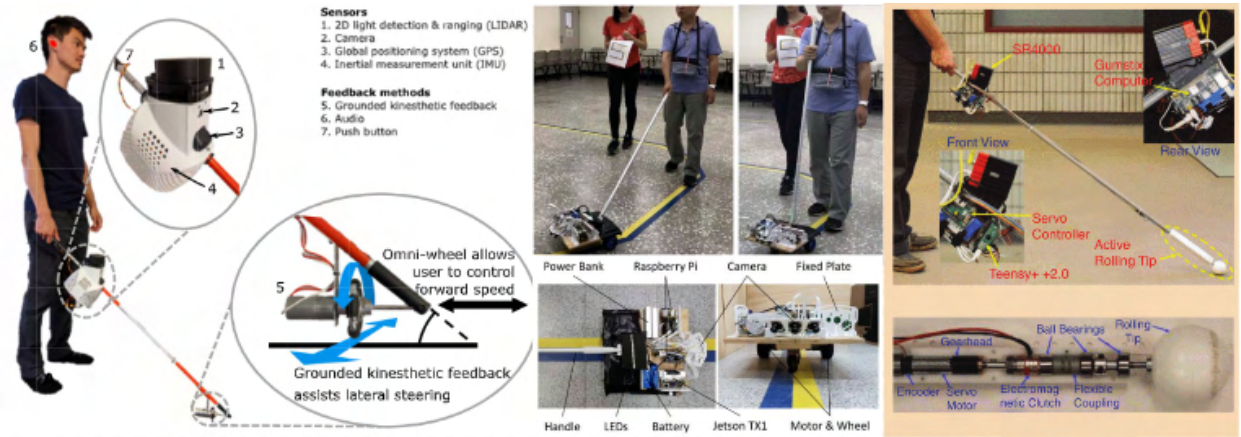
Most existing embodied systems are geometrically competent. They can avoid walls, follow waypoints, or maintain a lane. That is important, but it is not enough. A system that avoids obstacles still may not know which seat is socially appropriate, which shelf region contains pasta, or which repeated aisle it is standing in. The same gap appears across all chapters of this thesis. Chapter 3 studies social context for navigation goal selection. Chapter 4 studies semantic and human behavioral context for fine-grain product retrieval. Chapter 5 studies semantic localization in quasi-static shelf environments. Chapter 6 studies rich language observations for localization in aliased spaces. Chapter 7 studies multimodal knowledge extraction and semantic topology as a shared representation for search, localization, zone reasoning, and route communication.

The literature can therefore be organized around four needs. First, navigation guidance systems need to move from obstacle avoidance and waypoint following to context-aware goal selection and interaction in social environments. Second, manipulation guidance systems need to help users act in the near-field haptic space where small spatial errors matter. Third, localization systems need semantic evidence that remains useful when geometry is repetitive and the world changes. Fourth, multimodal spatial interactive systems need representations that connect visual observations, language, topology, and metric space. The following sections review these areas and identify the gaps that motivate the technical chapters.

2.2 Navigation Guidance in Social and Human-Centered Spaces

Navigation guidance for people with visual impairments has been studied across many robots and form factors. The dominant goals have been obstacle avoidance, path following, collision reduction, and waypoint navigation. These goals are foundational, but human navigation in public spaces also depends on implicit context. A user does not only need to reach any empty chair. They may want a chair that is closer, more private, away from strangers, or less likely to be occupied by someone else's belongings. That distinction turns navigation into goal selection under social and semantic constraints.

2.2.1 Actuated and Perceptive Canes



a) Slade et al. (2021) [27] b) Chuang et al. (2018) [11] c) Ye et al. (2016) [35]

Figure 2.1: Examples of wheeled cane designs ranging from sensors near the holding point on the cane to the bottom of the cane.

Smart canes and actuated canes preserve a familiar assistive form factor while adding sensing, computation, and feedback. Slade et al. [106] developed a wheeled augmented cane that can avoid obstacles and follow waypoints. Their design augments the cane end with an omnidirectional wheel that lets the user control forward speed and receive grounded kinesthetic steering feedback. The system also uses audio to provide richer information about objects. The researchers worked with three visually impaired participants to co-design the system and tested it with blindfolded and visually impaired users. They found grounded kinesthetic feedback to be faster and more accurate than coin-motor vibrotactile feedback and audio feedback for reducing heading error. Their system also increased walking speed for visually impaired participants by $18 \pm 7\%$ compared to earlier electronic travel aids based on vibrotactile feedback [35, 59, 133]. At the same time, the powered wheel can reduce the local environmental feedback that makes a white cane useful when the electronics fail or when the user wants direct tactile information.

Chuang et al. [29] augmented a cane with a custom mobile robot at the end. The robot used onboard compute, three cameras, and a three-wheel drive system. Guidance was delivered through grounded kinesthetic feedback and simple audio commands such as “Turn Left”, “Go Straight”,

and “Turn Right”. Their system focused on following man-made trails with visual markings. They compared computer vision methods for predicting control from multiple cameras and tested the final system with ten blind and visually impaired participants on an S-shaped trail. Users generally rated the system positively for real-time operation, but rated it poorly on portability, interaction, and speed.

Ye et al. [130] presented the co-robotic cane, another wheeled cane that provides grounded kinesthetic guidance. The cane can operate in an active navigation mode or passive object-detection mode. It uses user compliance to infer intent and switch modes automatically. In active mode, it estimates pose through visual odometry, iterative closest point, and pose graph optimization. The system performed better than dead reckoning in some low-feature environments, but in feature-rich environments it was comparable to dead reckoning. This matters for real-world navigation because crowded public places often contain clutter, moving people, repeated geometry, and partial occlusions.

These actuated cane systems show that the cane can become a robotic guidance platform. They also show a tradeoff that appears repeatedly in assistive robotics. More actuation and instrumentation can improve guidance, but it can also reduce portability, reduce passive cane utility, add weight, or require users to learn a new interaction style. In Chapter 3, the cane form factor is retained while perception, social goal scoring, and haptic feedback are added without trying to replace the cane’s basic local sensing role.

2.2.2 Non-Cane Robots for Navigation Assistance

A large body of navigation guidance research uses non-cane form factors such as carts, suitcase robots, walkers, drones, wheeled mobile robots, and quadrupeds. These platforms can carry more sensors and computation than a cane, but they also raise questions about portability, cost, public acceptance, and trust.

Gharpure et al. [43] explored grocery shopping assistance with a cart-based robot by mounting a shopping basket on a mobile robot. Their work is important because it addressed both locomotor

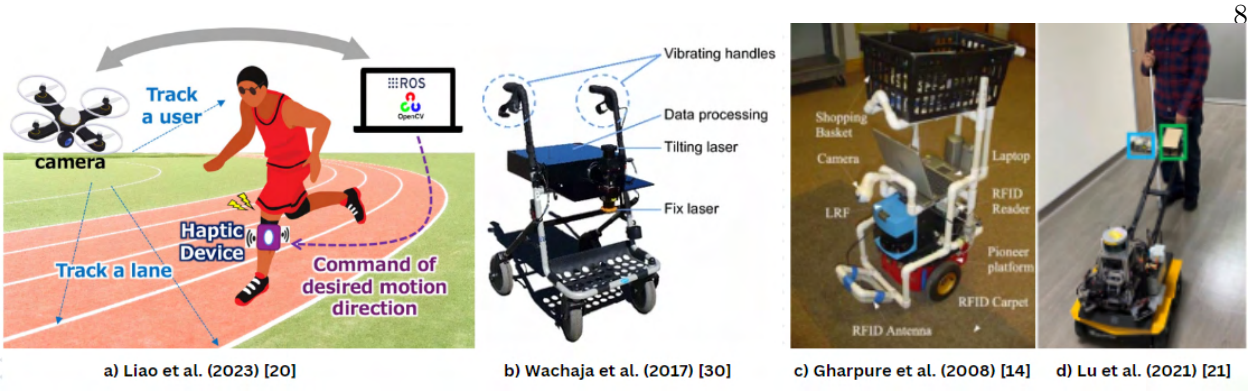


Figure 2.2: Researchers have worked with commercial robots, retrofitted robots, and new robotic systems to provide navigation assistance.

space and haptic space. Locomotor space is the area that requires bodily movement and route guidance. Haptic space is the immediate reachable space around the user. For navigation, their system used Markov localization with RFID tags installed in the store. For product interaction, it used an omnidirectional barcode scanner. They also studied browsing, typing, and speech as product-selection methods and found speech to have the lowest mean selection time and better subjective workload scores. The work made clear that grocery shopping requires navigation, product selection, and near-field interaction, but it also showed that store-level instrumentation creates scalability issues.

Lu et al. [79] used a Clearpath Jackal UGV to provide grounded kinesthetic feedback through a rigid harness. Their system used ultrawide-bandwidth beacons as localization landmarks and used audio to convey semantic information about locations. The system combined an Extended Kalman Filter with UWB landmarks and used deep reinforcement learning for dynamic obstacle avoidance. In a study with eight blind and visually impaired participants, the UWB-based solution outperformed SLAM-based baselines. This result is useful because it shows how difficult uninstrumented localization can be in dynamic settings. It also shows why systems that depend on UWB or RFID are hard to scale outside controlled deployments.

Guerreiro et al. [52] designed CaBot, an autonomous suitcase-shaped robot with a motorized wheel, LiDAR, a laptop, stereo camera, and tactile handle. It guides users using grounded

kinesthetic feedback, vibrotactile handle cues, and speech. It uses AMCL with a 2D LiDAR for localization and can announce nearby points of interest when they are manually annotated on a map. In a study with ten blind and visually impaired participants, users gave high ratings for safety, confidence, and trust. The CaBot platform later became a more compact suitcase form factor, making it easier to tow when inactive and more acceptable in public spaces.

The CaBot hardware also supported targeted real-world systems. BBeep [60] used the suitcase robot to support blind users in crowded airport environments by issuing pre-emptive sound notifications to nearby pedestrians when future collisions were predicted. In an airport user study with six blind and visually impaired participants, sound timing affected pedestrian trajectories. Another system used the same platform for museum exploration [61]. Blind visitors could use the robot to navigate toward exhibits, hear short audio descriptions while moving, and request more detailed explanations through a phone. The system relied on a manually defined topological route map created with museum staff. These systems show the value of route structure, points of interest, and semantic descriptions, but they still depend on prebuilt maps and curated knowledge.

Walker-based systems target users who need both visual and mobility support. Wachaja et al. [114] augmented an off-the-shelf walker with LiDARs, vibration motors, and a terrain classifier. Their controller predicted the future pose of the walker by modeling user delay, then issued vibration commands for go straight, left, right, and goal reached. In tests with blindfolded participants, many runs were shorter or faster with the predictive controller, although users preferred the non-predictive controller because it produced fewer signals. This highlights a recurring issue in assistive guidance. A technically better controller can be less desirable if the interaction becomes too frequent or mentally demanding.

Drones have also been explored for guidance. Liao et al. [72] proposed a drone-based system for running guidance. The drone observed a runner's movement and environment, predicted the direction needed to stay on track, and transmitted signals to a lower-leg haptic device. In outdoor running tests with seven blindfolded participants, the system guided runners to stay on track more often than no assistance and did not reduce running speed. At the same time, drones create safety

and trust concerns. Chang et al. [23] showed that users can exhibit excessive trust when interacting with drone systems, which raises safety concerns for assistive deployments in shared spaces.

Quadrupeds are attractive because they can traverse varied terrain, but their usability for blind guidance is still uncertain. Xiao et al. [126] used a Mini Cheetah connected to the human user by a leash. The system guided blindfolded participants through unknown indoor environments and narrow spaces. DeFazio et al. [32] created a force-estimation controller for a robotic guide dog using tugs on a leash. The system estimated the direction and magnitude of external forces and used them for navigation commands. These works show the promise of physical human-robot interaction, but quadrupeds can be noisy, bulky, and hard to handle when unpowered. Wang et al. [120] found that commercial quadrupeds had disadvantages in usability and trust compared to wheeled robots. For assistive technologies that must be carried, stored, used on public transit, or trusted in quiet indoor spaces, the form factor is not a side issue.

2.2.3 Non-Actuated and Wearable Guidance Systems

Non-actuated systems are not robots in the traditional sense, but they provide important lessons for perception, planning, and feedback. Kuribayashi et al. [70] proposed a smartphone-based corridor navigation system using a LiDAR-equipped phone. The system detects obstacles and intersections from a 2D grid map, plans paths with A*, detects intersections with YOLOv3, and provides audio and vibration feedback. In a study with fourteen blind participants, the system improved obstacle avoidance and intersection detection, but it also slowed users because they had to stop and scan.

Saha et al. [100] studied the last few meters of wayfinding, where GPS is not precise enough and where finding the right doorway is difficult. Their Landmark AI app had channels for scanning objects, reading signs, and using a reference image to find a previously visited place. Participants wanted precise directional guidance and distances to objects, but they also wanted less unnecessary information while walking. This tension is central to the rest of this thesis. A system must sense rich context, but it should not overload the user with everything it sees.

Nasser et al. [85] studied thermotactile feedback on a cane grip. They compared different Peltier module configurations and later compared thermal feedback with vibrotactile feedback. Users often preferred thermal cues because they felt lower task load, but participants suggested combining thermal and vibration cues. Wang et al. [118] developed a wearable neck device and vibration belt for free-space guidance, obstacle avoidance, and target-object navigation such as finding an empty chair. In their study with eleven blind users, the system reduced collisions during chair finding and other navigation tasks. These systems show that guidance modality matters as much as planning quality.

2.2.4 Social Navigation and Goal Selection

Much assistive navigation work focuses on avoiding obstacles or following a path, but social environments require more. Wang et al. [118] identified finding empty chairs in crowded areas as a high-priority mobility task for blind and visually impaired users, and also reported that the white cane has low utility for that task. Finding a chair is not just a detection problem. A user often wants a seat that is appropriate with respect to people, walls, personal belongings, privacy, and interpersonal distance.

Psychology work gives this problem structure. Staats and Groot [108] found that people choose seats in ways that optimize privacy and intimacy. In public seating, a chair near a wall can feel more private. A chair too close to a stranger may feel socially uncomfortable. A chair near a backpack or mug may be empty but implicitly taken. These are not obstacle labels. They are social cues that require interpreting objects and people in relation to the candidate goal.

Chapter 3 builds on this gap by treating social goal selection as part of robotic guidance. The cane system detects seats, humans, and objects, scores candidate chairs using convenience, anchoring, and proximity cues, plans a path, and conveys the plan through speech and vibrotactile feedback. The key shift is that the system is not only asking where it can move safely. It asks which goal is more socially appropriate. This is the first example in the thesis of using implicit environmental context as a computational input to guidance.

2.3 Manipulation Guidance in Near-Field and Shelf Environments

Many real-world tasks involve both navigation and near-field manipulation. Indoor navigation may end with operating an elevator, opening a door, choosing a seat, using a vending machine, or handling a trash bin. Grocery shopping requires moving through the store, finding an aisle, standing in front of the right shelf, retrieving the product, and examining it. Once a person is near the target, small spatial errors become important. The relevant space is the user's haptic space, the area that can be reached and explored without bodily translation [43, 46]. In dense shelves, tactile search can be slow and unreliable because products are similar, tightly packed, and often identified by visual labels.

Figure 2.3: Many common tasks involve operating and manipulating tools and items that are hard to work with without vision.

2.3.1 Robotic and Human-in-the-Loop Manipulation Guidance

Manipulation guidance remains less explored than navigation guidance. Bonani et al. [21] studied how blind users perceive robots and what functions they want from assistive robots. In a focus group with twenty blind and visually impaired participants, users preferred smaller portable form factors and wanted help with navigation and housekeeping tasks. The same work used a Baxter

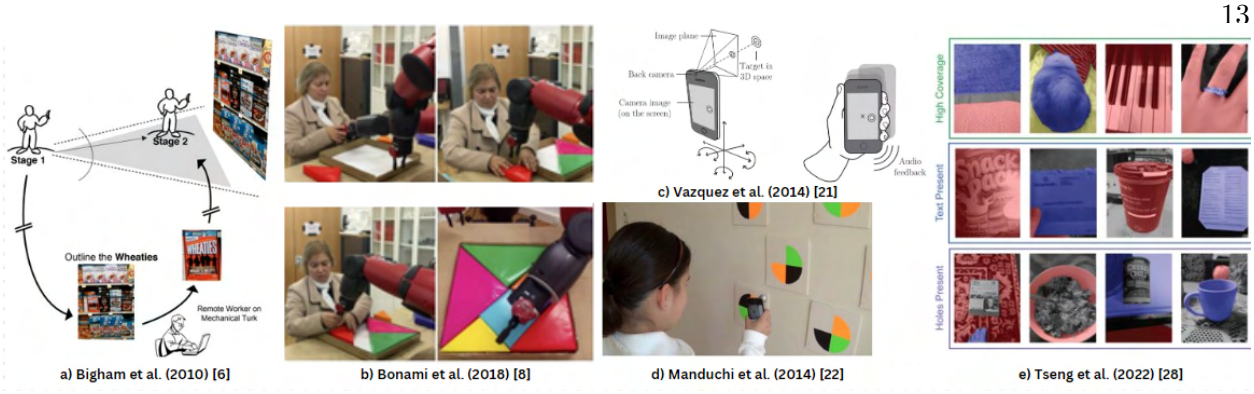


Figure 2.4: Prior work in manipulation guidance has not witnessed many traditional robotic form factors.

arm in a Wizard-of-Oz study for a Tangram assembly task. A collaborative assistive robot condition used voice and active kinesthetic guidance, while a voice-only condition gave all instructions at once. The collaborative condition performed better in average placement time and subjective metrics such as warmth and perceived competence. This suggests that physical guidance can help, but large manipulators are impractical for many daily tasks.

Bigham et al. [19] presented LocateIt, a crowd-assisted system that helps blind users find specific objects from phone images. Remote workers outline the target object, and the system uses the outline and color histogram matching to guide the user interactively from the phone. The method can work for many object types and does not require a barcode, but it relies on remote humans and active network connectivity. Be My Eyes [36] makes a similar tradeoff. Human help is flexible and often effective, but it has privacy, availability, scalability, and language-access barriers.

Vázquez et al. [113] proposed an assisted photography framework to help blind users aim a camera. Their system uses saliency maps and regions of interest to provide interactive feedback before selecting the best image for post-processing. The study found speech feedback to be efficient and preferred for usefulness. Manduchi et al. [81] highlighted a limitation of frame-by-frame acoustic guidance from a handheld camera. As a user approaches a target, maintaining the target in the field of view becomes harder. The robotic implication is important. Near-field guidance benefits from a persistent representation in a stable frame, not only from instantaneous camera feedback.

2.3.2 Product Identification and Grocery Retrieval

Product identification creates another challenge. Grocery stores contain a large number of products and approximately 30,000 new products are introduced each year in the United States market alone [87]. Fixed-class classifiers can work only for a limited product database. For example, Feng et al. [37] trained a classifier for 1329 products, but maintaining such a classifier for long-tail grocery inventory is difficult. Barcode scanning is reliable once a product is in the user's hand [43, 47, 86], but it does not solve search over a shelf because the user usually cannot scan the barcode until after retrieval.

Prior grocery assistant systems often focus on locomotor navigation rather than product retrieval [43, 67–69, 86]. Many use RFID tags, barcodes, or other environmental augmentation. Yuan et al. [132] showed that grocery shopping also includes overlooked tasks such as identifying products that are running short, itemizing needed products, and organizing purchased products at home. Saha et al. [100] studied the last few meters of wayfinding. Zhao et al. [137] studied grocery assistance for people with low vision. Together, these works show that shopping is not one task. It is a sequence of navigation, search, interaction, and verification problems.

Chapter 4 focuses on one underexplored subtask, fine-grain product retrieval from a shelf. ShelfHelp uses a cane-mounted RGB-D camera and odometry camera, detects products, associates observations over time, selects a target, and plans verbal manipulation guidance with an MDP. The system learns a mapping between verbal commands and human hand movement, then uses that model to issue commands such as moving left, right, up, down, forward, or backward. The important point for this literature chapter is not only that ShelfHelp retrieves products. It shows that manipulation guidance requires scene semantics, spatial precision, and human behavioral modeling. It also exposes a limitation that later motivates localization work. ShelfHelp assumes the user is already standing in front of the correct shelf.

2.4 Semantic Localization in Quasi-Static Environments

Navigation and manipulation guidance both depend on knowing where the system is. In many assistive settings, especially grocery stores, localization is not a solved prerequisite. A user may enter a store, approach a repeated aisle, or lose track of their location after a turn. The robot or wearable system must recover its global pose without relying on GPS, wheel odometry, LiDAR, or pre-installed beacons. This is difficult because human-centered indoor environments often have repeated geometry, repeated semantics, changing object placements, and long-tail object categories.

2.4.1 Geometry-Based Localization and Its Limits

Particle filter localization methods such as Monte Carlo Localization and Adaptive Monte Carlo Localization remain standard for onboard robot localization because they represent multi-modal beliefs and fuse motion with sensor likelihoods [1, 38]. Depth-camera and LiDAR-based localization can be accurate when geometry is distinctive [20, 122]. Recent surveys still emphasize that global LiDAR localization struggles under perceptual aliasing, dynamic objects, and map change [131]. Learned implicit representations and area-graph methods address parts of this problem [66, 127], but they often remain tied to geometric distinctiveness or heavy compute.

Grocery stores make this problem especially clear. Long shelves, repeated aisles, similar endcaps, and moving shoppers can make many locations geometrically plausible. Earlier grocery systems therefore often used environmental augmentation. RoboCart and related systems identified many computational challenges of grocery shopping and used RFID or similar infrastructure to support localization and product access [43, 67, 77]. These systems were influential, but installing and maintaining tags or beacons across stores is a major adoption barrier.

2.4.2 Semantic Localization and Long-Term Spatial Evidence

Semantic localization augments geometry with object, text, place, or layout cues. Semantic SLAM and floor-plan alignment methods use objects and room structure to localize in prior indoor

maps [**?**, 140]. Text cues such as signs and readable labels can serve as distinctive landmarks in structured indoor spaces [141]. Other methods model object existence probabilistically for long-term localization under change [2], or use visual positioning with semantic and geometric constraints [24, 31, 98]. These works show that semantics can break geometric symmetries.

The remaining problem is that many semantic localization methods assume stable object identities or fixed detector vocabularies. This assumption is fragile in quasi-static environments. A quasi-static environment has a stable global layout, but local semantic content changes over time. Products move, shelves are restocked, objects are occluded, and temporary items appear. The challenge is not only to detect objects. The localization system must decide which semantic cues are reliable evidence and which ones should be treated as noisy or transient.

ShelfAware, developed in Chapter 5, addresses this problem by modeling shelf semantics as probabilistic distributions over object counts and arrangements [5]. Instead of treating each semantic observation as a fixed landmark, ShelfAware uses distributional evidence from shelf contents within a particle filter. It combines semantic similarity with depth evidence and uses inverse semantic proposals to seed particles around likely global poses. This makes the filter more robust to restocking, missing items, and noisy observations while remaining compatible with low-cost wearable or cart-mounted sensors.

2.4.3 Rich Language Semantics for Aliased Spaces

ShelfAware shows that shelf semantics can support localization, but category-count vectors are still limited by the detector vocabulary. In a repetitive office, knowing that a view contains a chair may not disambiguate the location because many places contain chairs. In a grocery aisle, knowing that a view contains spice may not be enough because many shelves contain spices. VLM-GLoc, developed in Chapter 6, expands the semantic representation using rich natural-language observations. A vision-language model can describe color, material, object state, readable text, brand names, packaging, and contextual permanence. A label such as “chair” can become “black mesh chair with orange cushion”. A generic product category can become a brand-specific or

package-specific cue.

Recent foundation-model localization systems show that language-conditioned models can support spatial grounding. Vision-and-language navigation connects language and embodied movement [12, 25, 65]. VLM-based localization methods use VLMs for point-cloud grounding or 3D reasoning [56, 95]. Open-vocabulary mapping systems such as ConceptGraphs use foundation models to build queryable 3D scene representations [51]. These works are valuable, but many assume known localization, direct coordinate regression, or heavy 3D representations.

VLM-GLoc takes a different view. The VLM does not output coordinates. It produces semantic evidence for a probabilistic filter. Language observations are embedded and compared against a semantic map, then geometry and odometry are fused inside Monte Carlo Localization. This preserves the uncertainty handling and recovery behavior of particle filtering while using VLMs as rich semantic observation generators. The result connects new multimodal perception to a classical localization backbone rather than replacing localization with a one-shot prediction.

2.5 Multimodal Spatial Interactive Systems

The previous sections focused on guidance and localization. A broader question is what representation should connect these capabilities. A human-centered embodied system may need to search for a goal, localize from a single observation, identify a semantic zone, and communicate a route in natural language. A purely metric map is not enough for those tasks. A dense visual model may contain useful information, but it can be heavy, brittle under scene change, and hard to convert into instructions. A useful representation should connect metric space, topology, semantics, and language.

2.5.1 Vision-and-Language Navigation and Instruction Generation

Vision-and-Language Navigation requires an embodied agent to interpret natural language instructions and navigate from visual observations [12]. Early benchmarks operated over discrete panoramic graphs, while VLN-CE moved toward continuous environments with low-level motor

control [65, 94]. These benchmarks have pushed progress on language-conditioned navigation, but they often assume that good instructions already exist and that the agent is localized or operating inside a benchmark-specific structure.

Instruction generation addresses the opposite direction. Speaker models translate paths or image sequences into route instructions [39]. Newer systems such as NavRAG, GoViG, and NavComposer use scene descriptions, goal images, RGB sequences, and action traces to generate navigation language [53, 121, 125]. These systems are important for scaling data and connecting language to movement. However, sequence-based instruction generation can struggle to infer stable path geometry from images alone. It can reference salient but transient objects, miss definitive turn structure, or under-specify the topological transitions that people need in dense environments.

Spatial cognition research helps explain why this matters. People use landmarks, route segments, and meaningful transitions to understand space [44, 63, 76, 78]. Instructions such as "walk past the lentils on your left, then turn right" can be more useful than only metric commands like "walk 8.6 meters forward". Landmark-based instructions can reduce cognitive burden when the landmarks are stable, recognizable, and placed at useful decision points [41, 42, 107, 109]. This is closely related to assistive guidance because the user may need verbal or haptic information that is precise enough to follow but not overloaded with every detected object.

2.5.2 Scene Graphs, Open-Vocabulary Maps, and Semantic Topology

Scene graphs and open-vocabulary maps provide one path toward semantic spatial reasoning. ConceptGraphs and related 3D scene-graph planning systems build symbolic object-relation structures for querying and task planning [14, 51, 96]. These representations are powerful for structured reasoning, but they often compress visual observations into object nodes, relations, attributes, or serialized subgraphs. That can be helpful for question answering or planning, but route communication and localization also require geometry, topology, and view-grounded evidence.

GIST, developed in Chapter 7, is motivated by this gap. It transforms a consumer-grade mobile scan into a semantically annotated navigation topology. The pipeline converts RGB-D

data and odometry into a 2D occupancy map, extracts a topological graph over walkable space, selects informative keyframes and representative objects, and annotates them with a VLM. The resulting representation keeps deterministic geometric structure separate from higher-level semantic reasoning while anchoring both in a shared coordinate frame.

The key idea is that semantic topology can support several downstream tasks without rebuilding a new map for each one. In GIST, the same representation supports intent-aware semantic search, one-shot semantic localization, zone classification, and route generation. This makes it a capstone for the thesis narrative. The earlier chapters use context for specific tasks, such as socially preferred seats, product retrieval, and semantic localization. GIST studies how multimodal semantic context can be organized into a reusable spatial representation for human-AI interaction and embodied autonomy.

2.6 Other Important Aspects and Cautions for Assistive Technologies

The systems reviewed above focus on navigation guidance, manipulation guidance and enabling research in localization, and multimodal spatial interaction. However, transitioning these core algorithmic capabilities into practical, real-world assistive technologies requires addressing a broader set of sociotechnical challenges. While this thesis focuses on the enabling algorithms that allow embodied AI to operate in human-centered environments, deploying these systems safely and effectively necessitates consideration of the following critical themes:

Dataset creation. New datasets can accelerate assistive robotics in the same way they have accelerated many machine learning areas. Tseng et al. [111] showed that current algorithms often struggle to locate objects that have holes, are very small or very large, or lack text. Photos taken by blind and visually impaired users often contain these properties, which means assistive systems must handle images that differ from standard benchmark images.

Needfinding. Yuan et al. [132] conducted a study focused on grocery shopping activities by blind and visually impaired users. They identified often overlooked challenges such as identifying products that are running short, itemizing needed products, and organizing newly purchased prod-

ucts at home. Needfinding helps reveal the full task sequence rather than overemphasizing certain technical subproblems.

Personalization. Ohn-Bar et al. [89] showed significant variability between users in following turn-by-turn guidance. Interface timing and content may need to adapt to a user's walking pace, navigation skill, and preferred interaction style. This is relevant across the thesis because guidance must be technically correct and also usable in real time.

Privacy. Ahmed et al. [9] reported serious privacy concerns around current assistive technologies. This matters for camera-based systems, remote human assistance, shopping, and navigation in public spaces. Privacy cannot be treated as an afterthought, especially when systems collect images of the user's surroundings or reveal sensitive shopping goals.

Participatory design. Azenkot et al. [16] conducted participatory design work on building service robots. Participants wanted different assistance modes, options for stairs or elevators, varying levels of building detail, and behavior that does not draw extra attention. Participatory design helps avoid assumptions that can be unsafe, demeaning, or simply wrong for real users.

Disability simulation. Silverman et al. [103] and Williams et al. [123] showed that disability simulation can be inaccurate and can lead sighted people to underestimate the capabilities of blind people. Blindfolded studies can be useful for early safety and feasibility testing, but they cannot replace target-user studies. This limitation applies to several prototype evaluations in the field and should shape future evaluation of the systems in this dissertation.

2.7 Summary

Across the literature, a clear pattern emerges. Prior assistive navigation systems have made progress on obstacle avoidance, path following, and waypoint guidance, but they often do not reason about social appropriateness or goal meaning. Prior manipulation guidance systems show the value of verbal, kinesthetic, and human-in-the-loop assistance, but fine-grain autonomous product retrieval remains underexplored. Prior localization methods are strong when geometry is distinctive or when infrastructure is available, but quasi-static environments such as grocery stores break

many static map assumptions. Prior multimodal and vision-language systems can reason over rich observations, but they still need spatial grounding, topology, and reliable ways to connect semantics to robot pose and human-readable routes.

These gaps motivate four requirements for context-aware embodied AI systems in human-centered environments.

(1) **Interpret rich environmental context.** Systems should move beyond geometry and use social cues, object semantics, human behavior, and topological structure.

(2) **Operate without heavy environmental augmentation.** Systems should work in uninstrumented spaces without relying on RFID tags, UWB beacons, manually curated point-of-interest databases, or special markers.

(3) **Support guidance beyond obstacle avoidance.** Assistance should include intent-aware goal selection and richer route communication, not only safe motion.

(4) **Respect practical form factors and user experience.** The system should remain portable, low-profile, understandable, and sensitive to privacy, cognitive load, trust, and user preference.

The following chapters address these requirements through a sequence of systems. The first two systems study social context for navigation and human-movement-informed manipulation guidance. The next two systems study semantic localization using distributional semantics and rich VLM-characterized semantics. The final technical system studies semantic topology as a shared representation for various human-AI interaction tasks such as intent-aware search, zone categorization, and natural language route communication. Together, this thesis moves from assistive guidance to broader context-aware embodied AI systems that can perceive, model, reason, and communicate the structure of human-centered environments.

Chapter 3

A Novel Perceptive Robotic Cane for Socially-Aware Seat Selection

Publication note. This chapter is adapted from the paper **A novel perceptive robotic cane with haptic navigation for enabling vision-independent participation in the social dynamics of seat choice**, published in the 2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) [8].

3.1 Introduction

As established in the previous chapter, while assistive technologies for navigation have advanced, many systems struggle in complex real-world scenarios partly due to an underutilization of implicit environmental context. One crucial, yet often overlooked, aspect is the **social context** inherent in public spaces. Goal-based navigation in such places is critical for independent mobility, but simply reaching a destination, like an empty seat, is often insufficient. Sighted individuals implicitly navigate social dynamics, choosing locations based on factors like perceived privacy and appropriate interpersonal distance, capabilities largely unavailable to current assistive systems for people with visual impairments (PVI). Wang et al. [118] specifically identified finding empty chairs in crowded areas as a high-priority, yet poorly supported, task for BVI individuals using standard aids. Furthermore, psychological studies confirm that seating choices are strongly influenced by factors optimizing social comfort, such as intimacy and privacy [108].

This chapter addresses this gap by presenting a proof-of-concept perceptive robotic cane system (Figure 3.1) designed to bridge robotics and psychology. The system autonomously leverages

goal-based navigation assistance and environmental perception not just to find empty seats, but to specifically identify and guide a user towards **socially preferred seats** in unknown indoor environments. It combines computer vision, robotics (SLAM, object localization, path planning), and psychological insights regarding seat preference to score potential goals and plan an obstacle-free path. Guidance is provided via a novel 2-motor vibrotactile haptic feedback system integrated into the cane's grip.

Through a pilot user study, this work demonstrates the system's success in classifying seats based on social preference criteria (convenience, privacy, intimacy) and providing effective haptic navigation guidance towards them. This research represents a foundational step towards assistive technology that enables PVI individuals to more independently and appropriately engage in socially dynamic situations like seat choice, moving beyond basic waypoint navigation.

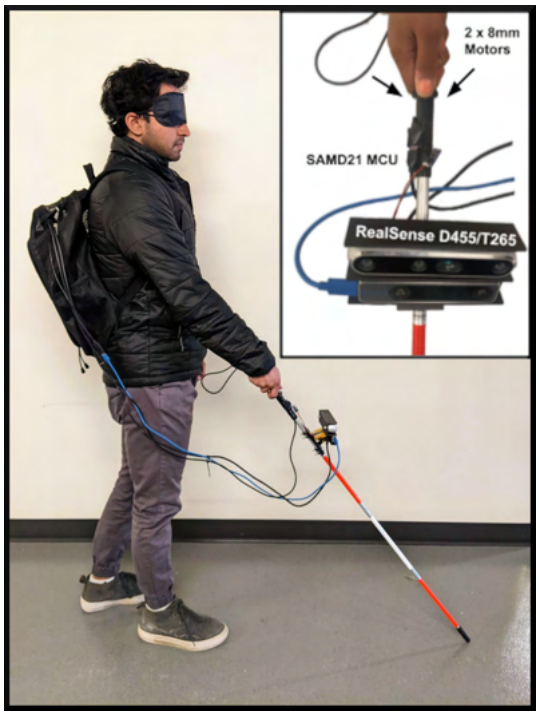


Figure 3.1: Robotic cane system developed for this work. It includes RealSense D455 and T265 cameras for SLAM and object localization, and vibrotactile motors for haptic navigation guidance.

Specifically, the contributions presented in this chapter are:

- A novel robotic cane system that leverages computer vision to enable socially preferred

autonomous goal selection and navigation in indoor spaces.

- An algorithm for social norm-aware chair selection, optimizing for convenience, intimacy, and privacy.
- A novel, intuitive vibrotactile feedback system providing navigation guidance through the robotic cane's grip.
- A pilot validation demonstrating the system's success in finding socially preferred seats and providing effective navigation guidance to novice users.

3.2 Related Work

Computer-aided assistive technology (AT) has been researched and developed for over 70 years [97], yet still does not have significant adoption from the BVI community.

3.2.1 Form Factors

Robotic navigation assistance has been investigated in various form factors, including mobile platforms, quadrupeds, wearables, and handheld cane-like devices. In the CaBot project from IBM Research [52], the team created an autonomous mobile robot in a suitcase form factor. Their platform is capable of indoor navigation and localization, driving next to its user, and leading them to their destination through a haptic interface. While this platform was an excellent testbed for haptics research, its final weight was over 50 pounds which substantially limited its portability and suitability for real-world deployment. CaBot also required a map of the indoor space a priori, which may not always be feasible. In work by Xiao et al. [126], a quadruped is used for navigation assistance in narrow spaces with a leash that accommodates slack, but is a loud and potentially disruptive alternative that is a substantial hindrance when unpowered. Wearable interfaces such as the ALVU [59, 118] can provide a less disruptive alternative, but imposes significant personal instrumentation and has limited capability due to its positioning on the person. A recent work [106]

developed a wheeled smart cane that can avoid obstacles and follow waypoints, but its powered wheel design significantly limits the feedback from the local environment that the cane can receive.

3.2.2 Smart Canes

Smart canes are a subset of assistive technologies that primarily rely on either ultrasonic sensors for obstacle detection and avoidance [88, 99, 110, 118] or computer vision to employ algorithms typically capable of path planning and goal-based wayfinding, as well as obstacle detection and avoidance [99]. The most popular smart cane systems have been developed by AssisTech[1] and WeWalk[2] . These products are equipped with a distance sensor and a single haptic motor and can only relay haptic feedback when there is an obstacle directly in front of the user and within a certain range. These devices do not support making or conveying navigational plans.

The majority of 'smart canes' surveyed utilized ultrasonic sensors for obstacle detection and communicated proximity and direction of the obstacle though vibrotactile motors [101], audio alerts [99], or a combination of the two [30, 82, 104, 115].

3.2.3 SLAM-based Navigational Aids

Chen et al. [26] developed a smart cane using Google Tango that performs mapping and localization, enabling path planning. Many SLAM approaches are prone to failures in dynamic environments [17], and recent research has made progress towards improving performance in such environments. Alcantarilla et al. [11] used dense scene flow to improve visual SLAM and demonstrated greater accuracy in real world experiments with BVI individuals. We chose to utilize online SLAM in our system to build a persistent spatial model of the scene, learning relevant navigation targets that can be utilized for providing better context and for conveying denser information than ultrasonic methods.

[1] https://assistech.iitd.ac.in/smartcane.php
[2] https://wewalk.io/en/

3.2.4 Feedback Mechanisms

Vibrotactile motors are used to provide programmatic feedback to a BVI user as a form of sensory substitution to convey visual information through touch. Users feel vibration through the white cane and are able to interpret changes in texture and elevation as well as obstacles through direct collision, however programmable feedback can be used to alert users of obstacles before collision and provide navigational guidance in ways a white cane can not. We have observed several encoding schemes throughout the literature that use vibrotactile motors to convey visual information, such as obstacles and their distance and navigational guidance using relative position distributed across the body as directional cues [104]. Notably, Nasser et al., developed the ThermalCane [85], a smart cane which encodes directional cues (go, stop, u-turn, left, and right) into thermotactile feedback by controlling changes in temperature through Peltier modules affixed to the cane's grip. While prior studies investigating the efficacy of vibrotactile feedback have varied wildly in their outcomes, this is largely due to variations in motor mounting, encoding schemes, and number of motors used. We opted to use vibrotactile feedback rather than thermotactile due to potential environmental issues that may effect the perception of temperature changes (e.g., wearing gloves on a cold day).

Motivated by Wang et al.'s [118] identification of 'chair finding' as the most important and desirable missing capability in white canes (and derivatives) and Staats and Groot's [108] finding that humans prefer to seat themselves in a way that optimizes privacy and intimacy, we created a novel perceptive robotic cane with vibrotactile navigation feedback. To the best of our knowledge, this is the first self-contained robotic system that can find socially preferred seats autonomously without any external human guidance. The presented system is simultaneously portable, useful without power, and benefits the user with situated environmental and social interactions that are infeasible with modern wearable and mobile robotics approaches.

3.3 Design Considerations

We have identified four considerations for the system design based on user feedback from two studies and observations we noted in research papers within our literature review. The first consideration, *modularity*, allows for iterative improvements without necessitating large integration costs. The computer vision, goal scoring algorithms, the sensors, the compute, and the vibration motors are swappable modules in the system. The second consideration, *usability*, involved a survey of work on modifications to white canes, many of which render the cane unusable for its original intended purpose. The addition of wheels [106, 112], requiring a fixed angle of the cane from the body [83,110], and excessive use of electronics along the cane's length [84] are identified as potential challenges to the user when operating the white cane with tapping or hovering techniques. This surfaced a third consideration, that *a robotic cane should still be able to function as a white cane for local environment feedback*, especially in case of system failure or loss of power. Fourth, *vibration motors should be collocated and perceptile through one fingertip* because of their high spatial acuity, which mitigates the spatial and temporal effects of perception when motors are distributed over greater distances across the body (e.g., palm, torso, thigh) [28]. This also allows for vibrations like those naturally experienced through the use of an uninstrumented cane to be perceptible because of the amount of unobstructed contact between the hand and cane, helping ensure the third design consideration is met.

3.4 System Design

We developed a robotic cane capable of identifying socially preferred seating [108] and planning an obstacle-free path to reach it. The system is capable of communicating with and guiding the user along the navigational path through highly localized yet distinguishable vibrotactile motors that are in contact with the tip of the thumb when placed on the cane's grip. The system follows a serial architecture with three primary components — perception, planning, and conveyance. They are designed to be modular so that they are easily replaceable as the supporting hardware and

foundational algorithms mature.

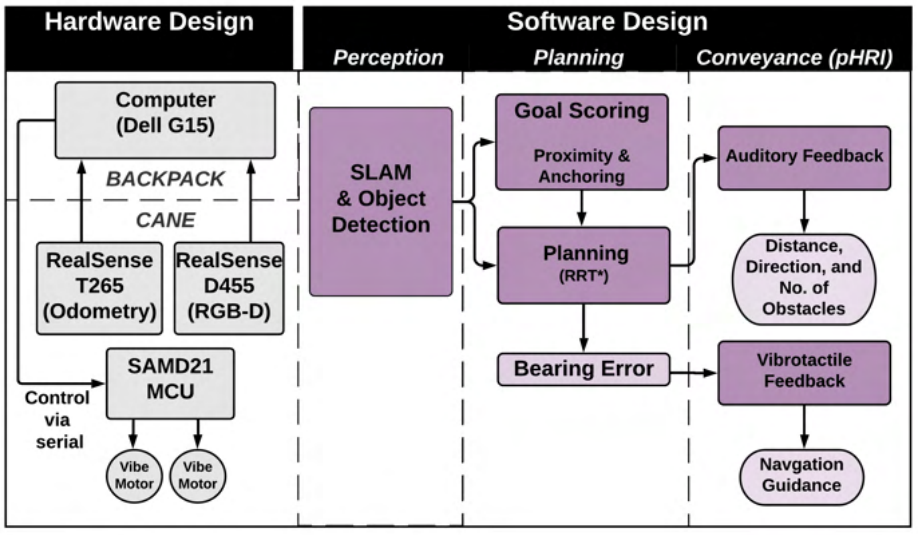


Figure 3.2: Design diagram. Perception and navigation algorithms are executed on a backpack-worn laptop, while all sensing and haptics are mounted on the cane.

3.4.1 Hardware Overview

The hardware as shown in the system diagram in Figure 3.2 consists of a cane-mounted RealSense T265 and RealSense D455, connected to a Dell G15 laptop with an RTX 3060 GPU carried in a backpack worn by the user. The RealSense T265 was chosen for its odometry and the RealSense D455 was chosen for RGB-depth sensing. Additionally, an Adafruit Trinket M0 SAMD21 microcontroller is connected to the laptop via USB to receive serial commands for controlling the vibration motors. The microcontroller was chosen for its small form factor. This extra hardware and its mounting support increases the weight of the cane by just (approximately) 0.33lbs, primarily located near the grip to minimize added fatigue.

3.4.2 Software Overview: Perception

The perception of the scene begins with a SLAM algorithm that creates an initial 2D occupancy grid from RGB+D camera input as the user scans the room by turning in place with gradual

movement. [3] The odometry provided by the Intel RealSense's IMU has low drift, allowing for very accurate user pose estimation. This enables accommodation of arbitrary handle tilt, as depth readings not within a specific height range can be discarded. The RGB+D data are also used to find objects and project them onto the 2D occupancy grid. The system uses Detectron2 for object detection and Mask-RCNN to obtain masks for classification. Classification masks are superimposed on the depth channel to estimate the depth of identified objects. Using the camera intrinsics, objects are mapped onto the 2D occupancy grid (Fig 3.3), completing the pipeline.

Challenge 1 (Data Association): Due to the system's high rate of movement, establishing proper data association between object observations and previously observed landmarks is a significant challenge. Particularly for objects that commonly exist in groups, such as chairs, it is important that the system be able to disambiguate between cases of nearby similar objects and a single object detected multiple times. For our reference implementation, we utilize an Extended Kalman Filter to model object positions as 2D Gaussians, providing a distribution against which to check new observations.

Challenge 2 (Exploration vs. Exploitation): Because the system is mapping its environment online and is intended to be usable in environments novel to the user, the decision of how to balance exploration of the space (scanning) with exploitation of the current map (choosing a target and navigating to it given the available free space) is critical for the user experience. As it is impossible to know whether there are more (possibly more desirable) objects of interest in the scene without performing the impractical step of fully exploring it, we parameterized the factors involved in this decision. The system utilizes a time-based and goal score threshold to determine when to move from exploration to exploitation.

3.4.3 Software Overview: Planning

We divide planning into two steps, goal selection and path planning:

[3] We use a publicly available SLAM implementation by RealSense at https://github.com/IntelRealSense/realsense-ros/tree/occupancy-mapping

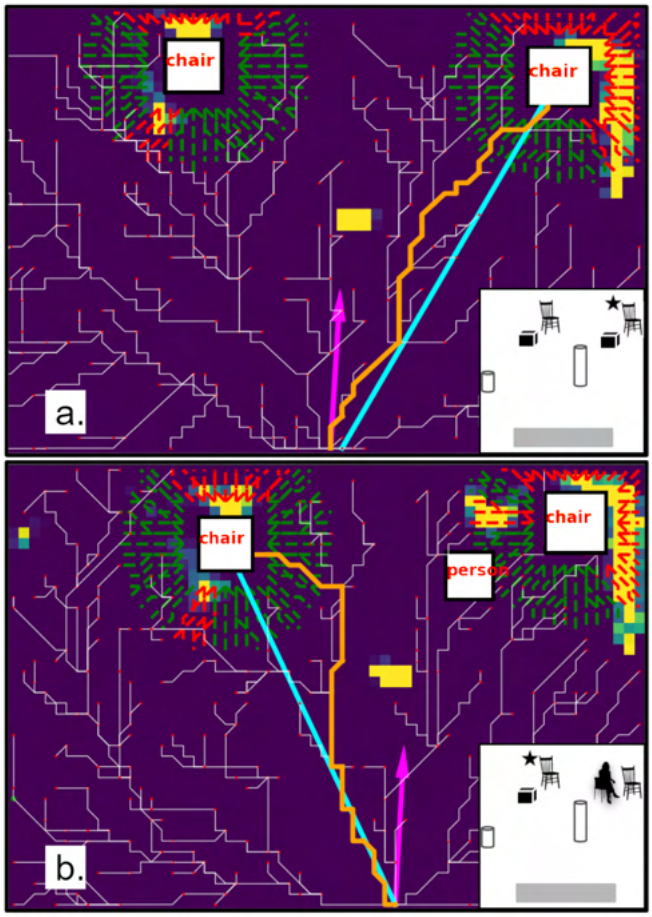


Figure 3.3: Visualization of live data from the system. 1) SLAM creates a 2D occupancy grid. Darker (purple) spaces indicate higher probability of free space. 2) Recognizable objects are projected onto the 2D grid with labels in white. 3) Goal scoring is done via the *anchor score* and *proximity score* to determine a navigation target. **Anchor score** rays are shown in green and red, with red indicating the rays contributing to a higher anchor score. 4) A path is found towards the goal using the Rapidly-exploring Random Tree (RRT*) shown in white. The unmodified RRT* path is shown in orange, and a pruned, optimized version of that path is shown in cyan. 5) The user's orientation is shown in magenta, which is used to generate the bearing error to be conveyed using the vibrotactile motors. (a) The system selects the right-hand side chair that has the *higher anchor score* due to walls on two sides, thus *increasing privacy*. (b) The system selects the left chair because there is a person close to the other chair, thus *decreasing intimacy*.

3.4.3.1 Goal Selection

As objects are detected and mapped by the system, they are considered as potential navigation goals. Each object of the target class (e.g., chair) is scored by an objective function accounting for levels of **convenience**, **privacy**, and **intimacy**. Ideal goals for the seating selection task

maximize convenience and privacy while minimizing intimacy. The system utilizes algorithms for evaluating **anchoring** to regulate privacy and **proximity** to balance convenience against intimacy. The objective function is defined as the sum of normalised anchor and proximity scores for each candidate goal object. A minimum threshold score (referenced in Challenge 2) can be used to inform the minimum score for a satisfactory goal candidate.

Anchor score: People prefer seats that are anchored in the environment to increase their privacy [108]. It is a common design choice for chairs to be anchored to large and contiguous objects, like a table that itself is anchored to a wall. Leveraging this, the system uses a novel anchor measurement ray-casting algorithm (Algorithm 1) to measure the relative anchoring of chairs using the 2D occupancy grid. The algorithm runs a sliding window of rays emanating from each detected chair in the scene (Green and red rays in Fig 3.3) to accumulate scores for each chair. Self-intersections with the chair being evaluated are ignored (defined by an `ignore_radius`), otherwise the rays emanating the center would not escape the chair itself (line 2). The sliding window approach assigns higher scores to contiguous obstacles (e.g., walls). A subset of rays is defined (line 7), counting how many of these rays intersect with occupied grid cells within `ray_cast_radius` of the object origin (line 9-13). If a window crosses the threshold (`wall_threshold`), we increase the `wall_counter` for the chair by 1 (line 14-15) (Fig 3.4). The normalized `wall_counter` value is returned as the object's anchor score (line 16-17). An example from real world data is visualized in Fig 3.3.

Algorithm 1: Anchor Score Procedure

Input: Chair Objects *Chairs*, 2D occupancy grid

Output: Normalised anchor scores

Parameters: `ray_cast_radius`, `window_size`, `ignore_radius`, `wall_threshold`

```
foreach chair ∈ Chairs do
    rays ← [ ] of rays cast from chair.position within ignore_radius to ray_cast_radius
    n_rays ← Total number of rays
    wall_counter ← 0
    for i ← 0 to n_rays − window_size do
        intercepted_rays ← 0
        for j ← i to i + window_size do
            /* ray is made up of cells that it spans over */
            ray ← rays[j]
            Iterate over ray
                if ray is intercepted by an occupied cell then
                    intercepted_rays ← intercepted_rays + 1
                    break
        if intercepted_rays/window_size > wall_threshold then
            wall_counter ← wall_counter + 1
    chair.anchor_score ← wall_counter
return Normalised anchor scores
```

Proximity score: People also prefer seats that are at a greater distance to others in order to decrease intimacy [108]. Considerations related to proximity are incorporated into the goal selection optimization process through a normalized linear cost model (Equation 3.1) with terms for **intimacy** ($C_{intimacy}$), **travel convenience** ($C_{convenience}$), **objects correlated with signs**

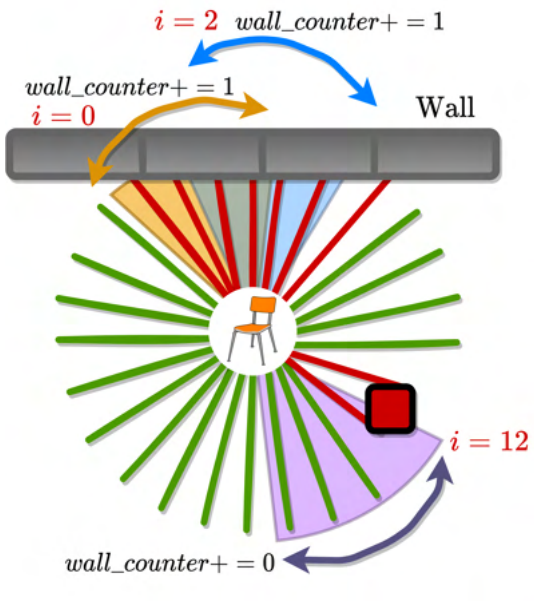


Figure 3.4: Anchor scores are calculated using a sliding window to track object-intersection density with radially cast rays. This method gives more votes to contiguous anchoring obstacles like walls. In the above example, the sliding window at $i = 0$ and $i = 2$ only contain object-intersecting rays (colored red, non-intersecting in green) by the wall and thus contribute 1 each to the *wall_counter*. Whereas the sliding window at $i = 12$ doesn't contain a sufficient density of object-intersecting rays and thus doesn't contribute to the *wall_counter*.

of occupancy ($C_{occupied}$), and **objects correlated with success** ($C_{success}$). The intimacy term penalizes goals that are closer to other humans, while the convenience term penalizes goals (g) that are further from the user. The object-occupancy term penalizes goals that are near objects likely to have owners, including backpacks or laptops, as these can be indicators that the goal is occupied or unavailable. Finally, the object-success term increases scores for goals that are close to alternatives or objects correlated with more selection (e.g., a table correlates with the existence of multiple chairs). $\mathcal{D}$ is used to represent the set of all detected objects or entities, with notation $\mathcal{D}_{human}$ indicating the subset of detections classified as humans, $\mathcal{D}_{occupied}$ indicating the subset of detections classified as object classes correlated with occupancy, and $\mathcal{D}_{success}$ indicating detections classified as object classes correlated with success. The model parameters are shown in Table 3.1, utilized in Equation 3.1, and visualized in Fig 3.5. While it is possible that this approach can still select an occupied goal if either the final 'acceptable score' threshold or $C_{intimacy}$ is kept very low,

in practice, we found that occupied goals (e.g., chairs) tend to receive very unfavorable scores due to $dist(\text{goal}, \text{human})$ being very small. Parameter values were manually tuned against reference scenes representative of normal use cases. Once computed for each candidate goal, the proximity scores are normalized. A sample result from real-world data is shown in Fig 3.3b.

Table 3.1: Parameters used for proximity scoring.

Parameter	Role
$C_{intimacy} = -6$	Penalizes nearness to other humans
$C_{convenience} = 1$	Rewards nearness with the user
$C_{occupied} = -1$	Penalizes nearness to failure-correlated objects
$C_{success} = 1$	Rewards nearness to success-correlated objects

$$S_{prox}(g, \mathcal{D}) = \frac{C_{convenience}}{dist(g, user)} + \sum_{d \in \mathcal{D}_{human}} \frac{C_{intimacy}}{dist(g, d)^2} + \sum_{d \in \mathcal{D}_{occupied}} \frac{C_{occupied}}{dist(g, d)^2} + \sum_{d \in \mathcal{D}_{success}} \frac{C_{success}}{dist(g, d)^2} \tag{3.1}$$

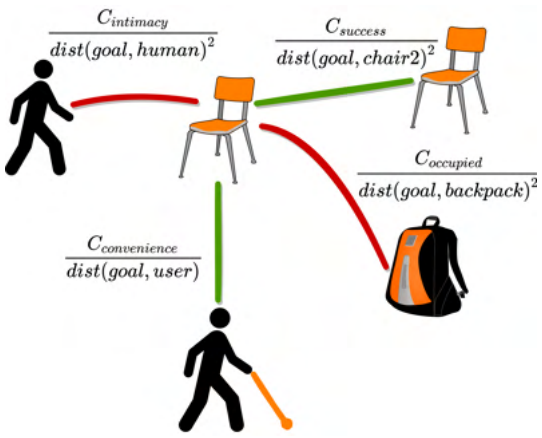


Figure 3.5: A chair's proximity score is affected by its relative position with respect to other socially meaningful objects, humans and the user. We add the scores shown by the green lines and subtract the ones shown by the red.

3.4.3.2 Path Planning

Once a goal is selected from the set of detected objects (e.g., the chair with the best overall score), RRT^* [58] is used to find a collision-free path from the user's current position. The resultant

path is successively pruned to avoid unnecessary motion through extraneous waypoints (Fig 3.3). We remove extraneous waypoints by recursively calculating the feasibility of the path between a waypoint's predecessor and successor.

3.4.4 Conveyance

Plans are conveyed to users through two modalities: a verbal goal overview and vibrotactile haptic guidance.

3.4.4.1 Verbal goal overview

Once the goal is found, the system generates a semantic description of the goal's relative location. The overview has the following template: *"{Goal Object} found about {} meters away in the {} o'clock direction"*, based on user patterns observed in popular human-guided assistive applications (e.g., **Be My Eyes**). The distance and direction estimates are calculated from the relative positioning of the last determined user pose and the goal. The verbal overview is presented to set user expectations about the navigation duration. The phrase *"with obstacles in the path"* is appended if there is not a collision-free straight-line path to the goal to set expectations regarding the complexity of the navigation plan. This overview is delivered via the laptop speaker, and precedes the vibrotactile guidance.

3.4.4.2 Vibrotactile guidance

The robotic cane utilizes a novel vibrotactile haptic communication system, providing navigational guidance through the grip. This is informed by prior work that has shown that two-point vibrotactile discrimination on the fingertips ranges from 2.1mm–8mm [92]. Accordingly, the robotic cane has two vibrotactile motors (8mm ERM) affixed on the grip such that during normal use the user will naturally position their thumb between them. This design removes the need for additional wearables (e.g., haptic belts) and minimizes the area of modification on the cane itself to maintain familiarity of feel.

Navigation information is relayed to the user by encoding waypoint bearing error into two distinct haptic animation patterns on each motor (left and right), creating five possible codes for conveyance: hard-right, soft-right, straight/no animation, soft-left, and hard-left. The design decision to have more than a single pattern per direction is rooted in literature indicating a preference for two levels of turn (e.g., *sharp left* and *soft left*) [123]. The soft-right and soft-left animations are programmed at one-half the vibration intensity of the hard-right and hard-left animation. The motors vibrate at 200 Hz. Intended use involves the user first correcting their bearing error by turning in-place in the direction of the corresponding vibration motor, then walking towards the next waypoint. A distinct animation pattern (both motors cycle between actuating at 100% duty cycle for 750ms and then at 0% duty cycle for 750ms for three seconds) signals the start and stop of the plan.

One significant challenge encountered during the design process was the observation that mounting vibrotactile motors directly to the cane causes the entire cane to vibrate, making it very challenging to tactilely discriminate between the vibration source as either left or right. Our system addresses this by mounting the wires of the motors directly to the cane grip, but allowing the motors to hover near the surface so that the cross-section of direct contact with the cane grip is negligible.

3.5 Pilot Study

We conducted a pilot study (n=6; 2 male, 4 female) with novice users for preliminary testing of the device, which included navigating through six scenarios while blindfolded. Participants were all sighted graduate students and had no prior experience with using a probe (e.g., white cane) to navigate. For testing at this stage, we follow prior work in performing preliminary validation using blindfolded participants [35,40,90,91,106,114] as a precursor to engaging with the BVI population.

The pilot tests took place in a configurable $12ft \times 17ft$ room as shown in Figure 3.6, representing more complex variants of room configurations used for similar studies of visually-impaired navigation [118]. The room ordering is randomized during the study to limit learning effects. Room

1 is designed to compare a user's proficiency in finding and navigating to a seat in a simple single chair scenario. Room 2 and Room 3 are designed to test the algorithmic goal-scoring capability, locating the more socially preferred seat based on proximity (which influences choices based on intimacy and convenience) and anchoring (which influences seat choice based on privacy). Cardboard boxes, bins, and suitcases are used as obstacles throughout each of the rooms to increase navigation difficulty.

During the experiment, users navigated through each room twice, using the robotic cane both with and without the verbal overview. The order of presentation of room layout and verbal condition was randomized, with users unaware of the number of unique layouts being presented. To begin a trial, the participant was blindfolded and led to the starting area (shown in gray in Figure 3.6) with random orientation.

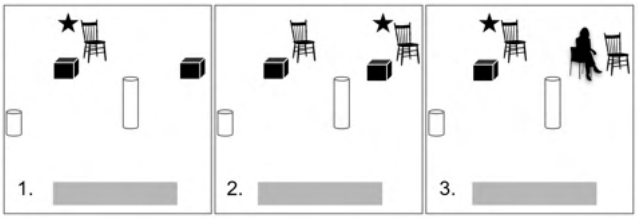


Figure 3.6: The three room layouts: The Room 1 layout includes one chair and four obstacles. The Room 2 layout includes one highly anchored chair, one unanchored chair, and four obstacles. Room 3 layout consists of a highly anchored chair next to an occupied chair, one unanchored chair, and three obstacles. Stars denote chairs that should be chosen according to anchoring/intimacy-based western social norms and the gray rectangles represent the starting locations.

3.5.1 Procedure

Each user completed six scenarios while blindfolded, each scenario comprised of a combination of Room (1,2,3) and robotic cane with verbal overview enabled/disabled. For each scenario, the blindfolded user was guided to a starting location. When given the verbal "START" cue, the user was instructed to begin by performing a scan of the room by rotating in place. Once a plan is developed, the system's haptic start sequence is conveyed to the user, indicating that they can begin navigating to a chair following the navigation guidance provided by the vibrotactile interface.

In between each scenario, the user is disoriented by being guided randomly around the testing area and while the room is rearranged into the configuration for the next scenario.

3.5.2 Results

Task Completion: Within each scenario, users were tasked with finding a suitable chair to sit in within 2.5 minutes. This task was considered successfully completed if the user navigated to a chair and said "FOUND". We obtained promising results from our algorithm that optimized for privacy, intimacy, and convenience. Users were able to find the highly anchored chairs in **10/12** scenarios. We saw a similar success rate for Room 3, which had a person (quietly) sitting in a chair next to the highly anchored chair. The system was able to guide the users to the chair that minimized intimacy in **10/12** scenarios, thus effectively demonstrating the proposed proximity scoring formula's ability to guide users toward socially appropriate seat choices. Even in the four scenarios where the system was not able to find the more socially preferred chair, users were still successful in finding **a** chair, thus achieving a **100%** success rate in finding a seat and **83.3%** success rate for finding a socially preferred seat. In these less socially appropriate cases, we observed that users had obtained diminished scan coverage of the room, triggering the system to stop further exploration of the scene and suggest the only available choice to avoid idling for too long.

Navigation Time: Navigation time is measured starting at the moment the user is able to begin moving, after the plan has been calculated and the start sequence has been communicated to them. The navigation timer is stopped once the user comes in contact with the goal using the robotic cane, to avoid confounds due to delays in providing the verbal "found" cue. We used a cut-off time of 2.5 minutes for each trial, but participants were able to find a seat within 45 seconds on average with none exceeding the cut-off.

Obstacle Avoidance: A summary of the average number of obstacles the users physically interacted with in each scenario is presented in Figure 3.7. Despite the fact that the rooms were populated with obstacles and walls, we observe that due to the vibrotactile navigation's effectiveness users were often able to avoid causing even a single collision while using the robotic cane. This

is an encouraging result as BVI individuals have expressed the desire to avoid cane contacts with other people [118]. Through this pilot study we have shown, with inexperienced users, that one benefit of our system is to help eliminate the socially undesirable situation of collisions with people seated in chairs (Room 3).

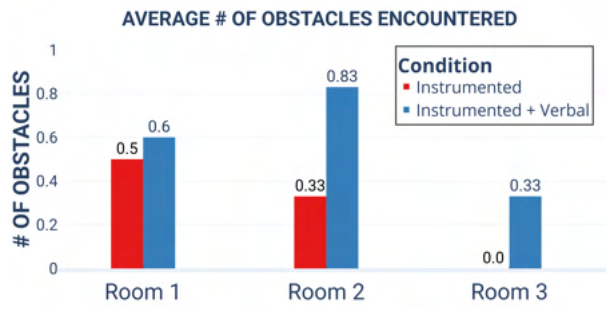


Figure 3.7: Even though the rooms had obstacles and walls, the novice users in our study often were able to avoid collisions with the environment entirely while navigating to their goal.

User Experience: We conducted post-activity surveys to measure aspects of the user experience focused on usability and confidence in independent navigation. We found that the users were confident accomplishing the task with our system, more so with the verbal overview enabled. Using a 5-point scale, participants rated their *confidence in navigation* at 4.83 ± 0.41 (verbal enabled) and 4.00 ± 0.63 (verbal disabled), their *confidence in travelling fast* at 4.33 ± 0.82 (verbal enabled) and 4.00 ± 0.63 (verbal disabled), and their *confidence in finding the goal* at 4.5 ± 0.84 (verbal enabled) and 3.83 ± 1.17 (verbal disabled). Users also rated the *verbal overview's helpfulness* at 4.67 ± 0.82 and the *ease of use* of the system at 4.17 ± 0.98 (verbal enabled) and 4.67 ± 0.52 (verbal disabled). The users perceived their performance to be adequate with little effort. Using the NASA Task Load Index survey's battery of questions (20-point scale), users rated their *performance* at 16.67 ± 3.39 (verbal enabled) and 17.00 ± 2.61 (verbal disabled), and their *effort* at 7.83 ± 4.12 (verbal enabled) and 4.67 ± 3.43 (verbal disabled).

Overall, pilot study participants rated the system as being highly usable and highly performant. In a post-activity interview, they mentioned the following:

- Verbal overview - *"I used the verbal instructions as a global plan to reorient myself", "Just*

having a mental picture of how far I need to go, that was good", "[The] head start with the verbal announcement was a little comforting/assuring" .

- Haptics - *"Hard and soft turns were easy to distinguish","The soft vs hard vibrations helped me a lot in determining how much to turn","[the] feedback was pretty fast".*

3.6 Discussion

We caution the reader to exercise care to avoid drawing conclusions about device readiness based solely on pilot tests with blindfolded, non-BVI participants, as these are not a reliable proxy for the (eventual) target population of this device [35, 103]. With this pilot study we have shown a proof of concept with regard to the hardware and software features of the system alongside a validation of the novel vibrotactile navigation interface, demonstrating that the cane-mounted system is able to construct a usable environment representation, plan within it, and effectively guide users to an autonomously determined desired goal location based on higher-order social features in real-time.

We observed that each user scanned the rooms uniquely, capturing various amounts of coverage which led to variations in the amount of information obtained for each room. We found early on that a much slower scan sequence did not allow the algorithm to view the complete scene within a reasonable time frame. To avoid an assumed annoyance to the user, the algorithm calculates a path to the best available chair ('local optimum') after a fixed timeout is reached, even if this means that the entire area hasn't been scanned and the 'globally optimal' chair hasn't been detected. This leads us to believe there are opportunities to improve user experience by exposing some of the system's inner workings to the user by allowing them to select their own scan time.

Interestingly, we also observed that all users performed best (as rated by task completion, reduced navigation time, and number of obstacles avoided) when using the robotic cane without verbal overview, yet 5/6 users stated they preferred the robotic cane **with** the verbal overview. It also became clear that the kinesthetic feedback from the cane resulting from the few object

collisions that occurred was a valuable signal, reinforcing the importance of maintaining this core capability of the white cane form factor. These observations will need to be further explored within a user study with BVI individuals.

3.7 Conclusion

This chapter presented a novel perceptive robotic cane system designed to incorporate **social context** into assistive navigation. By combining environmental perception with psychological models of seating preference (privacy, intimacy, convenience) and utilizing an intuitive vibrotactile guidance system, we demonstrated the feasibility of autonomously guiding users towards socially preferred goals in indoor environments. The pilot study with blindfolded novice users validated the core components, showing high task success rates, effective social goal selection based on the scoring algorithms, positive overall usability feedback, and higher mean confidence ratings when verbal and haptic feedback were combined.

This work represents a foundational step towards assistive systems that understand and operate within the social nuances of public spaces. It directly addresses the gap identified in Chapter 2 concerning the underutilization of social context in assistive navigation. While successfully demonstrating guidance based on social norms, this system focused primarily on reaching a target location. Many real-world tasks, however, require not just reaching a vicinity but also performing **fine-grained interaction and manipulation** within the haptic space, such as retrieving a specific item from a cluttered area. This necessitates incorporating different forms of context, specifically semantic understanding of objects and precise physical spatial reasoning. The next chapter details the "ShelfHelp" system, which tackles these challenges in the context of assistive grocery shopping, focusing on manipulation guidance leveraging semantic and physical cues.

Chapter 4

ShelfHelp: Grasp Guidance Integrating Semantic and Human Behavioral Context

Publication note. This chapter is adapted from the paper **ShelfHelp: Empowering Humans to Perform Vision-Independent Manipulation Tasks with a Socially Assistive Robotic Cane**, published in the Proceedings of the 2023 International Conference on Autonomous Agents and Multiagent Systems (AAMAS) [7].

4.1 Introduction

The ability to shop independently, especially in grocery stores, is important for maintaining a high quality of life. However, this can be particularly challenging for people with visual impairments (PVI). Stores carry thousands of products, with approximately 30,000 new products introduced each year in the US market alone [87], presenting a significant challenge even for modern computer vision solutions. While Chapter 3 addressed socially-aware navigation, many daily tasks require fine-grained manipulation in cluttered spaces, a capability less explored in assistive robotics.

Through this work, we present "ShelfHelp," a proof-of-concept socially assistive robotic system extending the perceptive cane platform introduced earlier. ShelfHelp aims to enhance the capabilities of instrumented canes, traditionally meant for navigation, by adding support for manipulation within the domain of shopping. We propose novel technical solutions including a visual product locator algorithm designed specifically for grocery store environments and a novel planner that autonomously issues verbal manipulation guidance commands to guide the user during prod-

uct retrieval. The system (Figure 4.1) leverages computer vision, robotics planning, and insights derived from human guidance strategies.

Figure 4.1: ShelfHelp includes a robotic cane equipped with RealSense D455 and T265 cameras. The system is powered through a laptop in a backpack. *Left:* The system used as a navigational device. It uses audio and haptic feedback for navigation guidance. *Right:* The system used as a manipulation device (focus of this chapter). It uses audio for manipulation guidance.

Currently, when encountering difficulty with shopping, PVI individuals often rely on sighted human assistance. This lack of independence can have negative connotations, including a potential loss of privacy [9]. Some PVI shoppers have indicated they are unwilling to use store staff for items requiring discretion, like medicine or personal hygiene products [9, 67, 69]. Our work seeks to alleviate this dependence and mitigate privacy concerns. PVI individuals have also expressed a need for systems to help locate items both in stores and at home [9]. Furthermore, environments like grocery stores, with densely packed similar items, pose challenges for purely tactile exploration and differentiation.

Grocery shopping primarily consists of three main subtasks: navigation, product retrieval, and product examination. This work focuses specifically on **product retrieval**, operating within the user's **haptic space**, defined in contrast to the *locomotor space* requiring bodily movement [46]. We define fine-grain manipulation guidance as assistance that brings the user's hand close enough

to the desired object for it to be within their haptic space and accessible for grasping, minimizing exhaustive tactile search.

We developed and evaluated two distinct autonomous verbal guidance modes, comparing their performance to a human assistance baseline through a human subjects study. The results demonstrate ShelfHelp's success in locating specific products and providing effective manipulation guidance to novice users. Encouraging findings validate the system's efficiency and effectiveness, supported by positive subjective user feedback on metrics including competence, intelligence, and ease of use.

Specifically, the contributions presented in this chapter are:

- A robotic cane system that leverages computer vision and human behavioral modeling to assist in independent grocery shopping for PVI.
- A modular two-stage computer vision pipeline to locate desired products based on semantic identity in a grocery setting.
- A novel fine-grain manipulation guidance system, learned partially from human data, that optimizes for guide time and command count without compromising legibility.
- A pilot study validating the system's success in locating products and providing effective guidance with novice users, comparing against human assistance.
- Findings on user preferences regarding planner properties and the desire for contextual grounding in guidance.

4.2 Related Work

4.2.1 Manipulation guidance

Manipulation guidance is an area that has been explored within the robotics community for over a decade. Vasquez et al. [113] showed that saliency maps could be used to find regions of interest (ROI) and directed users' hand to the ROI. They found that their verbal commands' efficacy

suffered because they did not utilize a global frame of reference. Bonani et al. [21] showed promise for the concept with an experimenter-controlled teleoperated system and Bigham et al. [19] did so with a Mechanical Turk-based system, but fully autonomous implementations were outside the scope of their contributions. Their manipulation guidance guided people to the general direction of the product but the system didn't focus on fine-grain manipulation guidance which is important in many scenarios, for example a kitchen countertop or where tactile differentiability makes exhaustive search inefficient such as grocery stores which have similar items densely situated. The most popular solution in this problem domain is a human-powered service called Be My Eyes [36], but this service suffers from scalability issues due to its reliance on human availability, is not readily available in developing countries, requires an active data connection, and introduces nigh-unavoidable privacy concerns. We present a novel verbal guidance solution wherein we learn a mapping of language commands to human hand movements and map them to actions within a Markov Decision Process (MDP) that can be solved with well-established reinforcement learning techniques, informing our guidance of the user.

4.2.2 Product identification

Existing techniques with a fixed number of output classes work with a limited database of products in a grocery store, for example, Feng et al. [37] trained a system for classifying 1329 products. These techniques may be impractical for the scale required of a solution for this domain because of the sheer amount of products [87] available and the slight variations they come in. It is infeasible to require creation and maintenance of labeled data and the training of an object classifier. Existing technology that is reliable in product identification uses bar code scanning [43, 47]. This often needs internet connectivity and can only identify a product once it is in the person's possession, making it inefficient for search over the grocery shelves. Our work is inspired by the body of work in **instance retrieval**, where the task is to find a target image in the scene [27]. Our proposed product identification system does not require re-training and works offline, making it scalable and practical.

4.2.3 Grocery assistant systems

Recent work on grocery assistant systems focuses largely on navigation inside the store [43, 67–69, 86], also known as the locomotor space of the user. Solutions that rely on environmental augmentation, such as the addition of RFID tags and barcodes, introduce barriers to adoption as they are inapplicable in uninstrumented domains. Barcodes alone can not reliably be used to locate a product from a dense cluster of products on a grocery shelf, as the shopper can typically only scan a barcode once they have retrieved the product. Prior research [43] has also focused on text-based product selection as an input method. Researchers have also focused on other issues around shopping, including identifying products that users are running out of and organizing newly purchased products at home [132], as well as "the last few meters" way-finding problem [100] and solutions for people with low vision [137]. We focus on an unsolved research area that primarily considers the haptic space of the user.

To summarize, our solution addresses 1) the problem of locating a desired product and 2) the challenge of providing effective fine-grain verbal guidance to reach and grasp the product.

4.3 System Design

ShelfHelp extends the capabilities of an existing robotic cane originally developed as a navigational assistance device. This was a design choice to re-purpose an existing hardware system since many of the navigational assistance prototypes have the necessary sensing and compute (Figure 4.1). The system is capable of locating desired products and verbally guiding the user to retrieve them. The software system follows a serial architecture composed of three main components - *perception, planning,* and *guidance*.

4.3.1 Hardware System

The hardware (Fig 4.1) consists of a cane-mounted RealSense D455 for RGB-Depth sensing and RealSense T265 for odometry. The sensors are connected to a Dell G15 laptop with an RTX

3060 GPU carried in a backpack worn by the user. As a navigational assistance device, the system is held with a standard cane grip but as a manipulation assistance device, a different, less collision prone grasp is used behind the sensors.

4.3.2 Software System

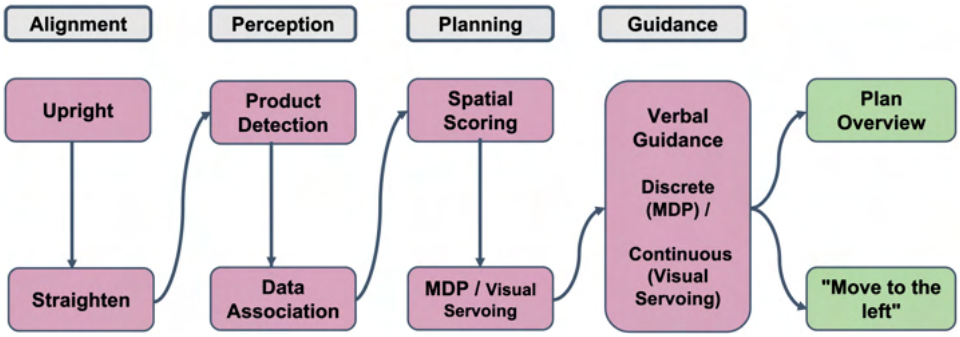


Figure 4.2: System Diagram. Alignment, perception, planning, and verbal conveyance are executed on a backpack-worn laptop, while all the sensing is mounted on the cane.

Our work can best be partitioned into sections regarding innovations in perception, solutions to the data association problem for maintaining a consistent and persistent mapping of product detections over time, product selection, fine-grain manipulation planning to reach the selected product, and methods for conveying this manipulation plan to the human user to complete the task.

4.3.2.1 Alignment

In order for the system to issue commands in the user's egocentric frame of reference, our system issues verbal commands to align the user and the system with respect to the shelf. We use the T265 camera's IMU to guide the user to point the system straight at the shelf in an upright way. We also align the user so that they directly oppose the shelf normal. This is done by locating the shelf after camera upright alignment via Hough transform. The system issues commands to turn in place until the largest horizontal line in the Hough transform is almost completely horizontal in the camera frame.

4.3.2.2 Product Detection

Algorithm 2: Product Detection Procedure

Input: scene, target_product, camera_pose

Output: best_product

Parameters: `similarity_threshold`

1 $boundingbox\_list \leftarrow region\_proposal(scene)$

2 $target\_feature \leftarrow feature\_extractor(target\_product)$

3 $similar\_instances \leftarrow []$

4 **foreach** $bb \in boundingbox\_list$ **do**

5 $\quad bb\_feature \leftarrow feature\_extractor(bb)$

6 $\quad bb\_score \leftarrow similarity(target\_feature, bb\_feature)$

7 $\quad$ /* *Scene has RGB and depth information* */

8 $\quad bb\_pose \leftarrow camera\_to\_world(bb, scene, camera\_pose)$

9 $\quad$ **if** $bb\_score \geq$ `similarity_threshold` **then**

10 $\quad\quad similar\_instances.append(bb\_pose, bb\_score)$

11 $similar\_products \leftarrow data\_association(similar\_instances)$

12 $best\_product \leftarrow spatial\_scoring(similar\_products)$

13 **return best_product**

The entire product detection pipeline works in realtime making it appropriate for real world usage. To use this system, we require that the user has only a single image of the product that they want to find. This image can be acquired in a number of ways, for example by taking a picture the first time it is purchased or by downloading an image from the internet. We have developed a novel two-stage product search system (Algorithm 2).

- In the first stage, our method proposes regions in form of bounding boxes that are most likely to contain **any** product (line 1 of Algorithm 2). We train the YoloV5 network on the SKU-110K dataset [45] to create a product detector. This network is robust and generates

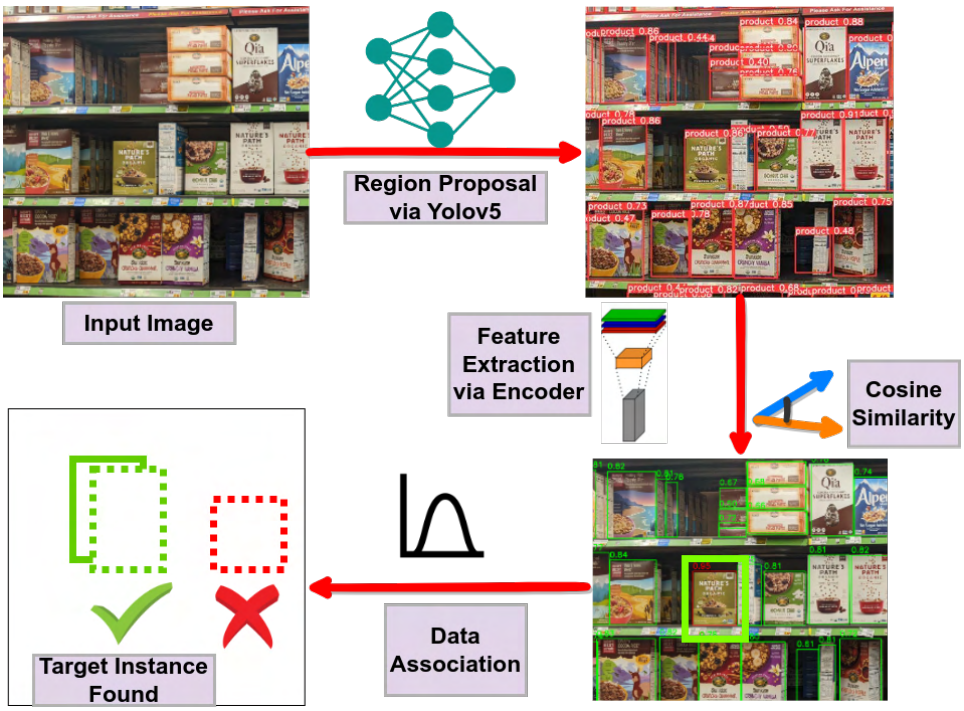


Figure 4.3: Our product search algorithm can reliably locate desired products on a grocery shelf. Regions with a high likelihood of containing any product are proposed in the first stage. The features of these regions are then compared against the target product image. Our data association solution is used to identify whether detections from incoming camera frames are new or re-detections of existing products. The above image shows our algorithm operating within an actual grocery store, where the product classification aspect of this work has been tested and validated. The data association and manipulation assistance components were validated within a lab-based study.

region proposals with 0.91-precision and 0.77-recall upon cross-validation with the SKU-110K dataset. Figure 4.3 (upper-right) shows a sample result from a real grocery store.

- In the second stage, an encoder is used as a feature extractor (line 2 and 5) that matches the features of the proposed regions and the target image, finding the best possible match (Fig 4.3). We do this by training an autoencoder on MS-COCO color images [73] and then utilizing the encoder portion as the feature extractor (Fig 4.4). This method doesn't require any retraining and works in real-time. For each new image added to the database, we pass it through the frozen encoder and save the generated feature vector on disk. This avoids requiring any retraining of the autoencoder. The encoder finds the latent space representation of the target product image and each of the proposed regions. We compare the

representation vectors using *cosine similarity* to find closer vectors (line 6). We empirically determine a similarity score threshold (0.6) that captures satisfactory performance across real-world environments, but this value can easily be fine-tuned in case there is a significant distribution shift between the evaluation and deployment environments.

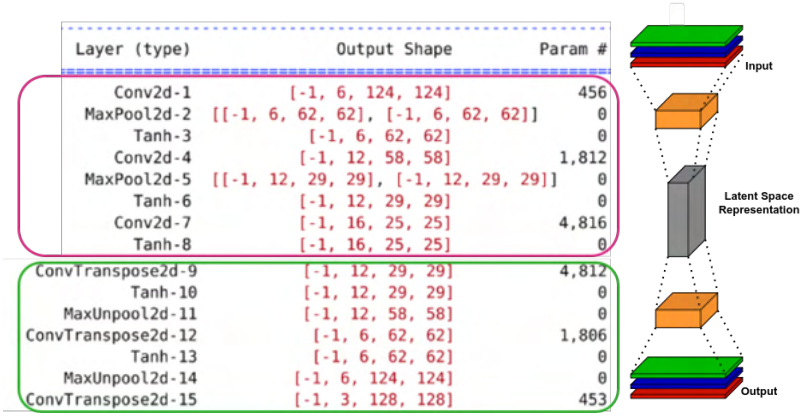


Figure 4.4: The architecture of the *autoencoder* trained for feature extraction. ShelfHelp uses the *encoder* portion (top layers outlined in red) as the feature extractor that creates a latent space representation of the images.

To transform the information from the camera frame to a fixed global frame, we use pose information obtained by a cane-mounted RealSense T265 (which has minimal drift in indoor settings) running an onboard Simultaneous Localization and Mapping (SLAM) algorithm, and fuse it with the depth information obtained from a cane-mounted RealSense D455 (line 8). We use a Gaussian Mixture Model (GMM) to refine detections and distinguish between the foreground and background depth information, as the bounding boxes can contain significant background pixels in case a product and its bounding box are not overlapping significantly.

4.3.2.3 Data Association

We use data association techniques to identify each instance of the same product uniquely across subsequent frames of camera capture (line 11). This is particularly challenging because similar products exist in groups. We do this by defining each product instance as a multivariate Gaussian defined by the tuple $p = \{x_g, y_g, z_g, w, h\}$ where x_g, y_g, z_g is the 3D location of the product

in a global frame and w, h are width and height in meters. Accounting for the width and the height helps us discard some incorrect matches, as incorrectly proposed regions with our method not only have to have similar features but also similar shape to be incorrectly labeled (Fig. 4.3 - lower left). The IMU data from the T265 sensor helps to calculate the object's pose in a global frame of reference and helps to create a persistent "map" of the product location. This way we can align the verbal directional commands with respect to the current hand pose and avoid the drawback of formulating verbal commands generated with targets located only in the camera frame of view, which is sensitive to hand movements [113]. Product instance information is updated using a rolling mean over associated detections. We also employ a *lazy deletion strategy* to delete instances that have not been seen a sufficient number of times (sparse detections) or recently (old detections).

4.3.2.4 Spatial Scoring

ShelfHelp then scores each detected product instance of the target product and picks the one with the highest score as its planning goal (line 12). It considers the rolling similarity with the target image and the spatial information. This allows the system to utilize important information that is absent without an explicit physics model, namely selecting an instance from the top level of stacked items to minimize the risk of toppling (Fig 4.5).

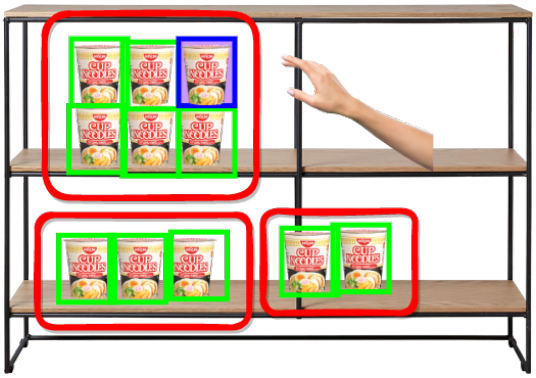


Figure 4.5: The spatial scoring system clusters all the found instances spatially and gives preference to the closest cluster to the current hand pose. Ties are broken arbitrarily.

4.3.2.5 Planning

We have developed two different guidance mechanisms to provide verbal instruction once the target has been located: *1) continuous*, for example, *"keep on going right"... "stop"* and *2) discrete*, for example, *"move 6 inches to the right"*. The decision to create a discrete guidance mode was inspired by study results showing that PVI have been known to perceive length units better than sighted people [13] and the criteria of minimizing verbal feedback for the task as research has shown that PVI prefer not being provided with excessive information [105].

4.3.2.6 Continuous Planner

The continuous guidance operates by calculating the relative position of the target and the device (Fig. 4.1), providing continuous cues along each individual axis of movement until the next is aligned. It generates the commands with the following template: *"keep on going* {*direction*}*"* where direction can be {*left, right, up, down, forward, backward*}. The system issues the command *"keep on going"* if it notices the hand slow down sufficiently in anticipation of the next command. Once the error on the current guidance axis is lower than a threshold the system issues the command *"stop"*.

4.3.2.7 Dataset from Human Demonstrations:

To develop the discrete guidance method, we collected a dataset mapping verbal movement commands from a fixed command-set, recording participants' net hand movements upon reacting to that command (Fig 4.6). We formed 36 discrete commands and issued 1250 instances of the commands in total to 25 volunteers (50 commands per person) while they were blindfolded. The hand movement data were recorded using an OptiTrack motion capture system. Figure 4.6 shows a sample from this dataset illustrating commands pertaining to *left* movement. We can see that the hand movement caused by these commands is close to a Gaussian distribution and the *mean* increases for the verbal commands that convey a larger movement. We fitted Gaussians to

characterize the movement caused by each command as

$$X_c \sim \mathcal{N}(\mu_c, \sigma_c^2)$$

where μ_c and σ_c are the mean and standard deviation of the movement caused by command c.

4.3.2.8 Discrete Planner

Through the human-subjects demonstrations, we find that the hand movement following the commands was not deterministic. Moreover, a greedy approach would not guarantee an optimal solution, akin to the 0-1 Knapsack problem. We solve this sequential decision-making problem by modeling it as an MDP formulated with (S, A, T, R) where S is the set of states in the MDP, A is the set of actions, T is a stochastic transition function describing the action-based state transition dynamics of the model, and R is a reward function.

- S is the tuple $(\Delta x, \Delta y, \Delta z, axis)$ where the first three terms are the difference in distance of the target and the hand pose, and **axis** defines the axis of motion for the previous command which could be any of three values corresponding to the X (horizontal), Y (vertical), or Z (depth) axis and a direction {*left, right, up, down, forward, backward*}. This formulation is dependent on the relative distance between the target and the hand and thus it finds a plan for all the potential states that can be encountered. We discretized the states at 5 cm resolution and considered a cuboid region of 0.8m as the operational space for the human hand. The system assumes the hand to be at the boundary of our cuboid in case it was outside of the region. In practice, a region bigger than this produced the same policy for regions that were further away.

- A is the set of discrete verbal commands such as *"Move 6 inches to the left"*.

- T is calculated from X_c as the movement caused by each command is not deterministic. We use the Euclidean difference between the states and Gaussian X_c associated with the command c to find out the probability of the transition between those states when command c is issued.

- The reward function R encourages reaching the target and discourages issuing superfluous commands. It also discourages a sequence of commands that could be illegible or frustrating by penalizing axis changes. Table 4.1 shows the reward values and their description. The *axis_order_reward* rewards adherence to a prescribed ordering of guidance with respect to the axes. For example, we set the initial guidance axis as *vertical* and the planner rewards transitioning from the *vertical* axis to the *horizontal* axis and not the *depth* axis. This helps to hone in on the product on the XY plane and then reach out for the product in the *depth* axis. The *living_penalty* penalizes excessive number of commands. The *interleaving_penalty* penalizes excessive interleaving of axes that could make the guidance illegible. It is necessary and suitable to transition once the distance to the target on the current axis of guidance is reduced and short enough. We ensure this by setting *necessary_transition_reward* inversely proportional to the distance remaining (error) on the current guidance axis.

Table 4.1: Outline of MDP Reward values

Reward	**Description**
$goal\_state\_reward = +10000$	reward for reaching the goal
$living\_penalty = -10$	penalty for using a command
$interleave\_penalty = -100$	penalty for changing axis of command
$axis\_order\_reward = +100$	reward for transitioning from vertical to horizontal axis
$necessary\_transition\_reward =$ $(0.001 + \text{current axis error})^{-1}$	reward for axis transition when current axis error is removed

The MDP is then solved using value iteration to generate a general reaching policy that can be queried online to guide the user toward arbitrary target locations.

4.3.2.9 Guidance

The verbal guidance has two components. It begins with a plan overview that is followed by hand movement instructions.

1) Plan Overview: The actual guidance is preceded by a plan overview which serves to set

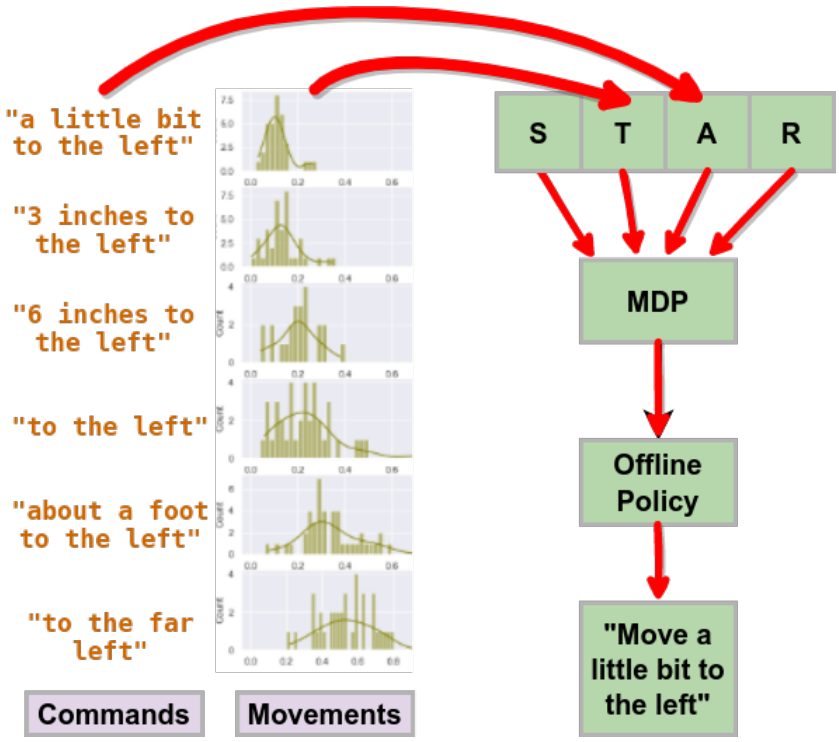


Figure 4.6: *(Left to right)* A sample of discrete commands. The movement (in meters) each command caused. MDP and solution definition. We train a model of human hand movement from demonstrations that inform the transition probabilities T. S defines the state space, A defines the discrete set of verbal actions, and R is the reward function. A policy is learned offline that can be used across reaching tasks.

the expectations for the general direction of the forthcoming instructions. We do so by computing the initial heading direction to the target. We project and discretize the heading direction to the target onto the plane of the shelf and use clock-based direction. The overview has the following template: *"I found the product at about { } o'clock direction"*.

2) Hand Movement Instructions: The plan overview is followed by guidance instructions. The instructions to convey to the user (actions) are computed online when using the continuous guidance mode and queried online from the policy learned in the discrete guidance mode. Based on the relative position of the target and the hand, a command is formulated (or retrieved from the policy) and issued aloud from a speaker that is part of the robotic cane system. In an effort to reduce frustration, the system issues new commands only when the user's hand has slowed down sufficiently to show that they are ready for the next command. The user is asked to grasp the

target object with their non-occupied hand if they are close to it with the system.

4.4 Pilot Study

We conducted an IRB-approved study with novice users (n=15) for system testing and validation. Participants were all sighted students who were blindfolded for the testing. At this stage of design and technical readiness, we follow prior work in performing preliminary validations using blindfolded users [35, 40, 90, 91, 106, 114] as a precursor to engaging with the PVI community.

4.4.1 Hypotheses

We test the following hypotheses in our comparison of the two planner modes.

- **H1**: Participants will retrieve products with a lesser number of commands in the discrete guidance mode compared to the continuous guidance mode.
- **H2**: Participants will retrieve the products in less time with the discrete guidance mode as compared to the continuous guidance mode, once the products are located.
- **H3**: Participants will retrieve the products with less net hand movement with the discrete guidance mode compared to the continuous guidance mode.

4.4.2 Experiment Design

We use the experimental setup shown in Figure 4.7 to test the system. While we test the system as a whole, we do so with a specific focus on the two proposed guidance planners - 1) Continuous and 2) Discrete. As a baseline, we also compare our system against a human caller (university research staff separate from the experimenters) guiding over a video call. The caller gave freeform verbal guidance with the aim of helping the participant reach the goal on the shelf. The users performed each of these 3 conditions (continuous, discrete, and human) 5 times. We had a set of 5 different products for each of the conditions and the same set of 5 products with the same spatial configuration was used for each condition. The products were picked from the top

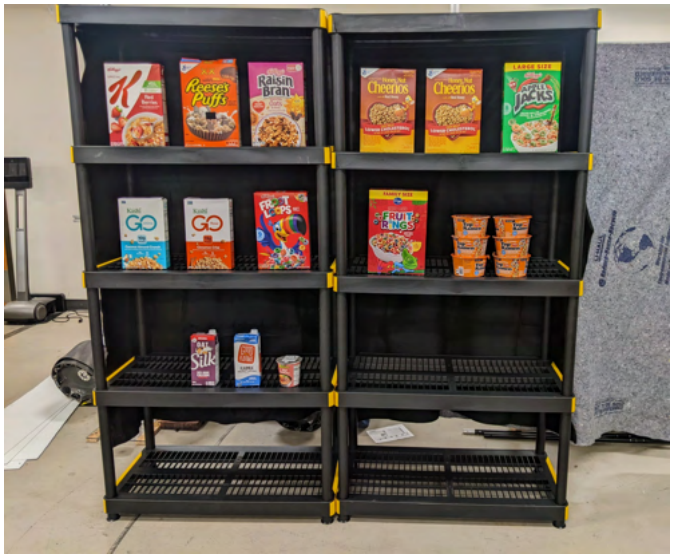


Figure 4.7: The experimental setup approximating a grocery store shelf, used for evaluating the efficacy of our manipulation guidance system.

two rows. We randomly shuffled the 3 conditions, counterbalancing as needed to achieve parity of orderings to avoid training/familiarity biases. We ensured that the users have the shelf in the frame at the starting pose and they were approximately 0.9m to 1.5 away from the target product in all the runs of the experiment. Each user was oriented on how to use the system, how to hold it, and how many runs they would perform. We asked them to keep their locomotion to a minimum and primarily move their hand. We also informed them about the alignment process and how to interpret the alignment guidance instructions that precede the actual guidance. The orientation process took about 3-4 minutes on average for each user.

4.4.3 Results

4.4.3.1 System Success Rate

Each participant retrieved 5 different products in each condition *(continuous, discrete, human)*. Perhaps unsurprisingly, the human condition was successful **75/75** times. We observed a few scenarios where participants picked up the wrong product initially but were corrected by the human caller. The product locator system was the same for both the *continuous* and the *discrete*

modes. The product detection failed **21/150** times in locating the desired product in the participant's first attempted scan of the shelf. This mostly happened when the desired product was not in the camera frame. If similar enough products are in frame (above threshold), the system selects the most visually similar product it can find. In some cases, when the desired target product is not facing straight or is occluded, the system might locate a visually similar product that may be incorrect (Figure 4.9). While inconvenient, these errors are ultimately correctable using existing work (e.g., barcode based methods). The system showed promise in dealing with overexposure and blur due to the data association as shown in the example (Figure 4.8). Both of our planners guided

Figure 4.8: A sample product detection result that was robust to blurring and overexposure with the help of *data association*.

the user to the desired product **150/150** (100%) times. In the continuous mode, the participant picked up the adjacent item **8/75** times, and in the discrete mode that happened **6/75** times. It is important to note that the guidance algorithm took the participant close to the product, but since the users used their non-dominant hand to pick up the product they picked up a product immediately adjacent to the desired one (which was usually just 1-2 inches away). We include these data in our subsequent analysis, as the high level guidance was still successful in minimizing the exhaustive search required in the haptic space of the participant.

4.4.3.2 Hypotheses Testing

Post-hoc comparisons using Tukey's HSD test revealed that participants retrieved the products with significantly fewer commands in the discrete mode as compared to the continuous mode, **confirming H1** (Figure 4.10). In fact, the number of commands was similar for the human caller

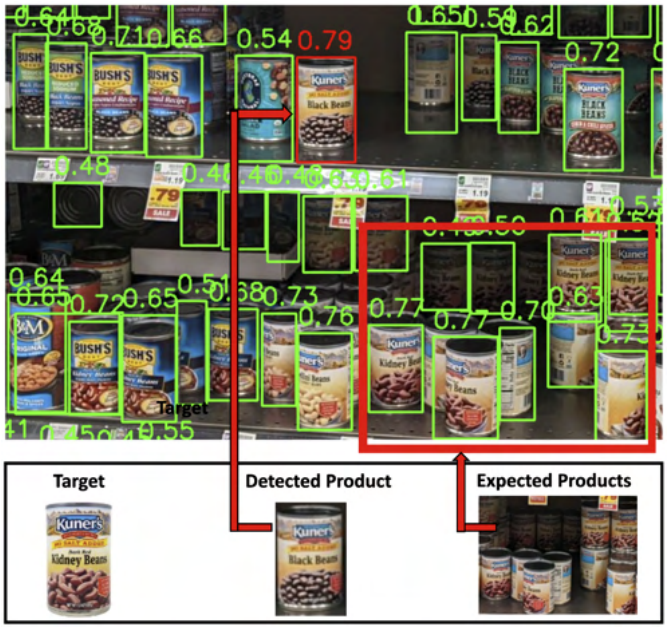


Figure 4.9: The product detection algorithm in a real grocery store. The system detects an incorrect product which is visually similar to the target. The expected products were either not facing straight or occluded partially by price tags hampering their visual similarity with the target.

and the discrete mode. Using the same test we found that the participants could retrieve the products significantly faster in the discrete mode compared to the continuous mode, thus **confirming H2** (Figure 4.11). The guidance time for the discrete was also similar to the human caller. We ran a TOST (two one-sided test) which **confirms distributional equivalence for the number of commands and guidance time between the discrete planner and human caller**. Although the same doesn't hold true if we consider the total time which includes: alignment, searching for the product, reporting the plan, and guiding. We can see (Figure 4.13) that both proposed modes incur some time in aligning and reporting the plan. We did not classify all of these activities for the human caller because they would interleave these instructions into their guidance, so we classify the time prior to providing actual movement instructions as 'search time'. We did not find any statistical difference between the net hand movement caused by either planner, thus we **could not confirm H3**. To our surprise, we see that the human caller had significantly higher net movement (Figure 4.12). We think that this is due to the fact that the human caller relied on tracking the

dominant hand of the participant in order to give them instructions with respect to the dominant hand position. The dominant hand would often go out of the cell phone camera frame and the human caller would have to issue instructions in order to gauge where the hand is causing these superfluous movements.

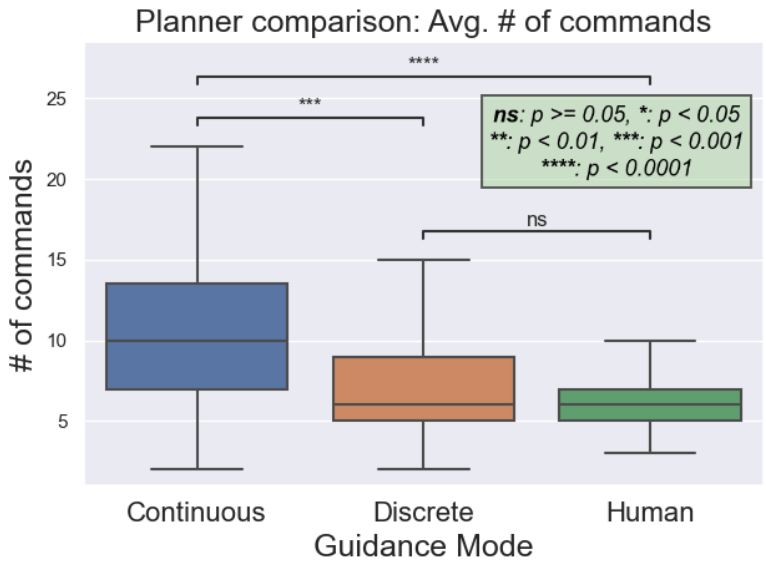


Figure 4.10: Average number of commands used by the different planners.

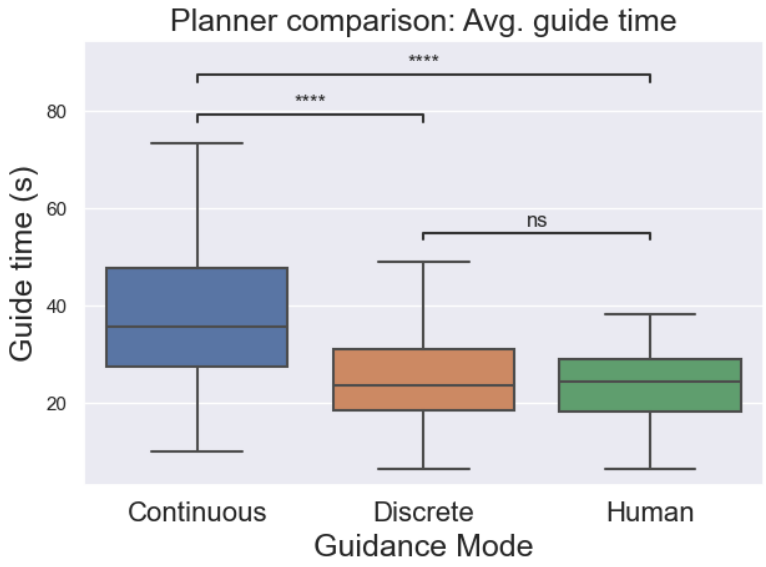


Figure 4.11: Average guide time used by the different planners.

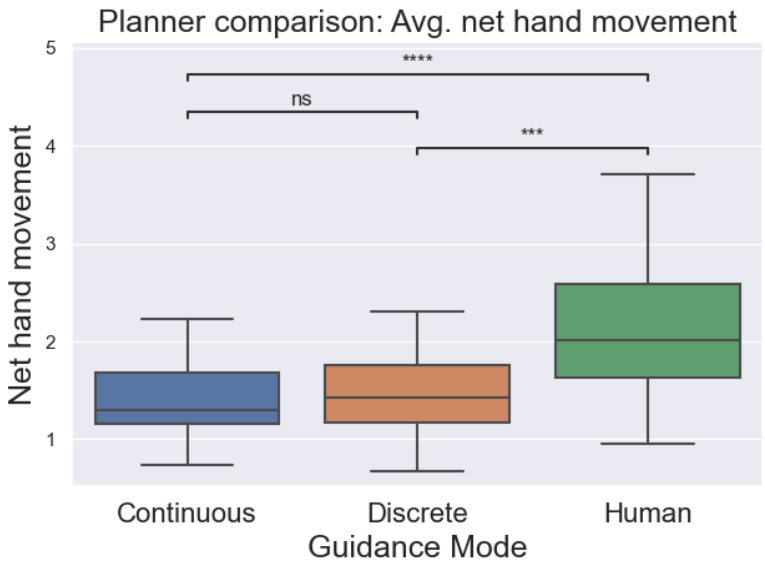


Figure 4.12: Average net hand movement caused by the different planners.

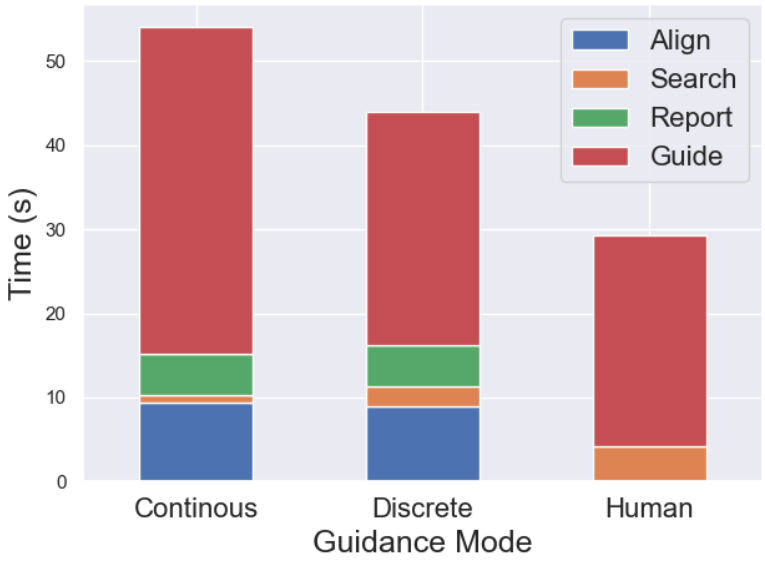


Figure 4.13: Average time taken by each planner. The alignment, search, and report durations are similar for Continuous and Discrete but the guidance time is lower for Discrete, comparable to human levels.

4.4.3.3 Subjective Evaluation

We administered a survey after each condition. Participants rated both of our planners high on metrics concerning: human-like, interactive, competent, and intelligent (Figure 4.14). The two planners and the human performance were not statistically different on the competence and intelligence metrics. Both of the planners were rated less humanlike than our human baseline. The discrete planner was rated slightly lower in the interactive metric which could be because the discrete planner did not provide any affirmations, which is an area of future investigation. The participants rated the plan overview's helpfulness as 4.2 (std 1.01) out of 5, indicating that this component was valuable to them. The large standard deviation is attributed to two participants rating this feature very low and in the exit interview they mentioned that it was hard and confusing for them to understand the overview template. We noticed participants mentioning a clear preference for one or the other planner. Some participants who liked the continuous mode mentioned that they liked that it provided some affirmation in the form of *"Stop"* command. We also noticed the human caller using some kind of affirmation along with the movement instructions, for example, *"Alright"*, *"Good"*, *"That one"*. Participants who liked the discrete planner more mentioned that they liked getting precise movement instructions and not relying on the system to stop them without much certainty about how much they would have to move. Participants also rated ShelfHelp favorably on metrics such as *ease of use, confidence, mental demand, temporal demand, frustration.*

4.5 Discussion

We caution the reader to exercise care to avoid drawing strong conclusions about device readiness based solely on pilot tests with blindfolded, non-PVI participants, as these are not necessarily a reliable proxy for the (eventual) target population of this device [35, 103], but we find the preliminary results promising and indicative of value in further investigation. With this pilot study, we have shown a proof-of-concept with regard to the hardware and software features of our proposed self-contained system (i.e., not dependent on external compute or data connection),

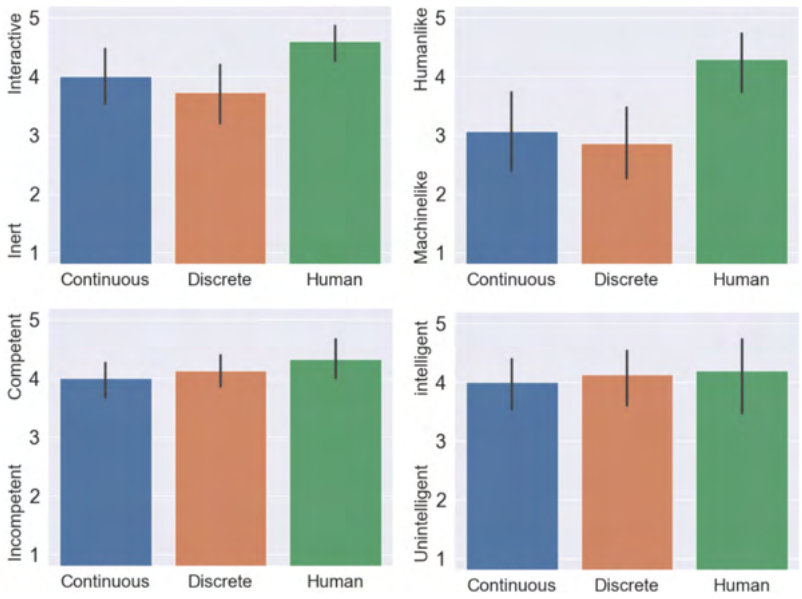


Figure 4.14: Average rating on subjective metrics: *(clockwise from upper-left) Interactive, Humanlike, Intelligent, Competent.* Both planners performed well on all the metrics compared to the human guide except for *Humanlike*.

alongside a validation of a novel product locator system coupled with a novel fine-grain manipulation guidance system, demonstrating that the system is able to locate a target product, plan to reach it, and effectively guide retrieval in real-time. The discrete planner performs quantitatively better than the continuous and is on par with the human in our study. The continuous planner elicited a positive response as well, in part due to the affirmations it provides by using the *"Stop"* command. It could be beneficial to add this feature to the discrete planner, effectively grounding the user during the execution of the plan. The product locator system relies on visual features which do not effectively leverage all of the available semantic information on product packaging, and a future direction would be real-time incorporation of semantic information as well.

4.6 Conclusion

This chapter presented "ShelfHelp," a system demonstrating how assistive robotics can provide **fine-grained grasp guidance** by integrating environmental context. Specifically, it combined a vision pipeline using object identity (**semantic context**) with an MDP-based planner informed

by human demonstrations (**human movement context**) to generate effective verbal guidance for product retrieval in cluttered settings. The pilot study validated that the system could successfully locate products and guide users, with the discrete, human-informed planner performing comparably to, and sometimes more efficiently than, direct human assistance in key metrics like guidance time and command count. The user feedback highlighted a desire for guidance with stronger **contextual grounding**.

While the fine-grained manipulation and targeted assistance demonstrated in this chapter highlight the immediate utility of context-aware assistive devices, these systems inherently rely on a critical assumption: that the agent already possesses a precise understanding of its global position within the environment. During the development and evaluation of these assistive frameworks in massive, highly aliased spaces like grocery stores, we encountered a fundamental limitation. Fine-grained assistance is only possible once the system has successfully navigated to the correct general vicinity. This motivates the work in the next chapter, "ShelfAware," which directly tackles the development of novel semantic localization techniques designed for these challenging, semantically-rich environments.

Chapter 5

ShelfAware: Real-Time Visual-Inertial Semantic Localization in Quasi-Static Environments with Low-Cost Sensors

Publication note. This chapter is adapted from the accepted IEEE Robotics and Automation Letters paper **ShelfAware: Real-Time Visual-Inertial Semantic Localization in Quasi-Static Environments with Low-Cost Sensors** [5].

5.1 Introduction

The preceding chapters established that interpreting social and semantic contexts is crucial for effective human-robot interaction and assistive guidance. However, as identified at the conclusion of Chapter 4, deploying such assistive technologies in the wild is bottlenecked by the system's ability to localize itself within complex, repetitive, and geometrically aliased environments. Transitioning to autonomous navigation requires rethinking how spatial knowledge is represented and parsed.

This chapter introduces an enabling technology designed to solve this localization bottleneck: a probabilistic semantic localization framework. Rather than relying on external beacons or static geometric landmarks, this approach utilizes the quasi-static distribution of semantic objects, such as product arrangements on store shelves. To ensure these systems remain accessible and deployable without reliance on heavy computational infrastructure, this framework is designed and validated to run efficiently on low-compute, laptop-class platforms. By treating semantics as probabilistic distributions, we provide the foundational spatial awareness necessary for both uninstrumented assistive devices and general autonomous robotic navigation stacks.

5.2 Domain Motivation

Many real-world indoor environments, such as retail stores and warehouses, are **quasi-static**: their global geometric layout (e.g., walls, shelving units) remains permanent over long horizons, but their local semantic contents change continually. In these settings, traditional geometry-based particle filters like Adaptive Monte Carlo Localization (AMCL) [38] suffer from severe perceptual aliasing. The combination of visually repetitive aisle geometry, dynamic clutter, and the noisy observations typical of low-cost wearable or mobile sensors routinely violates static map assumptions and causes rapid particle impoverishment [43]. Consequently, robust, start-anytime global localization in these GPS-denied environments remains a critical open problem for autonomous and assistive systems.

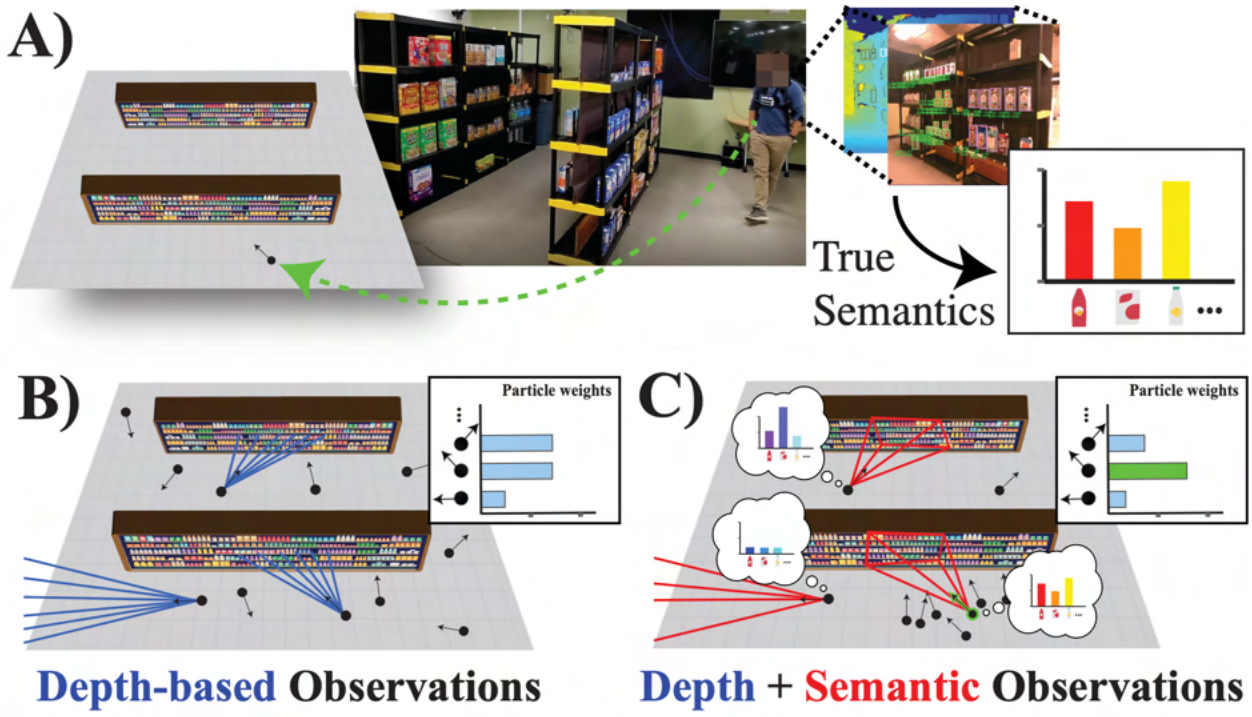


Figure 5.1: An overview of ShelfAware. A) A retail-like environment. B) Depth-based observation models in particle filtering rely solely on geometric features, which are ambiguous in long, repetitive aisles and lead to weak particle discrimination. C) ShelfAware injects particles based on semantic cues, enabling more distinctive and robust particle weighting combined with the depth observation model and improved global localization in retail-like environments.

To address these challenges, we present ShelfAware (Fig. 5.1), a semantic particle filter

tailored for robust global localization in quasi-static environments. Unlike prior approaches that treat semantics as fixed, discrete landmarks [**?**, 127, 140, 141], ShelfAware models semantic features as probabilistic distributions over object counts and spatial arrangements. This representation captures the intrinsic variability of real-world environments, preserving the statistical structure necessary for stable observation-driven localization even as local object configurations continuously fluctuate.

We evaluate ShelfAware across two domains: a semantically dense mock retail environment enabling rigorous motion-capture ground truth, and a 3,500 sq. ft. operational grocery store demonstrating open-world scalability. Experiments on consumer-grade wearable and cart-mounted sensors highlight the system's robustness for practical deployment across both autonomous service robots and assistive devices.

Our core technical contributions are twofold:

- A semantic mapping and observation design that encodes object collections as statistical distributions over counts and arrangement, providing inherent robustness to semantic perturbations and flux.
- A real-time, inverse observation model-based particle filter that leverages this representation on low-cost, vision-based hardware to efficiently propose and evaluate high-quality global pose hypotheses.

5.3 Related Work

5.3.1 Vision and Semantics-Based Localization in Quasi-Static Environments

Particle filter-based localization methods such as Monte Carlo Localization (MCL) and its adaptive variant AMCL [38] remain standard for onboard robot localization due to their scalability and integration with modern navigation frameworks [1]. However, their reliance on static geometric maps and feature-rich depth data makes them brittle in quasi-static or dynamic environments where geometry is repetitive or transient. This limitation has driven research toward semantic

localization, where environmental understanding extends beyond geometry to include object-level cues and contextual information [131].

Earlier work in semantic localization augments geometric maps with explicit, object- or region-level labels that serve as long-term landmarks for place recognition and drift reduction [**?**,127,140]. Text-based cues, such as signage or packaging, have also proven effective as distinctive features in structured indoor spaces [141]. More recent efforts adopt implicit semantic representations, encoding spatial and semantic structure jointly in learned neural fields [66], enabling continuous observation models compatible with particle filtering. Yet, both explicit and implicit methods often assume stable semantic identities, assumptions that rarely hold in quasi-static domains like warehouses or retail stores, where local semantics fluctuate even when global geometry remains constant.

Despite these advances, few approaches explicitly address semantic volatility: the gradual yet continual change in object distributions and arrangements that characterizes quasi-static environments. Traditional semantic-SLAM systems [**?**, 31] and semantic visual positioning frameworks [24, 98] demonstrate the promise of integrating semantics into localization, but their models typically treat detected landmarks as fixed or sparsely varying entities. This mismatch between model assumptions and real-world semantic drift leads to degraded performance in settings with restocking, occlusions, or partial observability. Recent deep visual localization methods [34, 75] demonstrate impressive accuracy but require server-grade compute, precluding real-time use on wearable edge devices. Furthermore, while modern visual SLAM [10] excels at local tracking, it typically requires extensive exploration to achieve global localization against a prior map, limiting the instantaneous recovery needed for on-demand assistance.

5.3.2 Applications in Assistive and Human-Centered Robotics

Quasi-static indoor environments such as retail stores also appear prominently in assistive and human-centered robotics. Many assistive navigation systems for people with visual impairments depend on reliable global localization to support guidance, object retrieval, or wayfinding [19,67,77].

However, prior systems often sidestep this challenge, relying instead on environmental instrumentation (e.g., RFID tags [77], Bluetooth beacons) or assuming that the user is already localized within a specific aisle or region [7]. Recent conversational and multimodal assistance frameworks [54, 57] have improved interaction but still rely on external positioning aids. ShelfAware complements this body of work by targeting the underlying localization problem directly, enabling global localization without external infrastructure, and doing so in a semantically dynamic environment representative of those faced by both assistive and service robots.

In contrast to prior methods that treat semantics as static landmarks, ShelfAware models them as statistical distributions over object counts and arrangements, enabling localization that is both robust to semantic flux and compatible with low-cost visual sensing. This probabilistic treatment of semantics situates ShelfAware within the broader context of semantic particle filtering, extending its applicability beyond assistive scenarios to general quasi-static domains.

5.4 Method

The objective of our proposed method is to achieve robust global localization in quasi-static indoor environments. These environments, which include warehouses, retail spaces, and laboratories, challenge conventional geometric localization methods due to visually repetitive structures, dynamic occlusions, and semantic drift [43]. ShelfAware addresses these challenges by fusing geometric and semantic cues within a probabilistic particle filtering framework tailored for vision-based sensing, and deployable on wearable or compact platforms that preclude LiDAR or wheel odometry.

5.4.1 Semantic Particle Filter Overview

ShelfAware builds on the MCL framework [38], augmenting standard depth likelihoods with a probabilistic semantic observation model that remains informative under semantic variability. Each particle represents a pose hypothesis $\mathbf{x}_t^{(i)}$ with weight $\mathbf{w}_t^{(i)}$, updated via motion, geometric (depth), and semantic observations (Fig. 5.4).

Unlike approaches that treat detected objects as fixed landmarks, ShelfAware models the

semantic state of the environment as distributions over class counts and coarse spatial arrangement. At runtime, the system forms a compact semantic observation vector from the live RGB-D frame and compares it to expected semantic signatures derived from a hybrid map (Sec. 5.4.2). The resulting semantic similarity acts both as (i) a **forward** observation likelihood for weighting particles and (ii) a query metric for an **inverse** semantic model that proposes high-quality pose hypotheses when global localization or recovery is needed (Sec. 5.4.5).

The joint observation model factors geometric and semantic likelihoods as such:

$$p(\mathbf{z}_t \mid \mathbf{x}_t, m) = \eta \; p_d(\mathbf{z}_t^d \mid \mathbf{x}_t, m_d) \; p_s(\mathbf{z}_t^s \mid \mathbf{x}_t, m_s), \tag{5.1}$$

where η is a normalizer, m_d and m_s are the geometric and semantic maps respectively, $\mathbf{z}_t = (\mathbf{z}_t^d, \mathbf{z}_t^s)$ are the depth and semantic observations (Secs. 5.4.3, 5.4.4). This factorization assumes conditional independence: given pose $\mathbf{x}_t$, depth $\mathbf{z}_t^d$ and semantic $\mathbf{z}_t^s$ measurements have independent noise.

5.4.2 Semantic Mapping

ShelfAware maintains a hybrid map combining geometric structure with semantic information. First, we construct a standard 2D occupancy grid map of traversable space using GMapping [48], with 10×10 cm resolution. This resolution supports precise ray casting for the depth observation model within the MCL framework and aligns with the accuracy needed for potential downstream manipulation tasks [7].

Second, we build a semantic map overlaid on the occupancy grid. The semantic map discretizes the volume into 20×20 cm cells in (x, y) and 30 cm in z to balance computational efficiency with the physical scale of objects and the environment. Each voxel stores a running distribution over observed object **classes** and their **counts**, yielding a coarse, distributional representation that captures typical arrangements without assuming fixed landmark identities.

To demonstrate that our system can work with different vision pipelines, we populate this semantic map using two distinct pipelines, depending on the evaluation environment:

Pipeline A: Supervised Closed-Set Semantics For rigorous quantitative benchmark-

ing in a controlled mock environment with fixed and known number of classes, we utilize a YOLOv9 detector fine-tuned on the SKU-110K dataset [45] to extract generic "shelf item" bounding boxes. A ResNet50-based classifier then assigns these detections to one of $C = 14$ manually defined application-level classes.

Pipeline B: Self-Supervised Open-Set Semantics To demonstrate zero-shot scalability for real-world deployment in a 3,500 sq. ft. operational grocery store, we eliminate manual annotation entirely. Using the same class-agnostic detector, we extract a 768-dimensional vector for each bounding box crop via a pre-trained DINOv3 Vision Transformer. We then cluster these features using K-Means, automatically defining $C = 60$ (obtained via grid-search) semantic categories (Fig. 5.2:right).

3D Projection and Map Integration Regardless of the chosen perception pipeline, detected objects are projected into the map frame using the RGB-D depth channel and camera calibration.

Given pixel coordinates $\tilde{\boldsymbol{u}} = [u, v, 1]^\top$ at the bounding box's **median** depth Z_c, the 3D world position is $\boldsymbol{X}_w = \boldsymbol{t}_{wc} + Z_c \boldsymbol{R}_{wc} \boldsymbol{K}^{-1} \tilde{\boldsymbol{u}}$, where $\boldsymbol{K}$ is the camera intrinsic matrix, and $\boldsymbol{R}_{wc}, \boldsymbol{t}_{wc}$ denote the camera-to-world transform.

To mitigate depth noise and vision errors, we refine each estimated product position along the camera ray using Bresenham's line algorithm [64] until it aligns with the nearest occupied occupancy map cell, preventing minor misalignments. Figure 5.2 visualizes the resulting semantic layer overlaid on the occupancy grid, for clarity, the plot shows only the dominant class per cell, while the map stores full per-class count distributions.

5.4.3 Semantic Observation Vector and Semantic Similarity

ShelfAware generates a semantic observation $\mathbf{z}_t^s$ from the live camera view (Fig. 5.3). This observation concatenates three sub-vectors: (i) a class-count vector $\mathbf{v}_c$ (what is present), (ii) a mean range vector $\mathbf{v}_d$ (how far), and (iii) a mean bearing vector $\mathbf{v}_\theta$ (in which direction), each aggregated over detections for the classes visible at time t.

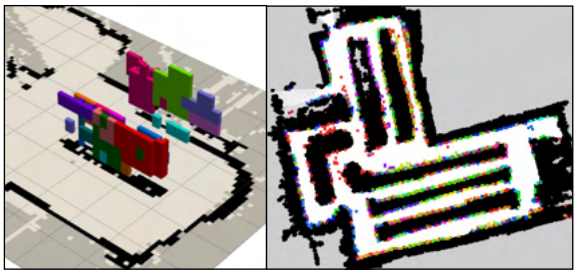


Figure 5.2: The semantic map overlaid on the occupancy grid. Each voxel stores a distribution over object class counts. Ray casting on this semantic layer yields the expected semantic vector $\mathbf{v}_{\text{sem}}$ comprising class counts, distances, and angles. *Left:* Mock store, *Right:* Real Store.

To compare live and expected semantic observations, we define a composite similarity

$$S(\mathbf{z}^s, \hat{\mathbf{z}}^s) = \alpha S_{\text{counts}} + \beta S_{\text{distance}} + \gamma S_{\text{angle}}, \tag{5.2}$$

with $\alpha, \beta, \gamma \geq 0$ and $\alpha + \beta + \gamma = 1$. This score is used both as (i) a forward semantic likelihood $p_s(\mathbf{z}_t^s \mid \mathbf{x}_t, m_s)$ in (5.1) and (ii) a query metric for the inverse model (Sec. 5.4.5).

Counts. We normalize $\mathbf{v}_c$ to form a categorical distribution over observed classes and compare it with the expected distribution via Jensen-Shannon divergence (JSD). Defining P and Q as the normalized count distributions and $M = \frac{1}{2}(P + Q)$, we set $S_{\text{counts}} = 1 - \text{JSD}$ with $\text{JSD}(P \parallel Q) = \sqrt{\frac{1}{2}\sum_{i=1}^{C}\left(P(i)\log\frac{P(i)}{M(i)} + Q(i)\log\frac{Q(i)}{M(i)}\right)}$. This choice is symmetric, bounded, and robust to sparsity in per-frame detections. By comparing normalized distributions rather than absolute counts, this JSD formulation provides inherent robustness to perception failures. Transient false negatives (e.g., missed bounding boxes due to motion blur) or false positives do not catastrophically invalidate the observation, but rather smoothly degrade the similarity score, preventing abrupt particle depletion.

Distances and Angles. The spatial components $\mathbf{v}_d$ and $\mathbf{v}_\theta$ are compared using L2 distances. For unbounded ranges, we map the L2 error d_{distance} to $S_{\text{distance}} = 1/(1 + d_{\text{distance}})$. For angles, which lie within the camera field-of-view (FOV), we use $S_{\text{angle}} = 1 - (d_{\text{angle}}/\text{FOV})$.

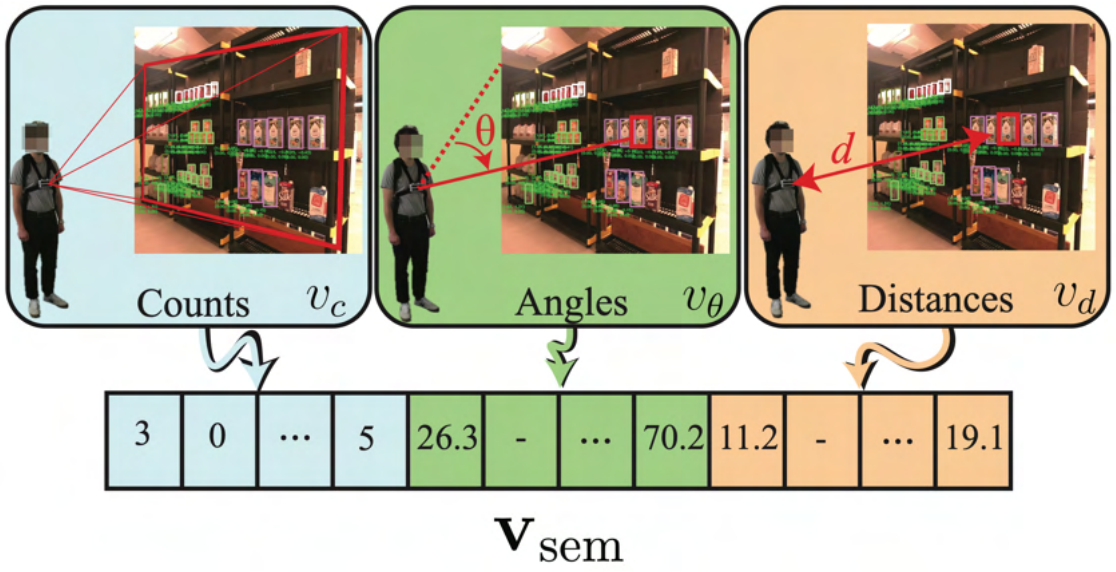


Figure 5.3: Semantic vector $\mathbf{v}_{\text{sem}} = [\mathbf{v}_c, \mathbf{v}_\theta, \mathbf{v}_d]$. The count vector $\mathbf{v}_c$ captures the number of items detected in each class, $\mathbf{v}_\theta$ and $\mathbf{v}_d$ capture mean relative bearings and ranges for those.

5.4.4 Depth Observation Model

To incorporate geometric information, we synthesize a 2D laser scan from a central horizontal band of the depth image. The depth observation at time t is $\mathbf{z}_t^d = \{z_t^{d,(k)}\}_{k=1}^K$. Given the expected range $\hat{z}^{(k)}(\mathbf{x}_t, m_d)$ for the k-th beam (via occupancy-map ray casting), we use the beam end-point mixture [38]:

$$p_{\text{hit}}(z_t^{d,(k)} \mid \mathbf{x}_t, m_d) = \mathcal{N}\big(z_t^{d,(k)},\, \hat{z}^{(k)}(\mathbf{x}_t, m_d), \sigma_{\text{hit}}^2\big), \tag{5.3}$$

$$p_{\text{short}}(z_t^{d,(k)} \mid \mathbf{x}_t, m_d) = \lambda e^{-\lambda z_t^{d,(k)}}\, \mathbf{1}_{[0,\hat{z}^{(k)}(\mathbf{x}_t, m_d)]}(z_t^{d,(k)}), \tag{5.4}$$

$$p_{\text{max}}(z_t^{d,(k)}) = \delta(z_t^{d,(k)} - z_{\text{max}}), \quad p_{\text{rand}}(z_t^{d,(k)}) = 1/z_{\text{max}}, \tag{5.5}$$

$$p_d(z_t^{d,(k)} \mid \mathbf{x}_t, m_d) = w_{\text{h}}\, p_{\text{hit}} + w_{\text{s}}\, p_{\text{short}} + w_{\text{m}}\, p_{\text{max}} + w_{\text{r}}\, p_{\text{rand}}, \tag{5.6}$$

where m_d represents the 2D occupancy map, $z_{\text{max}} = 6.0$ m is the maximum reliable sensor range, $\sigma_{\text{hit}} = 2.0$ is the standard deviation of the expected depth, and $\lambda = 0.1$ is the intrinsic decay parameter for short readings. The empirically tuned mixture weights ($w_{\text{h}} = 0.85$, $w_{\text{s}} = 0.05$, $w_{\text{m}} = 0.05$, $w_{\text{r}} = 0.05$) define the relative probability of a valid hit, a short reading, a max-range reading, and a random noise reading, respectively, and are constrained to sum to one. Assuming

conditional independence across beams, the joint likelihood for $\mathbf{z}_t^d$ is

$$p_d(\mathbf{z}_t^d \mid \mathbf{x}_t, m_d) = \prod_{k=1}^{K} p_d\big(z_t^{d,(k)} \mid \mathbf{x}_t, m_d\big). \tag{5.7}$$

We compute $\hat{z}^{(k)}(\mathbf{x}_t, m_d)$ efficiently using CDDT-accelerated ray casting [116], which runs on the CPU and leaves GPU resources available for the semantic perception pipeline.

5.4.5 Localization with an Inverse Semantic Model

ShelfAware integrates the depth and semantic models within an MCL filter, maintaining weighted particles $\{(\boldsymbol{x}_t^{(i)}, \boldsymbol{w}_t^{(i)})\}_{i=1}^{N}$ over the pose belief. The method's key innovation is an **inverse semantic model** that proposes high-quality pose hypotheses directly from live semantic observations, enabling fast **global localization** and recovery from tracking failures without special-case handling.

Offline Pre-computation We precompute expected semantic observations $\hat{\boldsymbol{z}}^s(\boldsymbol{x}, m_s)$ for discretized poses $\boldsymbol{x} = (x, y, \theta)$ over the free space (10cm cells, 36 orientation bins). For each pose, we ray cast the 3D semantic map (Sec. 5.4.2) to obtain the expected semantic vector. All vectors are cached in a hashmap (76 MB for our store environment). We also build a reverse index `class_to_poses` (2.1 MB) mapping each product class to the set of poses from which that class is visible, enabling efficient candidate pruning at runtime.

Online Localization Loop. Algorithm 3 summarizes the online filter. Particles are propagated using VIO-based motion. Each iteration forms a live semantic vector $\boldsymbol{z}_t^s$ and estimates an expected semantic view $\hat{\boldsymbol{z}}_t^s$ at the current pose estimate $\hat{\boldsymbol{x}}$. We query the inverse model when the semantic similarity falls below the threshold and the count subvector contains sufficient mass: $S(\boldsymbol{z}_t^s, \hat{\boldsymbol{z}}_t^s) < \tau_{\text{sim}}$ and $\|\boldsymbol{z}_t^{s,\text{count}}\|_1 > \tau_\kappa$. In this case, we (i) use `class_to_poses` to form a candidate set as the union of poses associated with the observed classes, (ii) score candidates by S (Eq. 5.2), (iii) inject particles at the top-k poses, and (iv) reweight the entire set with both depth (Sec. 5.4.4) and forward semantic likelihoods (Eq. 5.1). If the semantic similarity is acceptable, or if the observation contains insufficient semantic mass, we update using only the depth likelihood for

efficiency. This procedure enables rapid convergence from unknown initial conditions and robust recovery from drift.

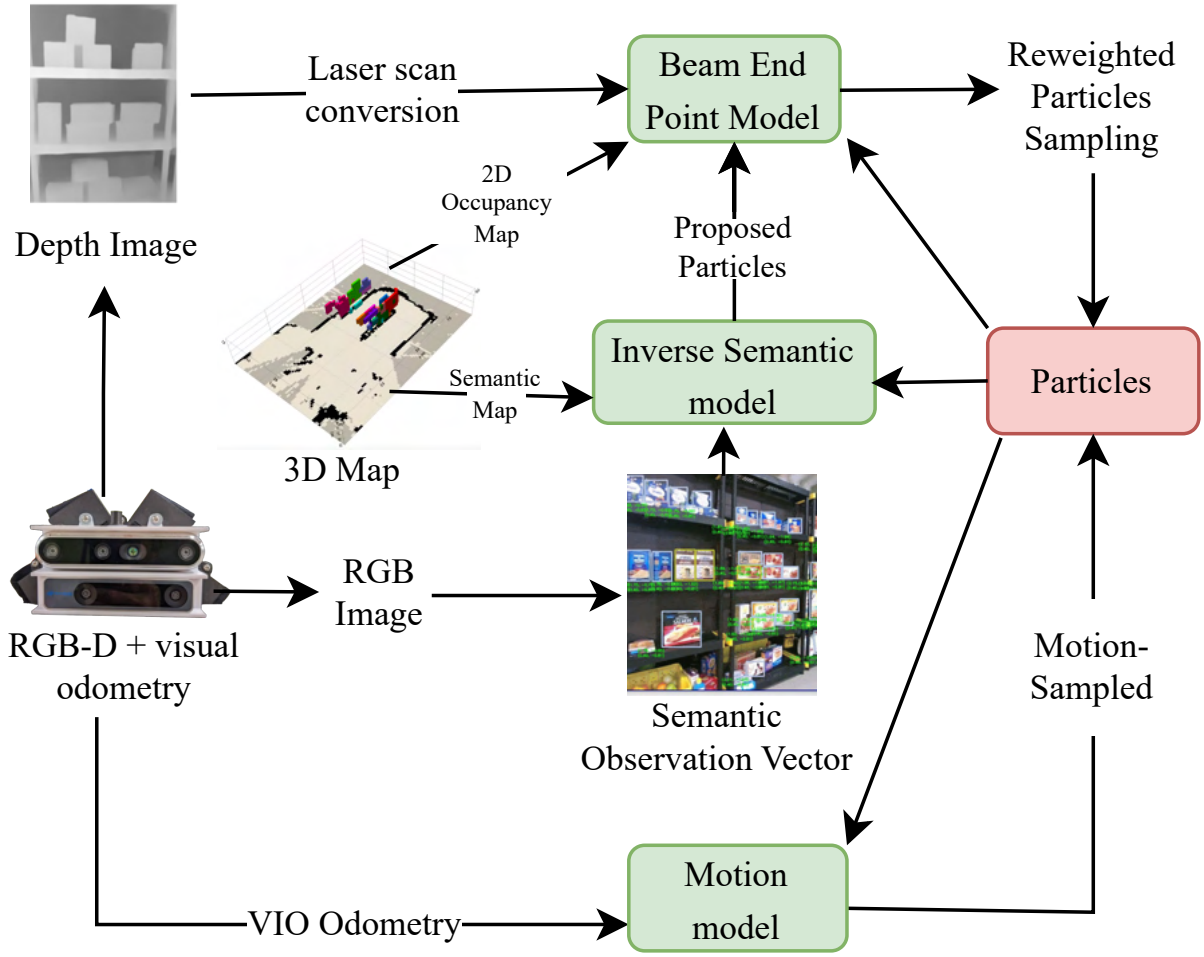


Figure 5.4: Data-flow diagram for ShelfAware. The semantic particle filter fuses depth likelihood with semantic likelihood and uses an inverse semantic model to propose high-quality particles for global localization and recovery.

Algorithm 3: Online Localization with ShelfAware

Input: Sensor data: image i_t, depth d_t, odometry o_t, precomputed semantic vectors $\hat{\mathbf{z}}^s(\mathbf{x})$

Output: Updated particle set P_{t+1}

Initialize particle set P_t uniformly over free space

$W_t \leftarrow (1/N, \ldots, 1/N)$

while true do

  $P_t \leftarrow \text{MotionModel}(P_t, o_t)$

  $z_t^s \leftarrow \text{GenerateSemanticVector}(i_t, d_t)$

  $\hat{x} \leftarrow \text{ExpectedPose}(P_t, W_t)$

  $\hat{z}_t^s \leftarrow \text{CalculateExpectedSemanticObs}(\hat{x})$

  $\text{sim} \leftarrow \text{CalculateSemanticSimilarity}(z_t^s, \hat{z}_t^s)$

  if sim $< \tau_{\textbf{sim}}$ **and** $\|z_t^{s,\textbf{count}}\|_1 > \tau_\kappa$ **then**

    $C \leftarrow \text{GetObservedClasses}(\mathbf{z}_t^s)$

    $\text{Candidates} \leftarrow \bigcup_{c \in C} \texttt{class\_to\_poses}[c]$

    $\text{Scores} \leftarrow [S(\mathbf{z}_t^s, \hat{\mathbf{z}}^s(\mathbf{x})) \text{ for } \mathbf{x} \in \text{Candidates}]$

    $\text{TopPoses} \leftarrow \text{GetTopK}(\text{Candidates}, \text{Scores}, k)$

    $P_t \leftarrow \text{InjectParticles}(P_t, \text{TopPoses})$

    $W_{\text{depth}} \leftarrow \text{DepthObservationModel}(P_t, d_t)$

    $W_{\text{sem}} \leftarrow \text{SemanticObservationModel}(P_t, i_t, d_t)$

    $W_t \leftarrow W_{\text{depth}} \odot W_{\text{sem}}$ `// element-wise product`

  else

    $W_t \leftarrow \text{DepthObservationModel}(P_t, d_t)$

  $W_t \leftarrow \text{NormalizeWeights}(W_t)$

  $P_t \leftarrow \text{Resample}(P_t)$

  $W_t \leftarrow (1/N, \ldots, 1/N)$

  $P_{t+1} \leftarrow P_t$

5.4.6 Hardware and Form Factors

Figure 5.5: ShelfAware hardware. A lightweight two-camera system with a 3D-printed mount was used throughout our experiments (top). This design allowed evaluation across a wearable chest mount (left) and a cart-mounted setup (right).

To meet the constraints of low-profile, wearable, or cart-mounted platforms, ShelfAware uses compact vision sensors: (i) an Intel RealSense D455 RGB-D camera (∼103 g) for color and depth, and (ii) an Intel RealSense T265 VIO camera (∼60 g) for odometry. Both are housed in a single 3D-printed mount and connected via USB to a Dell G15 laptop. This configuration supports two form factors: a chest-mounted wearable (laptop in a backpack) and a cart-mounted setup (Fig. 5.5). As discussed in Secs. 5.4.2–5.4.5, the system exploits CPU-based CDDT ray casting [116] to reserve GPU resources for real-time semantic perception, enabling practical deployment without LiDAR or wheel encoders.

5.5 Experimental Evaluation

We evaluate ShelfAware's ability to perform reliable, real-time global localization in quasi-static, semantically dense, and GPS-denied environments using compact vision sensors. Specifically, we assess its robustness to dynamic occlusions and sparse semantics, and its form factor sensitivity across wearable and cart-mounted configurations. We adopt these form factors to reflect practical compute constraints and to stress-test the method under depth-camera noise (lower scan frequency/point density than LiDAR) [74], intentionally avoiding the usability barriers of handheld devices [62] and the social stigma of head-mounted systems [71]. These evaluation goals reflect the start-anytime and recover-anytime operation demanded by practical deployments and shared-control use cases [15, 55, 135].

5.5.1 Experimental Setup

We conduct our controlled evaluation in a mock grocery store stocked with 150 products grouped into 14 object categories across nine shelves and three aisles. All experiments ran on a Dell G15 laptop (Intel Core i7-11800H (2.3 GHz), NVIDIA RTX 3060, 6GB VRAM, 32GB RAM) using the same vision-only sensor suite described in Sec. 5.4.6: an Intel RealSense D455 RGB-D camera and a RealSense T265 VIO camera (Fig. 5.5). Ground-truth 2D poses were obtained using an OptiTrack motion-capture system. We aligned the OptiTrack frame to the map frame offline by time-synchronizing mapping trajectories and solving for the rigid transform via RANSAC [33], enabling direct comparison of estimated and true poses. RGB-D streams were recorded at 30Hz and VIO at 200Hz (from the T265 IMU). This evaluation setting stresses depth-camera uncertainties that degrade purely geometric localization methods [74].

We considered four conditions to measure and characterize the robustness of the proposed method:

(1) **Cart:** Cameras mounted on a cart (Fig.5.5-right) [86].

(2) **Wearable:** Cameras chest-mounted, laptop carried in a backpack (Fig.5.5-left), stressing

odometry noise introduced by gait.

(3) **Dynamic Obstacles:** A person intermittently walked in front of the cameras to occlude vision.

(4) **Sparse Semantics:** To mimic depleted stock, we randomly removed 25% and 50% of products from shelves, reducing category signal.

We collected 5 trajectory sequences per condition (20 total, S1–S20), each ~ 40 s in duration. Between mapping and localization runs, we perturbed 20% of shelf contents to induce semantic drift. Unless stated otherwise, particles were initialized **uniformly over free space** to require true global localization [67, 86].

5.5.2 Perception Pipeline Configurations

We utilize the two perception pipelines introduced in Sec. 5.4.2. For the mock store (Pipeline A), a YOLOv9 detector (fine-tuned on the SKU-110K dataset [45]) proposes generic bounding boxes, which a ResNet50 classifier maps into 14 object categories (trained on 5,699 environment-specific frames). Grouping thousands of underlying SKU types into a small, fixed category set provides robustness against local inventory churn. Conversely, for the real-world operational store (Pipeline B), we combine YOLOv9's class-agnostic bounding box proposals with a zero-shot DINOv3 Vision Transformer and K-Means clustering. This open-vocabulary approach automatically extracts 60 semantic categories from the generic detections in a purely self-supervised manner, requiring zero environment-specific training or manual annotation to adapt to new inventory.

5.5.3 Baselines and Metrics

Baselines. We compare against three methods: (i) the standard ROS implementation of AMCL [1], (ii) an ablated ShelfAware variant that removes the semantic likelihood, yielding a pure geometric MCL baseline [38], and (iii) a Fixed Quantity Landmark (FQL) baseline. The FQL baseline represents standard explicit semantic localization approaches [140, 141] by treating

detected object classes as having fixed, deterministic counts. For fairness, all methods used **1,500 particles**, identical VIO odometry and RGB-D stream.

Metrics. Following established conventions in the literature, we evaluate ShelfAware via:

- **Global localization success.** A trial is a success if localization convergence occurs within the first 95% of the trajectory and remains converged until the end.
- **Convergence time (s).** Time from start to the beginning of the final convergence for successful trials.
- **Tracking ATE.** Absolute Trajectory Error (translation/rotation RMSE) computed after final convergence until end-of-sequence.

Convergence criterion. Anthropometric studies indicate a standing reach of ∼0.7–0.9 m [102], thus, localization errors below 0.7 m ensure targets remain within the reachable workspace. We declare convergence when the estimated pose is within 0.7 m translation and $\pi/4$ rad rotation of ground truth and stays within that threshold. We tuned semantic similarity weights by grid search and report results for $\alpha = 0.4, \beta = 0.4, \gamma = 0.2$ (Sec.5.4.3).

Table 5.1: Global localization results across setups in the mock store environment. Success is % of 25 trials per setup. The convergence time and RMSE is calculated only for successful convergences.

Method	Cart			Wearable			Dynamic			Degraded/Sparse			**Overall G%**
	Success	Time (s)	RMSE (m/rad)	Success	Time (s)	RMSE (m/rad)	Success	Time (s)	RMSE (m/rad)	Success	Time (s)	RMSE (m/rad)	
MCL	16%	1.06	0.41/0.27	12%	0.13	0.37/0.49	16%	0.13	0.25/0.04	16%	0.05	0.36/0.26	15%
AMCL	20%	17.41	**0.18**/**0.02**	0%	-	-	0%	-	-	20%	3.36	0.34/**0.05**	10%
FQL	96%	1.98	0.38/0.20	96%	4.14	**0.42**/**0.30**	80%	2.75	**0.36**/0.22	88%	4.07	0.42/**0.23**	90%
ShelfAware	**100%**	**0.27**	0.37/0.25	**100%**	**1.96**	0.36/0.42	**100%**	**0.29**	0.39/**0.15**	**88%**	**1.83**	**0.31**/0.34	**97%**

Table 5.2: Tracking Success Rate (T%) in the mock store.

Method	**Cart**	**Wearable**	**Dynamic**	**Sparse**	**All**	**T-RMSE**
MCL	8%	8%	12%	12%	10%	0.27/0.13
AMCL	17%	0%	0%	20%	10%	**0.25**/0.14
FQL	**60%**	60%	68%	48%	59%	0.32/0.10
ShelfAware	**60%**	**72%**	**80%**	**52%**	**66%**	0.29/**0.09**

5.5.4 Results: Global Localization

Table 5.1 (four conditions, 25 trials each) summarizes success, convergence time, and RMSE. ShelfAware achieved a **97%** overall success rate across 100 trials, versus **15%** for MCL and **10%** for AMCL. The mean time-to-convergence across all setups was **1.07s**. ShelfAware obtained the lowest translational RMSE in two out of four conditions and the highest success percentage in all four conditions.

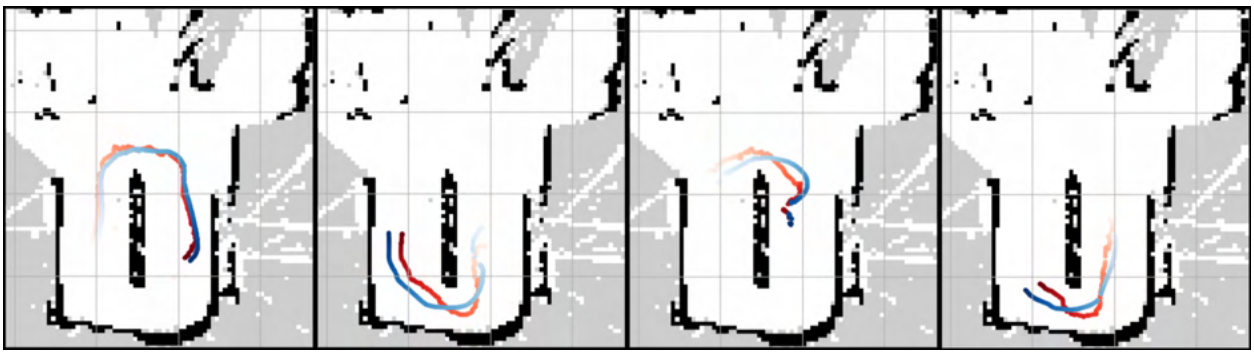


Figure 5.6: Examples of ground truth pose in blue and our estimated pose in red. Lighter to darker denotes the temporal progression of the trajectories. Each grid cell is 2m×2m.

5.5.5 Results: Tracking and Robustness

To evaluate continuous tracking performance, we measure the Tracking Success Rate (T%), defined as the percentage of trials that converge and remain strictly within the convergence thresholds until the end. Table 5.2 reports the T% and the average tracking RMSE (T-RMSE) across all conditions.

ShelfAware achieved an overall tracking success rate of **66%**, outperforming the deterministic FQL baseline (59%) and vastly exceeding the purely geometric MCL and AMCL methods (10%). ShelfAware was particularly robust to dynamic occlusion, achieving 100% global-localization success and 80% tracking success. Sparse semantics remained the most difficult condition: global success fell to 88%, tying FQL, while tracking success fell to 52%, compared with 48% for FQL. Thus, the distributional model retained a modest advantage under semantic depletion but did not eliminate

the resulting performance degradation. Qualitative trajectories (Fig.5.6) show rapid correction in repetitive aisles and robustness to occlusion bursts. These results mirror the global-localization advantage, demonstrating that probabilistic semantic tie-breaking maintains spatial consistency despite active dynamic obstacles and severe stock depletion.

The full pipeline ran at 9.6Hz on a standard consumer laptop with a mid-tier GPU, achieving real-time throughput without LIDAR or wheel encoders.

5.5.6 Real-World Open-Vocabulary Evaluation

To demonstrate ShelfAware's scalability and its independence from closed-set object detectors, we deployed the open-vocabulary pipeline (Pipeline B, Sec. 5.4.2) using the wearable form factor in a 3,500 sq. ft. operational specialty grocery store, generating a highly tractable 380 MB semantic map (Fig. 5.2, right). Because deploying a motion capture system was infeasible in a public retail space, ground truth was established via optimized RTAB-Map SLAM trajectories from the mapping phase. From this, we sampled 10 random evaluation trajectories ($\sim$40 seconds each at a normal walking pace). To ensure a fair comparison, we conducted a grid search over filter parameters for all baselines, reporting the highest-performing configurations.

Table 5.3 presents these results. In this large, geometrically aliased environment, purely geometric methods (MCL, AMCL) collapse ($\leq$ 8% success). Using the semantic vision pipeline alongside depth observations, ShelfAware successfully achieved global localization in 62% of trials, significantly outperforming all baselines. We note that ShelfAware requires more time to achieve initial global convergence in this setting than the FQL baseline. This is an expected computational trade-off: to achieve its higher success rate across a 3,500 sq. ft. space, ShelfAware's inverse semantic model must evaluate and inject particle proposals from a vastly larger candidate pose space.

We also evaluated an upper-bound condition using ground-truth (GT) semantic vectors obtained by raycasting the semantic map from the known RTAB-Map poses. With depth disabled, ShelfAware achieved 98% global-localization success and 92% tracking success, compared with 82%

Table 5.3: Real-world grocery store evaluation. "Upper-Bound Semantics" uses ground-truth semantic observations, with the depth setting indicated for each method, while "Real Vision" uses the live RGB image. Metrics include Global Localization Success (G%), Tracking Success (T%), Time to Global Convergence (G-Time), and translational/rotational RMSE (m/rad).

Method	**G%**	**T%**	**G-RMSE**	**T-RMSE**	**G-Time (s)**
MCL	8%	6%	**0.30**/0.25	**0.31**/0.15	17.4
AMCL	6%	4%	0.36/0.13	0.41/**0.03**	4.1
Real Vision (RGB + Depth)					
FQL	42%	38%	0.41/**0.09**	0.37/0.06	**2.9**
ShelfAware	**62%**	**40%**	0.34/0.12	**0.31**/0.10	6.2
Upper-Bound Semantics (GT Semantics)					
FQL (No Depth)	82%	70%	0.41/0.31	0.14/0.07	**1.6**
ShelfAware + Depth	80%	74%	**0.35/0.19**	**0.05/0.01**	2.9
ShelfAware, No Depth	**98%**	**92%**	0.35/0.27	0.19/0.06	3.2

and 70%, respectively, for FQL under the same GT-semantic, no-depth setting. This comparison supports the benefit of modeling semantic observations as continuous distributions rather than fixed quantities when both methods receive the same semantic evidence.

In a corresponding ShelfAware run with depth enabled, global and tracking success were 80% and 74%, respectively. The depth-enabled condition nevertheless had lower tracking RMSE among its successful trials (0.05/0.01 versus 0.19/0.06 m/rad). Thus, the saved results suggest a trade-off: geometric reweighting produced more precise estimates when it converged, but reduced the frequency of successful convergence in this geometrically aliased environment. Because the two runs were generated separately and their exact stochastic semantic draws and random seeds were not preserved, we treat this as a same-protocol comparison rather than an isolated causal ablation of depth.

5.5.7 Discussion and Limitations

Implications for assistive devices for People with Visual Impairment (PVI): Although our core contribution is a general method for vision-based localization in quasi-static environments, the results have direct implications for assistive navigation. First, **start-anytime** op-

eration is crucial in shared-control assistive use: users often want to invoke assistance on demand rather than run a fully autonomous device continuously. ShelfAware's rapid global localization (mean 1.07s across scenarios, Table 5.1) and its ability to recover from lost tracking (Table 5.2) align with these constraints by enabling on-demand pose estimation and re-localization without external infrastructure. Second, our chosen form factors reflect evidence on acceptance and usability: cane-mounted sensors face handling barriers and are often undesired by users [62] and head-mounted systems are frequently rejected for bulk and stigma [71], while cart-mounted or less bulky wearable solutions have shown promise [86]. These findings motivate the chest-mounted and cart configurations in Fig. 5.5 and Sec. 5.4.6. Finally, a class-level semantic representation is a practical fit for retail navigation: detectors can see many **classes**, but mapping detections into a compact set is both stable under SKU churn and sufficiently distinctive for localization, as evidenced by the high success rates in Table 5.1.

Path to an assistive navigation stack: ShelfAware provides spatial grounding that upstream guidance and downstream interaction modules can exploit. In shopping contexts, prior work has focused on product retrieval or fine-grained identification near the correct shelf [7, 86]. Our results address the prerequisite of reliably **reaching** the correct aisle/shelf in semantically dynamic, geometrically repetitive layouts. Integrating ShelfAware with wayfinding, obstacle avoidance, and product-retrieval interfaces (e.g., speech or haptics) is a natural next step toward end-to-end assistive experiences. Importantly, because our approach requires only vision sensors and VIO, it avoids environmental augmentation (e.g., RFID/beacons) and aligns with infrastructure-free deployments [19, 67, 77, 86].

Limitations: Four limitations warrant discussion. (i) **Large-scale drift and map updates.** Our inverse proposal mechanism can overcome moderate semantic drift by re-injecting particles, provided sufficient semantic cues remain visible in the live scene. However, massive inventory reorganizations outstrip this capacity. ShelfAware currently lacks an automated update mechanism and significant changes require rebuilding the semantic map from scratch. (ii) **Perception failures.** Prolonged occlusions, motion blur, or severe category sparsity can reduce the

information mass in the semantic vector, delaying inverse proposals and weakening forward likelihoods. (iii) **Computational complexity.** Precomputing the inverse semantic cache is strictly offline and scales linearly with the number of discrete states. Online, evaluating K depth beams and C semantic classes per particle scales linearly as $\mathcal{O}(N)$, since K and C are small constants. Recovering via particle injection takes only $\mathcal{O}(1)$ hash-map lookups. (iv) **Human-factors validation.** Our form-factor choices are informed by prior literature and informal feedback, but we did not conduct PVI user studies, measuring usability and trust with PVI participants, including multi-hour battery tests and interaction design, is essential future work.

5.6 Conclusion

In this work we propose **ShelfAware**, a novel method to address the challenging problem of vision-based localization in **quasi-static** indoor environments, where geometry is repetitive and local semantics evolve. ShelfAware models semantics as **distributional evidence over object categories** and couples this representation with an inverse semantic proposal mechanism inside an MCL framework, enabling the filter to remain informative under semantic drift and to generate targeted global pose hypotheses when needed.

Our core contributions are twofold: (i) a semantic mapping and observation design that encodes category counts and coarse spatial structure as probabilistic distributions, and (ii) a real-time particle filter that fuses depth likelihoods with this semantic similarity, utilizing precomputed semantic viewpoints for inverse pose proposals. We demonstrate the efficacy through extensive empirical evaluation, achieving robust **global localization** and **tracking** across both a highly controlled mock setup and a real-world grocery store.

In our mock environment, ShelfAware achieved **97%** overall global-localization success across four conditions vastly outperforming geometric baselines (15% MCL, 10% AMCL) and deterministic Fixed Quantity Landmark baselines (90% FQL). Furthermore, when deployed in a massive 3,500 sq. ft. retail store using an open-vocabulary vision pipeline, it successfully globally localized in **62%** of trials (compared to FQL's 42%), proving its scalability and robustness to severe perception

noise. An upper-bound ablation revealed that ShelfAware extracts significantly more localization signal from perfect semantic data than deterministic methods when provided with ground-truth semantics, ShelfAware achieved **98%** global success and **92%** tracking success, compared to just **82%** and **70%** for the FQL.

Beyond retail, the method applies to service and mobile robots in warehouses and offices, where quasi-static semantics and ambiguous geometry are common. In assistive contexts, the ability to **start anytime** and **relocalize** quickly supports shared-control and infrastructure-free deployment. Future work includes larger-scale evaluations across diverse layouts, embedded implementations, and online maintenance of the semantic layer, further advancing practical, vision-only localization in dynamic real-world environments.

Chapter 6

VLM-GLoc: Vision-Language Model Enhanced Monte Carlo Localization for Robust Semantic Global Localization in Cluttered Quasi-Static Environments

Publication note. This chapter is adapted from the paper **VLM-GLoc: Vision-Language Model Enhanced Monte Carlo Localization for Robust Semantic Global Localization in Cluttered Quasi-Static Environments**, which is currently under review and available as a preprint [6].

Chapter overview. Global localization in geometrically aliased, quasi-static environments such as grocery stores, offices, schools, and hospitals poses a significant challenge for mobile robots. Grocery stores with parallel aisles and a long-tailed distribution of products, as well as offices and labs with repetitive furniture such as chairs, desks, monitors, and doors, exemplify common indoor environments that present geometric and even semantic ambiguity. Traditional approaches rely either on distinct geometric features or on domain-specific vision pipelines that struggle with long-tail semantic distributions and transient visual clutter. We present VLM-GLoc, a method for hierarchical semantic Monte Carlo Localization (MCL) that leverages open-vocabulary Vision-Language Models (VLMs) as a unified semantic observation front-end. We hypothesize a three-fold benefit from VLMs: (1) extracting highly discriminative rich text features, (2) implicit quality filtering of blurry or dynamic objects, and (3) permanence reasoning for targeted data augmentation. We introduce an inverse semantic proposal mechanism that seeds particles via text-to-map retrieval. Evaluated across two real-world environments with different characteristics and two different platforms: a 3,500 sq. ft. grocery store with a cellphone and a 3,700 sq. ft. lab space with a quadruped, VLM-

GLoc achieves 70% and 74% global localization success respectively, substantially outperforming traditional geometry-only and domain-specific baselines.

6.1 Introduction

Global localization (recovering a robot's pose within a known map) is a prerequisite for true autonomy. While classical Adaptive Monte Carlo Localization (AMCL) is the **de facto** standard in robotics, it frequently fails in **geometrically aliased** environments. In spaces like grocery store aisles, office corridors, or repetitive laboratory cubicles, the local occupancy structure is nearly identical across different global regions. Semantic information offers a natural solution: humans distinguish locations not by wall geometry, but by **what they see**. However, existing semantic localization systems [5,140] typically rely on top-level class vectors requiring domain-specific object detectors. This proves brittle against long-tail distributions. In repetitive offices and labs, a generic "chair", "monitor", or "cabinet" label lacks the granularity needed for disambiguation.

We present *VLM-GLoc*, a hierarchical Monte Carlo Localization framework that replaces rigid visual pipelines with a unified Vision-Language Model (VLM) interface. Rather than asking the VLM to directly regress metric coordinates, VLM-GLoc uses it as a rich semantic observation generator. These natural-language observations are embedded by a lightweight sentence encoder and compared against a pre-built semantic map via cosine similarity. Our central hypothesis is that VLMs provide a three-fold advantage for probabilistic localization: *1. Rich Text Features:* Moving beyond simple class labels to highly discriminative descriptions (e.g., from "cabinet" to "yellow metal flammables cabinet"). *2. Implicit Quality Filtering:* The ability to label and suppress noisy detections, transient objects (e.g., jacket, cup), and dynamic actors (e.g., people, cart). *3. Permanence Reasoning & Augmentation:* Classifying items as static, quasi-static, or dynamic, which enables targeted training data augmentation (semantic dropout) without physically altering the scene.

To exploit these advantages without suffering premature geometric collapse, VLM-GLoc employs a hierarchical tracking strategy. We introduce an **inverse semantic proposal** mechanism

where a single query image is inverted into a sparse set of likely pose hypotheses to seed the particle filter. Semantics are used first to narrow down the global search space, while geometry is incorporated as a secondary consistency signal, either through semantic-candidate reranking or through post-convergence range reweighting, depending on the domain configuration.

The contributions of this work are: (1) A VLM-based semantic MCL method that uses open-vocabulary text embeddings as an observation model rather than for direct pose regression, and (2) An inverse semantic proposal mechanism and hierarchical fusion strategy to resolve geometric aliasing. Both are validated within two real-world domains: a 3,500 sq. ft. grocery store and a 3,700 sq. ft. laboratory, demonstrating that a single language interface substantially outperforms both geometry-only and domain-specific baselines.

6.2 Related Work

Probabilistic and geometric global localization. Particle-filter localization remains a standard formulation for start-anytime robot pose estimation because it represents multi-modal beliefs and naturally fuses motion and sensor likelihoods. KLD-sampling and AMCL made this family of methods practical for robot navigation [1,38]. Depth-camera and lidar localization can be accurate when the environment has distinctive geometry [20,122], but recent surveys emphasize that global lidar localization still struggles under perceptual aliasing, dynamics, and map change [131]. Learned implicit or area-graph representations address some of these challenges [66, 127], but they still depend primarily on geometric structure. VLM-GLoc keeps the probabilistic strengths of MCL while adding a language-rich observation channel for places where geometry alone admits many plausible modes.

Semantic and long-term localization. Semantic localization augments geometry with object, text, or layout cues. Semantic SLAM and floor-plan alignment methods use objects and room structure to localize in prior indoor maps [**?**, 140]. Other work exploits text spotting and named landmarks for robust onboard localization [141], or models object existence probabilistically to support long-term localization under change [2]. These approaches show that semantic information can

break geometric symmetries, but their representations are often fixed to a detector vocabulary or a specific landmark type. VLM-GLoc instead uses natural-language scene descriptions, preserving attributes such as color, material, readable text, packaging, and object state.

Retail and mobile localization. Grocery stores pose unique localization challenges due to repetitive shelving geometry and a massive, dynamic semantic space. While prior work [5] demonstrates that fixed-class shelf semantics support mobile localization, relying on predefined category vectors inherently limits the representation. VLM-GLoc expands this paradigm by fusing open-vocabulary product descriptions with a generalized particle filter. By moving beyond fixed ontologies, our approach capitalizes on the highly discriminative value of specific product names, brands, and packaging to resolve global ambiguity.

Open-vocabulary maps and learned visual localization. Recent foundation-model systems build open-vocabulary 3D scene graphs for perception and planning [51]. Vision-and-language navigation studies language-conditioned movement and route following [12, 25, 65], but generally assumes known localization rather than solving it within a fixed prior map. Recent work pushes toward direct localization, utilizing VLMs for 3D reasoning and point-cloud grounding [56, 95]. These methods show that language-conditioned foundation models can support spatial grounding. VLM-GLoc takes a complementary robotics view: the VLM never outputs coordinates. It produces semantic evidence for a probabilistic filter, while multi-modal belief maintenance, odometry fusion, hierarchical lidar consistency, and recovery from dynamic or quasi-static scene changes remain inside MCL.

6.3 Problem Formulation

We assume a prior 2D occupancy map. First, VLM-GLoc builds an offline semantic map. At test time, the robot receives an RGB-D observation, an odometry increment, and optionally lidar measurements. The goal is global localization: estimate the robot pose $\mathbf{x}_t = (x_t, y_t, \theta_t)$ in the prior

Figure 6.1: **VLM-GLoc online localization pipeline.** Evaluated across two domains (quadruped & cellphone), RGB images are VLM-annotated and encoded via a MiniLM that is LoRA-finetuned with quasi-static dropout augmentation for temporal robustness. Comparing this against the semantic map seeds initial particles via inverse proposal. A hierarchical filter then fuses odometry, semantics, and range data to predict the final pose.

map, without assuming a known initial pose. Let

$$\mathcal{M}_{\text{text}} = \{(p_i, T_i, \mathbf{e}_i)\}_{i=1}^{N}$$

be the semantic map over discrete pose cells p_i, where T_i is the VLM-generated scene description expected from pose p_i and $\mathbf{e}_i = f_\phi(T_i)$ is its normalized text embedding. For simplicity, we write $\mathcal{M} = \mathcal{M}_{\text{text}}$ below. Given a query image I_t, a VLM front end produces a structured text observation T_t, and a text encoder produces a normalized query embedding $\mathbf{e}_t$. The semantic likelihood for a particle x is computed by comparing $\mathbf{e}_t$ to the expected map embedding visible from x: $s_{\text{sem}}(x_t) = \cos(\mathbf{e}_t, E_{\mathcal{M}}(x_t))$ where $E_{\mathcal{M}}(x_t)$ is obtained by lookup or ray-casting on the semantic map.

6.4 VLM-GLoc

VLM-GLoc has an offline semantic mapping stage and an online localization stage.

6.4.1 Offline Semantic Map Construction

To build the offline map, VLM-GLoc first extracts objects from a mapping traversal. For each frame, it extracts bounding boxes in normalized coordinates and uses a VLM to generate natural-language labels (5-10 words detailing color, material, object type, and visible text). The

VLM also outputs a detection confidence score, following prior work on verbalized uncertainty estimation in VLMs [49, 128, 129] and a permanence classification: `[Static]` for structural elements, `[Quasi-Static]` for movable furniture, and `[Dynamic]` for transient objects. Low-confidence detections (< 0.3) and `[Dynamic]` items are filtered out.

Next, for each discretized free-space pose $p_i = (x_i, y_i, \theta_i)$, VLM-GLoc raycasts the camera field of view into the occupancy map up to a maximum range $r_{\max}$. All VLM-annotated objects visible from that pose are aggregated with their descriptions concatenated into a unified scene string and encoded via a sentence transformer f_ϕ (a LoRA fine-tuned MiniLM) to produce the embedding $\mathbf{e}_i \in \mathbb{R}^{384}$, yielding a discrete semantic map $\mathcal{M}_{\text{text}} = \{(p_i, T_i, \mathbf{e}_i)\}_{i=1}^{N}$.

The pose discretization differs by domain (Section 6.9.1). While the algorithmic interface is identical across domains, the low-level perception adapts to different platforms. In the lab domain, we use a Unitree Go1 quadruped (Figure 6.3) with an Intel RealSense D455 and Velodyne VLP-16, relying on the VLM to segment and describe repetitive furniture/equipment. In the grocery domain, we use a consumer-grade smartphone and pre-propose shelf items using a YOLO-V9 detector [117] fine-tuned on SKU-110K [45], using the VLM to extract product names, brands, and packaging details.

6.4.2 Quasi-Static Data Augmentation

The base sentence encoder is a frozen MiniLM-L6, a 22M-parameter model well-suited for resource constrained computing platforms. We train a LoRA fine-tuned adapter using triplet loss on pose pairs generated from the offline semantic map. During the lab domain mapping, we queried the VLM twice per pose to

Figure 6.2: **Semantic maps.** [Left] Lab map with repeated classes such as chairs, monitors, and desks. [Right] Grocery map with long-tail products.

obtain varied descriptions of the same

scene, which served as positive pairs. Conversely, scenes separated by a distance greater than a threshold T were utilized as negative pairs. For the grocery domain, we observed that the frozen MiniLM and the LoRA-finetuned model yield similar performance because the products naturally provide strong, distinct long-tail semantics. The lab, however, heavily benefits from fine-tuning because many regions share repeated object classes, the discriminative signal being subtle scene attributes and relative spatial arrangements.

To ensure the model remains robust against common environmental changes, we augment the positive text description pairs during fine-tuning using targeted dropouts. A quasi-static dropout simulates temporal change by removing transient items such as jackets, mugs, backpacks, and movable chairs, from the descriptions. These augmentations effectively simulate month-scale scene dynamics without requiring any physical rearrangement of the lab space.

6.4.3 Online Localization

Figure 6.1 summarizes the online localization pipeline, detailed step-by-step in Algorithm 4.

VLM Observation Generation. At test time, live semantic frames undergo the identical extraction, filtering, and concatenation process described in the offline mapping stage. The resulting scene string is passed through f_ϕ to produce the query embedding $\mathbf{e}_{\text{query}}$, ensuring the live observation and the prior map share a unified, highly discriminative feature space (e.g., precisely matching a "yellow metal flammables storage cabinet" rather than a generic "cabinet").

Inverse Semantic Pose Proposal. The online algorithm begins with an inverse semantic observation proposal, which generates rich semantic pose candidates to quickly narrow the global search space. We compare the query embedding $\mathbf{e}_t$ against all map embeddings and retain the top K pose modes: $\mathcal{H}_t = \text{TopK}_{p_i \in \mathcal{M}} \cos(\mathbf{e}_t, \mathbf{e}_i)$ These modes define a proposal distribution: $q(\mathbf{x}_t \mid I_t) = \sum_{k=1}^{K} \pi_k \mathcal{N}(\mathbf{x}_t, \mu_k, \Sigma_k)$ where μ_k is the pose of the retrieved map cell and $\pi_k \propto \exp(s_k/\tau)$, with $s_k = \cos(\mathbf{e}_t, \mathbf{e}_k)$. The temperature τ controls how sharply particles concentrate around the highest-scoring semantic matches. This proposal step replaces a fraction of the initially uniform

particles with samples drawn near the strongest semantic modes.

Hierarchical Semantic MCL. After initialization, particles are propagated via odometry and reweighted using both semantic and range likelihoods:

$$\mathbf{x}_t^{(n)} \sim p(\mathbf{x}_t|\mathbf{x}_{t-1}^{(n)}, u_t), \tag{6.1}$$

$$w_t^{(n)} \propto w_{t-1}^{(n)} L_{\text{sem}}(I_t|\mathbf{x}_t^{(n)})^{\alpha_t} L_{\text{range}}(z_t|\mathbf{x}_t^{(n)})^{\beta_t}. \tag{6.2}$$

This factorization assumes that, conditioned on the robot pose and the map, the semantic image observation I_t and the geometric range observation z_t provide independent measurement evidence. The hierarchical nature of VLM-GLoc is controlled by scheduling the exponent weights α_t and β_t. During initial global localization, we enforce a strict semantic hierarchy ($\alpha_t \gg \beta_t$). The semantic likelihoods remain sharp while geometry is kept deliberately soft, preventing premature convergence to incorrect poses in geometrically aliased regions (e.g., identical aisles or workstations). In the grocery domain, once the particle cloud has spatially converged, the range weight β_t is increased. In the lab domain, online range reweighting remains disabled ($\beta_t = 0$). Lidar is instead used to rerank the highest-scoring semantic candidates before particle seeding. We define this spatial convergence as the moment the median particle spread $\sigma_t = \text{median}_n(\|\mathbf{x}_t^{(n)} \text{median}(\mathbf{x}_t^{(1:N)})\|_2)$ falls below a threshold $\sigma_{\text{conv}} = 3.0\,\text{m}$. These strategies use geometry to refine or verify the leading semantic pose hypotheses while avoiding premature convergence to incorrect modes.

The semantic likelihood is evaluated as temperature-scaled cosine similarity: $L_{\text{sem}}(I_t|\mathbf{x}_t^{(n)}) = \exp\left(\tau \cos(\mathbf{e}_t, E_{\mathcal{M}}(\mathbf{x}_t^{(n)}))\right)$. The geometric likelihood evaluates a 2D range observation (derived from the depth camera or lidar) against a distance transform of the occupied map cells. Finally, to select the best multi-modal hypotheses, VLM-GLoc estimates the final tracked pose by clustering high-weight particles via DBSCAN and returning the dominant weighted mode.

6.5 Experimental Setup

Figure 6.3: Go1 with RealSense D455 and Velodyne VLP-16.

We collected 20 test trajectories per domain. To evaluate robustness against temporal changes, the lab domain trajectories we evaluate against were recorded one month after the initial mapping. During this period, several items, including bikes, whiteboards, robots, computer equipment, and chairs were moved or introduced, providing a realistic scenario for testing quasi-static resilience. Due to logistical constraints in the public retail space, the grocery trajectories were recorded later on the same day as the initial mapping. Key implementation parameters, including semantic encoder hyperparameters and particle filter configurations, are detailed in Section 6.9.1.

Domain 1 (Lab): The 3,700 sq. ft. research facility is traversed by a Unitree Go1 quadruped (Figure 6.3). The environment is visually and geometrically repetitive, containing identical cubicles, desks, monitors, chairs, cabinets, and doors. Localization utilizes a SLAM Toolbox [80] occupancy map paired with our discrete semantic map.

Domain 2 (Grocery store): The grocery domain is a 3,500 sq. ft. international grocery store captured using a consumer-grade smartphone (RGB-D data). The store consists of parallel shelving units densely packed with long-tail products. The geometry is massively aliased, as multiple aisles produce nearly identical lidar range measurements. This means localization must rely entirely on the discriminative semantic fingerprints provided by product names, brands, and

packaging.

Ground Truth. We utilize an offline map-aligned SLAM reference trajectory for both domains to ensure a high-fidelity baseline. We recorded exhaustive traversals achieving full map coverage. Because robust Iterative Closest Point (ICP) [18] alignment requires this extensive spatial overlap, we use it strictly offline to align these rebuilt environment maps against the prior mapping traversals, obtaining highly accurate translation and rotation matrices. We then chunked these aligned traversals with random starting points into the individual test trajectories used for evaluation.

Baselines. AMCL: The AMCL baseline uses the same particle-filter infrastructure but without semantic observations. It uses range likelihoods and odometry only.

FCS-MCL (Fixed-Class Semantic MCL): To evaluate fixed-class semantic localization, we implemented the ShelfAware [5] baseline (tailored for the grocery domain), which alongside [140] represents the standard semantic paradigm. This categorical particle filter captures the spatial distribution of these classes (using predefined category counts, mean distances, and mean relative angles) to model high-level semantic layouts. Adapting a domain-specific vision front-end, the grocery state vector tracks 20 product categories (e.g., rice, lentils, spices). For the lab, it uses a vocabulary like chair, desk, monitor, sign, cabinet, door, and robot.

Metrics: Following established conventions in the literature [5, 140, 141], we define **convergence** as the estimated pose falling within 0.7 m translation and $\pi/4$ rad rotation of the ground truth. To decouple initial pose estimation from continuous state tracking, we evaluate VLM-GLoc via:

Global localization success: A trial is successful if initial convergence occurs within the first 95% of the trajectory.

Global localization error (G-error): The absolute translation and rotation error measured at the initial convergence, capturing the accuracy of the system the moment global ambiguity is resolved.

Whole Trajectory ATE (W-ATE): The Absolute Trajectory Error (translation and rotation

RMSE) computed from the moment of initial global convergence until the absolute end of the sequence.

Stable tracking success: A trial is successful if it contains a final tracking that begins before the 95% cutoff and remains strictly within the convergence threshold until the end of the sequence.

Stable Tracking ATE (T-ATE): The Absolute Trajectory Error (translation and rotation RMSE) computed over the successfully tracked trajectory.

6.6 Results

Hypothesis 1: Rich Text Features Resolve Aliasing: We first demonstrate that rich semantics resolve severe geometric aliasing better than both pure geometry and top-level class vectors across two distinct domains. We run each trajectory 5 times to account for stochasticity and report the mean. Table 6.1 reports the unseen lab data comparison against AMCL and FCS-MCL on the exact same trajectory chunks. AMCL succeeds on only 4% of the runs due to repeating cubicles. FCS-MCL succeeds on 10%, its class vector erases the specific details (material, text, color, state) that distinguish repeated lab objects. In contrast, VLM-GLoc reaches 74% global localization success. The tracking success numbers remain lower than global localization success. However, the lab tracking trajectories are often visually close even when some frames fail the threshold. When we evaluate tracking using an ATE threshold, meaning small trajectory violations are allowed as long as the mean error remains under the threshold, the stable tracking success improves from 46% to 60%.

Table 6.1: Lab evaluation after one month of mapping. Metrics include Global Localization (Global %) and Tracking Success (Tracking %). All errors are reported as translation (m) / rotation (rad).

Method	Global %	G-error (t/r)	W-ATE (t/r)	Tracking %	T-ATE (t/r)
AMCL	4%	**0.45 / 0.12**	6.72 / 1.44	0%	n/a
FCS-MCL	10%	0.50 / 0.32	0.79 / 0.29	0%	n/a
VLM-GLoc	**74%**	0.51 / 0.23	**0.66 / 0.20**	**46%**	**0.41 / 0.13**

Table 6.2 shows similar findings in the grocery domain. Even against a competitive class-

vector baseline utilizing 20 grocery categories, VLM-GLoc improves global localization success from 30% to 70%. While the depth profiles of parallel aisles are identical, the long-tail language of product brands, packaging, and ingredient names provides a unique semantic fingerprint that allows the filter to converge reliably. While baselines occasionally report marginally lower tracking errors (T-ATE), this can be attributed to the fact that methods like AMCL and FCS-MCL succeed far less often. In contrast, VLM-GLoc maintains comparable RMSEs while successfully localizing on a vastly larger set, demonstrating a significant contribution to system reliability. Sensitivity analysis (Section 6.9.5) reveals that minor relaxations of the spatial convergence thresholds boost VLM-GLoc's global localization success to 95%. Isolating translation error accounts for the mirrored semantic viewpoints prevalent in the grocery domain, yielding significant gains not observed in baseline methods.

Hypothesis 2: Implicit Quality Filtering: We hypothesize that VLMs act as cognitive filters, suppressing noisy visual evidence. To validate this, we conducted an ablation in the grocery domain where blurry or unreadable product crops were intentionally retained in both the map and the query text. These low-quality images are particularly prevalent in this domain due to the handheld cellphone capture. As shown in Table 6.3, forcing the system to utilize these noisy observations degrades global localization by a 29% relative drop. Retaining unreadable items makes the semantics noisier, delays convergence, and prevents the filter from anchoring to reliable landmarks.

Hypothesis 3: Permanence Reasoning & Training Data Generation via Augmentation: Finally, we demonstrate within the lab domain that VLM-GLoc can characterize transient features to improve localization. As highlighted in Table 6.3, relying on a frozen MiniLM encoder results in an 18.75% relative drop in global localization success compared to our LoRA finetuned adapter, which is explicitly trained with targeted text-dropout augmentation that removes quasi-static items (jackets, backpacks, movable chairs). Furthermore, entirely removing VLM-generated permanence and quality tags from the descriptions causes a measurable 8% relative drop in success, proving that reasoning over these cues yields richer semantics crucial for robust localization.

Table 6.2: Grocery store evaluation over 20 trajectories in a 3,500 sq. ft. space.

Method	Global %	G-error (t/r)	W-ATE (t/r)	Track %	T-ATE (t/r)
AMCL	20%	0.50 / **0.10**	2.08 / 0.59	5%	0.31 / **0.15**
FCS-MCL	30%	**0.30** / 0.26	0.82 / **0.39**	25%	**0.30** / 0.16
VLM-GLoc	**70%**	0.32 / 0.31	**0.77** / 0.64	**65%**	0.31 / 0.18

Table 6.3: Ablation impacts on global localization success across domains.

Condition	Global % Δ
Retained blurry crops (Quality Filtering)	-29% (grocery)
Removed permanence tags (Permanence Reasoning)	-8% (lab)
LoRA with Augmentation vs. Frozen MiniLM (No Fine-tuning)	-18.75% (lab)

6.7 Qualitative Analysis

Inverse Semantic Proposal. As shown in Figure 6.4, inverse semantic retrieval provides strong global pose estimates before temporal particle filtering. The qualitative results highlight the resilience of our semantic proposal against common real-world failure modes. For instance, while mirrored viewpoints induce some local orientation scatter, the semantic hypotheses still concentrate in the correct global region. Furthermore, under severe perceptual ambiguity, such as featureless areas with identical doors, the system degrades gracefully, proposing multiple plausible regions. VLM-GLoc handles month-scale quasi-static changes well. Although items shifted between mapping and testing, the unified scene descriptions maintain enough contextual overlap to retrieve the correct area, showing applicability to long-term deployment in active spaces.

Trajectory Examples. As a counterpart in the retail domain, Figure 6.5 illustrates example grocery trajectories with ground truth and VLM-GLoc pose estimates. In this domain, semantic convergence and tracking are usually aligned because product identities sharply disambiguate repeated aisles.

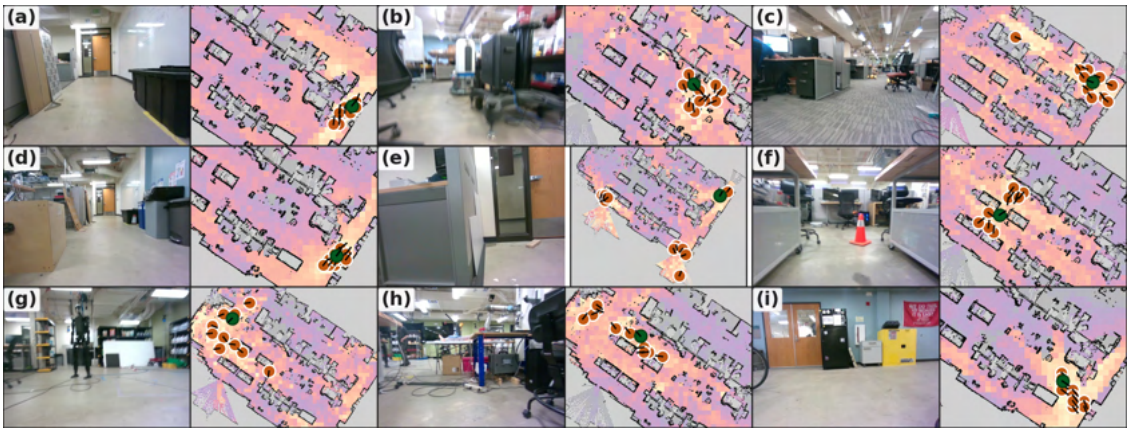


Figure 6.4: **Top-10 inverse semantic pose proposals.** For each live RGB observation (left), orange markers denote the top-10 retrieved semantic pose hypotheses within the prior map (right), while the green marker indicates the ground truth pose. The underlying heatmap visualizes the dense semantic cosine similarity, with lighter (yellow) regions representing higher retrieval scores. The proposal mechanism successfully anchors the robot despite severe motion blur (b), geometrically aliased or mirrored viewpoints (c, f, g, h), repeated-door ambiguity (e), and month-scale scene changes such as relocated objects or new clutter (f, g).

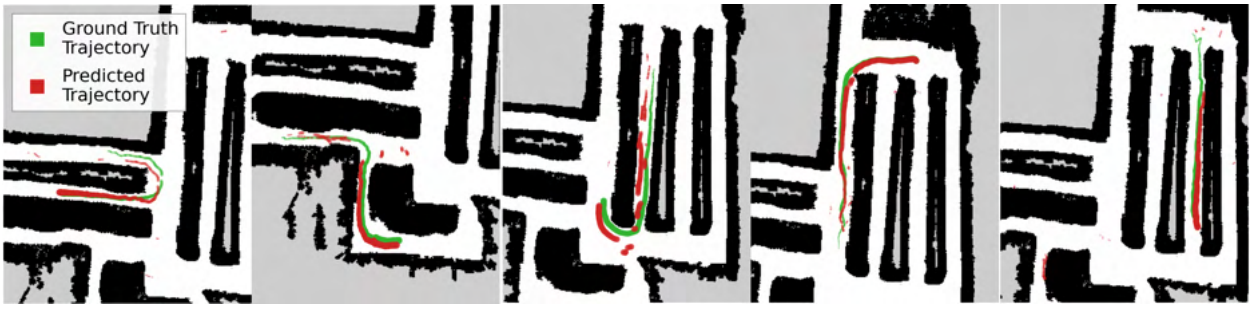


Figure 6.5: Example grocery trajectories with ground truth and VLM-GLoc pose estimates. Thinner to thicker correspond to the temporal progression of the trajectory.

6.8 Discussion

VLMs provide more than object names. The primary value of the VLM front end lies beyond simple object categorization. It preserves fine-grained, open-vocabulary evidence: colors, materials, readable text, packaging, and object state. It also filters low-quality evidence by characterizing noisy observations. This is why VLM-GLoc outperforms class-vector semantic baselines even though both rely on "semantics." For example, a standard detector label such as 'cabinet' loses warning text and material context, 'chair' loses the mesh back and orange cushion details,

and 'sign' loses the actual text. Section 6.9.8 provides high-similarity query/map description pairs demonstrating how this fine-grained evidence yields strong map retrieval scores.

Domain-general interface, domain-specific front ends. Both domains produce the exact same interface to the localizer: structured text and normalized embeddings. While previous baselines require a different class vocabulary and scoring vector engineered specifically for each domain, VLM-GLoc removes the need for a dedicated vision front end for each domain.

Hardware Accessibility and Form Factor. The grocery evaluation demonstrates that robust localization can be achieved using a standard cellphone as the primary sensor suite. This lowers the hardware barrier to entry, making the framework applicable for lightweight mobile robots and cellphone-based assistive navigation technologies.

Runtime and Practical Deployment. While the cached particle filter runs at 8 Hz (with Go1 sensors) and 44 Hz (with cellphone sensors), live synchronous cloud-VLM API calls yield an effective online rate of roughly 0.21 Hz for the lab and 0.91 Hz for the grocery store (using a semantic update interval of 5 frames, which was standard across our experiments). This sparse update rate is practical for resolving global ambiguity. During tracking, VLM-GLoc fuses these intermittent semantic observations with high-frequency geometric filtering and odometry to maintain smooth pose estimates. Full latency profiling is detailed in Section 6.9.2. Despite the latency bottleneck introduced by cloud APIs, the resulting accuracy and unified semantic front-end make combining VLMs within a probabilistic framework highly promising. Notably, while the grocery pipeline still utilizes a generic product class detector at the front, VLM-GLoc entirely removes the need for domain-specific clustering, spatial aggregation rules, and dense feature extractors required by prior systems.

Limitations: Our method inherently relies on rich semantic features to disambiguate scenes, meaning performance degrades in visually barren environments. Furthermore, balancing particle convergence for smooth pose estimation against the need to explore new hypotheses remains challenging. When likelihoods drop, it is difficult to determine whether the cause is physical scene changes, a lack of local features, or if particles have drifted into an incorrect mode requiring

global re-exploration. Finally, while the current cloud-based VLM latency restricts high-frequency semantic tracking, rapid advancements in lightweight, on-device VLMs promise to alleviate this constraint.

Conclusion: VLM-GLoc reframes VLMs as semantic observation generators for probabilistic robot localization, replacing brittle, domain-specific object detectors with a unified natural language interface. By converting query images into language-rich pose proposals and hierarchical semantic likelihoods, the system resolves severe global ambiguity in environments where geometry alone fails. Across vastly different domains: from repetitive laboratory cubicles navigated by a quadruped to dense retail aisles mapped via a consumer cellphone, the identical semantic MCL framework achieves robust global localization. VLM-GLoc demonstrates that rich open-vocabulary descriptions, implicit quality filtering, and targeted quasi-static reasoning offer a highly portable, hardware-agnostic solution for autonomous indoor navigation.

6.9 Supplementary Implementation and Analysis Details

6.9.1 Implementation and Reproducibility Details

Table 6.4 records the main parameters for the headline runs.

Table 6.4: Key implementation parameters.

Component	Value
Compute	Dell G15 laptop (Intel Core i7-11800H (2.3 GHz), NVIDIA RTX 3060, 6GB VRAM, 32GB RAM)
Lab semantic map	SLAM Toolbox occupancy map at 0.05 m/px, semantic grid spacing 0.50 m, 24 heading bins, 80.5° camera FOV, 10 m raycast, objects retained after ≥ 3 mapping observations
Grocery semantic map	Occupancy map at 0.05 m/px, semantic grid spacing 0.50 m, 8 heading bins, 60° camera FOV, 6 m raycast
Semantic encoder	MiniLM-L6-v2 text encoder with 384-d embeddings, lab uses triplet LoRA, grocery uses the frozen encoder
Lab LoRA training	Triplet loss, margin 0.5, LoRA rank 16, alpha 32, module dropout 0.1, 30 epoch schedule, batch size 32, learning rate 2×10^{-5}, $T = 3m$
Lab text augmentation	Positive-pair augmentation with quasi-static dropout, dynamic removal, item shuffle, subset-drop, composed perturbations, and exhaustive quasi-static variants, quasi-static and subset dropout remove 1–2 items
Lab localization	Semantic update every 5 frames, top-400 semantic proposal, candidate temperature 10, semantic seed noise $\sigma_{xy} = 0.30$ m, $\sigma_\theta = 0.16$ rad, DBSCAN pose estimate with $\epsilon = 1.0$ m, lidar weight 0.10
Grocery localization	Semantic update every 5 frames, semantic temperature 10, adaptive depth/lidar cap 0.50, base lidar weight 0.10, DBSCAN pose estimate with $\epsilon = 1.0$ m
Semantic/range weights	Post-convergence schedule with $\alpha_t = 1$. Before spatial convergence, $\beta_t = 0$, after $q_{50} < 3.0\,\text{m}$, $\beta_t = 0.10$, where q_{50} is the median particle distance from the particle-cloud spatial median. For the lab headline setting, regular online range reweighting is disabled ($\beta_t = 0$), lidar is used instead for semantic-candidate reranking with $\beta_{\text{rerank}} = 1.0$ and top-$k = 10$ lidar pose estimation.
Particle filter	KLD-adaptive MCL with 1500-particle cap and 200-particle floor, motion noise $(0.05, 0.02, 0.02, 0.01)$, laser $\sigma_{\text{hit}} = 0.30$ m
VLM	`gemini-3-flash-preview` (Temp: 0.3)
Pose estimate	Dominant DBSCAN-weighted particle mode over position and circular heading

6.9.2 Runtime and Latency Profiling

Table 6.5: Runtime summary. Cached particle filter (PF) Hz excludes live VLM annotation. Live annotation uses a semantic interval of 5 for both domains to estimate practical online Hz.

Domain	Cached PF Hz	Live VLM Mean Latency	Est. Online Hz
Lab	8.64	23.44 s	0.21
Grocery	44.27	5.36 s	0.91

6.9.3 Dominant-Mode Pose Estimate

The posterior of the particle filter can remain multi-modal. We therefore do not estimate pose by a global weighted mean over all particles. Instead, we cluster high-weight particles with DBSCAN over position and circular heading features and return the dominant weighted mode. This better matches the localization problem: the robot should report one coherent pose hypothesis, not the average of two different aisles or two mirrored lab regions.

6.9.4 Online Localization Algorithm

Algorithm 4: VLM-GLoc Online Localization

Input: Semantic map $\mathcal{M}$, particles $\{\mathbf{x}^{(n)}, w^{(n)}\}$, odometry u_t, image I_t, range z_t

Output: Dominant-mode pose estimate $\hat{\mathbf{x}}_t$

Propagate particles with odometry motion model $p(\mathbf{x}_t|\mathbf{x}_{t-1}, u_t)$

if semantic frame is available then

  $T_t \leftarrow \text{VLMAnnotation}(I_t)$

  $\mathbf{e}_t \leftarrow \text{TextEncoder}(T_t)$

  if not initialized then

    $\mathcal{H}_t \leftarrow \text{TopK}_{p_i \in \mathcal{M}} \cos(\mathbf{e}_t, \mathbf{e}_i)$

    Seed a fraction of particles from $q(\mathbf{x}_t|I_t)$ around $\mathcal{H}_t$

  Reweight particles by $L_{\text{sem}}(I_t|\mathbf{x}_t)^{\alpha_t}$

if range observation is available then

  Reweight or rerank particles by $L_{\text{range}}(z_t|\mathbf{x}_t)^{\beta_t}$

Normalize weights

Resample when effective particle count is low

$\hat{\mathbf{x}}_t \leftarrow$ dominant DBSCAN weighted mode

Return $\hat{\mathbf{x}}_t$

6.9.5 Convergence Sensitivity Analysis

As discussed in Section 6, the strict $\pi/4$ rotation threshold occasionally penalizes valid localizations where particles observe the exact same semantic and range targets from mirrored viewpoints (a common occurrence in parallel grocery aisles). Table 6.6 demonstrates that when evaluating purely on translation, or when slightly relaxing the strict spatial threshold, VLM-GLoc's success rate climbs significantly, whereas baselines fail to see similar improvements.

Table 6.6: Sensitivity of grocery store localization and tracking success to relaxed spatial thresholds (0.7 m / 0.8 m / 0.9 m). The heading threshold is fixed at $\pi/4$ rad for Joint metrics. Table 6.2 shows results corresponding to the highlighted numbers and threshold.

Method	Translation-Only (%)		Joint Translation + Rotation (%)	
	Global Success %	Tracking Success %	Global Success %	Tracking Success %
AMCL	30 / 30 / 30	10 / 15 / 15	20 / 20 / 20	5 / 10 / 10
FCS-MCL	35 / 35 / 35	20 / 30 / 30	30 / 30 / 30	25 / 25 / 25
VLM-GLoc	**85 / 90 / 95**	**70 / 75 / 80**	**70 / 75 / 80**	**65 / 70 / 70**

6.9.6 Ground Truth Acquisition Details

To establish ground truth, we initially evaluated a ceiling-mounted AprilTag constellation in the lab domain. However, disagreements among simultaneously visible tags produced systematic pose uncertainties on the order of 0.75 m, which is too high for rigorous evaluation. Furthermore, deploying such physical infrastructure in the public grocery store was logistically infeasible. Therefore, we utilized the offline map-aligned SLAM reference trajectories described in Section 5, which provided a highly accurate and consistent baseline across both domains without requiring physical environmental modifications.

6.9.7 Rich VLM Description Examples

Table 6.7: Qualitative comparison between generic object classes and VLM-GLoc's tagged, open-vocabulary descriptions. `[Dynamic]` `[Blurry]` objects are filtered before map/query embedding so they are not present in the following.

Domain	**Generic Object Class**	**VLM Rich Description**
Lab	Chair	`[Quasi-Static]` black mesh office chair with orange fabric seat cushion and wheels.
Lab	Cabinet	`[Quasi-Static]` yellow metal `JUSTRITE` flammables storage cabinet with warning labels.
Lab	Door	`[Static]` light-brown wooden door with rectangular glass panel, silver handle, and metal kick plate.
Lab	Whiteboard	`[Static]` large whiteboard with checkerboard AprilTag markers and handwritten `VLA` notes.
Lab	Robot	`[Quasi-Static]` white-blue mobile robot base beside a red robotic arm, blue cables.
Grocery	Lentils	Swad Yellow Vatana (Peas), Swad, Lentils.
Grocery	Condiment	Mother's Recipe Mango Pickle, Mother's Recipe, Condiments.
Grocery	Oil	Jiva Organic Mustard Oil, Jiva, Oil.
Grocery	Spice	MDH Pav Bhaji Masala, MDH, Spices.

6.9.8 High-Similarity Description Pairs

Table 6.8: VLM description pairs from held-out lab queries and the semantic text map. The top rows are good semantic matches, rows below the separator are weaker/confusable pairs that share generic object classes but differ in fine-grained attributes or scene context. Similarity is cosine score from the lab LoRA encoder. For grocery product descriptions, blurry/unreadable and dynamic detections are filtered before embedding.

Query description excerpt	Raycasted map description excerpt	Sim.
`[Quasi-Static]` red fabric banner with white text `WE DO THIS NOT BECAUSE IT IS EASY`, `[Quasi-Static]` yellow metal cabinet with `JUSTRITE` and `FLAMMABLES` labels, gray rolling cabinet.	`[Quasi-Static]` red fabric banner with matching white text, `[Quasi-Static]` yellow safety cabinet with `FLAMMABLES KEEP FIRE AWAY`, nearby wall signs and wooden door.	0.813
`[Static]` brown wooden double doors with glass panels, `[Quasi-Static]` black refrigerator with papers, `[Quasi-Static]` white emergency cabinet, yellow cabinet.	`[Quasi-Static]` yellow flammables storage cabinet with warning labels, red banner, `[Static]` light-brown door with metal kick plate, blue wall fixtures.	0.782
`[Quasi-Static]` tan plywood crate with latch, silver wire shelving, cardboard sheets, gray industrial cabinet, `[Static]` light-brown door.	`[Static]` light-brown door with glass window, `[Quasi-Static]` gray storage cabinet, blue recycling bin, black storage bin, whiteboard/storage area.	0.768
Weaker pairs sharing generic object classes		
`[Static]` light-brown wood-grain door with silver kick plate, dark metal door frame, `[Quasi-Static]` cubicle partition and gray storage cabinet.	`[Static]` brown wooden door with blue-white sign, `[Static]` orange wooden door, `[Static]` fire-extinguisher cabinet, repeated wall boards.	0.605
`[Quasi-Static]` wood and metal workbench, gray storage cabinet, red/yellow storage bins, black fabric office chair, backpack, fire-extinguisher cabinet.	`[Quasi-Static]` black monitor on a workbench, computer tower, gray storage cabinets, black padded rolling stool, yellow storage bin.	0.526
`[Quasi-Static]` large monitor on desk, gray cubicle desk, blue chair, black computer tower, cardboard boxes, trash bin.	`[Quasi-Static]` black mesh chair with orange seat, gray storage cabinet, black framed monitor, wood desk edge, repeated workstation objects.	0.430

6.9.9 Gemini Annotation Prompts

Lab / office object annotation prompt.

```
Detect all distinct objects in this lab/office image. For each object, provide:
1. A bounding box [ymin, xmin, ymax, xmax] in normalized 0-1000 coordinates
2. A SHORT label (5-15 words max): color + material + object type. Include any readable text.
3. Permanence: [S]=structural, [QS]=quasi-static furniture/equipment, [D]=dynamic people
4. Confidence (0.0-1.0): how certain you are this is a real, distinct object
(not blur/shadow/artifact)

Return as JSON array:
[{"box": [y1, x1, y2, x2], "label": "brief identity", "perm": "S/QS/D", "conf": 0.9}]

Rules:
- Do NOT include position in image (left, right, foreground)
- Do NOT include spatial relationships to other objects
- Do NOT detect plain floor or plain ceiling
- DO detect: walls with signs/text, distinctive fixtures, furniture, equipment, signs,
↪  monitors
- Be SPECIFIC: not "chair" but "black mesh office chair with orange cushion"
- Read and include any visible TEXT on signs, labels, room numbers
- Ignore motion blur artifacts and infrared dot patterns, only detect clearly visible objects
- conf should be LOW for blurry/unclear objects
```

Grocery product-crop annotation prompt.

```
You are analyzing a grid of {n} product images detected on shelves in an Indian grocery
↪  store.
Each image is labeled with an ID (C0, C1, etc.) and is a crop from a shelf detector.

For EACH product crop, identify it and provide:
{
  "id": "C0",
  "product_name": "specific product name (brand + product if readable)",
  "brand": "brand name if visible, else null",
  "category": "e.g., Spices, Lentils, Beverages, Snacks, Oil, Tea, Rice, Flour, Condiments,
  Sweets, Frozen, Dairy, Personal Care, Household",
  "is_blurry": true or false
}

Rules:
- Be specific: say "MDH Chana Masala" not just "spice". Say "Laxmi Moong Dal" not just
↪  "lentils".
- If the crop is too blurry to identify the product, set is_blurry to true
and still fill your best guess.
- If there are NO identifiable products (e.g., empty shelf, wall, floor, ceiling), return:
  {"products": [], "suitable_for_localization": false}
- Otherwise set "suitable_for_localization": true

Return ONLY valid JSON with a "products" array and "suitable_for_localization" boolean.
No markdown fences.
```

Figure 6.6: Gemini prompts used for VLM annotation in the lab and grocery domains. The lab prompt operates on full RGB frames and requests object boxes, rich labels, permanence tags, and confidence. The grocery prompt operates on YOLO-detected product crop grids and requests product identity, category, and blur filtering metadata.

Chapter 7

GIST: Multimodal Knowledge Extraction and Spatial Grounding via Intelligent Semantic Topology

Chapter overview. Dense, quasi-static environments such as retail stores pose a spatial grounding challenge for humans and embodied AI systems. Visual features become stale with inventory perturbations, and long-tail semantics defeat fixed-class perception. Vision-Language Models bring rich semantic priors to these spaces but lack a representation that connects what they see to where things are. In this work, we investigate what minimal spatial representation makes VLM-generated guidance groundable and propose semantic topology: a traversability graph extracted from a single consumer mobile scan, annotated with image-grounded semantic anchors bound to graph edges in a shared coordinate frame. We formalize this representation and show that one instance of it supports four downstream human-AI interaction tasks without task-specific maps: intent-aware semantic search with zone-level fallback, one-shot text-embedding global localization, semantic zone classification, and landmark-rich egocentric route instruction. Controlled ablations judged by two independent held-out LLMs isolate the necessary components: computed route structure and line-of-sight-validated landmark text drive instruction quality, while rendered map images contribute compact visual grounding but do not substitute for the topology. GIST outperforms visual-sequence and coordinate-only baselines, and a formative pilot offers preliminary real-world signal. Our results show that for quasi-static spaces the representational bottleneck for spatial language generation is explicit geometry and validated semantics rather than richer visual input to the model.

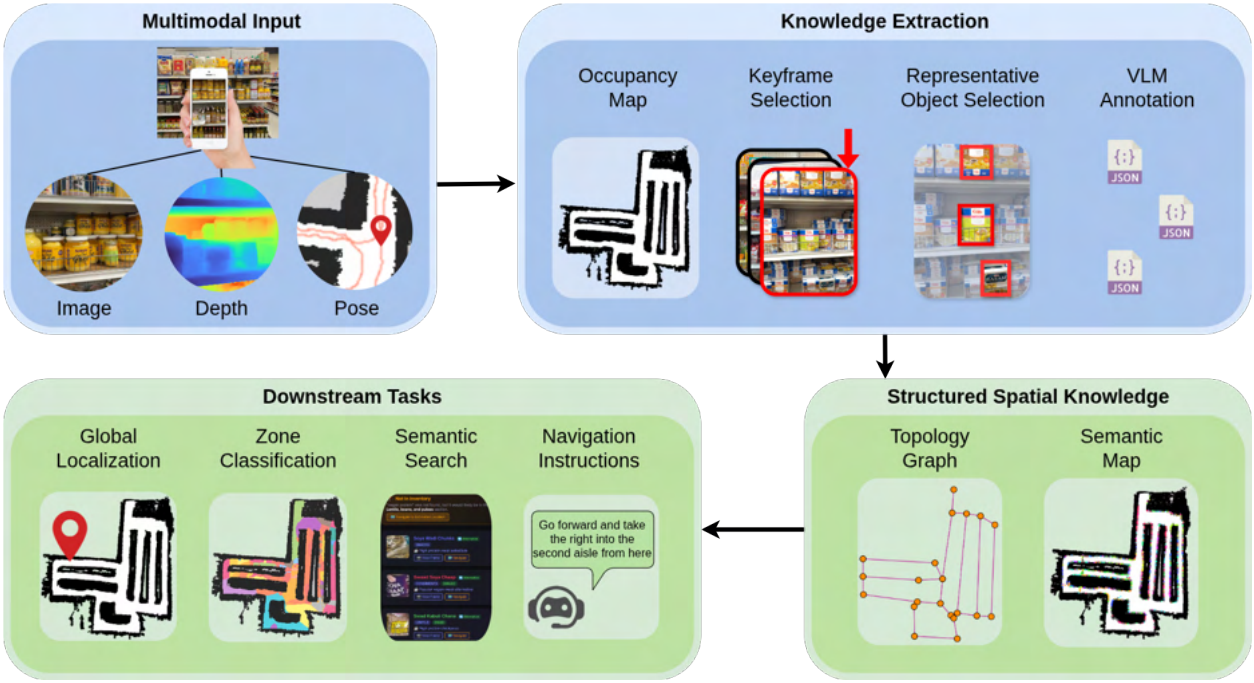


Figure 7.1: The **GIST** multimodal knowledge-extraction architecture. Raw multimodal inputs are distilled into structured spatial knowledge that supports intent-aware semantic search, coarse global pose initialization, zone inference, and spatially grounded route language.

7.1 Introduction

Navigating and searching dense, cluttered environments poses a spatial-grounding challenge for both humans and autonomous embodied AI. Conventional occupancy-grid localization systems represent free and occupied space together with metric geometry [1], but they do not encode the semantic structure people use for search, orientation, and route understanding [44, 76, 78]. In dense quasi-static spaces, useful directions also depend on stable landmarks and meaningful topological transitions rather than metric commands alone [63, 78].

These environments change gradually: inventory is restocked, products move locally, and temporary occlusions appear even when the global layout remains stable. They also contain long-tail, culturally specific items whose visual similarity produces perceptual aliasing. Recent Vision-and-Language Navigation systems perform well in benchmarked and synthetic domains [25, 65], but

real-world studies continue to expose weaknesses when targets are occluded, visually repetitive, or absent from the current view [3, 119, 124].

Sequence-based instruction generators such as NavComposer [53] are an important route-language paradigm, but their native protocol conditions on RGB observations sampled along a future trajectory. This input can expose useful visible landmarks, yet it does not explicitly encode stable route geometry, metric turn structure, or a persistent semantic goal map.

We introduce **GIST** (**G**rounded **I**ntelligent **S**emantic **T**opology), a semantic-topology framework constructed from a single consumer mobile scan (Figure 7.1). GIST converts RGB-D data and odometry into a 2D occupancy map, derives a traversability graph over walkable space, and registers VLM-labeled products as semantic anchors in the same metric frame. The representation separates deterministic route geometry from generative semantic interpretation while retaining source-image evidence for each anchor.

Together, these components form an integrated semantic-topology representation that supports four downstream Human-AI interaction tasks: (1) *Intent-Driven Search & Missing Goal Estimation*; (2) one-shot *Semantic Localization*; (3) *Zone Classification*; and (4) *Semantic-Topology Routing* that converts graph paths into view-independent, landmark-rich language. The current implementation constructs this representation offline for a quasi-static environment.

Across 15 grocery-store route scenarios, Full GIST and GIST Text perform similarly (3.93 versus 3.85, $p = .260$), while Full GIST outperforms visual-sequence and naive geometric prompting baselines on average. Controlled ablations show that the largest losses arise when topology-derived route text and occupancy-validated landmark text are removed. A preliminary in-situ pilot with five sighted graduate students yielded 8/10 successful verbal-guidance trials.

7.2 Related Work

GIST facilitates several critical human-AI and embodied AI tasks, the following subsections present related work foundational to these domains.

7.2.1 Vision-and-Language Navigation (VLN)

VLN requires an embodied agent to interpret natural language instructions and navigate 3D environments solely from visual observations [12]. Early benchmarks operated in discrete graph-based settings where agents teleport between panoramic viewpoints [94]. While VLN-CE [65] pushed toward continuous environments requiring low-level motor control, these systems still assume high-quality instructions are already provided.

Automated instruction generation has emerged to scale VLN training data. Early speaker models used LSTM-based networks to translate panoramic sequences into directions [39], lacking spatial specificity. Recent models like NavRAG [121] utilize scene description trees to provide fine-grained context. Goal-conditioned generation models like GoViG [125] attempt to produce instructions from only a first-person observation and a goal image. NavComposer [53] generates instructions from RGB image sequences combined with discrete action traces. However, because sequence-to-sequence methods bypass explicit topological reasoning, they struggle to infer path geometry from images alone. This results in instructions that reference visually salient but transient objects (*e.g.*, red crates, shopping carts) and cannot reliably specify definitive turn sequences, exact metric distances, or macro-level navigation instructions.

Scene Graphs

Scene-graph approaches such as ConceptGraphs and 3D scene-graph planning systems construct symbolic object-relation representations for querying and task planning [14, 51, 96]. These representations are powerful for relational reasoning, but they typically emphasize object nodes and symbolic relations. GIST instead centers a 2D occupancy map and a route-legible traversability graph, then binds image-grounded semantic anchors to reachable graph locations. This exposes metric distance, turns, edge headings, and line-of-sight landmarks directly for search, localization, and route communication.

7.2.2 Localization in Quasi-Static Environments

Particle filter-based localization methods (e.g., AMCL [1, 38]) remain standard for onboard robotics. However, their reliance on static geometric maps makes them notoriously brittle in quasi-static environments like warehouses and retail stores, where local semantics fluctuate even when global geometry remains constant [131].

Semantic SLAM systems [**?**,140] often assume landmark stability, an assumption that weakens when products shift or fluctuate in quantity. Recent probabilistic approaches such as **ShelfAware** [5] model shelf semantics as distributions over object counts. Deep visual localization and implicit neural representations [34,66,75] offer complementary appearance-based approaches. GIST investigates whether open-vocabulary text semantics can retrieve a coarse global pose mode for later geometric or temporal fusion.

7.2.3 Assistive Guidance Systems

Navigating large retail spaces or warehouses presents a challenge for people with visual impairments, involving macro-navigation in the locomotor space and micro-navigation in the haptic space [43].

Macro-Navigation and Product Search: Recent work focuses largely on navigation inside the store [67–69]. Solutions that rely on environmental augmentation, such as RFID tags [77] or Bluetooth beacons, introduce high maintenance overhead and barriers to adoption. For product identification, existing techniques using fixed-class object detectors [37] or barcode scanning [47,86] are impractical for the sheer volume of long-tail retail items [87]. While researchers have explored conversational agents [54, 57] and the "last few meters" way-finding problem [100], these systems often still rely on BLE beacons, underscoring a critical need for uninstrumented, scalable semantic search.

Manipulation Guidance and Social Dynamics: Once a user is in the vicinity of a product, manipulation guidance helps them find regions of interest [21, 113]. Crowdsourced assistance

like Be My Eyes [36] or VizWiz [19] relies on human availability. Some researchers have focused on the product retrieval sub-task by providing verbal manipulation guidance for product retrieval off the shelf [4, 7]. However, these systems inherently assume the user is already perfectly localized in front of the correct shelf [139]. GIST bridges this gap by creating intermediate spatial knowledge that focuses on solving the prerequisite locomotor navigation problem by searching for desired goals and estimating the current pose of the system.

7.3 Multimodal Knowledge Extraction

Our offline pipeline converts consumer mobile RGB-D data into a reusable semantic topology without manual annotation.

A semantic topology of an environment is a tuple $T = (G, A, \Pi)$ where: $G = (V, E)$ is a traversability graph embedded in the metric frame of a 2D occupancy map $\mathcal{M}$. Nodes V are junctions and turn points (heading change $> \theta_{\text{turn}}$) of the morphological skeleton of free space; edges E connect nodes and are certified traversable by line-of-sight checks against $\mathcal{M}$ at full resolution.

$A = \{(x_i, \ell_i, \phi_i)\}$ is a set of semantic anchors: metric positions $x_i \in \mathbb{R}^2$ obtained by depth reprojection and shelf-boundary refinement, each carrying an open-vocabulary label ℓ_i (name, brand, category, packaging) produced by VLM annotation of a quality-filtered image crop, and a pointer ϕ_i to that source keyframe.

$\Pi : A \to E$ is a binding operator mapping each anchor to the perpendicular projection of x_i onto its nearest edge. Π induces, for any query, a virtual node on the graph at the projection point, and assigns each anchor a side (left/right of edge heading via cross-product). Π is what makes semantics routable since every anchor is a reachable goal and an orientable landmark.

T is defined by three properties that neighboring representations lack jointly. (1) Deterministic geometry with generative semantics: all traversability and distance facts derive from $G \subset \mathcal{M}$ and are never produced by a language model, bounding the physical consequences of semantic hallucination. (2) Image-grounded anchors: unlike 3D scene graphs [14, 51, 96], which serialize observations into discrete nodes, relations, and attributes before language-model consumption, anchors in A pre-

serve alignment with the multimodal priors of modern VLMs via ϕ_i. (3) Route-legible structure: unlike metric occupancy maps or dense feature maps, G directly encodes junctions, turns, and edge headings from which egocentric instructions (e.g., "second aisle on your right") are computed rather than inferred. Whereas scene graphs optimize for relational queries and task planning, semantic topology optimizes for spatial communication (localization, search, and route language) in a shared frame.

7.3.1 Mobile Data to 2D Occupancy

We captured a real-world, 3,500 sq ft international grocery store using a consumer mobile device equipped with LiDAR. A 14.5-minute traversal yielded 52,006 RGB-D frames, which were subsampled 6× to 8,668 frames (∼10 fps) alongside 6-DoF ARKit visual-inertial odometry. To ensure lightweight processing and downstream scalability, we explicitly avoid computationally heavy classical SLAM backends. Instead, we project a 2D occupancy grid (0.05 m/pixel) by applying a height-slice to the raw mobile point cloud.

7.3.2 Informative Keyframe & Object Extraction

To reduce multimodal input processing and annotation cost, we first perform keyframe selection. Each full RGB frame is embedded via DINOv3 (`vitb16`, 768-dim). A sequential cosine filter retains only frames whose similarity to the previously accepted keyframe falls below a 0.85 threshold, condensing the sequence to 660 visually distinct keyframes (a 92.4% frame reduction) to eliminate redundancy.

Each keyframe is processed by a YOLOv9 model fine-tuned on the SKU-110K dataset [45] for dense shelf products. Because tightly-packed retail shelves generate an extreme density of bounding boxes for repetitive products, we filter the detections to maintain only crops that maximize image quality and semantic diversity. Specifically, we compute the Laplacian variance (sharpness) for every crop and employ a **Feature-Space Strategy**: from each frame, we select the absolute sharpest crop alongside the crop exhibiting the maximum DINOv3 cosine distance to ensure maximum visual

diversity. Finally, fresh produce (fruits and vegetables), which are rarely placed on traditional shelves, are extracted via a secondary pre-trained YOLOv9-COCO pass.

7.3.3 Semantic Map

To optimize VLM inference throughput and rate-limiting, these representative crops are arranged into 5×5 mosaic grid images (39 grids total). These mosaics are processed by a unified VLM (`gemini-3-flash-preview`) to extract the product's `name`, `brand`, `packaging_type`, and `category`. The VLM is also tasked to classify the crop as blurry or sharp. We only use the sharp crops for downstream tasks (such as localization, zone classification, and as landmarks for natural language generation) that are sensitive to classification errors.

Object Position Refinement: Object map positions are computed via depth-based unprojection, mapping the bounding box center through the synchronized depth map into the ARKit world coordinate frame. Specifically, given the homogeneous pixel coordinates $\tilde{\boldsymbol{u}} = [u, v, 1]^\top$ at the bounding box's **median** depth Z_c, the map position $\boldsymbol{X}_w$ is calculated as:

$$\boldsymbol{X}_w = \boldsymbol{t}_{wc} + Z_c \boldsymbol{R}_{wc} \boldsymbol{K}^{-1} \tilde{\boldsymbol{u}} \tag{7.1}$$

where $\boldsymbol{K}$ represents the camera intrinsic matrix, and $\boldsymbol{R}_{wc}$ and $\boldsymbol{t}_{wc}$ denote the camera-to-world rotation and translation, respectively. Because consumer depth sensors frequently suffer from noise near shelf edges due to quasi-static products, these raw unprojected points occasionally fall inside structural obstacles. To correct this, we implement Bresenham's line algorithm [64] along the camera line-of-sight to drag-or-pull the object by casting a ray from the camera origin through the product toward the nearest occupied pixel and free space boundary (the shelf wall). This refinement snaps products to plausible shelf-adjacent locations, improving usefulness for search and route planning.

7.3.4 Topology

We construct a navigation graph via morphological skeletonization of the occupancy map [134]. Skeleton pixels form a pixel-level graph from which we extract junctions (pixel degree ≥ 3). Turn points are detected using a sliding-window angular change detector (turn threshold=60°). Junctions and turns within a small threshold radius are merged via hierarchical clustering, and short dead-end spurs are pruned. Bresenham line-of-sight checks on the high-resolution occupancy grid ensure all edges are traversable. Non-traversable edges are removed. To bind semantic products to the geometric topology graph, we compute the perpendicular projection of the product coordinate onto the nearest graph edge. We dynamically insert a **virtual node** at this precise projection point during pathfinding. This reduces reach distance dramatically, ensuring the extracted A* path travels deep into the correct aisle and arrives within 0.1–0.5 m of the physical target. This also allows our system to figure out the side of aisle the product is on.

7.4 Downstream Human-AI Tasks

The extracted semantic topology serves as a shared foundation for multiple downstream interaction tasks. To ensure architectural simplicity and consistency, all generative reasoning tasks are powered by the same `gemini-3-flash-preview` foundation model.

7.4.1 Zone Classification

To provide high-level semantic context, all discovered products are classified into 18 specific zones (e.g., `Lentils`, `Spices`) via a single VLM prompt analyzing the unique string metadata. A KD-Tree is constructed from all mapped product coordinates. For every free-space pixel in the occupancy map, an inverse-distance-squared weighted vote of the $k = 5$ nearest products assigns a continuous semantic zone overlay to the walkable floor plan (Figure 7.3).

This continuous overlay exposes higher level semantic organization not explicitly encoded in the floor plan. For instance, the system generated a dense clustering of heterogeneous categories

near the top-left aisle region. Ground-truth verification of the raw frames confirmed this region corresponds to the store's dedicated "Organic" section. This demonstrates the architecture's capacity to autonomously discover emergent macro-structures from micro-level semantic distributions.

7.4.2 Intent-Aware Search & Category Estimation

Standard keyword matching is often inadequate for long-tail, culturally unique, and dense inventory where spelling variations and linguistic substitutions are common. To resolve this, we

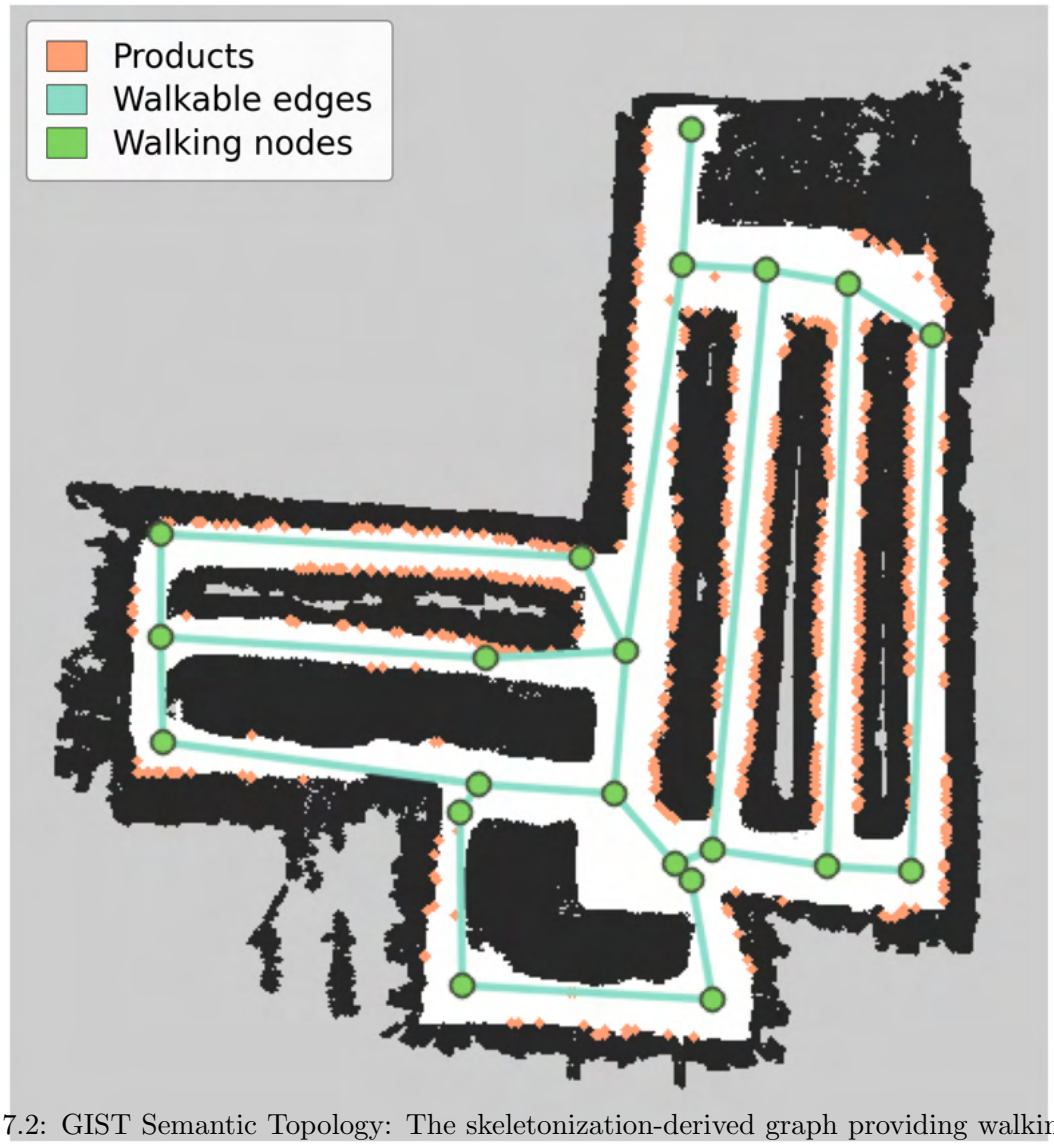


Figure 7.2: GIST Semantic Topology: The skeletonization-derived graph providing walking nodes (green) and traversable edges (teal), overlaid with localized products (orange).

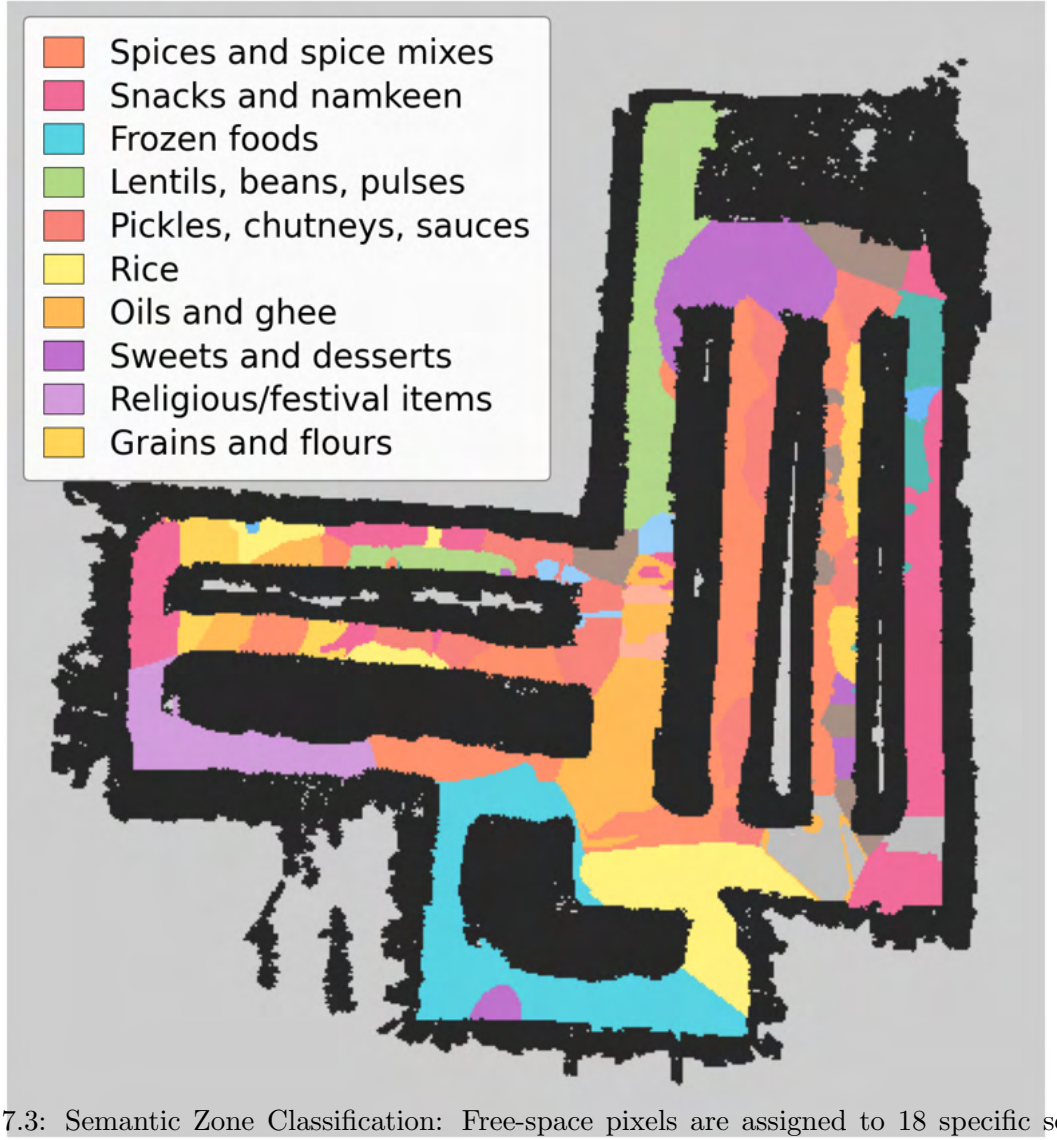


Figure 7.3: Semantic Zone Classification: Free-space pixels are assigned to 18 specific semantic zones via KD-tree voting over the localized product positions (legend displays the top 10 zones by area).

leverage the VLM to perform intent-aware semantic matching against our structured product map, deployed via an interactive web interface. As shown in Figure 7.4 (Right), abstract queries like "vegan protein" return not just exact matches, but items categorized as `Alternative` or `Related`, accompanied by generative reasoning explaining the selection. The system explicitly predicts the encompassing semantic zone (e.g., `Lentils, beans, and pulses`). By calculating the spatial median of the largest contiguous pixel cluster for that zone on the semantic map, it routes the user

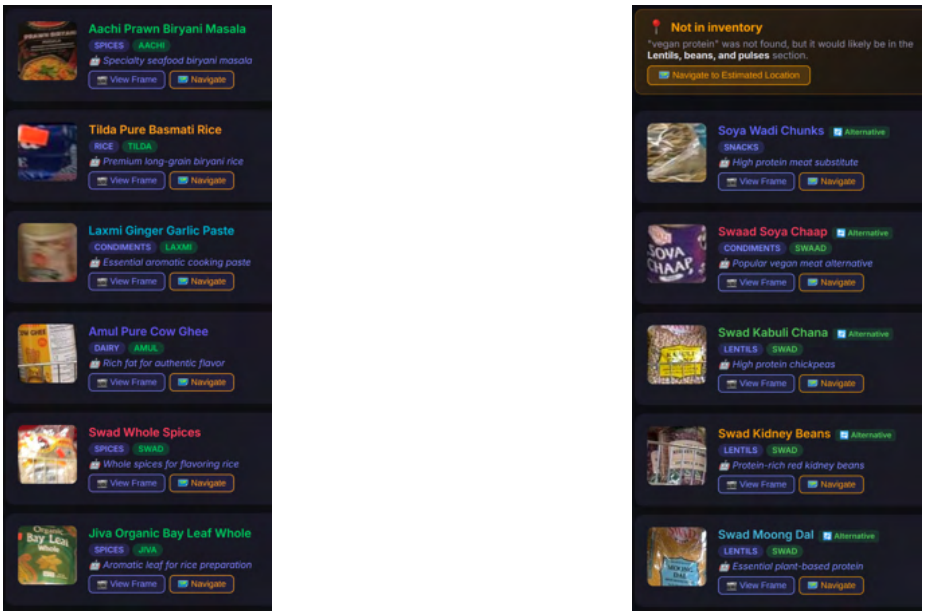


Figure 7.4: Intent-aware search via the web interface. **Left:** A multi-item recipe query ("biryani ingredients") identifies distinct ingredients. **Right:** An abstract query ("vegan protein") demonstrating how the VLM infers alternative semantic categories and zones most likely to contain user query products.

to the correct general aisle rather than failing outright if a specific product is unmapped.

This graceful fallback via **Category Estimation** is critical in practice because physical perception via mobile scanning inherently suffers from field-of-view (FOV) clipping (e.g., bottom-shelf items obscured by narrow aisles) and dynamic occlusions during the initial mapping phase. Furthermore, this intent-aware architecture natively supports complex, multi-goal queries. As seen in the "biryani ingredients" query (Figure 7.4, Left), the VLM effectively decomposes a prepared dish into its key distinct components—biryani spices, basmati rice, garlic paste, and ghee—and simultaneously locates or estimates the topological zones for all required items, laying the groundwork for optimized multi-stop path planning.

7.4.3 One-Shot Text-Based Semantic Localizer

To study whether the semantic layer can provide coarse global pose hypotheses, we construct a one-shot text-based localizer (Algorithm 5). Offline, the free-space occupancy grid is discretized into 0.5 m cells and eight orientation bins. For each pose, 20 camera rays are cast against the semantic map; visible anchor labels are deduplicated, alphabetically sorted, concatenated, and encoded with Qwen3-Embedding-0.6B. These normalized embeddings form a pose-indexed semantic map $\mathcal{M}_{text}$.

At query time, YOLOv9 crops from a single RGB image are labeled by the VLM and embedded with the same encoder. Cosine similarity returns ranked pose hypotheses. In addition to Top-1, we form a spatially coherent local mode around the highest-ranked hypothesis and report the unweighted mean of its candidate positions as the cluster-mean estimate. The hypothesis count and clustering radius are listed in Table 7.1. The cluster mean provides a pose proposal for downstream fusion with odometry and depth.

Algorithm 5: One-Shot Text-Based Semantic Localization

Input: Query RGB image I_q, precomputed pose map $\mathcal{M}_{text}$, hypothesis count K, cluster radius r_c

Output: Top-1 pose H_1 and cluster-mean position $\bar{\boldsymbol{x}}$

$C_{crops} \leftarrow \text{YOLOv9}(I_q)$

$L_q \leftarrow \text{SortAlphabetically}(\text{Unique}(\text{VLM}(C_{crops})))$

$E_q \leftarrow \text{QwenEmbed}(L_q)$

$\mathcal{S} \leftarrow \emptyset$

for each pose p_i **and cached embedding** $E_i \in \mathcal{M}_{text}$ **do**

  $\mathcal{S} \leftarrow \mathcal{S} \cup \{(p_i, \text{CosineSimilarity}(E_q, E_i))\}$

$H_1 \leftarrow \arg\max(\mathcal{S})$

$\mathcal{C} \leftarrow \{p_i \in \text{TopK}(\mathcal{S}, K) : \|\boldsymbol{x}_i - \boldsymbol{x}_{H_1}\|_2 \leq r_c\}$

$\bar{\boldsymbol{x}} \leftarrow |\mathcal{C}|^{-1} \sum_{p_i \in \mathcal{C}} \boldsymbol{x}_i$

return $(H_1, \bar{\boldsymbol{x}})$

7.4.4 Semantic-Topology Routing

We compute an A* path over G and serialize it into egocentric route text. Heading changes greater than 30° produce turn commands, collinear edges are merged into forward distances, and the total graph distance is reported explicitly. To obtain route-relevant landmarks, anchors near the path are tested for line of sight against $\mathcal{M}$, deduplicated by category, and serialized with category/name, distance along the route, and left/right side relative to the current edge heading.

These topology-derived route segments and occupancy-validated landmarks form the core routing prompt. Full GIST additionally supplies one rendered image containing the occupancy background, cyan topology graph, red A* path, start/goal markers, entrance label, and scale bar. The controlled ablation in Section 7.6.3 evaluates the image's incremental contribution while holding the structured text constant. By default, the prompt emphasizes body-relative left/right guidance, consistent with prior work on localized verbal wayfinding [41, 42]; cardinal cues could instead be exposed as a configurable user preference.

7.5 Evaluation Setup

The full suite of pipeline parameters enabling exact reproducibility is detailed in Table 7.1. To comprehensively evaluate the system, we defined 15 navigation scenarios across the 3,500 sq ft store (Table 7.2), ranging from simple to complex, multi-turn paths.

7.5.1 Baseline Configurations and Prompts

We compare Full GIST with a text-only GIST condition and two external prompting paradigms:

- **Full GIST**: one rendered image containing the occupancy background, topology graph, A* path, start/goal markers, entrance label, and scale bar, plus target identity, total graph distance, egocentric route segments, and line-of-sight-validated landmark text.
- **GIST Text**: the same target identity, total graph distance, route segments, and validated landmark text as Full GIST, but no image.

Table 7.1: Implementation & Reproducibility Parameters

Component	Hyperparameter / Configuration
Data Capture	Consumer LiDAR Smartphone (iPhone)
RGB-D Input	8,668 frames, 960×720 RGB, 256×192 Depth
2D Map	0.05m/px resolution
Keyframe Filter	DINOv3 (`vitb16-pretrain`), Cosine ≤ 0.85
Product Detection	YOLOv9 (SKU-110K), conf ≥ 0.25
Produce Detect.	YOLOv9 (COCO)
VLM Engine	`gemini-3-flash-preview` (Temp: 0.3)
VLM Batching	5×5 representative mosaic grids
Skeletonization	Zhang-Suen morphological thinning
Localization Encoder	Qwen3-Embedding-0.6B
Localization Grid	0.5m cells, 45° orientation bins, 60° FOV, 6m range, 20 rays
Cluster-Mean Estimate	$K = 100$ highest-ranking hypotheses, $r_c = 4\,\text{m}$ around Top-1, unweighted coordinate mean

Table 7.2: The 15 Benchmark Navigation Scenarios used for evaluation, spanning culturally specific product categories.

ID	Target Product	Dist.	Turns	ID	Target Product	Dist.	Turns
s01	Laxmi Chana Dal	2.9m	0	s14	Badshah Chatpat Masala	8.6m	3
s09	Swad Blackeye Peas	8.4m	2	s15	McVitie's Digestives	8.6m	3
s10	Mother's Recipe Pickle	4.7m	1	s04	Badshah Pav Bhaji	11.2m	0
s11	Patak's Hot Chili Pickle	13.7m	2	s05	Swad Soya Wadi	13.5m	1
s02	Aachi Biryani Masala	6.5m	1	s06	Swad Organic Urad Dal	14.2m	1
s12	Pani Puri Kit	6.5m	2	s07	Guntur Chilli Powder	17.1m	2
s03	Haldiram's Moong Dal	8.6m	1	s08	Deep Aloo Paratha	19.4m	2
s13	Parle Krackjack	8.6m	3				

- **NavComposer-style [53]**: up to six first-person RGB keyframes selected from the future A* trajectory (start, turn points, and final frame), paired with walk/turn/stop actions and the target product. It receives no occupancy map, topology graph, metric route segments, or validated landmark list.

- **Naive Gemini**: target product, store dimensions, entrance start description, straight-line distance, coarse allocentric direction, and optimal path length, but no image, topology, route segments, landmarks, zones, or obstacle information.

NavComposer follows its native input protocol, with images and actions sampled along the

route before instruction generation. The controlled component ablation below holds the generation pipeline and scenarios fixed while varying GIST inputs.

7.6 Results and Discussion

7.6.1 Semantic Localization Accuracy

We evaluated the semantic localizer on 20 curated shelf-facing frames from the mapping traversal, using ARKit translation as ground truth. Table 7.3 reports Top-1 and the runtime-available local cluster-mean estimator.

Table 7.3: Semantic localization translation error (mean ± population SD). The 14-frame subset contains the cases whose top-ranked semantic mode was manually identified as lying in the correct aisle or zone.

Estimator	**All 20 Frames**	**Correct Semantic Mode (14/20)**
Top-1 pose	5.31 ± 4.80 m	2.43 ± 1.11 m
Local cluster mean	4.79 ± 4.73 m	**1.92 ± 1.36 m**

The all-frame Top-1 result reflects the effect of semantic aliasing. On the 14 frames whose retrieved semantic mode lies in the correct aisle or zone, Top-1 error is 2.43 m and the local cluster mean reduces the error to 1.92 m. The cluster mean is the unweighted centroid of the spatially coherent high-ranking hypotheses around Top-1; its exact hyperparameters are reported in Table 7.1. Because this subset is identified after checking the retrieved semantic mode, these values characterize local precision after coarse semantic disambiguation.

Qualitative analysis reveals mirrored or repeated viewpoints with nearly identical visible product semantics (Figure 7.5). The ranked semantic hypotheses can initialize a particle filter or another temporal estimator that fuses them with odometry or depth.

7.6.2 Independent Multi-Criteria LLM Evaluation

Reference-based metrics such as BLEU and ROUGE are poorly aligned with the properties that matter for grounded route following [136]. We therefore use the *LLM-as-a-judge* paradigm [50,

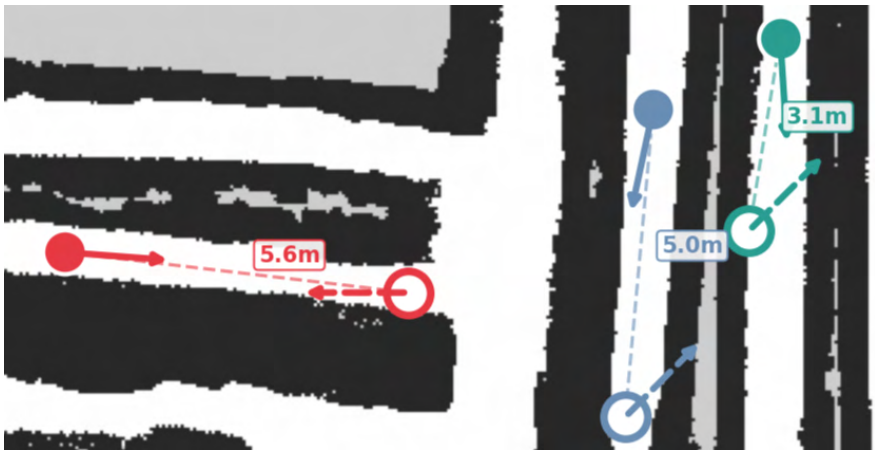


Figure 7.5: Semantic aliasing in localization. Ground-truth poses (solid) and Top-1 predicted poses (hollow) can be physically separated while observing similar semantic targets from mirrored viewpoints.

138] with two held-out model families, GPT-5.5 and Claude Opus 4.8, to score the same 15 scenarios on five higher-is-better criteria:

- **Egocentric Clarity:** Ensuring body-relative directions (left/right) rather than allocentric global frames [63].
- **Landmark Utility:** Strategic use of structural, permanent semantics rather than transient clutter [107].
- **Cognitive Load:** Synthesizing instructions into concise, logically chunked memory blocks [109].
- **Safety & Completeness:** Unambiguously specifying turns and defining definitive termination states [76].
- **View-Independent Guidance:** Creating view-independent directions that do not rely on visual pointing UIs [44].

Scores are averaged across judges within each scenario. Error bars are 95% nonparametric bootstrap confidence intervals over the 15 paired scenarios, and comparisons use two-sided paired Wilcoxon signed-rank tests. A bracket is displayed only when both judges are individually significant in the same direction; its stars use the weaker of the two judge-specific p values.

Full GIST scores 3.93 on the five-criterion average, compared with 3.85 for GIST Text, 3.14 for

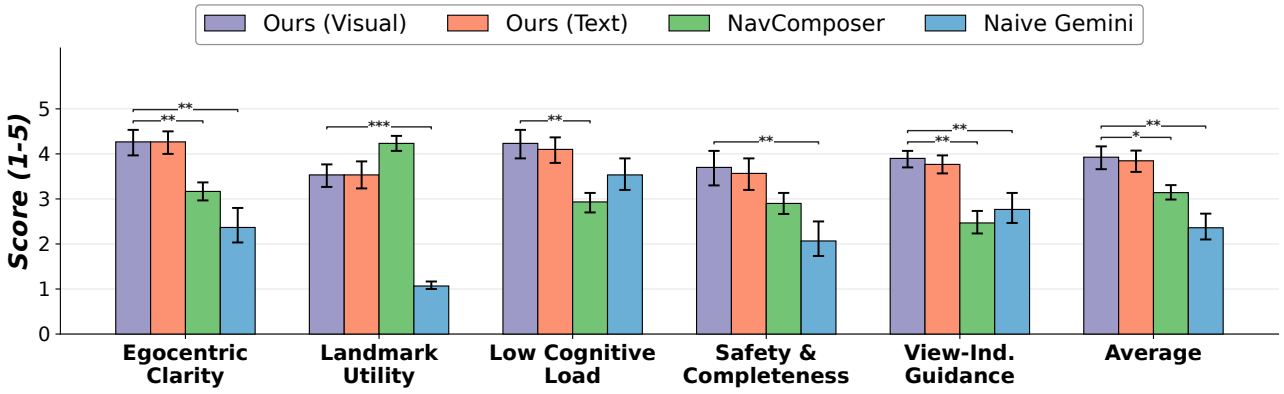


Figure 7.6: Held-out route instruction evaluation over 15 scenarios. For each scenario, the scores from GPT-5.5 and Claude Opus 4.8 are averaged. Bars show the mean across scenarios, and higher scores are better for all criteria. Error bars show 95% confidence intervals. Brackets appear only when two-sided paired Wilcoxon signed-rank tests are significant for both judges in the same direction, stars use the weaker judge-specific significance level ($*p < .05$, $**p < .01$, $***p < .001$).

NavComposer, and 2.36 for Naive Gemini. Full GIST and GIST Text are not significantly different ($\Delta = +0.08$, $p = .260$), showing that total route distance, egocentric route segments, and line-of-sight landmark text carry most of the measured guidance value. Full GIST exceeds NavComposer on average with replicated judge significance ($\Delta = +0.79$, weaker $p = .012$), including replicated gains in egocentric clarity, low cognitive load, and view-independent guidance. It also exceeds Naive Gemini on average ($\Delta = +1.57$, weaker $p = .002$), with replicated gains in egocentric clarity, landmark utility, safety and completeness, and view-independent guidance. NavComposer has higher Landmark Utility in the joint scores; this effect is not replicated across both judges, and its future-path RGB sequence provides direct access to visible landmarks.

7.6.3 Controlled Component Ablation

To evaluate the contribution of individual routing inputs, we generated fresh instructions for seven controlled removals from Full GIST. Each condition uses the same 15 scenarios and is evaluated by held-out GPT-5.5 and Claude Opus 4.8 judges. Table 7.5 reports the score drop from Full GIST after averaging all five criteria and both judges.

Table 7.4: Representative instructions for two scenarios. NavComposer receives future-path RGB keyframes and can reference visible scene content; GIST derives route structure and validated semantic landmarks from the shared topology and occupancy map.

Method	Mother's Recipe Pickle	Swad Blackeye Peas
Full GIST	Walk straight ahead for about 3 meters and then turn left into the first aisle. Continue for 2 meters, passing the Canned Goods on your right side. The Mother's Recipe Pickle is located just ahead.	Walk straight for 3 meters, turn left and walk for 2 meters, passing the Canned Goods on your right. Turn right into the aisle and walk 4 meters to find the Swad Blackeye Peas on the right shelf.
NavComposer [53]	Walk forward past the shelves of bagged lentils and beans until you reach the shopping cart. Turn left toward the yellow wall, and you will find the Mother's Recipe Pickle on the shelving unit to your right.	Walk forward past the shelves of jars and pickles until you reach the intersection with the shopping cart and boxes of red cup noodles. Turn left and then take an immediate right into the aisle with bags of beans and lentils, passing the decorative statues on the top shelf. Continue forward to find the Swad Blackeye Peas located on the lower shelves.
Naive Gemini	Walk straight ahead from the entrance for about five meters. You will find the Mother's Recipe Pickle directly in front of you.	Walk straight ahead into the store, then turn right and take an immediate left to find the Swad Blackeye Peas.

Table 7.5: Controlled component ablation for GIST route generation. Score drops are measured relative to Full GIST over 15 scenarios. The 95% confidence intervals are nonparametric bootstrap intervals. A p value is reported only when both judges are individually significant in the same direction; the weaker judge-specific value is shown.

Removed from Full GIST	**Score Drop**	**95% CI**	p
Route Text + Landmark Text	1.45	[1.11, 1.79]	.001
Route Text + Topology/Path Overlay Image	1.30	[0.95, 1.62]	.001
Landmark Text	1.07	[0.92, 1.23]	$< .001$
Route Text	0.69	[0.32, 1.15]	.018
Occupancy Map Image	0.43	[0.05, 0.99]	n.s.
Full Visual Image	0.14	[−0.01, 0.31]	n.s.
Topology/Path Overlay Image	0.11	[0.01, 0.21]	n.s.

Removing route text, landmark text, and computed distance while retaining the full image produces the largest drop ($\Delta = 1.45$). Removing route text together with the topology/path overlay also produces a large drop ($\Delta = 1.30$). Landmark text and route text each contribute independently, with drops of 1.07 and 0.69, respectively.

Removing the full rendered image while retaining target identity, route distance, route segments, and validated landmark text produces a smaller, non-significant drop ($\Delta = 0.14$). The aggregate effects of removing only the topology/path overlay or only the occupancy background are also not replicated across both judges. The occupancy background nevertheless improves Landmark Utility ($\Delta = 0.67$, weaker $p = .042$). These results identify the structured route and landmark information as the largest contributors to the aggregate score; topology and occupancy also provide the upstream computations used to generate that information.

7.6.4 Preliminary Pilot Evaluation

We conducted a formative in-situ pilot with five sighted graduate-student evaluators aged 21–35 in the target grocery store. Each evaluator completed two navigation tasks from the entrance. The application's visual route and product overlays were disabled so that participants relied on semantic search results and generated verbal instructions.

Table 7.6: Preliminary pilot outcomes ($N = 5$, two trials per participant).

Evaluator	**Target Product**	**Time (s)**	**Outcome**
E1	Dahi	75	Success
	Marigold Biscuit	35	Success
E2	Parle G	47	Success
	Schezwan Sauce	32	Success
E3	Okra	26	Success
	Ghee	93	Success (1 wrong turn)
E4	Ragi	N/A	Failed (FOV clip)
	Gud (Jaggery)	N/A	Failed (FOV clip)
E5	Rajma	15	Success
	Frozen Chapatti	64	Success (1 wrong turn)

Eight of ten trials succeeded (80%; exact 95% binomial CI [44.4%, 97.5%]), with a mean

completion time of 48.4 s among successful trials. Success was self-reported upon reaching the physical destination. The two failures involved bottom-shelf items that were outside the chest-level scan's field of view. Participants also noted that the validity of left/right language depends on matching the assumed starting orientation.

This small pilot provides initial evidence that sighted evaluators could often follow the generated verbal guidance in this store. It did not include a no-system condition or Blind/Low-Vision participants; larger studies with target users are needed to assess usability and accessibility.

7.7 Discussion and Limitations

Representation and environment scope. GIST is evaluated in one 3,500 sq. ft. grocery store using an offline representation of a quasi-static environment. Future work will examine online semantic updates and transfer to additional domains.

Perception and localization. Chest-level scanning can miss extreme top- or bottom-shelf items. Repeated or mirrored semantics produce large all-frame localization errors, and the 14-frame correct-semantic-mode subset was identified manually. The localizer is best suited as a coarse initializer or proposal distribution for temporal geometric fusion.

Routing evaluation. Held-out LLM judges provide controlled paired comparisons of instruction quality. Human route-following studies and route-grounded metrics are needed to assess physical execution and safety. NavComposer receives future-path images and actions, which should be considered when interpreting the baseline comparison.

Human evaluation and responsibility. The pilot contains five sighted students, ten trials. The current pipeline also uses cloud foundation models for parts of annotation and generation, creating privacy, latency, and cost concerns. Larger participatory studies with Blind/Low-Vision users and explicit uncertainty, privacy, and failure-recovery design are future work.

7.8 Conclusion

GIST demonstrates how one semantic-topological intermediate representation can support intent-aware search, zone inference, coarse semantic localization, and route-language generation in a dense grocery-store environment. The controlled ablation identifies topology-derived route structure and occupancy-validated landmark text as the largest measured contributors. The rendered image complements this scaffold as a compact, interpretable grounding interface.

Chapter 8

Conclusion and Future Work

8.1 Summary of Contributions

This dissertation developed context-aware embodied AI methods for human-centered environments. The central claim is that practical guidance and autonomy require understanding of environmental context. A robot or assistive device needs to reason about social cues, object semantics, human behavior, language-rich scene observations, and spatial structure in order to translate these systems from lab to real-world settings. The thesis starts with assistive guidance as an important testbed, then studies semantic localization as an enabling layer for autonomy, and finally studies rich environmental representations for key human-AI interaction tasks.

8.1.1 Chapter-wise contributions

Chapter 3 presented a perceptive robotic cane for socially aware seat selection. The system used environmental perception and social preference models to guide a user toward seats that better matched privacy, intimacy, and convenience criteria. This chapter showed that even a common goal-centric navigation task can depend on social context. The task is not only to find an empty chair but a chair that fits the social requirements.

Chapter 4 presented ShelfHelp, a robotic cane system for object retrieval. It showed that product localization and spatial verbal command planning can support fine-grained grasp guidance in cluttered grocery shelves. It also showed that product retrieval is a complicated computer vision problem and human hand movement models can help form optimal policies to guide humans to

acquire desired objects in cluttered scenes.

Chapter 5 presented ShelfAware, a real-time semantic localization method for quasi-static environments. ShelfAware modeled shelf semantics as distributional evidence and used inverse semantic proposals inside Monte Carlo Localization. The chapter showed that repeated geometry in grocery stores can be disambiguated by noisy semantic patterns. This addresses a prerequisite for assistive guidance near a known shelf: the system must know that the user is in front of the correct shelf before fine-grained retrieval can begin.

Chapter 6 presented VLM-GLoc, which extended the localization research with language-rich observations from vision-language models. Instead of relying only on fixed object classes, VLM-GLoc used natural-language scene descriptions to support global localization in cluttered indoor environments. This chapter showed that VLMs can be used as semantic observation generators inside a probabilistic localization system that can generate rich, discriminative descriptions, characterize object permanence, and provide quality filtering essential for robust localization from grocery stores to office spaces.

Chapter 7 presented GIST, which transforms multimodal semantic annotations into a shared semantic-topology representation. GIST supported intent-aware search, zone classification, one-shot semantic localization, and natural language routing. This chapter explores how rich environmental representations can support key downstream human-AI interaction tasks, from search, to situational awareness, to guidance.

Together, these chapters support the thesis statement. Context-aware embodied AI systems can use implicit environmental cues to produce more useful guidance and more reliable spatial grounding in complex human-centered environments.

8.2 Themes Across the Dissertation

Assistive guidance exposes the need for context. Chapters 3 and 4 use assistive technology as a testbed because many everyday environment have contexts that are only accessible visually. In Chapter 3, success was not merely reaching a chair but reaching a socially preferred

chair. In Chapter 4, success was not only detecting a product. It was helping the user move their hand to acquire that object optimally. These systems show that task success should be defined using the context of the user, the scene, and the interaction.

Semantic localization is an important prerequisite for both assistive guidance and autonomy. The assistive guidance systems in the early chapters assume that the user or device is already near the relevant goal region. That assumption does not hold when the system must work across a full store, office, lab, or campus and anytime the system is turned on. Chapters 5 and 6 address this gap by treating semantic evidence as a source of global localization information. ShelfAware shows that distributional shelf semantics can help disambiguate repeated grocery aisles. VLM-GLoc shows that language-rich observations can provide a more robust and generalizable semantic paradigm across grocery and office environments. In both cases, semantics are not used as fixed landmarks. They are uncertain observations that can be fused with motion and geometry inside a probabilistic localization system.

Rich environment representations supports interaction. Chapter 7 shows that a rich map should support more than localization. It should also help a system search for goals, identify semantic zones, and describe routes in a way that people can use. This theme also appears in the earlier chapters. The cane must communicate a selected seat through feedback. ShelfHelp must translate product coordinates into verbal hand guidance. GIST makes this problem explicit by grounding natural language route instructions in topology, landmarks, and metric structure.

These three themes give a common reading of the dissertation. The early chapters use assistive guidance to show why context matters. The middle chapters build semantic localization methods that make context usable for autonomy in repeated and changing spaces. The final chapter shows how rich environmental context can support important human-centric tasks: search, situational awareness, localization, and guidance.

8.3 Practical Lessons from Real-World System Development

Beyond the formal results, building and deploying the systems in this dissertation produced several practical lessons that are rarely reported in individual papers. Table 8.1 summarizes observations from prototyping, field deployments, and informal design feedback. They are intended as engineering guidance for researchers and students developing related embodied-AI and assistive-technology systems.

Table 8.1: Practical lessons from developing and deploying the systems in this dissertation.

Practical lesson	Implication for system design
Weight balance determines usability.	Mounting sensors on a cane created an unbalanced load and led to rapid fatigue during prototype use in some cases. Informal design conversations with blind users and prototype trials with blindfolded sighted participants reinforced the importance of balance. Sensors should be kept light and close to the hand or body, with batteries and compute moved off the cane when possible.
Sensor orientation constrains mapping.	Reliable mapping and semantic projection required the RGB-D sensors to remain approximately upright. This was difficult with handheld systems, easier with wearable mounts, and easiest with cart-mounted systems. Form factor, mounting, and frame-quality checks should therefore be designed together with the mapping pipeline. There are recent works emerging that correct for non-upright orientation.
Consumer LiDAR phones reduce integration cost.	LiDAR-equipped iPhone Pro models combined usable RGB-D sensing, ARKit visual-inertial odometry, onboard compute, battery, storage, and wireless communication in one comparatively inexpensive device. For rapid prototyping and data collection, this substantially reduced integration effort relative to custom robotic sensing suites.
More sensor data is not always better.	High-end LiDARs produced data rates that were difficult to justify for assistive devices and compute-limited mobile robots. In these settings, cellphone-class sensors and Intel RealSense cameras often provided a better trade-off among resolution, compute, power, bandwidth, and cost. Sensor selection should be driven by the end-to-end task rather than maximum raw data density.
2D semantic maps are often sufficient.	Many indoor navigation tasks ultimately plan motion on a horizontal plane. Retaining 3D information for perception and reprojection while maintaining a 2D occupancy and semantic map for localization, planning, and communication reduced computational and representational complexity.
Avoid IMU-only or gait-based odometry.	In the systems evaluated here, IMU preintegration accumulated substantially more drift than visual odometry or visual-inertial odometry. Estimating odometry directly from human walking motion was also difficult to calibrate and insufficiently accurate. IMU and gait cues are therefore better treated as auxiliary signals than as the sole odometry source.
Glass remains a sensing blind spot.	None of the depth cameras or LiDARs tested reliably detected glass obstructions. A missing depth return should not be interpreted as free space. Glass-heavy environments require complementary sensing, prior annotations, conservative planning, or explicit communication of uncertainty.
Benchmark SLAM backends in-domain.	For the indoor domains and sensor configurations used in this dissertation, SLAM Toolbox produced the most reliable 2D occupancy maps among the systems evaluated.

8.4 Limitations

The early assistive studies relied on blindfolded participants and pilot-scale evaluation. These studies were useful for early feasibility testing, but they do not replace studies with PVI participants. Future evaluations should study trust, comfort, fatigue, safety, social comfort, privacy, cognitive load, and preference in more realistic tasks.

ShelfAware and VLM-GLoc were evaluated as localization systems, not as complete end-to-end assistive navigation stacks. Their results show that semantic evidence can help recover pose in challenging environments, but a deployed system would still need route planning, user feedback, local obstacle handling, failure recovery, and safe interaction design.

The systems also rely on available semantic evidence. Performance can degrade in visually sparse areas, under occlusion, when shelves or objects fall outside the camera field of view, or when semantic observations are too noisy. This was visible in the later systems where bottom-shelf field-of-view clipping and repeated semantic layouts created failure cases.

Vision-language models add rich descriptions, but current cloud-based models introduce latency, cost, and privacy concerns. On-device models may reduce these concerns, but they still need evaluation in real-time robot settings. The system should also avoid treating VLM output as a source of geometric safety. In this dissertation, geometry and semantics are kept separate where possible, but deployment will need stronger checks around hallucination, privacy, and failure and uncertainty communication.

Finally, GIST includes an LLM-based evaluation of route instructions, but target-user evaluation is still future work. The route language must be tested with the people who would rely on it, especially in settings where wrong instructions can increase stress or create unsafe behavior.

8.5 Future Work

Integrated assistive navigation. The most direct next step is an integrated assistive navigation system that combines localization, semantic representations, and guidance. ShelfAware or VLM-

GLoc could provide pose estimates. GIST could provide route structure and language generation. ShelfHelp-style interaction could then support final product retrieval near the target shelf. Such a system would test whether the separate capabilities in this dissertation work together in a full task.

Long-term map maintenance. Human-centered environments change gradually. Grocery stores, laboratories, offices, and hospitals keep much of their global layout, but local objects move, products are restocked, and temporary clutter appears. Future work should focus on updating the semantic maps algorithmically and optimally by tracking confidence over time, and deciding when new scans are needed. It should also distinguish between stable landmarks, useful but changing cues, and transient objects that should not be used for guidance.

On-device and privacy-aware multimodal models. Future systems should reduce dependence on cloud VLM calls. Smaller on-device models could reduce latency, cost, and privacy risk. They may also allow more frequent semantic updates during motion. Lower latency is only useful if the model output remains reliable enough for localization, search, and route communication.

Target-user evaluation and participatory design. The systems developed with the findings of this dissertation should be evaluated with PVI participants in realistic tasks. These studies should include participatory design so that goals, form factors, feedback modes, and privacy choices reflect user priorities.

Broader deployment beyond assistive robotics. The same context-aware spatial representations can support service robots in retail stores, warehouses, hospitals, offices, schools, and campuses. These robots also need to know which objects are useful, social norms for these places, and which semantic cues provide better grounding. The long-term goal is a robot that understands a space in the way people do to interact with other people.

Bibliography

[1] AMCL nav2 documentation. `https://docs.nav2.org/configuration/packages/configuring-amcl.html`. Accessed 2025-09-21.

[2] Amanda Adkins, Taijing Chen, and Joydeep Biswas. Probabilistic object maps for long-term robot localization. In 2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pages 931–938. IEEE, 2022.

[3] Mohamed Aghzal, Xiang Yue, Erion Plaku, and Ziyu Yao. Evaluating vision-language models as evaluators in path planning. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 6886–6897, 2025.

[4] Shivendra Agrawal. Assistive robotics for empowering humans with visual impairments to independently perform day-to-day tasks. In Proceedings of the 2023 International Conference on Autonomous Agents and Multiagent Systems, pages 3023–3025, 2023.

[5] Shivendra Agrawal, Jake Brawer, Ashutosh Naik, Alessandro Roncone, and Bradley Hayes. Shelfaware: Real-time visual-inertial semantic localization in quasi-static environments with low-cost sensors. IEEE Robotics and Automation Letters, 2026.

[6] Shivendra Agrawal and Bradley Hayes. Vlm-gloc: Vision-language model enhanced monte carlo localization for robust semantic global localization in cluttered quasi-static environments. arXiv preprint arXiv:2605.30506, 2026.

[7] Shivendra Agrawal, Suresh Nayak, Ashutosh Naik, and Bradley Hayes. ShelfHelp: Empowering humans to perform vision-independent manipulation tasks with a socially assistive robotic cane. In Proceedings of the 2023 International Conference on Autonomous Agents and Multiagent Systems, pages 1514–1523, 2023.

[8] Shivendra Agrawal, Mary Etta West, and Bradley Hayes. A novel perceptive robotic cane with haptic navigation for enabling vision-independent participation in the social dynamics of seat choice. In 2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pages 9156–9163. IEEE, 2022.

[9] Tousif Ahmed, Roberto Hoyle, Kay Connelly, David Crandall, and Apu Kapadia. Privacy concerns and behaviors of people with visual impairments. In Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems, pages 3523–3532, 2015.

[10] Basheer Al-Tawil, Thorsten Hempel, Ahmed Abdelrahman, and Ayoub Al-Hamadi. A review of visual slam for robotics: Evolution, properties, and future applications. Frontiers in Robotics and AI, 11:1347985, 2024.

[11] Pablo F Alcantarilla, José J Yebes, Javier Almazán, and Luis M Bergasa. On combining visual SLAM and dense scene flow to increase the robustness of localization and mapping in dynamic environments. In 2012 IEEE International Conference on Robotics and Automation. IEEE, 2012.

[12] Peter Anderson, Qi Wu, Damien Teney, Jake Bruce, Mark Johnson, Niko Sünderhauf, Ian Reid, Stephen Gould, and Anton 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, pages 3674–3683, 2018.

[13] Yiannoula Andreou and Konstantinos T Kotsis. The estimation of length, surface area, and volume by blind and sighted children. In International Congress Series, volume 1282, pages 780–784. Elsevier, 2005.

[14] Iro Armeni, Zhi-Yang He, JunYoung Gwak, Amir R Zamir, Martin Fischer, Jitendra Malik, and Silvio Savarese. 3d scene graph: A structure for unified semantics, 3d space, and camera. In Proceedings of the IEEE/CVF international conference on computer vision, pages 5664–5673, 2019.

[15] Anika Arora, Lucas Nadolskis, Michael Beyeler, and Misha Sra. Visionai-shopping assistance for people with vision impairments. In 2024 IEEE International Symposium on Mixed and Augmented Reality Adjunct (ISMAR-Adjunct), pages 377–378. IEEE, 2024.

[16] Shiri Azenkot, Catherine Feng, and Maya Cakmak. Enabling building service robots to guide blind people a participatory design approach. In 2016 11th ACM/IEEE International Conference on Human-Robot Interaction (HRI), pages 3–10. IEEE, 2016.

[17] Samuel Bateman, Kyle Harlow, and Christoffer Heckman. Better together: Online probabilistic clique change detection in 3d landmark-based maps. In 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pages 4878–4885. IEEE, 2020.

[18] Paul J Besl and Neil D McKay. Method for registration of 3-d shapes. In Sensor fusion IV: control paradigms and data structures, volume 1611, pages 586–606. Spie, 1992.

[19] Jeffrey P Bigham, Chandrika Jayant, Andrew Miller, Brandyn White, and Tom Yeh. Vizwiz: Locateit-enabling blind people to locate objects in their environment. In 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition-Workshops, pages 65–72. IEEE, 2010.

[20] Joydeep Biswas and Manuela Veloso. Depth camera based indoor mobile robot localization and navigation. In 2012 IEEE International Conference on Robotics and Automation, pages 1697–1702, May 2012. ISSN: 1050-4729.

[21] Mayara Bonani, Raquel Oliveira, Filipa Correia, André Rodrigues, Tiago Guerreiro, and Ana Paiva. What my eyes can't see, a robot can show me: Exploring the collaboration between blind people and robots. In Proceedings of the 20th International ACM SIGACCESS Conference on Computers and Accessibility, pages 15–27, 2018.

[22] Rupert Bourne, Jaimie D Steinmetz, Seth Flaxman, Paul Svitil Briant, Hugh R Taylor, Serge Resnikoff, Robert James Casson, Amir Abdoli, Eman Abu-Gharbieh, Ashkan Afshin, et al. Trends in prevalence of blindness and distance and near vision impairment over 30 years: an

analysis for the global burden of disease study. The Lancet global health, 9(2):e130–e143, 2021.

[23] Christine T Chang, Matthew B Luebbers, Mitchell Hebert, and Bradley Hayes. Human non-compliance with robot spatial ownership communicated via augmented reality: Implications for human-robot teaming safety. In 2023 IEEE International Conference on Robotics and Automation (ICRA), pages 9785–9792. IEEE, 2023.

[24] Haoye Chen, Yingzhi Zhang, Kailun Yang, Manuel Martinez, Karin Müller, and Rainer Stiefelhagen. Can we unify perception and localization in assisted navigation? an indoor semantic visual positioning system for visually impaired people. In International Conference on Computers Helping People with Special Needs, pages 97–104. Springer, 2020.

[25] Jiaqi Chen, Bingqian Lin, Ran Xu, Zhenhua Chai, Xiaodan Liang, and Kwan-Yee Wong. Mapgpt: Map-guided prompting with adaptive path planning for vision-and-language navigation. In Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 9796–9810, 2024.

[26] Qingtian Chen, Muhammad Khan, Christina Tsangouri, Christopher Yang, Bing Li, Jizhong Xiao, and Zhigang Zhu. Ccny smart cane. In 2017 IEEE 7th Annual International Conference on CYBER Technology in Automation, Control, and Intelligent Systems.

[27] Wei Chen, Yu Liu, Weiping Wang, Erwin Bakker, Theodoros Georgiou, Paul Fieguth, Li Liu, and Michael S Lew. Deep learning for instance retrieval: A survey. arXiv preprint arXiv:2101.11282, 2021.

[28] Roger W Cholewiak. The perception of tactile distance: Influences of body site, space, and time. Perception, 28(7):851–875, 1999. PMID: 10664778.

[29] Tzu-Kuan Chuang, Ni-Ching Lin, Jih-Shi Chen, Chen-Hao Hung, Yi-Wei Huang, Chunchih Teng, Haikun Huang, Lap-Fai Yu, Laura Giarré, and Hsueh-Cheng Wang. Deep trail-following robotic guide dog in pedestrian environments for people who are blind and visually impaired-learning from virtual and real worlds. In 2018 IEEE International Conference on Robotics and Automation (ICRA), pages 5849–5855. IEEE, 2018.

[30] Il Yong Chung, Sanghag Kim, and Kang Hyeon Rhee. The smart cane utilizing a smart phone for the visually impaired person. In 2014 IEEE 3rd Global Conference on Consumer Electronics (GCCE), pages 106–107, 2014.

[31] Ryan Crabb, Seyed Ali Cheraghi, and James M. Coughlan. A Lightweight Approach to Localization for Blind and Visually Impaired Travelers. Sensors, 23(5):2701, January 2023.

[32] David DeFazio, Eisuke Hirota, and Shiqi Zhang. Seeing-eye quadruped navigation with force responsive locomotion control, 09 2023.

[33] Konstantinos G Derpanis. Overview of the ransac algorithm. Image Rochester NY, 4(1):2–3, 2010.

[34] Siyan Dong, Shuzhe Wang, Shaohui Liu, Lulu Cai, Qingnan Fan, Juho Kannala, and Yanchao Yang. Reloc3r: Large-scale training of relative camera pose regression for generalizable, fast, and accurate visual localization. In Proceedings of the Computer Vision and Pattern Recognition Conference, 2025.

[35] Aline Darc dos Santos, Fausto Orsi Medola, Milton José Cinelli, Alejandro Rafael Garcia Ramirez, and Frode Eika Sandnes. Are electronic white canes better than traditional canes? a comparative study with blind and blindfolded participants. Universal Access in the Information Society, 20(1), 2020.

[36] Be My Eyes. Be my eyes, 2022.

[37] Chia-Hui Feng, Ju-Yen Hsieh, Yu-Hsiu Hung, Chung-Jen Chen, and Cheng-Hung Chen. Research on the visually impaired individuals shopping with artificial intelligence image recognition assistance. In International Conference on Human-Computer Interaction, pages 518–531. Springer, 2020.

[38] Dieter Fox. Kld-sampling: Adaptive particle filters. Advances in neural information processing systems, 14, 2001.

[39] Daniel Fried, Ronghang Hu, Volkan Cirik, Anna Rohrbach, Jacob Andreas, Louis-Philippe Morency, Taylor Berg-Kirkpatrick, Kate Saenko, Dan Klein, and Trevor Darrell. Speaker-follower models for vision-and-language navigation. Advances in neural information processing systems, 31, 2018.

[40] A Jin Fukasawa and Kazusihge Magatani. A navigation system for the visually impaired an intelligent white cane. In 2012 Annual International Conference of the IEEE Engineering in Medicine and Biology Society.

[41] Thomas Gallagher, Elyse Wise, Hoe Chee Yam, Binghao Li, Euan Ramsey-Stewart, Andrew G Dempster, and Chris Rizos. Indoor navigation for people who are blind or vision impaired: Where are we and where are we going? Journal of Location Based Services, 8(1):54–73, 2014.

[42] Florence Gaunet and Xavier Briffault. Exploring the functional specifications of a localized wayfinding verbal aid for blind pedestrians: Simple and structured urban areas. Human-Computer Interaction, 20(3):267–314, 2005.

[43] Chaitanya P Gharpure and Vladimir A Kulyukin. Robot-assisted shopping for the blind: issues in spatial cognition and product selection. Intelligent Service Robotics, 1(3):237–251, 2008.

[44] Nicholas A Giudice. Navigating without vision: Principles of blind spatial cognition. In Handbook of behavioral and cognitive geography, pages 260–288. Edward Elgar Publishing, 2018.

[45] Eran Goldman, Roei Herzig, Aviv Eisenschtat, Jacob Goldberger, and Tal Hassner. Precise detection in densely packed scenes. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pages 5227–5236, 2019.

[46] Reginald G Golledge, Roberta L Klatzky, and Jack M Loomis. Cognitive mapping and wayfinding by adults without vision. In The construction of cognitive maps, pages 215–246. Springer, 1996.

[47] Google. Lookout by google, 2022.

[48] Giorgio Grisetti, Cyrill Stachniss, and Wolfram Burgard. Improved techniques for grid mapping with rao-blackwellized particle filters. IEEE transactions on Robotics, 23(1):34–46, 2007.

[49] Tobias Groot and Matias Valdenegro-Toro. Overconfidence is key: Verbalized uncertainty evaluation in large language and vision-language models. In Proceedings of the 4th Workshop on Trustworthy Natural Language Processing (TrustNLP 2024), pages 145–171, 2024.

[50] Jiawei Gu, Xuhui Jiang, Zhichao Shi, Hexiang Tan, Xuehao Zhai, Chengjin Xu, Wei Li, Yinghan Shen, Shengjie Ma, and Honghao Liu. A survey on llm-as-a-judge. The Innovation, 2024.

[51] Qiao Gu, Ali Kuwajerwala, Sacha Morin, Krishna Murthy Jatavallabhula, Bipasha Sen, Aditya Agarwal, Corban Rivera, William Paul, Kirsty Ellis, Rama Chellappa, et al. Conceptgraphs: Open-vocabulary 3d scene graphs for perception and planning. In 2024 IEEE International Conference on Robotics and Automation (ICRA), pages 5021–5028. IEEE, 2024.

[52] João Guerreiro, Daisuke Sato, Saki Asakawa, Huixu Dong, Kris M. Kitani, and Chieko Asakawa. Cabot: Designing and evaluating an autonomous navigation robot for blind people. In The 21st ACM SIGACCESS Conference on Computers and Accessibility, NY, USA, 2019.

[53] Zongtao He, Liuyi Wang, Lu Chen, Chengju Liu, and Qijun Chen. Navcomposer: Composing language instructions for navigation trajectories through action-scene-object modularization. IEEE Transactions on Circuits and Systems for Video Technology, 2025.

[54] Rie Kamikubo, Hernisa Kacorri, and Chieko Asakawa. "we are at the mercy of others' opinion": Supporting blind people in recreational window shopping with ai-infused technology. In Proceedings of the 21st International Web for All Conference, pages 45–58, 2024.

[55] Rie Kamikubo, Seita Kayukawa, Yuka Kaniwa, Allan Wang, Hernisa Kacorri, Hironobu Takagi, and Chieko Asakawa. Beyond omakase: Designing shared control for navigation robots with blind people. In Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems.

[56] Shuhao Kang, Youqi Liao, Peijie Wang, Wenlong Liao, Qilin Zhang, Benjamin Busam, Xieyuanli Chen, and Yun Liu. Vlm-loc: Localization in point cloud maps via vision-language models. arXiv preprint arXiv:2603.09826, 2026.

[57] Yuka Kaniwa, Masaki Kuribayashi, Seita Kayukawa, Daisuke Sato, Hironobu Takagi, Chieko Asakawa, and Shigeo Morishima. Chitchatguide: How can a guidance system with large language models impact shopping mall experiences for people with visual impairments? In International Conference on Human-Computer Interaction with Mobile Devices and Services, 2024.

[58] Sertac Karaman and Emilio Frazzoli. Sampling-based algorithms for optimal motion planning. The international journal of robotics research, 30(7):846–894, 2011.

[59] Robert K. Katzschmann, Brandon Araki, and Daniela Rus. Safe local navigation for visually impaired users with a time-of-flight and haptic feedback device. IEEE Transactions on Neural Systems and Rehabilitation Engineering, 26(3):583–593, 2018.

[60] Seita Kayukawa, Keita Higuchi, João Guerreiro, Shigeo Morishima, Yoichi Sato, Kris Kitani, and Chieko Asakawa. Bbeep: A sonic collision avoidance system for blind travellers and nearby pedestrians. In Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems, pages 1–12, 2019.

[61] Seita Kayukawa, Daisuke Sato, Masayuki Murata, Tatsuya Ishihara, Hironobu Takagi, Shigeo Morishima, and Chieko Asakawa. Enhancing blind visitor's autonomy in a science museum using an autonomous navigation robot. In Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems, pages 1–14, 2023.

[62] Sung Yeon Kim and Kwangsu Cho. Usability and design guidelines of smart canes for users with visual impairments. international Journal of Design, 7(1), 2013.

[63] Roberta L Klatzky. Allocentric and egocentric spatial representations: Definitions, distinctions, and interconnections. In Spatial cognition: An interdisciplinary approach to representing and processing spatial knowledge, pages 1–17. Springer, 1998.

[64] P Koopman. Bresenham line-drawing algorithm. Forth Dimensions, 8(6):12–16, 1987.

[65] Jacob Krantz, Erik Wijmans, Arjun Majumdar, Dhruv Batra, and Stefan Lee. Beyond the nav-graph: Vision-and-language navigation in continuous environments. In European Conference on Computer Vision, pages 104–120. Springer, 2020.

[66] Haofei Kuang, Xieyuanli Chen, Tiziano Guadagnino, Nicky Zimmerman, Jens Behley, and Cyrill Stachniss. Ir-mcl: Implicit representation-based online global localization. IEEE Robotics and Automation Letters, 8(3):1627–1634, 2023.

[67] Vladimir Kulyukin, Chaitanya Gharpure, and John Nicholson. Robocart: Toward robot-assisted navigation of grocery stores by the visually impaired. In 2005 IEEE/RSJ International Conference on Intelligent Robots and Systems, pages 2845–2850. IEEE, 2005.

[68] Vladimir Kulyukin and Aliasgar Kutiyanawala. Accessible shopping systems for blind and visually impaired individuals: Design requirements and the state of the art. The Open Rehabilitation Journal, 3(1), 2010.

[69] Vladimir A Kulyukin and Chaitanya Gharpure. Ergonomics-for-one in a robotic shopping cart for the blind. In Proceedings of the 1st ACM SIGCHI/SIGART conference on Human-robot interaction, pages 142–149, 2006.

[70] Masaki Kuribayashi, Seita Kayukawa, Jayakorn Vongkulbhisal, Chieko Asakawa, Daisuke Sato, Hironobu Takagi, and Shigeo Morishima. Corridor-walker: Mobile indoor walking assistance for blind people to avoid obstacles and recognize intersections. Proc. ACM Hum.-Comput. Interact., 6(MHCI), sep 2022.

[71] Kyungjun Lee, Daisuke Sato, Saki Asakawa, Hernisa Kacorri, and Chieko Asakawa. Pedestrian detection with wearable cameras for the blind: A two-way perspective. In Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems, pages 1–12, 2020.

[72] Zhenyu Liao, Jose V. Salazar Luces, Ankit A. Ravankar, and Yasuhisa Hirata. Running guidance for visually impaired people using sensory augmentation technology based robotic system. IEEE Robotics and Automation Letters, 8(9):5323–5330, 2023.

[73] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In European conference on computer vision, pages 740–755. Springer, 2014.

[74] Xiaoye Liu, Zhenyu Zhang, Jim Peterson, and Shobhit Chandra. The effect of lidar data density on dem accuracy. In Proceedings of the 17th International Congress on Modelling and Simulation (MODSIM07). Modelling and Simulation Society of Australia and New Zealand, 2007.

[75] Thibaut Loiseau, Guillaume Bourmaud, and Vincent Lepetit. Alligat0r: Pre-training through co-visibility segmentation for relative camera pose regression. arXiv preprint arXiv:2503.07561, 2025.

[76] Jack M Loomis, Roberta L Klatzky, Reginald G Golledge, et al. Navigating without vision: basic and applied research. Optometry and vision science, 78(5):282–289, 2001.

[77] Diego López-de Ipiña, Tania Lorido, and Unai López. Blindshopping: enabling accessible shopping for visually impaired people through mobile technologies. In International Conference on Smart Homes and Health Telematics, pages 266–270. Springer, 2011.

[78] Kristin L Lovelace, Mary Hegarty, and Daniel R Montello. Elements of good route directions in familiar and unfamiliar environments. In International conference on spatial information theory, pages 65–82. Springer, 1999.

[79] Chen-Lung Lu, Zi-Yan Liu, Jui-Te Huang, Ching-I Huang, Bo-Hui Wang, Yi Chen, Nien-Hsin Wu, Hsueh-Cheng Wang, Laura Giarré, and Pei-Yi Kuo. Assistive navigation using deep reinforcement learning guiding robot with uwb/voice beacons and semantic feedbacks for blind and visually impaired people. Frontiers in Robotics and AI, 8:654132, 2021.

[80] Steve Macenski and Ivona Jambrecic. Slam toolbox: Slam for the dynamic world. Journal of Open Source Software, 6(61):2783, 2021.

[81] Roberto Manduchi and James M. Coughlan. The last meter: Blind visual guidance to a target. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '14, page 3113–3122, New York, NY, USA, 2014. Association for Computing Machinery.

[82] Rajesh Kannan Megalingam, Aparna Nambissan, Anu Thambi, Anjali Gopinath, and Megha Nandakumar. Sound and touch based smart cane: Better walking experience for visually challenged. In 2014 IEEE Canada International Humanitarian Technology Conference - (IHTC), pages 1–4, 2014.

[83] Vidula V. Meshram, Kailas Patil, Vishal A. Meshram, and Felix Che Shu. An astute assistive device for mobility and object recognition for visually impaired people. IEEE Trans. on Human-Machine Systems, 2019.

[84] Sharada Murali, R Shrivatsan, V Sreenivas, Srihaarika Vijjappu, S Joseph Gladwin, and R Rajavel. Smart walking cane for the visually challenged. In IEEE Region 10 Humanitarian Technology Conference, pages 1–4, 2016.

[85] Arshad Nasser, Kai-Ning Keng, and Kening Zhu. Thermalcane: Exploring thermotactile directional cues on cane-grip for non-visual navigation. In The 22nd International ACM SIGACCESS Conference on Computers and Accessibility, New York, NY, USA, 2020.

[86] John Nicholson, Vladimir Kulyukin, and Daniel Coster. Shoptalk: independent blind shopping through verbal route directions and barcode scans. The Open Rehabilitation Journal, 2(1):11–23, 2009.

[87] NielsenIQ. Bursting with new products, there's never been a better time for breakthrough innovation, 2019.

[88] Yoshihiro Niitsu, Toshiki Taniguchi, and Kouichiro Kawashima. Detection and notification of dangerous obstacles and places for visually impaired persons using a smart cane. In 2014 Seventh International Conference on Mobile Computing and Ubiquitous Networking (ICMU), pages 68–69, 2014.

[89] Eshed Ohn-Bar, João Guerreiro, Kris Kitani, and Chieko Asakawa. Variability in reactions to instructional guidance during smartphone-based assisted navigation of blind users. Proceedings of the ACM on interactive, mobile, wearable and ubiquitous technologies, 2(3):1–25, 2018.

[90] Ryuta Okazaki and Hiroyuki Kajimoto. Perceived distance from hitting with a stick is altered by overlapping vibration to holding hand. In CHI'14 Extended Abstracts on Human Factors in Computing Systems, pages 1903–1908. 2014.

[91] Emily E O'Brien, Aaron A Mohtar, Laura E Diment, and Karen J Reynolds. A detachable electronic device for use with a long white cane to assist with mobility. Assistive Technology, 26(4):219–226, 2014.

[92] C. A. Perez, C. A. Holzmann, and H. E. Jaeschke. Two-point vibrotactile discrimination related to parameters of pulse burst stimulus.

[93] Puppy In Training. How much does a guide dog cost? `https://puppyintraining.com/how-much-does-a-guide-dog-cost/`, 2023. Accessed: 2023-11-19.

[94] Yuankai Qi, Qi Wu, Peter Anderson, Xin Wang, William Yang Wang, Chunhua Shen, and Anton van den Hengel. Reverie: Remote embodied visual referring expression in real indoor environments. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 9982–9991, 2020.

[95] Kevin Qu, Haozhe Qi, Mihai Dusmanu, Mahdi Rad, Rui Wang, and Marc Pollefeys. Loc3r-vlm: Language-based localization and 3d reasoning with vision-language models, 2026.

[96] Krishan Rana, Jesse Haviland, Sourav Garg, Jad Abou-Chakra, Ian Reid, and Niko Suenderhauf. Sayplan: Grounding large language models using 3d scene graphs for scalable robot task planning. arXiv preprint arXiv:2307.06135, 2023.

[97] Santiago Real and Alvaro Araujo. Navigation systems for the blind and visually impaired: Past work, challenges, and open problems. Sensors, 2019.

[98] Chufan Rui, Yichen Liu, Junru Shen, Zhaobin Li, and Zaipeng Xie. A multi-sensory blind guidance system based on yolo and orb-slam. In 2021 IEEE International Conference on Progress in Informatics and Computing (PIC), pages 409–414. IEEE, 2021.

[99] M.F. Saaid, A.M. Mohammad, and M.S.A. Megat Ali. Smart cane with range notification for blind people. In 2016 IEEE International Conference on Automatic Control and Intelligent Systems (I2CACIS), 2016.

[100] Manaswi Saha, Alexander J Fiannaca, Melanie Kneisel, Edward Cutrell, and Meredith Ringel Morris. Closing the gap: Designing for the last-few-meters wayfinding problem for people with visual impairments. In The 21st international acm sigaccess conference on computers and accessibility, pages 222–235, 2019.

[101] Jayant Sakhardande, Pratik Pattanayak, and Mita Bhowmick. Smart cane assisted mobility for the visually impaired. International Journal of Electrical and Computer Engineering, 6(10):1262 – 1265, 2012.

[102] Arijit K Sengupta and Biman Das. Maximum reach envelope for the seated and standing male and female for industrial workstation design. Ergonomics, 43(9):1390–1404, 2000.

[103] Arielle M. Silverman, Jason D. Gwinn, and Leaf Van Boven. Stumbling in their shoes: Disability simulations reduce judged capabilities of disabled people. Social Psychological and Personality Science, 6(4):464–471, 2015.

[104] Bhupendra Singh and Monit Kapoor. Assistive cane for visually impaired persons for uneven surface detection with orientation restraint sensing. Sensor Review, 40(6):687–698, 2020.

[105] Milo Marsfeldt Skovfoged, Alexander Schiller Rasmussen, Lucie Kalova, Laura-Dora Daczo, and Hendrik Knoche. " is it there or not?" why augmented white canes do not need to provide detailed feedback about obstacles. In Adjunct Proceedings of the 2022 Nordic Human-Computer Interaction Conference, pages 1–5, 2022.

[106] Patrick Slade, Arjun Tambe, and Mykel J Kochenderfer. Multimodal sensing and intuitive steering assistance improve navigation and mobility for people with impaired vision. Science Robotics, 6(59), 2021.

[107] Molly E Sorrows and Stephen C Hirtle. The nature of landmarks for real and electronic spaces. In International conference on spatial information theory, pages 37–50. Springer, 1999.

[108] Henk Staats and Piet Groot. Seat choice in a crowded café: Effects of eye contact, distance, and anchoring. Frontiers in Psychology, 10:331, 2019.

[109] John Sweller and Paul Chandler. Evidence for cognitive load theory. Cognition and instruction, 8(4):351–362, 1991.

[110] Hotaka Takizawa, Shotaro Yamaguchi, Mayumi Aoyagi, Nobuo Ezaki, and Shinji Mizuno. Kinect cane: An assistive system for the visually impaired based on three-dimensional object recognition. In 2012 IEEE/SICE International Symposium on System Integration (SII), pages 740–745, 2012.

[111] Yu-Yun Tseng, Alexander Bell, and Danna Gurari. Vizwiz-fewshot: Locating objects in images taken by people with visual impairments, 2022.

[112] I. Ulrich and J. Borenstein. The guidecane-applying mobile robot technologies to assist the visually impaired. IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans, 2001.

[113] Marynel Vázquez and Aaron Steinfeld. An assisted photography framework to help visually impaired users properly aim a camera. ACM Transactions on Computer-Human Interaction (TOCHI), 21(5):1–29, 2014.

[114] Andreas Wachaja, Pratik Agarwal, Mathias Zink, Miguel Reyes Adame, Knut Möller, and Wolfram Burgard. Navigating blind people with walking impairments using a smart walker. Autonomous Robots, 41(3), 2017.

[115] Mohd Helmy Abd Wahab, Amirul A. Talib, Herdawatie A. Kadir, Ayob Johari, A. Noraziah, Roslina M. Sidek, and Ariffin A. Mutalib. Smart cane: Assistive cane for visually-impaired people, 2011.

[116] Corey H Walsh and Sertac Karaman. Cddt: Fast approximate 2d ray casting for accelerated localization. In 2018 IEEE International Conference on Robotics and Automation (ICRA), pages 3677–3684. IEEE, 2018.

[117] Chien-Yao Wang, I-Hau Yeh, and Hong-Yuan Mark Liao. Yolov9: Learning what you want to learn using programmable gradient information. In European conference on computer vision, pages 1–21. Springer, 2024.

[118] Hsueh-Cheng Wang, Robert K Katzschmann, Santani Teng, Brandon Araki, Laura Giarré, and Daniela Rus. Enabling independent navigation for visually impaired people through a wearable vision-based feedback system. In 2017 IEEE international conference on robotics and automation (ICRA), pages 6533–6540. IEEE, 2017.

[119] Liuyi Wang, Xinyuan Xia, Hui Zhao, Hanqing Wang, Tai Wang, Yilun Chen, Chengju Liu, Qijun Chen, and Jiangmiao Pang. Rethinking the embodied gap in vision-and-language navigation: A holistic study of physical and visual disparities. In Proceedings of the IEEE/CVF International Conference on Computer Vision, pages 9455–9465, 2025.

[120] Luyao Wang, Qihe Chen, Yan Zhang, Ziang Li, Tingmin Yan, Fan Wang, Guyue Zhou, and Jiangtao Gong. Can quadruped navigation robots be used as guide dogs? arXiv preprint arXiv:2210.08727, 2022.

[121] Zihan Wang, Yaohui Zhu, Gim Hee Lee, and Yachun Fan. Navrag: Generating user demand instructions for embodied navigation through retrieval-augmented llm. In Findings of the Association for Computational Linguistics: ACL 2025, pages 8430–8440, 2025.

[122] Yoshiaki Watanabe, Karinne Ramirez Amaro, Bahriye Ilhan, Taku Kinoshita, Thomas Bock, and Gordon Cheng. Robust localization with architectural floor plans and depth camera. In 2020 IEEE/SICE International Symposium on System Integration (SII), pages 133–138. IEEE, 2020.

[123] Michele A Williams, Caroline Galbraith, Shaun K Kane, and Amy Hurst. " just let the cane hit it" how the blind and sighted see navigation differently. In Proceedings of the 16th international ACM SIGACCESS conference on Computers & accessibility, pages 217–224, 2014.

[124] Tim Windecker, Manthan Patel, Moritz Reuss, Richard Schwarzkopf, Cesar Cadena, Rudolf Lioutikov, Marco Hutter, and Jonas Frey. Navitrace: Evaluating embodied navigation of vision-language models. arXiv preprint arXiv:2510.26909, 2025.

[125] Fengyi Wu, Yifei Dong, Zhi-Qi Cheng, Yilong Dai, Guangyu Chen, Hang Wang, Qi Dai, and Alexander G Hauptmann. Govig: Goal-conditioned visual navigation instruction generation. arXiv preprint arXiv:2508.09547, 2025.

[126] Anxing Xiao, Wenzhe Tong, Lizhi Yang, Jun Zeng, Zhongyu Li, and Koushil Sreenath. Robotic guide dog: Leading a human with leash-guided hybrid physical interaction, 2021.

[127] Fujing Xie and Sören Schwertfeger. Robust lifelong indoor lidar localization using the area graph. IEEE Robotics and Automation Letters, 9(1):531–538, 2023.

[128] Miao Xiong, Zhiyuan Hu, Xinyang Lu, Yifei Li, Jie Fu, Junxian He, and Bryan Hooi. Can llms express their uncertainty? an empirical evaluation of confidence elicitation in llms. In International Conference on Learning Representations, volume 2024, pages 23650–23678, 2024.

[129] Weihao Xuan, Qingcheng Zeng, Heli Qi, Junjue Wang, and Naoto Yokoya. Seeing is believing, but how much? a comprehensive analysis of verbalized calibration in vision-language models. In Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing, pages 1408–1450, 2025.

[130] Cang Ye, Soonhac Hong, Xiangfei Qian, and Wei Wu. Co-robotic cane: A new robotic navigation aid for the visually impaired. IEEE Systems, Man, and Cybernetics Magazine, 2(2):33–42, 2016.

[131] Huan Yin, Xuecheng Xu, Sha Lu, Xieyuanli Chen, Rong Xiong, Shaojie Shen, Cyrill Stachniss, and Yue Wang. A survey on global lidar localization: Challenges, advances and open problems. International Journal of Computer Vision, 132(8):3139–3171, 2024.

[132] Chien Wen Yuan, Benjamin V Hanrahan, Sooyeon Lee, Mary Beth Rosson, and John M Carroll. Constructing a holistic view of shopping with people with visual impairment: a participatory design approach. Universal Access in the Information Society, 18(1):127–140, 2019.

[133] He Zhang and Cang Ye. An indoor wayfinding system based on geometric features aided graph slam for the visually impaired. IEEE Transactions on Neural Systems and Rehabilitation Engineering, 25(9):1592–1604, 2017.

[134] T. Y. Zhang and C. Y. Suen. A fast parallel algorithm for thinning digital patterns. Communications of the ACM, 27(3):236–239, 1984.

[135] Yan Zhang, Ziang Li, Haole Guo, Luyao Wang, Qihe Chen, Wenjie Jiang, Mingming Fan, Guyue Zhou, and Jiangtao Gong. " i am the follower, also the boss": Exploring different levels of autonomy and machine forms of guiding robots for the visually impaired. In Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems, pages 1–22, 2023.

[136] Ming Zhao, Peter Anderson, Vihan Jain, Su Wang, Alexander Ku, Jason Baldridge, and Eugene Ie. On the evaluation of vision-and-language navigation instructions. In Proceedings

of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume, pages 1302–1316, 2021.

[137] Yuhang Zhao, Sarit Szpiro, Jonathan Knighten, and Shiri Azenkot. Cuesee: exploring visual cues for people with low vision to facilitate a visual search task. In Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing, pages 73–84, 2016.

[138] Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, et al. Judging llm-as-a-judge with mt-bench and chatbot arena. Advances in neural information processing systems, 36:46595–46623, 2023.

[139] Peter A Zientara, Sooyeon Lee, Gus H Smith, Rorry Brenner, Laurent Itti, Mary B Rosson, John M Carroll, Kevin M Irick, and Vijaykrishnan Narayanan. Third eye: A shopping assistant for the visually impaired. Computer, 50(2):16–24, 2017.

[140] Nicky Zimmerman, Tiziano Guadagnino, Xieyuanli Chen, Jens Behley, and Cyrill Stachniss. Long-term localization using semantic cues in floor plan maps. IEEE Robotics and Automation Letters, 8(1):176–183, 2022.

[141] Nicky Zimmerman, Louis Wiesmann, Tiziano Guadagnino, Thomas Läbe, Jens Behley, and Cyrill Stachniss. Robust onboard localization in changing environments exploiting text spotting. In 2022 IEEE/RSJ international conference on intelligent robots and systems (IROS), pages 917–924. IEEE, 2022.