Robotics and Autonomous Systems 203 (2026) 105535

Contents lists available at ScienceDirect

Robotics and Autonomous Systems

journal homepage: www.elsevier.com/locate/robot

Semantic Visual Simultaneous Localization and Mapping: A survey on state of the art, challenges, and future directions☆

Thanh Nguyen Canh [a,*], Haolan Zhang [a], Xiem HoangVan [b,*], Nak Young Chong [a,c]

[a] School of Information Science, Japan Advanced Institute of Science and Technology, 1-1 Asahidai, Nomi, 923-1292, Ishikawa, Japan
[b] Vietnam National University, University of Engineering and Technology, 144 Xuan Thuy, 10000, Hanoi, Viet Nam
[c] Department of Robotics, Hanyang University, 55 Hanyangdaehak-ro, Sangnok-gu, Ansan, 15588, Gyeonggi, Republic of Korea

ARTICLE INFO

Keywords:
Semantic vSLAM
Semantic mapping
Visual SLAM
Visual localization

ABSTRACT

Semantic Visual Simultaneous Localization and Mapping (Semantic vSLAM) is a critical area of research in robotics and computer vision, focusing on the simultaneous localization of robotic systems and the association of semantic information to construct the most accurate and comprehensive model of the surrounding environment. Since the first foundational work on Semantic vSLAM appeared more than two decades ago, the field has attracted increasing attention across various scientific communities. Despite its significance, the field lacks comprehensive surveys encompassing recent advances and persistent challenges. In response, this study provides a thorough examination of the state-of-the-art of Semantic vSLAM techniques, with the aim of illuminating current trends and key obstacles. Beginning with an in-depth exploration of the evolution of visual SLAM, this study outlines its strengths and unique characteristics while also critically assessing previous survey literature. Subsequently, a unified problem formulation and evaluation of the modular solution framework is proposed, which decomposes the problem into discrete stages, including visual localization, semantic feature extraction, mapping, data association, and loop closure optimization. Moreover, this study investigates alternative methodologies such as deep learning and the utilization of large language models, alongside a review of relevant research about contemporary SLAM datasets. Concluding with a discussion on potential future research directions, this study serves as a comprehensive resource for researchers seeking to navigate the complex landscape of Semantic vSLAM.

1. Introduction

Autonomous robotic systems play a vital role in diverse applications such as search and rescue, exploration, augmented reality, and autonomous navigation. These systems must possess a comprehensive understanding of their environment, which entails creating detailed maps, localizing themselves within these maps, and interpreting semantic information about their surroundings. Semantic Visual Simultaneous Localization and Mapping (Semantic vSLAM) addresses these challenges by integrating traditional SLAM capabilities with semantic perception, thereby enabling robots to construct detailed, high-level representations of their environment. Such semantic maps are essential for executing complex tasks efficiently and accurately while balancing computational and memory constraints.

Semantic vSLAM has gained increasing attention in the robotics and computer vision communities over the past decades. This interest stems from the growing demand for autonomous systems capable of operating in dynamic, unstructured, and complex environments. The field has expanded significantly, driven by advances in sensor technologies, computational capabilities, and artificial intelligence (AI). While this growth has broadened the scope of research, it has also introduced challenges related to integrating diverse methodologies and ensuring cross-disciplinary collaboration. This survey seeks to address these gaps by providing a unified perspective on Semantic vSLAM, analyzing key advancements, and identifying critical research challenges.

Semantic vSLAM is currently at a transformative juncture, propelled by novel developments in spatial perception and AI. Breakthroughs in deep learning have enabled robots to extract high-level semantic

☆ This work was supported in part by JST SPRING, Japan Grant Number JPMJSP2102 , and in part by the Asian Office of Aerospace Research and Development under Grant/Cooperative Agreement Award No. FA2386-25-1-4034.

* Corresponding authors.
E-mail addresses: thanhnc@jaist.ac.jp (T.N. Canh), haolan.z@jaist.ac.jp (H. Zhang), xiemhoang@vnu.edu.vn (X. HoangVan), nakyoung@jaist.ac.jp (N.Y. Chong).

https://doi.org/10.1016/j.robot.2026.105535
Received 7 August 2025; Received in revised form 12 December 2025; Accepted 10 May 2026
Available online 16 May 2026

features, enabling them to recognize objects, infer relationships, and interact more intelligently with their surroundings. These advances include neural network models for beyond-line-of-sight prediction, reasoning over dynamic environments, and processing deformable scenes. By leveraging these technologies, Semantic vSLAM has the potential to transcend traditional limitations and offer robust solutions for real-world applications.

However, the integration of semantic information introduces significant challenges. These include maintaining accuracy in dynamic settings, achieving real-time processing efficiency, and developing robust data fusion algorithms. Moreover, the field lacks standardized benchmarks and reproducible research practices, which are essential for evaluating and comparing different approaches. This survey aims to highlight these challenges, propose potential research directions, and emphasize the importance of interdisciplinary collaboration to advance the state-of-the-art in Semantic vSLAM.

1.1. Visual SLAM evolution

The concept of Simultaneous Localization and Mapping (SLAM) was first introduced by Smith and Cheeseman in the 1980s [1]. In addition, since the advent of mobile robots in the late 1960s, the goal of enabling them to execute tasks autonomously has remained a central theme in robotics research [2]. Since then, SLAM has become a cornerstone technology in robotics, enabling autonomous systems to navigate and interact with their environments. Inspired by human spatial perception – the ability to localize, map, and adapt to unfamiliar settings – robots equipped with SLAM algorithms can perform similar tasks using various sensors, such as cameras, LiDAR, and inertial measurement units (IMUs). This ability can be enhanced with acquired training and plays a crucial role in human cognition and robot control development. SLAM technology has found applications across diverse domains, including underwater vehicles (UWVs) [3], unmanned aerial vehicles (UAVs) [4], autonomous driving [5], service robots [6], augmented reality (AR) [7], and virtual reality (VR) [8]. These applications underscore the versatility and importance of SLAM in enabling autonomy.

Based on the type of sensors, traditional SLAM systems can be broadly categorized into LiDAR-based SLAM (L-SLAM) and vision-based SLAM (vSLAM). LiDAR sensors excel at precise distance measurements and high-frequency updates, making them ideal for applications requiring low-drift motion estimation [9,10]. However, LiDAR systems are costly and computationally intensive. In contrast, vSLAM leverages cameras to capture rich visual information, allowing for feature extraction and environmental understanding at lower cost [11,12]. Despite its advantages, vSLAM is sensitive to lighting conditions and struggles in textureless or dynamic environments. In recent days, various SLAM technologies have focused on vSLAM due to its low hardware cost, high accuracy in small scenes, and the ability to obtain rich environmental information. On the other hand, there are still many challenges, especially those that come with dynamic object movements and environments lacking textures.

To overcome these limitations, researchers have explored multi-sensor fusion approaches, combining cameras, IMUs, and GPS to improve localization robustness. Recent advancements in vSLAM have emphasized graph-based methodologies [13], in which the front end constructs graphs from sensor data and the back end optimizes them to determine the most probable pose configuration.

While traditional vSLAM focuses on geometric map construction and localization, it lacks semantic understanding. The integration of *semantic* information – enabled by advancements in deep learning – has opened new avenues for vSLAM. By extracting semantic features, researchers have improved pose estimation, map accuracy, and environmental understanding [14–16]. Semantic vSLAM systems enable robots to perform tasks such as semantic localization and mapping, significantly enhancing their adaptability to complex and dynamic environments. The rise of these systems represents a paradigm shift, bridging the gap between geometric and semantic approaches in SLAM. In tandem with its geometric foundation, SLAM has steadily evolved to incorporate higher-level information about the environment – termed *semantic information* – giving rise to the field of *Semantic vSLAM*. Fig. 1 illustrates a taxonomy based on the coupling level between geometric and semantic components.

Early SLAM solutions largely centered on geometric cues, such as points, lines, and planar structures, using various sensor modalities like cameras or LiDAR for feature extraction [11,17]. This approach, often referred to as traditional or *geometric SLAM*, excels when the environment remains static and well-defined. However, real-world settings inherently involve illumination changes, dynamic obstacles, and complex scenes with sparse textures. These conditions can undermine the robustness and accuracy of purely geometric methods, which typically rely on handcrafted features or geometric primitives. Consequently, the research community has increasingly focused on more holistic solutions that integrate vision sensors (*e.g.*, monocular, stereo, RGB-D cameras) and other sensor data, thereby enhancing SLAM's adaptability and reliability across diverse and challenging operational domains [2,17,18].

In parallel with these sensor-fusion strategies, breakthroughs in deep learning have led to significant leaps in *semantic understanding*. By leveraging neural networks for object detection, semantic segmentation, and scene recognition, SLAM systems can move beyond constructing raw geometric maps to encoding rich descriptive information about encountered objects and scene layouts [16,19]. This synergy – combining geometric estimation with semantic cues – paves the way for systems that can better handle moving objects, reinterpret ambiguous features, and enable robust long-term mapping in dynamic or partially known environments [14,15]. Notably, *Semantic vSLAM* not only locates and maps entities in the scene but also categorizes and attributes them with meaningful labels, thereby supporting higher-level tasks such as scene understanding, path planning, human–robot interaction, and task-oriented manipulation. In instances where the aim is to enhance localization, mapping, or both, these challenges are known as semantic localization, semantic mapping, and Semantic vSLAM, respectively.

Semantic Mapping augments traditional SLAM by embedding high-level contextual and categorical information about the environment into the spatial map, enabling a shift from purely geometric representations to cognitively meaningful scene understanding. This concept was first introduced by Dellaert and Bruemmer [19], highlighting the importance of associating semantic entities – such as object classes, scene layouts, and terrain types – with spatial locations. Unlike conventional occupancy grids or point clouds, semantic maps provide symbolic abstractions that facilitate complex reasoning, task planning, and human–robot interaction [4,20–22]. Recent advances in deep learning, particularly in semantic segmentation and instance-level object detection, have significantly improved the granularity and accuracy of semantic labels. This has enabled near-real-time semantic map construction, even in dynamic environments, using data from RGB, RGB-D, or multimodal sensors. Techniques such as Bayesian fusion of per-frame segmentation [15] and learned feature embeddings for instance tracking [23] have further enhanced map coherence and consistency over time.

However, despite its potential, semantic mapping introduces several open challenges that hinder scalable deployment in real-world robotics. One major concern is the increased computational complexity associated with maintaining and updating semantic labels, especially under limited hardware and power constraints [2,17,24]. Moreover, inaccurate or incomplete semantic predictions can introduce drift and inconsistency into the SLAM pipeline, particularly when these labels are used for loop closure or data association. Loop closure refers to the process of recognizing that a robot has returned to a previously visited location and using this information to correct accumulated drift errors

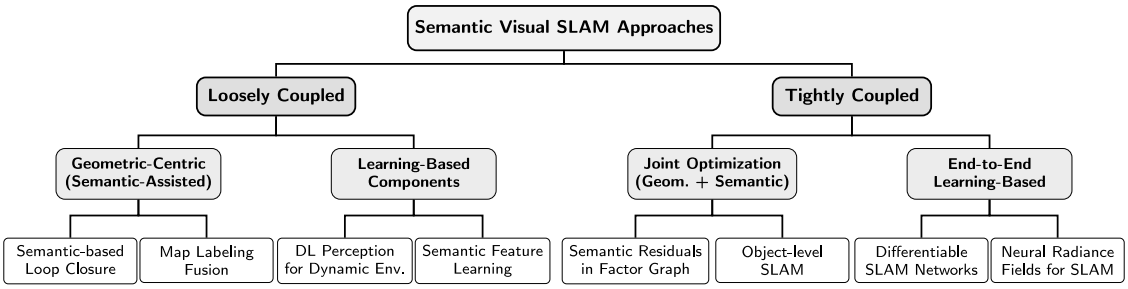

Fig. 1. Taxonomy of Semantic Visual SLAM approaches based on coupling level between geometric and semantic components.

in the trajectory estimate, thereby achieving global map consistency. Data association refers to the process of establishing correspondence between current sensor observations and existing elements in the map, determining whether a detected object or feature corresponds to a previously observed landmark or represents a new entity. Handling dynamic objects adds further complexity, as semantic information must be continuously validated and updated to prevent map corruption. Another critical limitation lies in the scarcity of standardized benchmarks and evaluation protocols focused specifically on the quality and consistency of semantic maps. This gap restricts fair comparison and reproducibility across approaches [19]. Additionally, memory-efficient representations, such as OctoMap-based semantic trees [25,26] or voxel hashing [27,28], must balance rich semantic encoding with storage constraints, especially for long-term autonomy or large-scale environments.

Semantic Localization aims to enhance pose estimation accuracy and robustness by incorporating high-level semantic information such as object classes, spatial relationships, and scene context, moving beyond low-level geometric features. This paradigm enables more reliable localization in perceptually ambiguous or dynamically changing environments, where traditional SLAM approaches based on handcrafted features often degrade. Semantically-aware localization leverages advances in deep neural networks (DNNs) for object detection, semantic segmentation, and scene classification to extract robust, discriminative features that are invariant to viewpoint and appearance changes. Early works such as SuMa++ [29,30] integrated semantic segmentation into the SLAM pipeline using a semantic ICP framework, demonstrating improved alignment accuracy in cluttered and unstructured environments. Probabilistic approaches have also been explored, where semantic information augments traditional observation models. For instance, extensions of the Markov localization framework [31,32] incorporated object-level semantics into the sensor likelihood model to improve robustness against perceptual aliasing.

A significant limitation of these early models lies in their deterministic treatment of semantic observations, which disregards the inherent uncertainty in perception systems. To mitigate this, Akai et al. [33] proposed a Bayesian localization approach that integrates supervised object recognition into a probabilistic graphical model. Their framework models classification uncertainty using a Dirichlet distribution, allowing for a more principled fusion of semantic cues. Subsequently, Akai [34] introduced a probabilistic treatment of depth regression errors, improving pose inference in scenes with noisy depth observations.

Recent efforts [35–38] have focused on leveraging dense semantic maps to enhance global consistency and robustness under appearance changes. These approaches often encode semantic priors into factor graph-based SLAM systems (a bipartite graphical model used for representing probabilistic inference problems) or utilize metric-semantic fusion for long-term localization. Nevertheless, many of these methods still depend on planar scene assumptions and handcrafted features, which limit their generalization to complex, 3D-structured environments. As highlighted in [39], such limitations can result in projection inconsistencies and scale-dependent feature degradation, underscoring the need for learning-based spatial representations capable of modeling geometric and semantic uncertainty jointly.

Semantic vSLAM unifies the principles of semantic mapping and semantic localization into a cohesive framework, enabling intelligent robotic systems. By fusing low-level geometric data with high-level semantic cues, Semantic vSLAM systems can achieve more accurate and robust localization based on object-level features, generate high-resolution and context-aware maps, and maintain compact and efficient storage representations. These systems exhibit improved robustness to occlusions, lighting variations, and viewpoint changes, while also facilitating semantic-level interaction with dynamic and unstructured environments [40,41]. Recent works have demonstrated the utility of Semantic vSLAM for active exploration [42], object-centric localization [43], and robust relocalization under appearance and structural changes [44]. In general, Semantic vSLAM can be decomposed into four primary sub-problems that define its core functionality:

1. **Semantic extraction and mapping**: Leveraging semantic segmentation, object detection, and scene parsing to extract meaningful features and incrementally integrate them into a globally consistent map.
2. **Semantic data association**: Establishing correspondences between semantic entities across time and viewpoints, often using probabilistic models, learned feature descriptors, or graph-based optimization to maintain label consistency.
3. **Semantic uncertainty and representation**: Modeling perceptual uncertainty from semantic predictions (*e.g.*, classification confidence, depth variance) using probabilistic frameworks such as Bayesian filters, Dirichlet distributions, or Gaussian Mixtures to ensure reliable integration.
4. **Semantic cost function**: Designing loss functions or objective terms that encode semantic priors and constraints – such as object geometry, label consistency, or topology – for use in optimization-based SLAM back-ends.

Finally, *Semantic vSLAM* extends traditional SLAM by simultaneously estimating the 3D geometry of a scene while attaching semantic labels to observed objects and structures. This unified framework integrates both metric and symbolic understanding of the environment, enabling robots not only to map where things are but also what they are. One of the earliest systems to leverage both spatial and semantic representations was proposed by Galindo et al. [80], which introduced a layered cognitive map architecture to incorporate conceptual knowledge into the mapping process. Contemporary Semantic vSLAM approaches typically extend the SLAM state vector to include semantic entities or incorporate semantic constraints into the optimization process. Some works introduce novel object-level representations, such as ellipsoids (quadrics) [54], planar landmarks [81], or mesh models [82], allowing compact and expressive 3D scene encoding. Others focus on semantic data association strategies to maintain map consistency under ambiguity and perceptual noise. Probabilistic models based on maximum likelihood estimation [41,83], max-mixture frameworks [84], and multi-hypothesis k-assignment methods [85]

Table 1
A detailed comparison of representative and state-of-the-art Semantic vSLAM algorithms across multiple criteria. **Architecture Descriptions:** *PG* — Optimizes robot poses as graph nodes with relative pose constraints as edges, *FG* — Generalized graphical model representing probabilistic relationships between variables (poses, landmarks, semantics) through factor nodes, *BA* — Joint nonlinear optimization of 3D structure and camera parameters by minimizing reprojection errors, *JO* — Simultaneous optimization of geometry, appearance, or semantics in a unified framework, commonly used in neural implicit and Gaussian splatting methods, *F2M* — Tracking approach that aligns each new frame directly against the accumulated 3D map rather than frame-to-frame. **Library Descriptions:** *g2o* [45] — General Graph Optimization framework for pose graph (a simplified factor graph specifically representing robot trajectory estimation, where nodes correspond to robot poses at different time steps and edges encode relative pose constraints) and bundle adjustment (a nonlinear optimization technique that jointly refines 3D structure and camera parameters by minimizing the sum of reprojection errors between observed image features and their predicted projection) optimization, *Ceres* [46] — Google's nonlinear least squares solver for large-scale optimization problems, *GTSAM* [47] — Georgia Tech Smoothing and Mapping library for factor graph-based probabilistic inference and optimization, *PyTorch* [48] — Deep learning framework commonly used for neural network-based SLAM components and end-to-end learning.

Reference	Year	SLAM Approach	Semantic Perception	Map Repr.	Dyn. Aware	Open Vocab.	Arch.	Library	Env.	Onl.	Loop Clos.	Large Scale	Public Avail.
Classic, Object-based, and Early Dense SLAM													
CNN-SLAM [49]	2017	DD	CNN DP	DM	–	–	PG	g2o	🏠	+	+	–	✓
SemanticFusion [15]	2017	DF	CNN SS	SM	–	–	F2M	_	🏠	+	–	–	✓
Co-Fusion [50]	2017	OF	CNN SS	SM	+	–	F2M	_	🏠	+	–	–	✓
VSO [51]	2018	F	SS	SpM	–	–	BA	g2o	🏠🌲	+	+	+	✗
DS-SLAM [52]	2018	F	SegNet	SP	+	–	PG+BA	ceres/g2o	🌲	+	+	+	✓
MaskFusion [53]	2018	OF	Mask R-CNN	SM	+	–	F2M	_	🏠	+	–	–	✓
CubeSLAM [40]	2019	OBA	O	3D Cuboids	+	–	PG+BA	GTSAM	🏠🌲	+	+	+	✓
QuadricSLAM [54]	2019	OBA	O	DQ	–	–	FG	GTSAM	🏠🌲	–	+	+	✓
EAO-SLAM [55]	2020	OBA	O	3D Cuboids	–	–	JO	GTSAM	🏠	+	–	+	✓
Fusion++ [23]	2018	OF	Mask R-CNN	TSDF	–	–	PG	g2o	🏠	+	+	–	✗
PanopticFusion [56]	2019	DF	P	PV	–	–	F2M	_	🏠	+	–	–	✗
Kimera [57]	2020	F	P	3D SeM	+	–	FG	GTSAM	🏠🌲	+	–	+	✓
Hydra [58]	2022	F	SS	3D SG	–	–	FG	GTSAM	🏠🌲	+	+	+	✓
Continuous Representation SLAM													
iMAP [59]	2021	I	C	MLP	–	–	JO	PyTorch	🏠	+	–	–	✓
NICE-SLAM [60]	2022	I	C	FeG+MLP	–	–	JO	PyTorch	🏠	+	–	–	✓
Point-SLAM [61]	2023	I	C	FeG+MLP	–	–	JO	PyTorch	🏠	+	–	–	✓
H2-Mapping [62]	2023	I	C	FeG+MLP	–	–	JO	PyTorch	🏠	+	–	+	✓
SNI-SLAM [63]	2024	I	SS	VGM+MLP	–	–	_	_	🏠	+	–	–	✓
H3-Mapping [64]	2024	I	C	FeG+MLP	–	–	JO	PyTorch	🏠	+	–	+	✓
DynaMoN [65]	2024	I	SS	FeG+MLP	+	–	JO	PyTorch	🏠	+	–	–	✓
SplaTAM [66]	2024	GS	C	3D G	–	–	JO	PyTorch	🏠	+	–	–	✓
NEDS-SLAM [67]	2024	GS	SS	SeG	–	–	JO	PyTorch	🏠	+	–	+	✗
GS-SLAM [68]	2024	GS	C	3D G	–	–	JO+BA	PyTorch	🏠	+	–	+	✓
SGS-SLAM [69]	2025	GS	S	SeG	+	–	JO	PyTorch	🏠	+	–	–	✓
SemGauss-SLAM [70]	2025	GS	SS	SeG	–	–	JO	PyTorch	🏠	+	–	+	✗
Foundation Model-based SLAM Systems													
ConceptFusion [71]	2023	DF	CLIP	DFF	–	+	F2M	PyTorch	🏠	–	–	–	✓
FM-Fusion [72]	2024	DF	SAM + CLIP	VGM	–	+	F2M	PyTorch	🏠	+	–	–	✓
ConceptGraphs [73]	2024	SG	CLIP	SM	–	+	PG	PyTorch	🏠	+	–	–	✓
LOSS-SLAM [74]	2024	OF	DINO	OM	–	+	FG	GTSAM	🏠	+	–	–	✗
LEXIS [75]	2024	LD	VLM	TM	–	+	PG	PyTorch	🏠	+	+	–	✗
Hier-SLAM++ [76]	2025	GS	LLM+GM	SeG	–	+	JO+PG	PyTorch	🏠	+	–	+	✗
Multi-Robot Semantic vSLAM Systems													
Kimera-Multi [77]	2022	F	SS	D SeM	–	–	PG	GTSAM	🏠🌲	+	+	+	✓
SlideSLAM [78]	2024	OF	IS	D OM	–	–	PG	GTSAM	🏠🌲	+	+	+	✓
HAMMER [79]	2025	GS	CLIP	SeG	–	+	PG	BA	🏠🌲	+	–	+	✗

BA: Bundle Adjustment	F: Feature based	JO: Joint Optimization	PV: Panoptic Voxels	TM: Topological Map
C: Color Only	FeG: Feature Grid	LD: Language Driven	SeG: Semantic Gaussian	TSDF: Truncated Signed Distance Function
D: Distributed	FG: Factor Graph	LLM: Large Language Model	SeM: Semantic Mesh	VLM: Vision Language Model
DD: Dense Direct	F2M: Frame to Model	MLP: Multi-Layer Perceptron	SfM: Structure from Motion	VGM: Voxel Grid Map
DF: Dense Fusion	G: Gaussians	O: Object Detection	SG: Scene Graph	+ : Present
DFF: Dense Feature Field	GM: Generation Model	OF: Object Fusion	SM: Surfer Map	– : Absent
DM: Dense Map	GS: Gaussian Splatting	OM: Object Map	SpM: Sparse Map	🏠: Indoor
DP: Depth Prediction	I: Neural Implicit	P: Panoptic Segmentation	SP: Sparse Point	🌲: Outdoor
DQ: Dual Quadrics	IS: Instance Segmentation	PG: Pose Graph	SS: Semantic Segmentation	🏠🌲: Indoor/Outdoor

have been developed to handle uncertain object detection and multi-instance association. Approaches can also be categorized based on how semantics are modeled: explicitly through symbolic labels or object poses, or implicitly via learned feature embeddings or volumetric priors. While explicit [4,86,87] representations facilitate interpretability and planning, but they often demand increased storage and suffer from brittleness under perceptual uncertainty. Implicit representations, learned through DNNs [60,88], offer improved robustness but pose challenges for explainability and optimization within SLAM back-ends.

Recently, 3D Gaussian Splatting has emerged as a promising representation for capturing scene geometry and appearance. 3D Gaussian Splatting is an explicit scene representation technique that models the environment as a collection of anisotropic 3D Gaussian primitives, each characterized by its position (mean), covariance matrix (defining shape and orientation), color, and opacity. This representation enables efficient differentiable rendering through a splatting operation that projects these 3D Gaussians onto the 2D image plane, allowing for real-time novel view synthesis and high-fidelity scene reconstruction.

Table 2
Comparison of topic coverage across related works.

Topic		*Kostavelis* [91]	*Sualeh* [92]	*Chen* [93]	*Chen* [94]	*Pu* [95]	*Wang* [96]	*Chen* [17]	*Our*
Introduction	Historical review	Briefly	Yes	Yes	Yes	Yes	Briefly	Yes	Yes
	Problem formulation	No	No	No	No	No	No	Briefly	Yes
Modular scheme	Semantic Extraction	Briefly	Briefly	Briefly	Yes	Yes	No	Yes	Yes
	Semantic Localization	No	No	No	No	Briefly	Briefly	No	Yes
	Semantic Mapping	Yes	No	Yes	Briefly	Yes	Yes	Briefly	Yes
	Semantic Data Association	No	No	No	No	Briefly	No	Yes	Yes
	Semantic Back-end Optimization	No	Briefly	Yes	No	Briefly	No	Yes	Yes
Alternative approaches	Learning-based and Foundation Model	No	No	No	No	No	No	No	Yes
	Continuous Representations	No	No	No	No	No	No	No	Yes
	Multi-Robot Systems	No	No	Briefly	No	Briefly	No	No	Yes
Open problems	Semantic benchmarks	Briefly	No	No	Briefly	No	No	No	Yes
	Active Semantic vSLAM	No	No	No	No	No	No	No	Yes
	Lifelong learning	No	No	No	No	No	Yes	Briefly	Yes
	Generalization & robustness	No	No	Briefly	No	Briefly	Briefly	Briefly	Briefly
	Reproducible research	No	No	No	No	No	No	No	Briefly
	Practical applications	Yes	No	No	Briefly	No	Briefly	No	Briefly

By modeling the scene as a collection of anisotropic 3D Gaussians, this approach supports continuous view synthesis and efficient rendering while offering potential as a compact SLAM representation. Its integration into SLAM frameworks remains an active area of research, with early efforts exploring its suitability for semantic reconstruction and localization [68,69,89,90]. A detailed comparison of these Semantic vSLAM systems is presented in Table 1, summarizing core characteristics such as sensor modality, representation type, semantic modeling technique, back-end architecture, and suitability for dynamic or large-scale environments.

1.2. Previous works

Several recent surveys [17,91,92,94–102] have investigated the progression of visual SLAM, particularly highlighting its evolution from traditional geometric approaches to those that incorporate deep learning and semantic scene understanding. However, the majority of these studies emphasize the importance and the limitations of classical vSLAM, especially in dynamic environments, and the potential of semantic cues to enhance scene interpretation. Few offer a systematic breakdown of Semantic vSLAM as a unified framework, and most do not examine its subcomponents in detail, such as semantic data association or loop closure optimization. Table 2 summarizes the themes covered by each study and contrasts them with the scope of this survey.

The earliest among them, by Kostavelis and Gasteratos [91], offers a comprehensive survey on "semantic mapping for mobile robots". This work analyzes the trends and main aspects of semantic mapping and identifies three core research challenges: (1) defining the minimal criteria for a map to be considered semantic, (2) developing semantic mapping evaluation metrics, and (3) integrating semantic maps for knowledge representation. However, this review primarily focuses on augmenting traditional 2D or 3D maps (*e.g.*, topological maps) with semantic labels and does not address components such as semantic localization, data association, or back-end optimization. More recent works [94,99,100] acknowledge the limitations of geometric SLAM and advocate for deep learning as a data-driven paradigm for sensor interpretation and scene understanding. These surveys trace the chronological evolution of SLAM technologies and paradigms but stop short of delving into the architecture or mathematical modeling of Semantic vSLAM systems. Other surveys [97,102,103] approach semantic mapping as a strategy to improve perceptual awareness in autonomous systems. These papers typically divide the semantic mapping pipeline into three stages: data acquisition, semantic and spatial fusion, and symbolic knowledge representation. However, they generally treat mapping and perception as decoupled modules rather than parts of an integrated SLAM framework.

Sualeh et al. [92] introduces the fundamentals of SLAM and provides a high-level overview of Semantic vSLAM. Their work is introductory and does not explore a formalized model or detailed submodules. Similarly, Pu et al. [95] highlight the challenges of vSLAM in dynamic environments, reviewing how deep learning improves the front-end (*e.g.*, dynamic object segmentation, feature robustness under varying illumination) and the back-end (*e.g.*, place recognition and loop detection). However, their work lacks a unified mathematical formulation of Semantic vSLAM and does not cover semantic data association or map optimization in depth. Wen et al. [98] study the influence of dynamic objects on visual SLAM and leverage Mask R-CNN to segment dynamic regions and initialize camera poses. Their approach combines photometric, geometric, and depth consistency to differentiate between static and dynamic features. Meanwhile, Chen et al. [17] surveys semantic object association methods – both probabilistic and non-probabilistic – and briefly mentions semantic localization in indoor and outdoor environments, but only at a high level. Most recently, Wang et al. [96] categorize Semantic vSLAM in dynamic scenes into four key areas: (1) dynamic object exclusion, (2) semantic tracking for localization, (3) semantic mapping under dynamic conditions, and (4) multi-sensor fusion. While this survey emphasizes the challenges of real-world deployment, it does not offer a unified treatment of Semantic vSLAM nor a detailed breakdown of how semantics are integrated into specific SLAM components such as data association, loop closure, or cost modeling.

In summary, existing surveys underscore the significance of integrating semantics into SLAM and outline historical trends and potential applications. However, they fall short of presenting Semantic vSLAM as a unified probabilistic framework or exploring the four critical components – semantic extraction and mapping, semantic data association, semantic uncertainty modeling, and semantic-aware cost functions – in a structured manner. Therefore, this paper aims to fill this gap by formulating Semantic vSLAM probabilistically and providing an in-depth discussion of its modular architecture and open challenges.

1.3. Paper structure

The remainder of this paper is structured as follows. Section 2 introduces the Semantic vSLAM problem, beginning with a unified probabilistic formulation and followed by the decomposition into key subproblems. Section 3 discusses semantic feature extraction from raw sensor data, including object detection, segmentation, and scene understanding. Section 4 focuses on semantic localization methods that leverage object-level and high-level contextual cues for robust pose estimation. Section 5 covers semantic mapping techniques, including map fusion strategies and handling dynamic environments. Section 6 explores semantic data association and integration, emphasizing probabilistic models and multi-instance tracking. Section 7 addresses loop

closure detection and global optimization incorporating semantic constraints. Section 8 presents learning-based approaches, including Deep Learning and foundation models. Section 9 reviews recent advances in continuous and implicit scene representations, such as neural fields and 3D Gaussian splatting. Section 10 extends the discussion to multi-robot Semantic vSLAM, focusing on collaborative semantic mapping and distributed optimization. Section 11 outlines open research challenges, including lifelong learning, zero-shot generalization, and standardization of semantic benchmarks. Finally, Section 12 concludes the paper with a summary of key insights and future directions for the field.

2. The semantic vSLAM problem

2.1. Problem formulation

Consider the Semantic vSLAM problem, where the robot's state is represented as $\mathcal{X} \triangleq \{\mathbf{X}_t\}_{t=1}^{T} \in SE(3)$ for $t = 1, \dots, T$, and evolves according to deterministic discrete-time kinematics: $\mathbf{X}_t := \begin{bmatrix} \mathbf{R}_t & \mathbf{p}_t \\ 0^T & 1 \end{bmatrix}$, where $\mathbf{R}_t \in SO(3)$ denotes the rotation matrix and $\mathbf{p}_t \in \mathbb{R}^3$ the translation vector of the robot pose.

Semantic vSLAM extends classical SLAM by jointly estimating the robot trajectory and a map enriched with semantic information. The goal is to estimate the trajectory $\mathcal{X}$, a set of landmark states $\mathcal{L} = \{L_m\}_{m=1}^{M}$, and semantic information $\mathcal{S} = \{\mathbf{S}^c\}_{c=1}^{C}$ corresponding to C semantic classes. We define the semantic map as $\mathcal{M} = \{\mathbf{M}_t\}_{t=1}^{T}$. These estimates are conditioned on a sequence of observation measurements $\mathcal{Z} = \{\mathbf{Z}_t\}_{t=1}^{T}$ and control inputs $\mathcal{U} = \{\mathbf{U}_t\}_{t=1}^{T}$.

To infer this state, the system maintains an internal *belief* or *information state* [104–106], denoted by $b_t(\mathbf{X}_t)$, representing the posterior probability distribution at time t:

$$b_t(\mathbf{X}_t) \triangleq p\left(\mathbf{X}_t, \underbrace{L_{1:t}, \mathbf{S}_{1:t}}_{\text{descriptor } \Theta}, \mathbf{M}_{1:t} \mid \mathbf{Z}_{1:t}, \mathbf{U}_{1:t-1}\right), \tag{1}$$

where $\mathbf{Z}_{1:t}$ and $\mathbf{U}_{1:t-1}$ denote the sequences of observations and controls, respectively. The *belief space* of probability destiny functions (pdf) over the set $\mathcal{X}$ is defined as:

$$\mathcal{B}(\mathcal{X}) \triangleq \left\{ b : \mathcal{X} \to \mathbb{R} \mid \int b(\mathbf{X}) d\mathbf{X} = 1,\ b(\mathbf{X}) \geq 0 \right\}. \tag{2}$$

In order to estimate the robot's pose and establish a map, agents must be capable of predicting posterior belief distributions, that is the pdf over $\mathcal{X}$ after taking observation $\mathbf{Z}_{t+1}$ and performing a control input $\mathbf{U}_t$:

$$b_{t+1}(\mathbf{X}_{t+1}) \triangleq p(\mathbf{X}_{t+1}, \Theta_{t+1}, \mathbf{M}_{t+1} \mid \mathbf{Z}_{t+1}, \mathbf{U}_t, b_t(\mathbf{X}_t)). \tag{3}$$

This problem can be factorized using Bayes' rule and the Markov assumption as:

$$\begin{aligned} p(\mathbf{X}_{1:t}, \mathbf{M}, \Theta_{1:t} \mid \mathbf{Z}_{1:t}, \mathbf{U}_{1:t}) \propto & \prod_{\tau=1}^{t} p(\mathbf{Z}_\tau \mid \mathbf{X}_\tau, \Theta_\tau, \mathbf{M}) \\ & \cdot p(\Theta_\tau \mid \mathbf{X}_\tau, \mathbf{M}) \cdot p(\mathbf{X}_\tau \mid \mathbf{X}_{\tau-1}, \mathbf{U}_\tau), \end{aligned} \tag{4}$$

where, $p(\mathbf{X}_\tau \mid \mathbf{X}_{\tau-1}, \mathbf{U}_\tau)$ is the motion model, $p(\mathbf{Z}_\tau \mid \mathbf{X}_\tau, \Theta_\tau, \mathbf{M})$ is the sensor likelihood (observation model), and $p(\Theta_\tau \mid \mathbf{X}_\tau, \mathbf{M})$ represents the semantic likelihood conditioned on the pose and map. In Semantic vSLAM, the map $\mathbf{M}$ is characterized by a collection of semantic landmarks.

$$\mathbf{M} = \{(l_i, c_i) \mid l_i \in \mathbb{R}^3, s_i^c \in C\}, \tag{5}$$

where l_i is the 3D location and s_i^c is the semantic class of landmark i, and $C = \{1, \dots, C\}$ is the set of possible semantic labels. The belief can be updated recursively via a Bayesian filter:

$$\begin{aligned} b_t(\mathbf{X}_t, \mathbf{M}, \Theta_t) \propto & p(\mathbf{Z}_t \mid \mathbf{X}_t, \mathbf{M}, \Theta_t) \int p(\mathbf{X}_t \mid \mathbf{X}_{t-1}, \mathbf{U}_t) \\ & \cdot b_{t-1}(\mathbf{X}_{t-1}, \mathbf{M}, \Theta_{t-1})\, d\mathbf{X}_{t-1}. \end{aligned} \tag{6}$$

In addition, Semantic vSLAM must also reason about *semantic uncertainty* of the system. A piece of semantic information can be extracted from keyframe t as $\mathbf{s}_k = (s_k^c, s_k^s, s_k^b) \in \mathbf{S}_t$ [41], consisting of a categorical label $s_k^c \in C$, a score confidence s_k^s, and a bonding box s_k^b. We may model its uncertainty using a distribution such as:

$$p(\mathbf{S}_t \mid \mathbf{Z}_t) = \text{Categorical}(\pi_t), \quad \pi_t = \text{softmax}(f_\theta(\mathbf{Z}_t)), \tag{7}$$

where $f_\theta(\cdot)$ is a deep network predicting semantic logits from sensor data [43,107,108]. The robot state is commonly assumed Gaussian with a pdf $b(\mathbf{X})$ having mean $\hat{\sigma}$ and covariance $\sum_x$ [109,110]. The *semantic data association* process infers a set of latent correspondences $\mathcal{D}$ linking current observations to map landmarks. The standard formulation follows a maximum likelihood objective:

$$\mathcal{X}_{ml}^*, \Theta_{ml}^* = \underset{\mathcal{X},\Theta}{\text{argmax}}\, p(\mathcal{Z} \mid \mathcal{X}, \Theta). \tag{8}$$

Some works also assume *maximum likelihood* (ML) observations [23,40,41,54,84,110,111], *i.e.* the semantic measurement data associations are independent across keyframes:

$$\begin{aligned} \mathcal{D}^* &= \underset{\mathcal{D}}{\text{argmax}}\, p(\mathcal{D} \mid \mathcal{X}^0, \Theta^0, \mathcal{Z}), \\ D_{i+1} &= \underset{\mathcal{D}}{\text{argmax}}\, p(\mathcal{D} \mid \mathcal{X}^i, \Theta^i, \mathcal{Z}). \end{aligned} \tag{9}$$

An Expectation-Maximization (EM) approach is often applied:

$$\begin{aligned} \mathcal{X}^{i+1}, \Theta^{i+1} &= \underset{\mathcal{X},\mathcal{D},\Theta}{\text{argmax}}\, p(\mathcal{Z} \mid \mathcal{X}, \mathcal{D}, \Theta), \\ \mathcal{X}^{i+1}, \Theta^{i+1} &= \underset{\mathcal{X},\Theta,\mathcal{D}}{\text{argmax}}\, \mathbb{E}_{\mathcal{D}}[\log p(\mathcal{Z} \mid \mathcal{X}, \Theta, \mathcal{D}) \mid \mathcal{X}^i, \Theta^i, \mathcal{Z}] \\ &= \underset{\mathcal{X},\Theta,\mathcal{D}}{\text{argmax}} \sum_{\mathcal{D} \in \mathbb{D}} p(\mathcal{D} \mid \mathcal{X}^i, \Theta^i, \mathcal{Z}) \log p(\mathcal{Z} \mid \mathcal{X}, \Theta, \mathcal{D}) \end{aligned} \tag{10}$$

2.2. Main subproblems

Although Semantic vSLAM is initially articulated as a unified approach, it is often decomposed into a modular sequence to improve manageability and system development. From a computational pipeline perspective, the five stages reflect the natural information flow in practical Semantic vSLAM systems, progressing from raw sensor data through semantic understanding to globally consistent map construction. In addition, from a pedagogical perspective, this modular breakdown provides a structured framework for organizing the extensive and diverse literature in Semantic vSLAM. By treating each stage as a distinct research area with its own challenges, methods, and evaluation criteria, we enable readers to navigate the complex landscape more effectively and identify specific areas of interest or potential contributions. The five main constituent subproblems, which are shown in Fig. 2 include:

1. *Semantic extraction*: this stage involves extracting high-level semantic information from raw sensory inputs $\mathbf{Z}_t$. Deep learning models such as convolutional neural networks (CNNs), semantic segmentation networks (*e.g.*, Mask R-CNN [108], DeepLab [112]), or transformers are typically used to generate semantic labels, object masks, or scene descriptions $\mathbf{S}_t$. The output includes categorical distributions, pixel-wise segmentations, and object-level bounding boxes.
2. *Semantic localization*: in this stage, semantic features $\mathbf{S}_t$ are used alongside geometric measurements to enhance the robot's pose estimation $\mathbf{X}_t$. Techniques such as semantic re-weighting of features, use of object-level landmarks, or probabilistic pose refinement using semantic masks are commonly applied.
3. *Semantic mapping*: this stage constructs and maintains a geometric-semantic map $\mathcal{M}$, where each element combines a 3D location l_i and semantic class c_i. The map can be built as sparse object-level landmarks, dense semantic voxel grids, or mesh-based reconstructions. Semantic fusion techniques update the map incrementally by aggregating multi-view observations and resolving class conflicts.

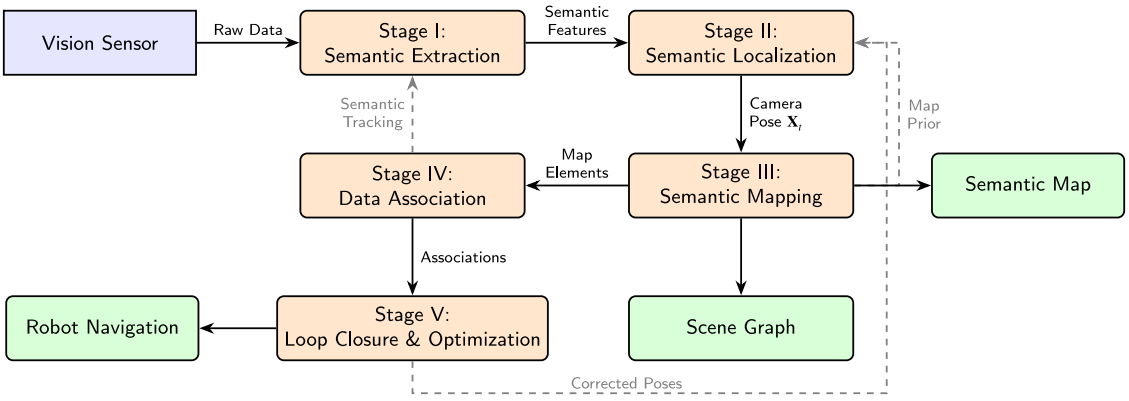

Fig. 2. Semantic vSLAM pipeline flowchart illustrating the information flow through five interconnected stages. Solid arrows indicate forward data flow, dashed arrows represent feedback connections that enable iterative refinement and global optimization.

4. *Semantic data association*: the final stage addresses the temporal consistency of semantic landmarks across frames. This includes matching detected objects to prior map entities, resolving ambiguity in class predictions. Probabilistic frameworks such as maximum-likelihood estimation or expectation–maximization are often employed.
5. *Semantic loop closure optimization*: past frames are revisited, and semantic landmarks are aligned to correct accumulated drift. Optimization is often done via bundle adjustment (BA) or factor graph smoothing.

2.3. Inter-module dependencies and trade-offs

While the five-stage decomposition provides a useful framework for understanding and organizing the Semantic vSLAM literature, it is essential to recognize that these modules are not independent in practice. Real-world systems exhibit tight coupling and bidirectional dependencies between stages, where the output of one module directly influences the performance of others. Understanding these interactions is crucial for designing robust systems and identifying potential failure modes. This subsection discusses both the synergistic gains that arise from effective module integration and the potential conflicts that can lead to system degradation.

2.3.1. Synergistic gains

The integration of semantic information throughout the SLAM pipeline yields substantial benefits that exceed what any individual module could achieve in isolation.

Semantic extraction enhance localization:. High-level semantic features provide robust cues for pose estimation that are inherently more invariant to illumination changes, viewpoint variations, and partial occlusions compared to traditional low-level geometric features [41,51]. By identifying object classes, the system can selectively weight or exclude features based on their expected reliability. Most critically, semantic segmentation enables the identification and filtering of potentially dynamic objects such as pedestrians, vehicles, and animals [52,113]. Excluding features associated with these dynamic classes significantly improves pose estimation accuracy in real-world environments where static world assumptions are frequently violated. Recent studies [114,115] demonstrate that combining semantic priors with geometric verification achieves superior dynamic object handling compared to either approach alone.

Semantic information improve data association:. One of the most significant synergies occurs between semantic extraction and data association [41,84]. Semantic class labels dramatically constrain the search space for correspondence matching. When a detected object is classified as belonging to a specific category, it only needs to be matched against existing landmarks of the same class, rather than all landmarks in the map. This constraint reduces computational complexity from $O(N)$ to $O(N_c)$, where $N_c \ll N$ represents landmarks of class c. Furthermore, semantic information helps resolve perceptual aliasing, a fundamental challenge in SLAM where geometrically similar but semantically distinct entities might otherwise be incorrectly associated [54,55]. For instance, two chairs with similar geometric profiles can be distinguished based on their spatial context or co-occurring objects, preventing erroneous associations that would corrupt the map.

Accurate localization enables consistent semantic mapping:. The quality of semantic maps depends critically on accurate pose estimation [15,57]. Precise camera pose ensures that semantic observation from multiple viewpoints is correctly projected and fused into a coherent 3D representation. When localization is accurate, Bayesian fusion of per-frame semantic predictions effectively reduces classification noise and resolves ambiguities through multi-view consistency [116,117]. Conversely, pose errors cause semantic labels from different frames to be projected to incorrect 3D locations, resulting in label conflicts, blurred object boundaries, and duplicate semantic entities.

Semantic landmarks strengthen loop closure:. Semantic information provides distinctive and stable features for recognizing previously visited places [118,119]. Object-level representations and scene graphs are inherently more robust to viewpoint variations than traditional appearance-based descriptors such as Bag-of-Words [120,121]. The spatial arrangement of semantic entities(*e.g.*, "a desk next to a window with a plant on the shelf") creates unique signatures that can identify locations even under significant appearance changes due to lighting, weather, or minor scene modifications. This semantic-geometric consistency checking substantially reduces false positive loop closures, which are particularly damaging to map quality [58,122].

2.3.2. Potential conflicts and failure modes

Despite the synergistic benefits, the tight coupling between modules also creates pathways for error propagation and system failure. Understanding these vulnerabilities is essential for developing robust Semantic vSLAM systems.

Semantic extraction errors corrupt localization:. When the localization module relies on semantic labels to filter features or weight observation, incorrect semantic extraction can significantly degrade pose estimation [96,113]. False positive dynamic detection occurs when static elements, such as parked cars, statues, or wall textures resembling humans, are incorrectly classified as dynamic objects. Valid features from these regions are then discarded, potentially leaving insufficient constraints for accurate pose estimation, particularly in feature-sparse environments. False negative dynamic detection is equally problematic: when actual moving objects are not recognized as dynamic, their features are included in the optimization, introducing erroneous constraints that cause trajectory drift [114,115]. The severity of this issue depends on the relative motion and feature density of the undetected dynamic object. Class confusion, where semantically meaningful but incorrect labels are assigned, can also mislead downstream processes. For example, confusing a glass door with a wall could cause the system to expect static features where significant appearance changes occur.

Localization drift causes semantic inconsistencies. Accumulated pose errors, particularly during extended operation without loop closures, progressively corrupt the semantic map [58,123]. As drift accumulates, semantic observations from recent frames are projected to locations inconsistent with earlier observations. This manifests as duplicated objects (the same physical entity appearing multiple times in the map), split objects (a single entity fragmented across multiple map regions), and boundary artifacts (inconsistent semantic labels at the interfaces between mapping sessions). These inconsistencies not only degrade map quality but can also mislead the data association module, compounding errors over time.

Computational trade-offs limit real-time performance:. The computational demands of semantic extraction, particularly when employing state-of-the-art deep neural networks, can conflict with real-time requirements for localization and mapping [124,125]. High-accuracy segmentation models often require significant GPU resources and introduce latency that may be unacceptable for time-critical applications. Systems must carefully balance the accuracy benefits of sophisticated semantic models against the processing delays they introduce. Common strategies include running semantic extraction at reduced frame rates [125], using lightweight network architectures [126], processing semantics in parallel threads [52], or selectively applying detailed analysis only to keyframes [127]. Each approach involves trade-offs between semantic richness, temporal consistency, and computational efficiency.

To provide a clear and well-organized presentation, and noting that a substantial amount of existing literature breaks down Semantic vSLAM into five distinct stages, we have opted to review each of these stages individually in Sections 3 to 7. In addition, this structure supports a robust theoretical foundation and allows extensions such as learning-based methods (Section 8), continuous scene representation (Section 9), and multi-robot collaboration (Section 10). It also creates a pathway for incorporating active decision-making for semantic exploration, bridging toward Active Semantic vSLAM.

3. Stage I: Semantic extraction in semantic vSLAM

Semantic extraction is the first and foundational stage in the Semantic vSLAM pipeline, which aims to transform noisy, ambiguous sensor data into structured, robust, accurate, and relevant semantic information for downstream tasks such as mapping, localization, and decision-making. This section delves into the core components of semantic extraction: discerning individual objects through object detection (A), achieving pixel-level understanding via object and instance segmentation (B), recognizing broader environmental contexts through place categorization (C), and refining object representations using contextual priors (D). Table 3 summarizes the key characteristics of different detection approaches and their integration strategies.

3.1. Object detection

Object detection involves predicting bounding boxes around each object and associated semantic labels (*e.g.,* car, human, chair). Before the dominance of deep learning, several traditional methods were prominent, such as the Histogram of Oriented Gradient (HOG) descriptor and Deformable Part Model (DPM). However, these methods are generally less accurate and less adaptable to diverse object classes, particularly in complex scenes. Deep learning, especially CNNs, has dramatically advanced object detection. These methods typically optimize a multi-task loss function, which combines losses for object classification and bounding box regression. Semantic vSLAM systems often utilize single-stage detectors due to their real-time inference capabilities. Among the most widely used are the YOLO series [128,137–144], which unify object classification and bounding box regression into a single forward pass. Another efficient variant is the SSD (Single Shot MultiBox Detector) [130], which uses multi-scale feature maps for detection.

To improve detection accuracy, especially in cluttered or dynamic scenes, many systems adopt two-stage architectures such as R-CNN [145], Fast R-CNN [146], and Faster R-CNN [129]. These methods use region proposal networks (RPNs) followed by refinement networks for accurate object localization and classification. Mask R-CNN [108] further extends Faster R-CNN to include object instance segmentation, which is especially beneficial for overlapping or deformable objects. Recent advances incorporate transformer architectures, exemplified by DETR [147] and DINO [148], which model long-range dependencies in the image and perform set-based global object prediction. These models have demonstrated strong performance in complex environments with occlusions and semantic ambiguity. Some approaches extend object detection into 3D by incorporating depth sensing or monocular depth estimation. Models such as VoxelNet [149], Frustum PointNets [150], and DenseFusion [151] integrate RGB-D to localize objects in 3D space.

In the context of Semantic vSLAM, detected objects serve as semantic landmarks, offering more stable and interpretable features than low-level keypoints [126,152–154]. Formally, given an input image I_t at time t over N_t objects in observation t, the object detector outputs a set of detections:

$$\mathbf{s}_k = (s_k^c, s_k^s, s_k^b)_{k=1}^{N_t} \in \mathbf{S}_t, \tag{11}$$

where s_k^c is the semantic class label, s_k^s is the confidence score, and $s_k^b \in \mathbb{R}^4$ denotes the shape information. These outputs are used to initialize or update object-level landmarks in the semantic map. Hence, several Semantic vSLAM works combine YOLO with state-of-the-art vSLAM methods, such as ORB-SLAM and RGBD-SLAM, for obstacle avoidance and UAV tracking [155], navigation guidance [156], and real-time decision-making in autonomous driving scenarios [22,157]. On the other hand, some works [158,159] integrate Mask R-CNN or Faster R-CNN as a semantic filter, which provides different semantic labels for the image context. Soares et al. [124] explored the trade-off between detection accuracy and inference speed by fusing YOLO and Mask R-CNN outputs in crowded environments. Shao et al. [158] and Zhang et al. [159] proposed systems where Faster R-CNN filters spurious keypoints for robust localization in dynamic environments. Furthermore, systems such as QuadricSLAM [54] and CubeSLAM [40] represent objects as parameterized shapes (quadrics, cuboids) and optimize their poses alongside the camera trajectory.

In summary, object detection is the critical front-end of Semantic vSLAM, bridging raw sensor data with symbolic world understanding. It is essential for semantic landmark identification [160,161], dynamic object handling [124,162], constructing object-aware semantic maps [22,157], and enabling task-specific applications [155,156].

Table 3
Comparison of Semantic Extraction Methods for Semantic vSLAM. This table summarizes the key approaches for extracting semantic information from visual inputs, comparing their output granularity, representative methods, computational requirements, and suitability for real-time SLAM applications.

Method type	Output	Representative works	Complexity	Real-Time	SLAM integration
Object Detection	2D bounding boxes with class labels	YOLO [128], Faster R-CNN [129], SSD [130]	Low–Medium	✓	Object-level landmarks, dynamic object filtering, coarse spatial reasoning
Semantic Segmentation	Dense pixel-wise class labels	FCN [107], DeepLab [131], SegNet [132]	Medium–High	✓/✗	Dense semantic maps, feature masking, scene understanding
Instance Segmentation	Per-instance masks with class labels	Mask R-CNN [108], SOLO [133], PointRend [134]	High	✗	Instance-level mapping, object tracking, multi-object SLAM
Panoptic Segmentation	Unified stuff and things segmentation	Panoptic FPN [135], Panoptic-DeepLab [136]	High	✗	Complete scene parsing, hierarchical mapping, stuff-things distinction

3.2. Object/instance segmentation

While object detection provides coarse bounding box localization of objects, *object* and *instance segmentation* provide a more granular scene understanding by assigning a class label to every pixel (semantic segmentation) or to individual object instances (instance segmentation). This pixel-level classification is invaluable for Semantic vSLAM for several reasons: (1) it enables more accurate geometric modeling of the environment, (2) it improves data association by reducing ambiguities in cluttered or overlapping scenarios, and (3) it provides richer semantic context for tasks such as dynamic object filtering and scene understanding. Popular semantic segmentation networks include U-Net [163], Bayesian SegNet [164], SegNet [132], PSPNet [165] and DeepLabv3+ [166]. For instance segmentation, approaches like Mask R-CNN [108], SOLO [133], and YOLACT [167] are commonly used. Other key methods include PANet [168], which improves information flow between layers, and HTC [169], which introduces multi-stage cascaded heads for refined segmentation. Recently, transformer-based models such as Mask2Former [170] and QueryInst [171] have demonstrated improved performance by leveraging global context and set-based reasoning for mask prediction.

One of the major challenges in SLAM is coping with dynamic objects. Instance segmentation allows for the explicit identification and masking of dynamic regions. For instance, DynaSLAM employs Mask R-CNN for foreground-background separation, ensuring that only static background points contribute to the SLAM backend optimization [113]. Similarly, DS-SLAM [52] uses semantic masks to segment out people and other moving objects, thus enhancing pose estimation stability. Formally, for a given image I_t at time t over N_t object in observation t, the segmentation model produces:

$$\mathbf{s}_k = (s_k^c, s_k^s, s_k^m)_{k=1}^{N_t} \in \mathbf{S}_t, \tag{12}$$

where s_k^c is the class label, o_k^s is the confidence score, and $s_k^m \in \{0,1\}^{H\times W}$ is the binary mask for the kth instance. Segmentation information can be fused with depth maps or point clouds to generate 3D semantic reconstructions. SemanticFusion [15], Fusion++ [23], MID-Fusion [172], and Voxblox [173] demonstrate how semantic masks can be incrementally fused with geometric data. Such dense semantic mapping enhances the robot's ability to reason about spatial relationships and object permanence in dynamic scenes. Beyond 2D segmentation, 3D instance segmentation is increasingly explored. Approaches such as PointGroup [174] and 3D-SIS [175] predict 3D instance masks from RGB-D data or point clouds, providing direct volumetric segmentation useful for dense semantic mapping.

Object and instance segmentation enhance the granularity and reliability of Semantic vSLAM. By providing detailed object contours and instance-level masks, they enable fine-grained reasoning about environmental geometry and semantics, bridging visual perception with 3D mapping and dynamic object management. Semantic map construction benefits from segmentation through voxel-based fusion, as in Bayesian updates by Stückler et al. [116], or the label-oriented voxelgrid filter by Shi et al. [176], and surfel (surface element)-based or segment-based maps [4].

In addition, panoptic segmentation unifies these two tasks, providing a more holistic scene understanding [135,177,178]. It assigns both a semantic label and a unique instance ID (if applicable) to every pixel in the image. The performance of panoptic segmentation is often measured by the panoptic quality metric and Intersection-over-Union (IoU) of predicted segment q and ground truth segment g, which captures both recognition and segmentation quality:

$$\mathcal{A}_{PQ} = \frac{\sum_{(q,g)\in TP} IoU(q,g)}{|TP| + \frac{1}{2}|FP| + \frac{1}{2}|FN|}, \tag{13}$$

where TP, FP, and FN are the set of true positives, false positives, and false negatives, respectively. Conceptually, PQ can be seen as the product of Segmentation Quality (SQ) and Recognition Quality (RQ): $\mathcal{A}_{PQ} = \mathcal{A}_{SP} \times \mathcal{A}_{RQ}$. Panoptic-SLAM [179] and PS-SLAM [180] are recent visual SLAM systems designed for dynamic environments that use panoptic segmentation networks to identify both static background "stuff" and dynamic "thing" instances. This allows for more precise dynamic feature filtering and the creation of more detailed semantic maps

3.3. Place categorization

Place categorization is a pivotal function within Semantic vSLAM, responsible for assigning high-level semantic labels (*e.g.* "corridor", "kitchen", "office", "street scene") to distinct areas within the robot's perceived environment [181–183]. Robust place categorization remains challenging due to environmental variability, including dynamic lighting, occlusions, and seasonal or viewpoint changes. Early techniques relied on holistic image features, attempting to capture the "gist" of a scene using global descriptors. The Bag-of-Words (BoW) model, which quantizes local image features (*e.g.,* SIFT, SURF, ORB) into a "visual vocabulary", became a foundational technique for compact place representation and recognition [184]. However, these handcrafted approaches lack invariance to real-world scene diversity. With the advent of deep learning, CNNs and, more recently, transformers have become the backbone of place categorization. Models like AlexNet [185], VGGNet [186], GoogLeNet [187], and ResNet [188], often pre-trained on large-scale datasets such as ImageNet [189], Places365 [190], and SUN [191], serve as powerful feature extractors or can be fine-tuned for specific scene recognition tasks. Scene-specific classifiers like PlaceNet [192], Places365 CNN [190], or transformer-based models such as ViT [193], and TransVPR [194] have improved the robustness of categorization across different lighting, seasons, and occlusions.

Formally, given an input image I_t to a probability distribution π_t over N_t objects in observation t, the place categorization output is:

$$\mathbf{s}_k = (\text{Categorical}(\pi_t))_{k=1}^{N_t} \in \mathbf{S}_t, \quad \pi_t = \text{softmax}(g_\phi(I_t)), \tag{14}$$

where g_ϕ is a deep scene recognition model, typically the output of the final layers of the CNN. Some systems fuse object-level and

scene-level to jointly enhance contextual reasoning and contextual understanding [195]. This multi-scale fusion supports discriminative feature learning and enables robust global localization even under perceptual aliasing.

In Semantic vSLAM, place categorization thus complements object-level observations by enabling semantic anchoring at the scene level, promoting spatial consistency and high-level task planning [91]. Its integration with geometric reasoning and memory models remains an active area of research. Several studies incorporate place recognition into SLAM, either as a topological constraint for global localization [196], or to augment loop closure [197] and support long-term navigation [198]. In addition, hierarchical Semantic vSLAM systems, such as those proposed by Bowman et al. [41] and Salas-Moreno et al. [111], often use place categories to structure spatial memory, enabling fast retrieval and reuse of semantic submaps. Some methods fuse place categories with scene graphs or room templates to construct hierarchical maps [199]. Recent SLAM frameworks like DS-SLAM [52], PSPNet-SLAM [200], and DRV-SLAM [115] embed place categorization into the SLAM pipeline to provide semantic cues in dynamic and ambiguous environments. For example, semantic loop closures can be performed when the predicted place label matches historical observations, even under significant viewpoint change.

To summarize, semantic extraction in SLAM transforms raw sensor inputs z_t into semantic observations y_t. This process is formally captured as:

$$p(y_t \mid z_t) = \mathcal{S}(z_t), \tag{15}$$

where $\mathcal{S}$ denotes a semantic encoder (*e.g.*, CNN, transformer) producing categorical labels, segmentation masks, and confidence distributions [107,108,201].

3.4. Object regularization via semantic context constraints

While detectors (*e.g.* YOLO [138], DETR [147]) provide initial labels, raw perceptions often lack the consistency and plausibility required for high-fidelity maps. Ignoring context frequently leads to geometric inaccuracies, ambiguities, and physical impossibilities. Object regularization addresses these issues by refining object attributes (pose, shape, class) through semantic context constraints, ensuring the map adheres to real-world physical and logical properties.

A key aspect of regularization is enforcing geometric and shape priors. Objects are often regularized by fitting predefined geometric primitives such as cuboids for furniture, spheres for smaller objects, or planes for walls and tables to their observed point clouds or masks. The parameters of these primitives (*e.g.*, dimensions, orientation) can be constrained based on typical values for an object's semantic class [202]. This process can be formulated as an optimization problem where, for an object with parameters $\mathbf{s}_m = (s_m^p, s_m^b)$ including object pose s_m^p and object shape s_m^b, and observation $\mathbf{Z}_t$, the goal is to find the maximum a posteriori (MAP) estimate:

$$\mathbf{s}_m = \underset{\mathbf{s}_m}{\operatorname{argmax}}(log\, P(\mathbf{Z}_t \mid \mathbf{s}_m) + log\, P(\mathbf{s}_m \mid c_i)), \tag{16}$$

where the likelihood term $P(\mathbf{Z}_t \mid \mathbf{s}_m)$ measures how well the models fit the data (*e.g.*, point-to-model distance). In contrast, the prior $P(\mathbf{s}_M \mid c_i)$ penalizes deviations from expected shapes and sizes for the object's class c_i.

Beyond individual object geometry, topological and relational constraints define the expected spatial arrangements between entities. This includes support relationships (*e.g.*, a "monitor" is on a "desk", a "chair" is on the "floor") and co-occurrence patterns. Such relationships can be modeled explicitly within a probabilistic framework or as factors in an optimization graph. Vasudevan and Siegwart [203] proposed a Bayesian framework where the joint probability of a place concept $\mathbf{s}_k$ depends not only on object counts c_i but also on inter-object relationships r_j (*e.g.*, distance, orientation):

$$P(\mathbf{s_k}, c_{1\dots n_1}, r_{1\dots n_2}) = P(\mathbf{s}_k) \cdot \prod_{i=1}^{n_1} P(c_i \mid \mathbf{s}_k) \cdot \prod_{j=1}^{n_2} P(r_j \mid \mathbf{s}_k). \tag{17}$$

This allows the system to reason about plausible scene configurations. Similarly, graph-based representations, like the semantic graphs proposed by Kong et al. [204], Singh and Leonard [205], and Muravyev et al. [199], can encode these topological constraints as edges between object nodes.

These constraints are typically enforced through several computational methods. In factor graph optimization, which is central to modern SLAM, object properties and their relationships are directly incorporated into the graph [206]. An object's pose $\mathbf{X}_j$ and shape information s_j^b become variables, and semantic constraints are added as factors. For instance, a relational factor between a cup O_j and a table O_k could be expressed as an energy term to be minimized:

$$E_r(\mathbf{X}_j, \mathbf{X}_k) = f_c(IsOn(X_j, s_j^b, \mathbf{X}_k, s_k^b,)), \tag{18}$$

where f_c function penalizes configurations that violate the geometric "IsOn" predicate. Patel et al. [207] describe this principle as "semantically guided local and global pose optimization".

Probabilistic fusion of multiple observations over time also provides powerful temporal regularization. As a robot observes an object from different viewpoints, inconsistencies in pose, shape, or semantic label can be averaged out. For example, voxel-based semantic maps can update the class probability distribution for a voxel v recursively. Building on the work of Stückler et al. (2012) [116], Nakajima et al. [117] proposed updating the class probability $P(s_k^c \mid \mathbf{M}_k, \mathbf{Z}_{1:t})$ for a map region $\mathbf{M}_k$ given measurement $\mathbf{Z}_1 : t$. For instance, using a log-odds representation $l_t(s_k^c \mid \mathbf{M}_k) = \log \frac{P(s_k^c \mid \mathbf{M}_k, \mathbf{Z}_{1:t})}{1-P(s_k^c \mid \mathbf{M}_k, \mathbf{M}_{1:t})}$, the update can be:

$$\begin{aligned} l_t(s_k^c \mid \mathbf{M}_k) = {} & l_{t-1}(s_k^c \mid \mathbf{M}_k) \\ & + \log \frac{P(s_k^c \mid \mathbf{M}_k, \mathbf{Z}_t)}{1 - P(s_k^c \mid \mathbf{M}_k, \mathbf{Z}_t)} - l_0(s_k^c \mid \mathbf{M}_k), \end{aligned} \tag{19}$$

where $l_0(s_k^c \mid \mathbf{M}_k)$ is the prior log-odds. This iterative refinement process regularizes the semantic map by ensuring temporal consistency. A similar principle is applied in S3M-SLAM [4], where semantic surfel labels are updated based on new observations.

In addition, scene parsing can be formulated as an energy minimization problem where an energy function defines the desirability of a particular labeling and geometric configuration of objects $\mathbf{L}_t = \{l_1, \dots, l_N\}$. One effective way to impose regularization is Conditional Random Fields (CRFs) [208,209], which model spatial or semantic dependencies between nearby object predictions. Given a set of initial object detections $\{\mathbf{s}_k\}_{k=1}^{N_t}$, the regularized label assignment $\mathbf{L}_t$ can be inferred by minimizing an energy function:

$$\mathbf{L}_t^* = \arg\min_{\mathbf{L}_t} \sum_k \psi_u(l_k \mid \mathbf{Z}_k) + \sum_{k,k'} \psi_p(l_k, l_{k'} \mid \mathbf{Z}_k, \mathbf{Z}_{k'}), \tag{20}$$

where $\psi_u(l_k)$ is the unary potential derived from object detector confidence (*e.g.*, softmax output), and $\psi_p(l_k, l_{k'})$ is the pairwise potential encouraging label smoothness or enforcing contextual compatibility. The pairwise term often captures semantic relationships, such as co-occurrence frequencies (*e.g.*, a keyboard is likely near a monitor) or spatial configurations (*e.g.*, a monitor is typically above a keyboard). In SLAM pipelines, these constraints can be learned from datasets or designed using geometric rules.

Beyond CRFs, recent work has incorporated graph neural networks (GNNs) to propagate semantic messages across object nodes. Each object proposal is treated as a node v_i with initial features f_i, and a GNN refines these via learned aggregation functions:

Table 4
Comparison of Semantic Localization Strategies in Semantic vSLAM. This table presents different approaches for incorporating semantic information into pose estimation, highlighting their core techniques, advantages, limitations, and representative implementations.

Strategy	Core technique	Advantages	Limitations	Representative works
Feature-based Approaches				
Semantic Feature Selection	Filter/weight features based on semantic class, exclude dynamic objects	Improved robustness in dynamic environments, reduced outliers	Dependent on segmentation accuracy, may discard useful features	DS-SLAM [52], DynaSLAM [113], Blitz-SLAM [114]
Semantic Feature Association	Use semantic labels to constrain feature matching	Reduced search space, improved matching precision	Class confusion propagates to localization	VSO [51], Kimera [57]
Object-based Approaches				
Object Landmark Localization	Use detected objects as landmarks for pose estimation	Compact representation, robust to viewpoint changes	Requires accurate object detection, limited in object-sparse scenes	CubeSLAM [40], QuadricSLAM [54], EAO-SLAM [55]
Joint Object-Pose Optimization	Simultaneously optimize camera poses and object parameters	Mutually beneficial refinement, globally consistent	High computational cost, complex optimization	SLAM++ [111], Fusion++ [23]

$$f_i' = \gamma\left(f_i, \bigoplus_{j\in\mathcal{N}(i)} \phi(f_i, f_j, e_{ij})\right), \tag{21}$$

where ϕ is the message function, e_{ij} encodes the edge (*e.g.*, spatial proximity), $\mathcal{N}(i)$ is the neighborhood, and γ is the update function. TopoNet [210] and Graph-SLAM [211] use topological graphs over semantic landmarks, improving generalization to novel environments by enforcing object-place co-regularity. Other approaches regularize landmark geometry using learned shape priors (*e.g.*, cubes, quadrics), helping constrain ill-posed object pose estimation problems [40,54].

On the other hand, DRV-SLAM [115] addressed the challenge of dynamic or cluttered environments by applying spatial attention between objects and background regions to refine ambiguous detections. In summary, semantic context regularization bridges low-level noisy detection with high-level scene priors, promoting map consistency and aiding downstream tasks like data association and loop closure.

4. Stage II: Semantic localization

Once semantic information has been extracted from sensor data, the next critical stage in Semantic vSLAM systems is Semantic Localization. This process leverages the extracted high-level semantic cues, such as object identities, class labels, and scene categories, to achieve more robust and accurate camera pose estimation than is possible with purely geometric features. Traditional visual localization relies on matching low-level features such as points or lines, which can be unreliable in textureless environments, under significant viewpoint or illumination changes, or in the presence of perceptual aliasing, where different places appear visually similar. Semantic localization addresses these shortcomings by grounding the estimation process in more stable, meaningful, and context-aware landmarks and information. Table 4 categorizes semantic localization strategies into feature-based, object-based, and learning-based approaches, highlighting their core techniques, advantages, limitations, and representative implementations.

Formally, given semantic observations $\mathbf{S}_t$ and geometric measurements $\mathbf{Z}_t$, the goal of semantic localization is to estimate the posterior distribution of the robot's pose $\mathbf{X}_t$:

$$p(\mathbf{X}_t \mid \mathbf{Z}_{1:t}, \mathbf{S}_{1:t}, \mathbf{U}_{1:t}) \propto p(\mathbf{Z}_t \mid \mathbf{X}_t, \mathbf{S}_t) \cdot p(\mathbf{X}_t \mid \mathbf{X}_{t-1}, \mathbf{U}_t), \tag{22}$$

where $\mathbf{u}_t$ represents control inputs. This approach leverages both geometric consistency and semantic coherence to improve localization robustness.

4.1. Semantic feature and landmark-based localization

One of the first approaches is semantic feature and landmark-based localization. This approach advances beyond low-level points by utilizing recognized objects as primary features for tracking and pose estimation, leveraging their rich appearance and geometry as persistent landmarks. VPS-SLAM [212] uses a semantic segmentation network to classify detected planes as 'floor', 'wall', or 'ceiling' and cues these large, stable structures as landmarks for robust localization, especially for aerial robots. Similarly, AVP-SLAM [213] is tailored for autonomous valet parking by using robust environmental semantic features like parking lines, guide signs, and speed bumps as landmarks, which are more stable and reliable than point features in texture-poor parking lot environments. Miller et al. [214] also demonstrated a system where semantic object detection was used for localization on a free-flying robot, showing improved robustness over feature matching. DRG-SLAM [82] combines point, line, and plane features, using semantics to handle dynamic scenes. Its pose optimization minimizes a weighted sum of error from these different geometric features:

$$E(T) = \sum E_{point} + \lambda_L \sum E_{line} + \lambda_P \sum E_{plane}. \tag{23}$$

In addition, semantics can guide the matching of traditional low-level features to establish more reliable, longer-term correspondences. Lianos et al. [51] improve tracking robustness by using semantic consistency to filter point matches between temporally distant frames. By projecting map points into current frames and pruning search candidates based on semantic labels, they establish reliable "medium-term constraints".

4.2. Segmentation-guided localization in dynamic environments

The presence of dynamic objects (*e.g.* pedestrians, vehicles) violates the static world assumption that underpins most traditional visual SLAM algorithms, leading to erroneous feature matching, incorrect pose estimation, and corrupted maps. The most common application is to use segmentation to identify and exclude pixels belonging to dynamic objects from the localization process. Semantic segmentation has become a key strategy for addressing these challenges, enabling robots to distinguish static from dynamic elements, thereby enhancing localization accuracy and robustness. Liu et al. [215] propose a semantic supervision method where features are ranked by significance. A feature point p_i belonging to a semantic class c_i is assigned a significance score $R(p_i)$ based on a predefined class weight $W_s(c_i)$:

$$R(p_i) = W_s(c_i). \tag{24}$$

This score, potentially refined by an attention mechanism, helps the system focus on features from more stable classes to improve pose estimation accuracy. These methods broadly fall into two categories: those that rely solely on semantic priors and those that fuse semantic information with geometric constraints to achieve more reliable motion detection.

The most direct approach to handling dynamic scenes is to use a pre-trained semantic or instance segmentation network to identify and exclude objects from classes that are *a priori* known to be dynamic. In this method, feature points detected on pixels labeled as 'person', 'car', 'bus', *etc.* are simply masked out and not used in the tracking or mapping threads. While efficient and simple, this method is often overly aggressive, discarding valid static features useful for localization. Moreover, it relies entirely on segmentation accuracy, limiting its ability to handle unknown or misclassified objects. To overcome these limitations, most state-of-the-art systems combine segmentation with geometric motion cues to verify true dynamism. This fusion enables the system to distinguish between a parked car (static) and a moving car (dynamic), resulting in more precise and reliable localization. This technique fuses semantic segmentation with optical flow to identify potentially dynamic regions. Notable implementations like SOF-SLAM [216] and PSPNet-SLAM [200] leverage this combined approach to build robust SLAM systems.

In addition, some researchers employ semantic information to exclude dynamic objects to enhance localization accuracy in dynamic environments. One of the first attempts [217] proposed a technique that uses semantic segmentation to identify moving vehicles in autonomous driving scenarios and treats them as outliers for removal. Based on this, SaD-SLAM [218], DGS-SLAM [219], Blitz-SLAM [114], Dynamic-DSO [220], DySt SLAM [221], and WF-SLAM [222] also leverage instance segmentation by combining epipolar geometry constraints to eliminate dynamic objects. Excluding features associated with moving objects helps the SLAM system reduce localization inaccuracies caused by these dynamic components. Yet, this strategy might inadvertently eliminate crucial features that could be advantageous for localization in specific contexts. To address this challenge, PLD-SLAM [223], YPD-SLAM [224], DRG-SLAM [82], and PLDS-SLAM [225] combine some novel features, such as point, line, and plane, to enhance the robustness of feature tracking using clustering methods and epipolar geometry. On the other hand, to achieve efficiency in computational complexity, several studies [125,127,226,227] extract keyframes and conduct semantic extraction only in these selected keyframes. These papers cluster dynamic points using K-means or random sample consensus (RANSAC)-based or using Bayesian filtering to exclude points with high dynamic probability. Another approach is to reduce the computational cost of deep learning by utilizing a light-weight semantic model. DRE-SLAM [96] and Crowd-SLAM [126] use YOLOv3/Tiny with clustering or domain-specific training to detect dynamic features and exclude them from tracking. YOLO-SLAM [152] and CFP-SLAM [228] leverage depth and epipolar geometry to filter out high-motion regions after YOLOv3/v5 detection. Other systems such as DO-SLAM [229], OVD-SLAM [230], and MVS-SLAM [231] extend ORB-SLAM2/3 with YOLOv5/v7 and geometric outlier rejection to improve localization robustness in highly dynamic scenes.

Furthermore, multi-view geometry provides powerful constraints for identifying dynamic points. For a feature point p_f observation in two frames, its 3D position can be triangulated. Its reprojection error e_k in a keyframe k can be calculated as:

$$e_k = u_k - \pi_k(T_{kw}, p_{fw}), \tag{25}$$

where u_k is a observed feature location, p_{kw} is the triangulated 3D point in world coordinates, T_{kw} is the camera pose, and π_k is the projection function. For points on dynamic objects, this reprojection error will be large. DynaSLAM [113] employs instance segmentation with geometric verification and inpainting, while MaskFusion [53] tracks and reconstructs multiple moving objects. To filter dynamic features, optical flow within semantic masks is compared with the camera's ego-motion; if they are inconsistent, the features are considered dynamic and removed. Detect-SLAM [232] similarly combines an SSD object detector with motion consistency checks. DynaNav-SVO [233] extends this by constructing a region-of-interest (ROI) from *a priori* static urban elements, ensuring reliable feature extraction. DS-SLAM [52] utilizes SegNet for semantic filtering, whereas DynaSLAM. Blitz-SLAM [114] implements a dual-phase strategy, initially interpreting scenes via deep learning and subsequently confirming them geometrically. CFP-SLAM [228] uses a hierarchical method grounded in object detection and motion classification, while SG-SLAM [234] combines semantic interpretation with geometric constraints within a graph-based framework. Zhang et al. [235] evaluate scene quality based on semantic extraction to optimize camera pose estimation. Dynamic elements d_t can corrupt pose and map estimation. We define a binary mask M_t indicating static regions:

$$d'_t = M_t \odot d_t, \tag{26}$$

where $\odot$ is the element-wise product. However, once dynamic objects are masked, they leave "*holes*" in the image. To address this, frameworks like DynaSLAM [113], IDV-SLAM [236], and Empty Cities [237] use background inpainting techniques, often powered by generative adversarial networks (GANs) [238], to fill in the regions occluded by dynamic objects. This restores a complete and static view of the scene, providing a more stable input for subsequent visual odometry and mapping tasks. To maintain real-time performance, RDS-SLAM [125] decouples segmentation into a parallel thread, allowing high-frequency tracking while periodically updating dynamic masks to prevent map corruption.

While traditional dynamic SLAM methods often treat moving objects as outliers to be excluded, recent research emphasizes tracking dynamic objects and jointly estimating their motion with camera poses. DynaSLAM II [239] and Dynamic SLAM [96] jointly optimize camera and object trajectories using semantic segmentation and motion models, generating both static and dynamic maps. DOT-SLAM [240] and PointSLOT [241] apply instance segmentation and object-based BA to track rigid object motions. Several systems combine semantic instance segmentation, optical flow, and photometric consistency for multi-object tracking and motion estimation [242]. For example, MOTSLAM [243], DE-SLAM [244], and TwistSLAM [245] use joint optimization frameworks that fuse semantic, geometric, and motion cues to estimate object and camera poses. SLAMANTIC [246] and VDO-SLAM [247] use dynamic confidence and motion constraints to enhance map reliability. More recent approaches, such as ClusterVO [248] and OTE-SLAM [249], integrate tracking algorithms ByteTrack [250] with semantic object detection, while also estimating dynamic object poses through twist parameterization or joint pose graphs. This provides a much richer understanding of the environment, which is essential for applications like collision avoidance and behavior prediction in autonomous driving.

5. Stage III: Semantic mapping

While semantic extraction provides the raw semantic labels for objects, regions, and places, Semantic Mapping is concerned with integrating this information into a persistent, structured, and queryable representation of the environment. As discussed in the review by Mascaro and Chi [256], building a rich and actionable model of the world is a crucial capability for autonomous systems, moving beyond simple geometric point clouds to denser, semantically-aware representations. This section reviews how semantic maps are represented 5.1, how semantic information is integrated and fused into these maps 5.2, and how to evaluate semantic uncertainty and formulate cost functions 5.3. Table 5 presents a comprehensive comparison of semantic map representations spanning discrete representations (sparse points, dense clouds, voxels, surfels, meshes), object-centric representations (bounding boxes, quadrics), and hierarchical representations (scene graphs, topological maps).

Table 5
Comparison of Semantic Map Representations in Semantic vSLAM. This table categorizes different map representation types, their data structures, semantic integration methods, memory efficiency, and typical application scenarios.

Representation	Data structure	Semantic integration	Memory	Scalability	Representative works
Discrete Representations					
Sparse Point Map	3D points with semantic labels	Per-point class labels, label propagation	Low	High	Dyna-SLAM [251], DS-SLAM [52]
Dense Point Cloud	Dense 3D points with RGB and semantics	Per-point semantic features, Bayesian fusion	High	Low	SemanticFusion [15], ElasticFusion [252]
Voxel Grid/TSDF	Regular 3D voxels with semantic probabilities	Per-voxel label voting, probabilistic fusion	Medium–High	Medium	Voxblox++ [253], Kimera [57]
Surfel Map	Surface elements with position, normal, semantics	Per-surfel semantic attributes, incremental update	Medium	Medium	SurfelMeshing [254], Co-Fusion [50]
Mesh Map	Triangular mesh with per-vertex/face semantics	Vertex labeling, texture mapping	Medium	Medium	Kimera [57], Hydra [58]
Object-Centric Representations					
3D Bounding Box	Oriented cuboids with class labels	Object-level semantics, instance ID	Very Low	High	CubeSLAM [40], EAO-SLAM [55]
Quadric Surface	Dual quadric ellipsoids	Object class and pose encoding	Very Low	High	QuadricSLAM [54]
Object-level TSDF	Per-object volumetric reconstruction	Instance-aware dense reconstruction	Medium	Medium	MaskFusion [53], Fusion++ [23]
Hierarchical and Topological Representations					
Scene Graph	Nodes (objects/places) with edges (relationships)	Semantic attributes, spatial relations	Low	High	Hydra [58], ConceptGraphs [73]
Topological Map	Graph of places with semantic descriptions	Place-level semantics, connectivity	Very Low	Very High	LEXIS [75], Questmaps [255]

5.1. Semantic map representations

The choice of semantic map representations is crucial as it dictates how information is stored, updated, accessed, and utilized by the robot. Different representations offer trade-offs in terms of expressiveness, scalability, memory efficiency, and suitability for particular tasks. Unlike geometric maps (*e.g.*, point clouds, occupancy grids), semantic maps encode high-level scene information. A semantic map can be formally represented as a tuple:

$$\mathcal{M}_i = \{(\mathbf{X}_i, \mathcal{F}_i, \mathbf{s}_i)\}_{i=1}^{N}, \tag{27}$$

where $X_i \in SE(3)$ denotes the pose of an element, $\mathbf{s}_i$ is the semantic information, and $\mathcal{F}_i$ is set of observed features.

5.1.1. Object-centric maps

These maps explicitly model the environment as a collection of distinct object instances. Each object is typically stored with its semantic class label, 3D pose (position and orientation), dimensions or shape parameters, and potentially other attributes like appearance features or functional affordances. In many SLAM systems, distinct and recognizable objects serve as semantic landmarks. These landmarks are associated with their 3D positions and class labels in the map. Civera et al. [202] proposed an early Semantic vSLAM that merged traditional point features with known 3D object models, where objects, once recognized, were inserted and tracked in the EKF SLAM map. SLAM++ [111] used a 3D CAD model to identify and render objects, building a map composed of full 3D object geometries. However, these systems are constrained by their reliance on pre-defined object templates. Several studies [160,257] estimate object poses and incorporate them into the map, effectively creating an object-level representation. However, they often incur high computational costs and pose a challenge to achieving real-time system performance.

These frameworks focus on object SLAM by modeling objects as dual quadric formulation [54,258], rectangular bounding volumes [40], specific objects [259], and estimating fine geometric models of objects using a CNN trained [258]. Dynamic object-level SLAM has been considered in [234,260]. Object-oriented semantic mapping, for example by Sünderhauf et al. [16] and Canh et al. [22] aims to create a meaningful map that contains instances of known objects. These methods provide object information for robotics tasks such as navigation or obstacle avoidance. To improve performance on these tasks, Mascaro et al. [261] present an individual object-level semantic mapping pipeline that integrates 3D instance segments and a refined diffusion scheme. Hu [262] also introduces a multi-level map, including inaccurate object modeling and limited object representation, based on geometric, plane, and object maps, particularly for constructing long-term consistent maps. Li et al. [263] model scene text as planar semantic features, effectively creating a map of text objects with geometric and semantic properties.

On the other hand, Fusion++ [23] uses 2D instance mask predictions and fuses them into the TSDF reconstruction within globally consistent loop-closed object SLAM maps. Voxblox++ [253] also combined geometric segmentation, instance-aware segmentation refinement, data association, and map integration to create an object-centric map. Following this line of research, TSDF++ [264] introduces a multi-object mapping approach that uses separate reconstruction volumes for each object, based on the TSDF formulation. The studies conducted by Asgharivaskasi and Atanasov [42,265] stand out as they introduce a multi-class (semantic) OctoMap. These works employ a closed-form lower bound on the Shannon mutual information between map and range-category observations to determine information-rich robot trajectories.

5.1.2. Volumetric semantic maps

These maps discretize 3D space into elementary units (points, surfels, or voxels) and store semantic information within each unit.

Voxel-based semantic maps:. These extend traditional occupancy grids like OctoMap by associating each occupied voxel (or OctoMap leaf node) not just with an occupancy probability, but also with a semantic label or a probability distribution over multiple semantic classes. Stückler et al. [116] proposed a Bayesian framework to fuse probabilistic object-class segmentations from multiple RGB-D views into such a voxel-based 3D semantic map. For a voxel v and class s^c, the

posterior probability $P(s^c \mid v, Z_{1:t})$ given measurements $Z_{1:t}$ is updated recursively:

$$P(s^c \mid v, \mathbf{Z}_{1:t}) \propto P(s^c \mid v, \mathbf{Z}_t) \frac{P(s^c \mid v, \mathbf{Z}_{1\ldots t-1})}{P(s^c \mid v)}. \tag{28}$$

Kochanov et al. [260] present a probabilistic mapping approach, which uses recursive Bayesian filtering to update voxel occupancy and semantic labeling based on observation and scene flow [176,266]. Other methods [160,267] also create an "improved Octomap" where voxels can store semantic object information. Dyna-SLAM [251] uses semantic and sparse flow cues to identify and classify dynamic objects and uses voxel block hashing for large-scale reconstruction. CNN-SLAM [49] predicts depth map using a CNN network and creates a dense semantic map based on the retraining of the network for semantic segmentation using soft-max later and cross entropy function. Kimera [57] introduced an adaptation of bundled raycasting to build a global 3D mesh using Voxblox and a TSDF model, and to semantically annotate it with 2D semantic labels, label propagation, and Bayesian updates. Li et al. [268] build a dense map upon ORB-SLAM and a semi-global stereo matching algorithm for disparity map generation. Shi et al. [176] introduced a "label-oriented voxelgrid filter" that ensures intra-frame spatial continuity and inter-frame spatiotemporal consistency when fusing 2D semantic labels into a 3D voxelized map. Liu et al. [72] integrate vision-language foundation models to improve the construction of a generalizable instance-aware semantic map. Qian et al. [269] introduces a novel probabilistic object state representation that models both stationarity and magnitude of geometric change for each object and a Bayesian update rule that incorporates geometric and semantic information for consistent map maintenance.

Surfel-based semantic maps:. Surfel representations model surfaces using a collection of 3D discs or ellipses. Semantic information can be attached to each surfel. Stuckler et al. [270] integrate object-class segmentation, SLAM, and semantic 3D fusion into a real-time operating semantic mapping system based on multi-resolution surfel maps. Canh et al. [4] in S3M, build a semantic sparse map for UAVs using a semantic surfel cloud, where each surfel sf_k stores geometric attributes (position p_k, normal n_k, radius r_k) and a semantic label distribution $L_k = \{P(s_1^c \mid sf_k), \ldots, P(s_N^c \mid sf_k)\}$. The semantic label is updated based on projected 2D segmentations based on the current frame's segmentation L_k^{obs}:

$$L_k^{new} = \eta \cdot L_k^{obs} + (1 - \eta) \cdot L_k^{old}. \tag{29}$$

Reddy [271] combines classical semantic segmentation and motion constraints to separate dynamic objects from static scenes, which starts with the computation of low-level features like SIFT descriptors, optical flow, and stereo disparity, and reconstructing the environment. Building on a similar idea, MaskFusion [53] tracks multiple moving objects even when they move independently from the camera and reconstructs them as surfel clouds. SemanticFusion [15] and Co-Fusion [50] utilize CNNs and state-of-the-art dense SLAM - ElasticFusion to predict semantics from multiple viewpoints and fuse them into a surfel map using a Bayesian update scheme. Morreale et al. [272] achieve dense mapping by iteratively collecting free-space tetrahedra and applying semantic simplification. Seichter et al. [273] extend Normal Distributions Transform (NDT) mapping by storing separate Gaussian distributions for each semantic class per cell, enhancing accuracy and consistency.

Panoptic semantic maps:. To address the limitation of lacking the ability to distinguish individual instances belonging to the same category in higher-level semantic understanding, panoptic representations aim to integrate class labels for background areas (stuff), while simultaneously segmenting and identifying distinct foreground objects (things) one by one. Nakajima et al. [117] describes the four components of the method: dense approach of InfiniTAMv3 [274] for SLAM, 2D semantic segmentation with a specifically designed CNN, incrementally building a geometric 3D map, and updating class probabilities assigned to each segment of the geometric 3D map. PanopticFusion [56] introduces a novel online volumetric semantic mapping system that densely predicts class labels of stuff and things based on 2D panoptic label prediction, panoptic label tracking, thing label probability integration, and online map regularization. Pham et al. [275] designs an inference-optimal segmentation from predictions of a DNNs, then progresses super-voxel clustering to achieve a real-time dense reconstruction of 3D indoor scenes. Yang and Liu [276] implemented geometric segmentation to discover novel scene elements and refine them using panoptic segmentation results from UPSNet [277], and then associated temporal data using VPSNet [278]. Panoptic Multi-TSDFs [279] introduced a hierarchical map representation and temporal hierarchy based on panoptic entities structured as submaps, then integrated measurements into active submaps using projective updates and TSDF weighting. PanopticNDT [280] presented an efficient and robust panoptic mapping approach based on occupancy NDT mapping by combining a subsequent mapping stage and a panoptic segmentation stage using EMSANet [273], which includes semantic and instance prediction, fusion, and confidence score generation. PanoRecon [281] addressed the challenge of realized online geometry reconstruction along with dense semantic and instance labeling by incrementally performing 3D geometric reconstruction and 3D panoptic segmentation in a view-independent 3D feature volume. EPRecon [282] achieved real-time performance by integrating a lightweight module for depth prior estimation in 3D volume and a novel panoptic features extraction strategy that combines voxel features and image features.

5.1.3. Topological-semantic maps

These maps represent environments as graphs where nodes correspond to places or regions (*e.g.*, rooms, corridors) and edges signify connectivity or traversability. This representation is particularly useful for high-level path planning and human–robot interaction. Bernuy and Solar [283] define the topological semantic map (TSM) as a graph structure that describes roads and their environments, including types, spatial positions, object lists, traversability indices, curvature indices for road nodes, and odometry based on semantic descriptions of images. Some studies [196,284,285] also address the challenge of autonomous driving applications by leveraging semantic observations of the environment, a TSM for storing selected semantic observations, and a topological localization algorithm that uses a Particle Filter to obtain the vehicle's pose in the TSM. Zhao et al. [286] utilized DNNs to identify indoor scenes without training in a specific environment and estimated the semantic region. Topomap can improve navigation task performance by simplifying the problem and reducing the computational cost of path-planning algorithms, while providing sufficient knowledge of traversable spaces [287–289].

3D scene representation or 3D scene understanding is a fundamental and essential problem in robotics. Several studies [73,255,290–292] focus on this challenge by constructing 3D scene graphs to model entire buildings and rooms, and to include semantics for objects (*e.g.*, shape, class, material). Armeni et al. [290] represent semantic information in a 3D four-layer structured scene graph based on 3D mesh models, registered RGB panoramas, and camera parameters. Kim et al. [291] propose a 3D scene graph framework for intelligent agents that integrates data processing via Adaptive Blurry Image Rejection (ABIR) and Keyframe Group Extraction (KGE) to reduce redundancy. It employs Spurious Detection Rejection (SDR) using 3D position and Word2Vec semantics to construct a directed graph, where nodes represent objects and edges represent their relations. Wu et al. [292] create a globally consistent scene graph by fusing node feature extraction, edge feature computation, graph neural network (GNN) feature propagation, and class prediction based on a feature-wise attention (FAT) mechanism, which re-weights individual latent features at each target node to handle missing neighboring points. Mehan et al. [255] present an indoor environment as a multi-channel occupancy representation based on room-mask prediction and transition-region detection, and

formulate it as a 3-class instance segmentation task using the Mask R-CNN architecture. Gu et al. [73] create object-based 3D mapping by class-agnostic 2D segmentation and estimating spatial relationships between objects by calculating the 3D bounding box IoU to obtain a similarity matrix. Soucsa and Bassani [293] consolidate deep visual features extracted by GoogLeNet [187] and shallow Multilayer Perceptron (MLP) to create representations of regions and use a moving average and visual habituation mechanism. Yang et al. [294] parse and vectorize indoor architecture from floor-plan raster maps, which leverage a learning-based hierarchical approach to identify a set of geometric primitives with semantics and use of mixed integer programming (MIP) to fuse primitives and their relationship information into vector graphics while enforcing high-level structural constraints. Upon this idea, Cao et al. [295] present a Context-Enhanced Full-Resolution Network (CEFRN) for floor-plan segmentation to improve the accuracy of topological semantic maps for the Partially Sighted or Visually Impaired (PSVI). In addition, some studies [119,122] use the topology of the object-level map to achieve the effectiveness and robustness of the loop closure process. Fredriksson et al. [296] identify and classify intersections for semantic and topological mapping, then find paths between intersections, dead ends, and pathways to unexplored areas to generate the robot paths. SENT-Map [297] also provides exciting potential for robots operating in complex human environments. PRISM-TopoMap [199] is a recent online topological mapping method that uses learnable multimodal place recognition to build and maintain the graph structure without relying on global metric coordinates.

5.1.4. Hybrid and hierarchical maps

These maps employ hierarchical maps that combine the strengths of different representations across multiple levels of abstraction, like metric, semantic, and topological, to bridge the gap between detailed and strategic planning. For example, executing high-level directives such as *"go to the living room and bring the coffee mug I left on the table"* requires developing a functional world model that captures the relationships among semantic entities and integrates real-world details across multiple levels of abstraction. The first attempt by Tomatis et al. [298] integrates metric and topological paradigms into a hybrid system for both localization and map building using two levels of abstraction in combination with the different types of environmental structures. A similar approach by Kuipers et al. [299] defines four distinct representations of spatial knowledge in the Spatial Semantic Hierarchy (SSH) for building the tree of topological maps on the basis of SSH. Galindo et al. [80] introduce a multi-hierarchical framework merging spatial (metric) and conceptual (semantic) hierarchies. Anchoring bridges these levels by linking abstract symbols to sensor data representing the same physical object. Drouilly et al. [300] develop a new scheme for updating maps of a large-scale dynamic environment using stable semantic representation, which updates semantic layers of the map directly and models changes in terms of static class occlusions caused by dynamic objects. Yang et al. [301] propose a hierarchical Conditional Random Field (CRF) model for joint grid label optimization, incorporating unary, pairwise, and high-order potentials. Unary potentials are derived from dilated CNN predictions, pairwise potentials utilize Gaussian kernels for color and spatial similarity, and high-order terms employ a robust P^n Potts model [302] to enforce label consistency within superpixel cliques.

Malleson et al. [303] introduce a hybrid 4D model approach that combines prior surfel-graph modeling with higher-resolution volumetric fusion, based on pairwise data, smoothness costs, label costs, intra-part fusion, and inter-part fusion. Luo and Chiou [304] combine a 2D occupancy grid map with an overlaid topological graph, which is organized by a hierarchical semantic organization structure, linking abstract concepts with tangible objects, while Wen et al. [305] introduce 3D semi-dense based on Mask R-CNN for navigation and semantic information storage. Yue et al. [306] propose a hierarchical framework for collaborative semantic 3D mapping based on multimodal semantic information fusion and collaborative semantic map fusion using the EM algorithm. Building upon this idea, Deng et al. [307] construct a graph model and apply a label diffusion method to generate an accurate hierarchical and dense semantic occupancy map. Rosinol et al. [308] combine 5 layers and hierarchical representations, including metric-semantic, objects and agents, places and structures, and rooms and buildings, into a single model: a 3D scene graph. Zhang et al. [309] introduce a human-inspired object search strategy that builds a metric-topological map and selects the optimal search node by considering object co-occurrence relationships and robot location. Recent papers by Hughes et al. [58,310] present a real-time spatial perception system for incremental construction of scene graph layers, including local mesh and object Euclidean Signed Distance Field (ESDF) generation, topological place subgraph extraction using a Generalized Voronoi Diagram, and room segmentation using a community-detection approach. H2-Mapping [62] and its improved version, H3-Mapping [64], leverage a hierarchical hybrid representation with a quasi-heterogeneous feature grid to achieve real-time performance and high-quality construction, reducing redundant feature grid allocation and improving texture training efficiency with limited sampling and training time. Wald et al. [311] presented a Scene Graph Prediction Network (SGPN) for generating a graph describing objects (nodes), including a class hierarchy and a set of attributes describing visual and physical appearance and their relationships (edges) with different categories of relationships including spatial/proximity relationships, support relationships, and comparative relationships. Khronos [123] addresses the Spatio-Temporal Metric-Semantic vSLAM (SMS) problem by constructing a dense 4D world model. It supports long-term autonomy through local estimation within an active window and global optimization using Truncated Least Squares (TLS) loss with Graduated Non-Convexity (GNC) [312] in GTSAM [47]. Following this approach, Clio [313] defined the task-driven 3D scene understanding problem, where the robot builds a compressed map representation and dynamically selects the level of detail and objects for its maps based on the task. This method utilized FastSAM [314] and CLIP [315] for segmentation, temporal association, and 3D reconstruction, and built upon Hydra [58] to construct 3D place primitives and describe how CLIP embeddings are associated with places.

5.2. Semantic fusion through multiple observations

A core challenge in semantic mapping is distilling noisy, often conflicting per-frame semantic predictions into a single, coherent, globally consistent 3D semantic map. Semantic fusion is the process that addresses this challenge by aggregating information from multiple viewpoints and observations over time. This process is fundamentally reliant on multi-view consistency, where a robust underlying SLAM system provides accurate camera poses and establishes frame correspondences. These correspondences enable semantic labels for the same 3D points or regions, observed from different perspectives, to be associated (in the data association step) and then fused, thereby reducing uncertainty and correcting errors in individual predictions.

The dominant approach for fusing semantic labels in dense representation (*e.g.,* voxels or surfels) is probabilistic. This method treats the labels of each map element as random variables and iteratively updates their class probability distributions using new observations. A foundational technique involves recursive Bayesian update [270]. For each map element v, a probability distribution $P(s_v^{cf})$ over the set of semantic classes s^c is maintained. When a new observation $\mathbf{Z}_t$ is registered, the semantic predictions from a CNN at the corresponding pixel location are used as the likelihood $P(Z_t \mid s_v^{cf} = s^c)$ to update the prior probability $P(s_v^{cf} = s^c)$. In log-odd form, this update for each class s^c is:

$$s_{t,v,c}^{cf} = s_{t-1,v,c}^{cf} + \log \frac{P(s_v^{cf} = s^c \mid \mathbf{Z}_t)}{1 - P(s_v^{cf} = s^c \mid \mathbf{Z}_t)}. \tag{30}$$

This fusion across multiple independent observations effectively averages out noise from the segmentation network, yielding a more accurate and stable semantic map. This principle is widely used in systems such as SemanticFusion [15], as well as in other voxel-based and segment-based maps. PanopticFusion [56] achieves this by maintaining separate probability distributions for semantic class (stuff) and instance IDs (things) for each voxel. For a voxel v, the semantic distribution $P_{\mathbf{s}}(s^c \mid v)$ is updated with a standard Bayesian update. The instance distribution $P_I(i \mid v)$ for instance i is updated differently to prevent instances from blending into each other. Specifically, it uses a non-normalized update and then assigns the voxel to the single instance with the highest probability. The update, for instance, i at voxel v given observation Z_t is:

$$P_t(i \mid v) = P(\mathbf{Z}_t \mid i) P_{i-1}(i \mid v). \tag{31}$$

In addition, several articles [23,53,123,251,264,316,317] use the IoU tracker [318], and define the overlap threshold to match predicted objects from the previous frame to the current frame. To enhance the ability of this method, some works [253,261] introduce a 3D IOU score, which calculates a pairwise of instance segments s_i^c with the number of points $|s_i|$ and object instance $\mathbf{s}_o$:

$$IOU(s_i^c, s^o) = \frac{\Pi(s_i^c, \mathbf{s}_o)}{|s_i| + \Pi(s_o^c, \mathbf{s}_o) + \sum_{i \neq i'} \Pi(s_{i'}^c, \mathbf{s}_o)}. \tag{32}$$

where $\Pi(s_i^c, s^o)$ is the total number of points belonging to object $_o$. Besides that, Mascaro et al. [261] propose a technique that first performs probabilistic fusion and then refines the result via label diffusion on a voxel graph. They construct a k-Nearest Neighbor (k-NN) graph of the voxels and formulate an energy minimization problem to find a consistent labeling s^{cf}. The energy function balances fidelity to the observed data with spatial smoothness:

$$Q(s^{cf}) = \sum_{i \in C} (p_i^r - p_i^f)^2 + \mu \sum_{i,j \in N_k} w_{ij} \left(\frac{p_i^r}{\sqrt{d_i}} - \frac{p_j^r}{\sqrt{d_j}} \right)^2, \tag{33}$$

where p_i^f is the initial fused probability for an observed voxel i, the set C, p_i^r is the refined distribution, and the second term regularizes the solution by encouraging neighboring voxels to have similar labels, normalized by their degree d. This diffusion process propagates strong semantic predictions from confident regions to less certain or unobserved regions, resulting in a more complete and consistent semantic map. Iro et al. [290] use a voting scheme for multi-view consistency, which defines the weight for each frame f_i based on robot position $\mathbf{x}_i$ as $w_{i,j} = \frac{\sum_{i,j} \|\mathbf{x}_i - f_{cj}\|}{\|\mathbf{x}_i - f_{cj}\|}$. However, this method does not account for object uncertainty, especially in complex environments. Popular methods [4,72,267,301,319,320] include Bayesian updates or other probabilistic aggregation that recursively combine new observations Z_t with prior belief:

$$P(\mathbf{s} \mid v, \mathbf{Z}_{1:t}) \propto P(\mathbf{s} \mid v, \mathbf{Z}_t) \cdot \frac{P(\mathbf{s} \mid v, \mathbf{Z}_{1:t-1})}{P(\mathbf{s} \mid v)}, \tag{34}$$

where $P(\mathbf{s} \mid v, \mathbf{Z}_t)$ is the semantic classifier output. This fusion reduces uncertainty and corrects individual frame errors, yielding coherent semantic maps. These studies [73,269,291] construct a pair of semantic and geometric similarity with respect to all objects in the map by calculating position similarity score R_{cp} and color similarity score R_{cc} and cut-off by a predefined threshold θ_{cutoff} and find the final fusion by an optimal strategy like Hungarian algorithm or greedy assignment. Alternative fusion methods include learned fusion networks [49], splat rendering for projective fusion, and an update scheme similar to previous surfel-based approaches [50], self-supervised feature fusion using deep vision transformers (DINO, CLIP), and generating open-set instance clusters and refining label certainty with repeated observation [71,74]. These methods also enforce spatial and temporal label consistency.

5.3. Semantic uncertainty and cost function

Beyond guiding feature selection, semantic information can be integrated directly into the mathematical core of localization and mapping. This is achieved by either (1) formulating probabilistic observation models that explicitly account for semantic uncertainty or (2) designing novel cost functions for optimization-based SLAM that penalize semantic and geometric inconsistencies. These two concepts are deeply intertwined, as the uncertainty model often informs the weighting of terms in the cost function.

5.3.1. Probabilistic observation models and semantic uncertainty

In probabilistic localization frameworks (*e.g.*, particle filters, Kalman filters), beliefs about the robot's state are updated using an observation likelihood model, $P(\mathcal{Z} \mid \mathcal{X}, \mathcal{M})$, which quantifies the probability of making a measurement $\mathbf{Z}$ for a pose $\mathbf{X}$ and map $\mathbf{M}$. Semantic localization enriches this by incorporating semantic measurements $\mathcal{S}$. However, semantic classifiers are imperfect, they produce probabilistic outputs and can make errors. Therefore, robustly handling the uncertainty of these semantic predictions is crucial. A naive approach might use only the highest-probability class prediction, discarding valuable uncertainty information [260]. Akai et al. [33] model the uncertainty of the entire class probability vector using Dirichlet distributions. This approach naturally down-weights uncertain measurements from ambiguous or occluded objects, resulting in robust pose estimation. The Dirichlet distribution $\mathbf{Dir}(\theta \mid \boldsymbol{\alpha})$ over a simplex of class probabilities θ and $\boldsymbol{\alpha}$ is the concentration parameter of the Dirichlet. In their framework, when an object is observed, the classifier outputs a probability vector $\pi_k = (\pi_{k,1}, \ldots, \pi_{k,C})$ over C classes. The core idea is to compute the likelihood of observing the specific vector π_k given the predicted probabilities from the map at the robot's hypothetical pose. This likelihood, $P(\pi \mid \mathcal{M}, \mathbf{X}_t)$, is used to update the weight of the particle filter. It is calculated by marginalizing over all possible tree class distributions θ_k:

$$P(\pi_k \mid \mathcal{M}, \mathbf{X}_t) = \int p(\pi_k \mid \theta_k) p(\theta_k \mid \mathcal{M}, \mathbf{X}_t) d\theta_k. \tag{35}$$

Assuming the likelihood $p(\theta_k \mid \mathcal{M}, \mathbf{X}_t)$ follows a Dirichlet distribution with parameters α_k, this integral has a closed-form solution:

$$P(\pi_k \mid \mathcal{M}, \mathbf{X}_t) = \frac{\Gamma\left(\sum_{c=1}^{C} \alpha_{k,c}\right)}{\prod_{c=1}^{C} \Gamma(\alpha_{k,c})} \prod_{c=1}^{C} \pi_{k,c}^{\alpha_{k,c}-1} \tag{36}$$

where $\Gamma(\cdot)$ is the Gamma function, and the Dirichlet parameters $\alpha_{k,c}$ are set based on the map's prediction $p_m(c)$ for that location and a concentration parameter λ that reflects confidence: $a_{k,c} = \lambda p_m(c) + 1$. In addition, active mapping aims to plan a robot's path to reduce the map's uncertainty most effectively. Lu et al. [321] quantify the semantic uncertainty of mapped semantic object instance $\mathbf{s}_i$ using the entropy of its fused class probability distribution $p(s^c|\mathbf{s}_i)$. The total semantic uncertainty of the map $\mathcal{M}$ is the sum of entropies over all objects:

$$\mathbb{H}(\mathcal{M}) = -\sum_{\mathbf{s}_i \in \mathcal{M}} \sum_{s^c \in C} p(s^c \mid \mathbf{s}_i) \log p(s^c \mid \mathbf{s}_i). \tag{37}$$

5.3.2. Semantic cost functions

In optimization-based SLAM, camera poses are typically found by minimizing a cost function, often a sum of geometric reprojection errors. A semantic cost function augments this objective with terms that penalize semantic inconsistencies between the current observation and the map. VSO [51] introduces a semantic consistency term into the optimization. In addition to the standard geometric reprojection error, it adds a semantic error term that penalizes assigning a 3D map point to a semantic class inconsistent with the image segmentation. The total cost function over a set of map points:

$$E = E_{\text{geo}} + \lambda E_s, \tag{38}$$

and E_s captures reprojection error in the semantic label space:

$$E_s = \sum_{k,i} e_s(k,i),$$
$$e_s(k,i) = \sum_{c \in C} w_i(c) \cdot log(p(\mathbf{s}_k \mid \mathbf{X}_k, \mathbf{M}_i, s_k^c = c)), \tag{39}$$

where $w_i(c)$ the class weight for landmark i. Direct semantic alignment approaches optimize over semantic probability maps, which can be more robust to lighting changes than photometric error alone [51]. Instead of merely aligning pixel intensities, methods such as SDVO [322] also use a direct semantic alignment approach, aligning semantic probability maps generated by segmentation networks, which are less susceptible to illumination changes. The semantic alignment error E_{sem} can be formulated as:

$$E_s = \sum_{i \in \mathcal{P}} w_{s^c} \|\mathbf{P}_s(i) - \mathbf{P}'_s(i')\|_\gamma, \tag{40}$$

where $\mathbf{P}_s(i)$ is the semantic probability vector at pixel i in the current frame, $\mathbf{P}'_s(i')$ is the projected semantic probability vector from the map at the corresponding pixel i', w_{s^c} is the heuristic weighting factor for the semantic s^c, and $\|\cdot\|_\gamma$ is the Huber norm. CNN-SLAM [49] enhances a direct SLAM method - LSD-SLAM [323] by integrating a depth map predicted by a CNN. The system's energy function includes a term that penalizes deviations from the CNN's depth prediction, but the prediction's uncertainty weights it. This allows the system to trust the geometric photometric alignment in high-gradient regions and fall back on the learned depth prior in textureless areas. Malleson et al. [303] formulated a semantic cost function based on pair-wise and MDL (minimum description length) formulation:

$$E_s = \sum_{i \in \mathcal{P}} c_{i,s_i^c} + \sum_{ij \in \mathcal{N}} V_{ij}(s_i^c, s_j^c) + \sum_{m \in \mathcal{M}} \text{MDL}_m, \tag{41}$$

where $\sum_{i \in \mathcal{P}} c_{i,s_i^c}$ is data cost, which combines the mean of Euclidean distance between input point tracks and modeled point tracks with an incompleteness penalty, $\sum_{ij \in \mathcal{N}} V_{ij}(s_i^c, s_j^c)$ is the smoothness cost, which calculates a cost for each edge ij in the total $\mathcal{N}$ edges in sufel graphs, and $\sum_{m \in \mathcal{M}} \text{MDL}_m$ is label cost, which adds a fixed cost for each label for each models in the set $\mathcal{M}$ with at least one point assigned to it.

On the other hand, when objects are represented as geometric primitives such as quadrics, planes, or text, the cost function can be formulated to optimize the parameters of these objects and the camera poses directly. Hosseinzadeh et al. [258] propose an object-aware BA where the cost function minimizes the reprojection error of both 3D points Z_j and points on the surface of object quadrics Q_k. The error for a point on a quadric is the distance between the observed 2D point $\mathbf{x}_{ik}$ and the projection of the 3D quadric:

$$E = \sum_{i,j} (\|\pi(\mathbf{X}_i, \mathbf{Z}_i)\|^2_{\sum_{ij}}) + \sum_{i,k} (\|\pi(\mathbf{X}_i, Q_k)\|^2_{\sum_{ik}}), \tag{42}$$

where $\pi(\mathbf{X}_i, \mathbf{Z}_i)$ is the projection of the 3D quadric Q_k into the camera frame with pose $\mathbf{X}_i$. TextSLAM [263,324] uses a cost function that combines standard point reprojection error E_{point} with a photometric error computed directly on planar text features E_{text}, anchoring pose estimates to these stable semantic landmarks. Yang et al. [294] also integrates the pixel-wise corner likelihood E_d^c of corner information and pixel-wise edge likelihood E_d^ϵ of edge information into a semantic cost function.

By explicitly modeling semantic uncertainty and incorporating semantic terms into optimization cost functions, these methods create a powerful synergy between geometric estimation and high-level scene understanding, leading to significant gains in robustness, accuracy, and overall capability.

6. Stage IV: Semantic data association

Data association is a crucial, often challenging, process of establishing correct correspondences between current sensor observations and elements already stored in the map. It is the fundamental problem of answering the question: "Is this object I see now the same physical object I have seen before?". While traditional data association relies on geometric and appearance-based cues, semantics provide a powerful additional layer of information, making this process significantly more robust and efficient. Table 6 compares geometric-based, semantic-constrained, probabilistic, and learning-based data association methods, covering their matching criteria, uncertainty handling, and computational characteristics.

Mathematically, the data association problem can be framed as finding the optimal assignment hypothesis, C^*. The goal is to find the most likely set of pairings between measurements and landmarks:

$$C^* = \underset{C}{\text{argmax}}\, P(C \mid \mathcal{Z}_t, \mathcal{M}_{t-1}). \tag{43}$$

However, this formulation obscures the problem's immense combinatorial complexity. For multiple measurements and landmarks, the number of possible association hypotheses is enormous, making a brute-force search computationally infeasible for any non-trivial scenario. This inherent complexity is compounded by several significant real-world challenges that introduce ambiguity, including perceptual aliasing, viewpoint and appearance variation, sensor noise and measurement uncertainty, and dynamic environments.

6.1. Geometric and appearance-based approach

Most modern SLAM systems use deterministic or score-based data association methods that are computationally efficient. These approaches typically combine multiple sources of information – semantic, geometric, and appearance – into a unified scoring function to determine the best match for an observation. This multi-cue approach mitigates the risk of relying on a single, potentially unreliable source of information. The fundamental idea is to first use the semantic class as a hard constraint to prune the search space drastically. A new observation is only compared against existing landmarks in the map. After this filtering, various cues are combined to score the remaining potential matches.

Sünderhauf et al. [16] and Zhang et al. [160] match 3D points in the landmark and temporary semantic object based on nearest neighbor search and k-d tree. This method, however, faces challenges in noisy or complex environments, where obstacles and the rapid motion of objects create uncertainty. Similarly, some studies [258,326] also match the projected landmarks with bounding boxes by using the Hungarian/Munkres [327] algorithm to find the optimal solution by minimizing the cost matrix. CubeSLAM [40] is a classic example that performs data association for both static and dynamic objects by finding the map candidate with the highest 2D bounding box IoU with the current detection. This is further refined by geometric distance checks and an optional appearance-similarity score, demonstrating a simple yet effective multi-cue approach. A sophisticated approach is to combine multiple cues in a weighted score explicitly. Mu et al. [328] introduce the concept of a nonparametric pose graph to address the challenge of data association and SLAM with an unknown number of objects and ambiguous data association. This method uses Dirichlet Process prior to model data associations $D_{\mathbf{s}_i}^{\mathbf{Z}_k}$, allowing for the inference of objects $\mathbf{s}_i$ and their association with observation $\mathbf{Z}_k$:

$$p(D_{\mathbf{s}_i}^{\mathbf{Z}_k} = i) = \text{DP}_i = \begin{cases} \frac{N_{\mathbf{Z}_i}}{\sum_i N_{\mathbf{Z}_i} + \alpha} & 1 \le i \le N_{\mathbf{s}_i}, \\ \frac{\alpha}{\sum_i N_{\mathbf{Z}_i} + \alpha} & i = N_{\mathbf{s}_i} + 1, \end{cases} \tag{44}$$

where $N_{\mathbf{Z}_i}$ is the number of observation of the ith-object in the total of $N_{\mathbf{s}_i}$ objects, and α is the concentration parameter of DP prior. Based on this paper, EAO-SLAM [55] proposes an "ensemble data association" score, which applies the Wilcoxon Rank-sum test to point cloud data and the Single-sample T-test to object centroids. The observation is associated with the map landmark that determines the null hypothesis

Table 6
Comparison of Semantic Data Association Methods in Semantic vSLAM. This table presents different approaches for establishing correspondences between observations and map elements, comparing their matching criteria, handling of uncertainty, and computational characteristics.

Method	Core approach	Advantages	Limitations	Representative works
Geometric-based Methods				
Nearest Neighbor	Match based on geometric distance (Euclidean, Mahalanobis)	Simple, computationally efficient	Sensitive to noise, ambiguous in cluttered scenes	LSD-SLAM [323]
ICP Variants	Iterative closest point alignment	Accurate for dense data, well-established	Local minima, requires good initialization	ElasticFusion [252], KinectFusion [325]
Semantic-Constrained Methods				
Class-Constrained Matching	Only match observations with the same semantic class	Reduced search space, fewer false matches	Fails with class confusion, misses cross-class associations	Bowman et al. [41], DS-SLAM [52]
Semantic Similarity Matching	Match based on semantic feature similarity (*e.g.*, CLIP embeddings)	Open-vocabulary capability, robust to appearance changes	Computationally expensive, requires feature extraction	ConceptFusion [71], FM-Fusion [72]
Probabilistic Methods				
Maximum Likelihood	Select association maximizing observation likelihood	Principled, handles measurement noise	Point estimate, ignores association uncertainty	CubeSLAM [40], Quadricslam [54]
Expectation-Maximization	Iteratively estimate associations and states	Handles soft assignments, principled uncertainty	May converge to local optima, computationally intensive	Bowman *et al.* [41], Doherty *et al.* [84]

by these scores. However, the performance of this method is highly dependent on the careful tuning of the pre-defined threshold, which may not be optimal across different environments or object types.

Chen et al. [259] employ a two-stage association strategy inspired by [328], combining FairMOT [329] for coarse geometric filtering with a Gaussian Mixture Model (GMM) [330] for fine-grained semantic matching. Although effective for distinguishing similar objects, this hierarchical approach risks premature rejection during the coarse stage. Separately, recent systems have begun leveraging learned shape priors. With the advent of deep learning, some systems use learned shape priors. DSP-SLAM [331] uses a deep shape decoder to predict a low-dimensional shape code for each object. Data association is then performed by finding the mapped object with the most similar shape code, making the association more robust to textureless objects. The framework proposed by Wu *etal*. [332] provides a comprehensive scoring function that combines 2D bounding box overlap, 3D object distance, and appearance similarity (from ORB features) to determine the best association. The object with the highest combined score is selected as the match. However, like other score-based methods, it relies on a challenging assignment, which can be brittle in ambiguous situations with multiple good candidates.

Geometric and appearance-based approaches typically achieve efficient computational performance for semantic data association in Semantic vSLAM. However, the primary weakness is that these methods make a single, "hard" decision for each association based on a score threshold. In ambiguous situations where multiple good candidates exist (*e.g.* two identical chairs nearby), this can lead to an incorrect match. Unlike probabilistic methods that can maintain multiple hypotheses, a single incorrect hard assignment can introduce a permanent error into the system. The performance of score-based methods is highly dependent on the careful tuning of weights and thresholds. These parameters are often hand-tuned for a specific environment or sensor setup and may not generalize well to new conditions, requiring tedious recalibration. The performance of the upstream object detector fundamentally limits these methods. A missed detection means no association can be made, and an incorrect class label will prevent a valid association by prematurely pruning the correct landmark from the search space at the first step.

6.2. Probabilistic approach

To overcome the brittleness of making "hard" decisions about the single best match, probabilistic frameworks compute a probability for every potential association. This approach is more robust to the inherent ambiguity in real-world scenes, as it does not have to commit to a single, possibly incorrect, correspondence. These methods are particularly well-suited to integration into factor graph-based SLAM systems, where they can be represented as soft, probabilistic constraints.

A highly effective method, pioneered by Bowman et al. [41], is to treat data association and landmark class labels as latent variables, utilizing the EM algorithm. The EM algorithm offers an alternative, iterative approach to probabilistic association between two steps: (1) E-Step: given the current best estimates for the states of the mapped objects, this step computes the probability, or "responsibility", $\gamma_j i$ that measurement j originated from object k. This step associates observed semantic entities $\mathbf{s}_t$ with existing map entries $\mathcal{M}$ and landmark position L_i. The association likelihood can be written as:

$$p(s_t^c \mid \mathbf{s}_t, \mathcal{M}) \propto \sum_i \delta(s_t^c = s_i^c) \cdot \mathbb{I}[\mathbf{s}_t \sim (\mathbf{l}_i, s_i^c)], \tag{45}$$

where δ is the Kronecker delta and $\mathbb{I}$ is an indicator function measuring similarity [41]. (2) M-Step: using these responsibilities as weights, this step updates the state of each object (*e.g.* pose, shape, velocity) by computing a weighted average over all measurements. This step is formulated as a nonlinear least squares problem by combining multiple factors, including semantic factors $f_{kj}^s(\mathbf{X}_i, L_j)$ for each landmark j, geometric factors $f_j^y(\mathcal{X})$, and inertial factors $f_i^I(\mathcal{X})$:

$$\begin{aligned}\hat{\mathbf{X}}_{1:T}, \hat{L}_{1:M} = \underset{\mathcal{X}, L_{1:M}}{\operatorname{argmin}} &\sum_{k=1}^{K}\sum_{j=1}^{M} f_{kj}^s(\mathcal{X}, L_{1:M}) \\ &+ \sum_{i=1}^{N_y} f_i^y(\mathcal{X}) + \sum_{t=1}^{T-1} f_t^I(\mathcal{X}).\end{aligned} \tag{46}$$

EM-Fusion [333] uses an EM framework for robustly tracking multiple dynamic objects, where the responsibilities help to handle occlusions and complex interactions. Parkison et al. [334] propose a Semantic ICP algorithm, where EM is used to find the optimal alignment between two point clouds. The semantic labels are incorporated into the E-step to influence the correspondence probabilities, making the alignment process more robust to poor initializations. The joint likelihood based on residual information $\mathcal{R}$ is formulated as:

$$p(\mathcal{R}, \mathcal{S}, \mathcal{D} \mid \mathcal{X}) \propto p(\mathcal{R}, \mathcal{S} \mid \mathcal{D}, \mathcal{X}) p(\mathcal{S} \mid \mathcal{D}, \mathcal{X}) p(\mathcal{D} \mid \mathcal{X}). \tag{47}$$

Doherty et al. [83] propose the use of multimodality for posterior inference over poses and landmarks, given a factor graph to accommodate non-Gaussian variables using nonparametric belief propagation,

which approximates the belief over continuous state variables using Gibbs sampling and kernel density:

$$p(\mathcal{X}, \mathcal{L} \mid \mathcal{X}) \propto \prod_{\varphi}(\mathcal{X}, \mathcal{L}, \mathcal{Z}) \prod_{\psi} \psi(\mathcal{X}, \mathcal{L}). \tag{48}$$

where φ and ψ are a measurement factor and a prior factor, respectively. Another approach by Doherty et al. [84] uses the "max-marginal" instead of the "sum-marginal" to mitigate the effects of data association errors while preserving the Gaussian nature of the problem.

$$p(\mathcal{X}, \mathcal{L} \mid \mathcal{X}) \propto \prod_{i} f_i(V_i), V_i \in \{\mathcal{X}, \mathcal{L}\}. \tag{49}$$

In a factor graph, this creates a single, non-Gaussian factor for each measurement that connects to the robot pose and all potential landmark matches. The negative log-likelihood of this mixture model is minimized as the factor's cost. This makes the system extremely robust to outliers and ambiguity. If a measurement is geometrically and semantically consistent with two different landmarks, it will contribute a partial constraint to both, avoiding a risky, hard assignment. This framework has been extended to multimodal SLAM, where semantic labels from different sensors are fused within the probabilistic association. However, this method requires evaluating each measurement against every plausible landmark in the map. In large-scale environments with thousands of landmarks, this can become a significant bottleneck. Michael et al. [85] directly address this challenge by proposing several approximations to make probabilistic data association scalable. These include considering only a local subset of landmarks for association (*e.g.*, within a radius of the robot) and using multi-resolution semantic maps to perform coarse-to-fine association. These optimizations significantly reduce the computational burden without sacrificing too much of the robustness that makes the probabilistic approach intensively powerful. Sünderhauf et al. [54,110] factor the conditional probability distribution as:

$$\begin{aligned} P(\mathcal{X}, \mathcal{Q} \mid \mathcal{Z}, \mathcal{L}) \propto & \prod_{i} P(\mathbf{X}_{i+1} \mid \mathbf{X}_i, \mathbf{Z}_i) \\ & \prod_{ijk} P(Q_j \mid \mathbf{X}_i, L_{ijk}), \end{aligned} \tag{50}$$

where $\mathcal{Q}$ and $\mathcal{L}$ are the set of quadric and line, respectively. The optimal X^*, Q^* are sought as optimal values based on maximum a posteriori (MAP). Zhang et al. [335] propose a hierarchical topic model in which an object is analogous to a document, semantic classes to topics, and visual features (*e.g.*, SIFT words) to words. Data association is then performed by calculating the probability that a new observation (a "document" of "words") belongs to an existing object "document" based on their shared "topic" and "word" distributions. This allows the system to reason about object identity at a higher level of abstraction.

In multi-robot SLAM, data association must also occur between the maps of different robots. Yue et al. [306] and Deng et al. [307] propose hierarchical frameworks where this association is based on both geometric consistency and semantic similarity. The semantic similarity between two matched objects or voxels, with class distributions $p_i(s^c)$ and $p_j(s^c)$, can be quantified using the Kullback–Leibler (KL) divergence:

$$D_{KL}(p_i \,||\, p_j) = \sum_{c} p_i(s^c) \log \frac{p_i(s^c)}{p_j(s^c)}. \tag{51}$$

A low KL divergence indicates high semantic similarity, increasing the probability of a correct association. In these collaborative mapping frameworks, a server agent can aggregate submaps from multiple robots, solve this large-scale data association problem to find the correct global alignment of all maps, and then fuse them into a unified global semantic map.

Despite their robustness, probabilistic data association frameworks face significant limitations. Performance relies heavily on upstream perception, missed detections or misclassifications can zero out association weights, preventing correct matches regardless of geometric proximity. Computational cost scales rapidly with map size, making real-time operation in object-rich environments difficult without approximations like local sub-maps, which trade robustness for speed. Initialization sensitivity in iterative methods like EM can lead to local minima and incorrect associations. Furthermore, environments with identical objects (*e.g.*, rows of chairs) remain challenging, as semantic and geometric cues may fail to distinguish instances, leading to ambiguous maps. Finally, dynamic occlusions can cause hypothesis explosions and track switching when motion models are insufficient.

6.3. Advanced data association contexts

Beyond associating static objects in a scene, data association methods must handle more complex scenarios, including dynamic objects, articulated models, and long-term changes. This requires more advanced reasoning and contextual understanding. In dynamic environments, data association is synonymous with tracking—maintaining a consistent label s_i^c for each object over time, regardless of its motion. DynaSLAM II [239] is a seminal work that tightly couples the tracking of multiple dynamic objects with the camera's ego-motion estimation. Data association involves matching new object segmentation from Mask R-CNN to existing object tracks. This is guided by a motion model (*e.g.*, constant velocity), which predicts the object's bounding box in the current frame. The IoU between the predicted and detected boxes is then used to establish the correspondence. These methods enhance accurate tracking and real-time performance, but their reliance on a simple motion model may fail for objects with abrupt or nonlinear motion, leading to track loss. MID-Fusion [172] formulates a cost matrix between newly observed object segments and existing dynamic object models in the map. The cost $C(\mathbf{s}_i, \mathbf{M}_k)$ for associating segment i with model k combines a geometric term and a semantic consistency term. While effective for robust camera tracking, continuous object information estimation, and real-world applications, the computational complexity of solving the assignment problem scales with the number of objects and detections, which can be a bottleneck in highly cluttered scenes.

Instead of relying on a top-down object detector, Iqbal and Gans [336] use a nonparametric clustering algorithm like DBSCAN applied to low-level feature points to group them into spatial clusters. A data association step then matches these geometric clusters to existing object models in the map. While this method can discover novel objects, it cannot assign them a semantic label without an additional classification step. Compositional Object SLAM [257] performs data association at the part level. This allows for robust tracking even when the object's overall configuration changes. However, this approach requires pre-defined part-based models for each object class, which can be difficult and time-consuming to create. PS-SLAM [180] associates panoptic "things" by verifying overlap with reprojected dynamic objects. High overlap links instances to existing tracks, while insufficient overlap initializes new ones. However, this strict reliance on segmentation quality renders the system vulnerable to failures caused by fragmented or incorrect masks.

In summary, advanced data association methods provide robust tracking and improved efficiency in complex environments. However, tracking methods for dynamic scenes often rely on simplified motion models. These models are insufficient for capturing complex, nonlinear, or abrupt motions, which can lead to lost tracks or incorrect associations. In addition, the requirement for *a priori* models and the lack of semantics in unsupervised methods limit the usefulness of the resulting map for higher-level semantic tasks, unless an additional classification stage is incorporated. The effectiveness of these specialized methods may be limited to the specific contexts for which they were designed. A part-based model for a chair will not help with a rigid object, and a dynamic object tracker adds unnecessary computational overhead in a static scene. Furthermore, the complexity of managing and associating data for a large number of dynamic, articulated, or collaboratively mapped objects remains a significant scalability challenge.

Table 7
Comparison of Semantic Loop Closure and Optimization Methods in Semantic vSLAM. This table summarizes different approaches for detecting revisited places and optimizing the global map, comparing their detection strategies, semantic utilization, and optimization frameworks.

Method	Core approach	Advantages	Limitations	Representative works
		Place Recognition Methods		
Global Descriptors	Holistic image representation (NetVLAD, GeM)	Viewpoint robust, compact representation	May miss fine-grained differences, training required	NetVLAD [192], CosPlace [338]
Semantic BoW	Bag-of-words with semantic object categories	Object-level matching, appearance invariant	Requires reliable object detection, coarse matching	Arshad *et al.* [120], Gawel *et al.* [339]
Scene Graph Matching	Match semantic scene graph structures	Structural similarity, relationship-aware	Graph matching complexity requires accurate graphs	Hydra [58], ConceptGraphs [73]
		Geometric Verification		
RANSAC-based	Robust estimation with outlier rejection	Effective outlier handling, widely applicable	Computational cost, may fail with high outlier ratio	ORB-SLAM [12], VINS-Mono [340]
Semantic Consistency Check	Verify semantic label consistency between candidates	Reduces false positives, efficient filtering	Dependent on segmentation quality	Yu *et al.* [118], Cao *et al.* [119]
		Back-End Optimization		
Pose Graph Optimization	Optimize poses with relative pose constraints	Efficient, decoupled from mapping	Does not refine landmarks, may accumulate errors	g2o [45], GTSAM [47]
Bundle Adjustment	Joint optimization of poses and 3D structure	Globally consistent, refines geometry	Computationally expensive, scalability issues	Ceres [341], iSAM2 [342]
Factor Graph with Semantics	Include semantic factors in optimization	Unified geometric-semantic optimization	Complex factor design, increased dimensionality	Bowman *et al.* [41], Rosinol *et al.* [57]
Semantic Map Correction	Propagate loop closure to semantic labels	Consistent semantic map after correction	Label re-fusion overhead, potential conflicts	Kimera [57], Hydra [58]

7. Stage V: Loop closure optimization for semantic vSLAM

While sequential tracking and mapping build a locally consistent view of the environment, the incremental nature of this process inevitably leads to the accumulation of minor errors. Over long trajectories, this results in significant drift, where the robot's estimated pose and the map become increasingly inaccurate and inconsistent. Loop closure is the process of recognizing a previously visited place, estimating accumulated drift, and correcting the map and trajectory to enforce global consistency. It is the critical step that transforms a locally accurate trajectory into a globally correct map. Traditionally, loop closure has been achieved using appearance-based place recognition methods like Bag-of-Words (BoW) [337], which are susceptible to perceptual aliasing and viewpoint changes. Semantics provides a higher-level, more robust source of information for both detecting loop closures and optimizing the map afterward. This section details how semantics enhance loop closure detection (Section 7.1) and the subsequent pose graph optimization (Section 7.2). Table 7 summarizes loop closure detection strategies and back-end optimization approaches, including place recognition methods, geometric verification techniques, and semantic-aware optimization frameworks

7.1. Semantic loop closing

Detecting a loop is the first and most critical step in achieving global consistency. Traditional place recognition methods, which rely on matching low-level visual features, are often brittle and prone to failure under significant viewpoint changes or perceptual aliasing (where different places appear visually similar). Semantic information provides a higher level of abstraction, enabling the system to recognize high-level scene understanding based on its meaning and structure, which is inherently more robust.

A common and effective strategy is to combine traditional appearance-based methods with a semantic consistency check, which leverages the speed of appearance-based matching while using semantics as a powerful verification step to reject false positives. First, an appearance-based method generates a set of loop closure candidates. Second, each candidate is scored for semantic consistency. CALC2.0 [343] computes a semantic score based on the similarity score of two corresponding observations to determine a loop closure. Arshad and Kim [120] propose a weighted fusion of similarity scores to improve robustness. The similarity score $R_s(\mathbf{Z}_i, \mathbf{Z}_j)$ for semantic loop closure can be defined as:

$$R_s(\mathbf{Z}_i, \mathbf{Z}_j) = \sum_{k=1}^{N} w_k \cdot \mathrm{R}(\mathbf{s}_{i,k}, \mathbf{s}_{j,k}), \tag{52}$$

where w_k is a weighting factor, and $\mathrm{R}(\cdot,\cdot)$ is a similarity metric like the Jaccard index and cosine similarity of semantic embeddings. Hu et al. [344] and SV-Loop [345] propose a similar fusion, where the final similarity is a weighted sum of the BoW score and a semantic score derived from the consistency of detected object labels. This strategy is also effective in specialized environments, such as a forest, where semantic cues like "trunk" and "canopy" are used to verify loop closures proposed by an appearance-based method. This hybrid approach maintains the efficiency of traditional methods while significantly improving precision by rejecting geometrically plausible but semantically inconsistent matches. It is still fundamentally reliant on the initial appearance-based candidate generation. If the BoW model fails to propose the correct loop closure candidate due to extreme viewpoint or illumination changes, the semantic check cannot recover from this failure.

To address this challenge, some studies integrate semantic information directly into the BoW model to generate more distinctive visual words, rather than treating semantics as a separate verification step. A "semantic visual vocabulary" is created where each visual word is associated not only with its appearance but also with a semantic label [346]. However, this approach is constrained by the pre-trained visual vocabulary's limitations, which limit its applicability in real-world environments. By clustering the description vector along with the robot's navigation component to enhance visual word generation, system accuracy improves at the expense of computational cost. Papapetros et al. [347] tightly integrate semantics into the Bag-of-Words (BoW) model by assigning the semantic class of the region where a feature was extracted to its visual word. This makes the descriptor more discriminative and helps resolve initial perceptual aliasing. However, this approach risks discarding useful information,

particularly for features at region boundaries or in areas with ambiguous segmentation. Alternatively, a 3D semantic graph shifts the paradigm from matching holistic scene appearance to matching the spatial arrangement of semantic objects, which is often more robust to viewpoint changes. In this graph, nodes represent object instances and edges define their spatial and topological relationships, allowing a place to be described as a "constellation" of objects. Loop closure is then detected by finding a structural match (an isomorphism) between the graph of the current scene and a previously constructed graph in the map. SemanticTopoLoop [295] builds a topological graph from object landmarks (quadrics) and calculates a similarity score between a query graph G_q and a candidate graph G_c:

$$R(G_q, G_c) = R_c + \rho R_s, \tag{53}$$

where R_c is the spatial semantic similarity and R_s is the topology structure similarity based on the consistency of topological relationships. SemanticLoop [118] and the work by Lin et al. [122] also use similar graph-matching approaches, which calculate matching reward based on vertex similarity and edge similarity. This approach achieves efficient drift correction under drastic scene and viewpoint changes, but accuracy and robustness can be affected by partially reconstructed areas and the challenges of the outdoor environment.

Qian et al. [121] introduce a 3D semantic covisibility graph where edges encode object co-visibility from keyframes, adding robust topological constraints. This structural approach relies on scene geometry rather than appearance, ensuring high robustness to viewpoint and illumination changes while handling minor environmental shifts (e.g., moved furniture) via partial graph matching. However, it may fail in sparse environments. Furthermore, because graph matching is NP-hard (Nondeterministic Polynomial-time), even with heuristics, it can be computationally expensive for very large graphs. Xiao et al. [348] apply a similar two-stage approach using 360 ° panoramic images. After candidate retrieval, they project the query image's semantic segments into the candidate image's frame. The final score combines visual similarity and the ratio of reprojection-consistent semantic pixels, effectively measuring how well the semantic layouts align. This method makes the place descriptor more viewpoint-invariant from the outset, but it requires specialized panoramic-camera hardware and does not apply to standard monocular or stereo SLAM systems. Kim and Kim [349] propose a method where, after an initial set of loop candidates is generated, a semantic consistency check is performed. They compute a semantic score based on the uniqueness score of semantic object labels between the query and candidate frames. Singh et al. [350] propose a two-level descriptor for this study. The top-level "Semantic-Geometric Descriptor" (SGD) is a histogram that encodes the types and relative poses of objects in the scene. The lower-level descriptor captures local geometric details. A coarse search is performed efficiently with low-dimensional SGD, and only the top candidates are then compared via the more expensive fine-grained matching. This approach is highly efficient and scalable, as it avoids costly geometric comparisons for the vast majority of non-matching keyframes. However, performance depends on the distinctiveness of the high-level semantic constellations. In environments with many repetitive yet semantically similar areas (*e.g.*, a street with identical bus stops), the top-level filter may be ineffective. Chen et al. [351] focus on this problem in industrial settings. Their method first performs a coarse loop closure detection using object detection to identify all objects in both the query and candidate frames. By comparing the object IDs, it explicitly identifies objects that have moved or been removed between the two views. These inconsistent instances are masked out, and the final geometric verification for the loop closure is performed only on the stable, static parts of the scene. In addition, to handle extreme viewpoint variations, PRIOR-SLAM [352] uses a novel view synthesis approach. It generates a canonical view of the scene from the current keyframe's observation. Loop closure is then performed by matching these canonical views, which are much more consistent across different robot trajectories than the original egocentric views. The similarity score is computed from a learned descriptor of the synthesized canonical image.

In summary, integrating semantics into loop closure detection represents a significant evolution from simple appearance matching to a more robust form of scene understanding. The various strategies, from using semantic labels to verify appearance-based candidates, building hierarchical descriptors for efficient search, to matching object constellations and scene graphs for viewpoint invariance, all contribute to a common goal: improving the accuracy and robustness of semantic loop closure detection. By providing a higher level of abstraction, semantics help overcome the key limitations of traditional methods, especially in challenging real-world scenarios involving perceptual aliasing, dynamic elements, and extreme viewpoint or illumination changes.

7.2. Semantically-informed pose graph optimization

Once a loop closure has been detected between the current robot pose $\mathbf{x}_i$ and a previous pose $\mathbf{x}_j$, a loop closure constraint is added to the pose graph. This constraint represents the relative transformation ΔT_{ij} between two poses. The final step is to optimize the pose graph to find the configuration of all robot poses that best satisfies both the sequential odometry constraints and the new loop closure constraints, thereby distributing the correction across the entire trajectory. Semantic enhances this back-end optimization by providing more robust constraints and a richer multi-layer optimization problem. A typical SLAM back-end is formulated as a factor graph, a graphical model in which nodes represent robot poses and landmark positions, and factors represent constraints between them. The optimization problem to find the set of poses $\mathbf{X}^*$ and landmark positions $\mathbf{L}^*$ that minimizes a non-linear least squares cost function is represented as:

$$\begin{aligned}\mathbf{X}^*, \mathbf{L}^* = \underset{\mathbf{X},\mathbf{L}}{\operatorname{argmin}} & \underbrace{\sum_{(i,j)\in\mathcal{E}_G} \|e(\mathbf{X}_i, \mathbf{X}_j), \Delta T_{ij}\|^2_{\Omega_{ij}}}_{\text{odometry constraints}} + \\ & \underbrace{\sum_{(k,l)\in\mathcal{E}_L} \|e(\mathbf{X}_k, \mathbf{X}_l), \Delta T_{kl}\|^2_{\Omega_{kl}}}_{\text{loop closure constraints}} + \\ & \underbrace{\sum_{(i,m)\in\mathcal{E}_M} \|e(\mathbf{X}_i, \mathbf{X}_m), \mathbf{z}_{im}\|^2_{\Omega_{im}}}_{\text{landmark measurement constraints}},\end{aligned} \tag{54}$$

where $e(\cdot)$ is an error function, and Ω denotes the information matrix (inverse covariance) representing the confidence in the constraint. When the objects are used as landmarks, they provide strong geometric constraints. If a loop closure is established by matching several objects, the relative transformation can be computed by finding the rigid body transform that best aligns these matched objects. Furthermore, objects with known shapes (*e.g.*, from a CAD model or a learned prior) can be used to add object pose constraints to the graph. The error term for an object observation is the difference between the object's properties and the projection of its 3D model into the camera frame. VSO [51] includes a semantic error term in its optimization that penalizes assigning a 3D map point to a semantic class that is inconsistent with the image segmentation, effectively adding a soft semantic constraint to the graph. SemanticLoop [118] uses alignment from its graph matching to generate a highly reliable pose constraint for the loop closure. Iqbal and Gans [336] use nonparametric clustering to form object-level landmarks from low-level features. These object landmarks are then used as nodes in the pose graph optimization. This approach enables object-level constraints without relying on a pre-trained object detector, making it adaptable to novel environments. VPS-SLAM [212] is designed for aerial robots and uses semantic segmentation to identify these planes. It adds planar constraints to the pose graph, where the error term is the point-to-plane distance between observed 3D

points and the corresponding plane model in the map. This provides strong geometric stability, especially for correcting drift in height and orientation.

In addition, richer semantic representations, such as 3D scene graphs, enable a transition from a single-layer pose graph to a more complex, hierarchical optimization problem. Hydra [58] builds a hierarchical system with a three-layered graph: a low-level graph of agent poses and features, a mid-level graph of object and place nodes, and a high-level graph of building structures. The optimization problem then involves minimizing errors across all these layers simultaneously. This includes not only the standard geometric errors but also inter-layer consistency errors, such as ensuring that an object node is contained within the correct room node. Agrawal et al. [353] propose a "hybrid" optimization that combines the continuous optimization of the pose graph with a discrete optimization over the semantic graph. After standard pose graph optimization, a second stage optimizes the semantic labels and relationships in the scene graph to ensure consistency with the newly optimized geometry. This iterative process allows geometric and semantic information to refine each other mutually. However, the optimization problem is significantly more complex. Solving multi-layered, hybrid continuous/discrete optimization problems in real-time is a significant computational challenge and an active area of research.

In summary, semantically-informed pose graph optimization enriches the traditional SLAM back-end by adding a wealth of new constraints derived from a high-level understanding of the scene. By incorporating features from objects, planes, and topological relationships, these methods can achieve global accuracy and semantic consistency that are unattainable with purely geometric approaches.

8. Deep learning-based and foundation models approaches

While the previous sections have discussed the integration of semantic modules into traditional SLAM architectures, recent years have witnessed a significant evolution toward learning-based approaches. These methods harness the representational power of deep learning (DL) and, more recently, foundation models (FMs) to improve or even replace key SLAM components, offering more flexible and generalizable solutions. The approaches range from supervised learning on specific tasks to open-world reasoning enabled by vision-language models.

8.1. Deep learning (DL)

Deep learning techniques, particularly convolutional and recurrent neural networks, are now widely adopted as perception front-ends in SLAM systems. These models are trained on large-scale annotated datasets to predict semantics, depth, optical flow, and ego-motion. In a generalized form, learning-based SLAM can be expressed as:

$$(\hat{\mathcal{X}}_t, \hat{\mathcal{M}}) = f_\theta(\mathcal{Z}_{1:t}, \mathcal{U}_{1:t}), \tag{55}$$

where f_θ is a deep neural network, $\mathcal{Z}$ denotes sensor observations, and $\mathcal{U}$ control inputs [354]. In monocular settings, estimating the world scale and handling dynamic objects are major challenges. CNN-SLAM [49] was a pioneering work that integrated a CNN-based depth prediction network directly into a direct LSD-SLAM [323] framework. The system's energy function combines a standard photometric error with a depth consistency error weighted by the uncertainty of the CNN's prediction. This allows the system to trust the geometric photometric alignment in high-gradient regions and fall back on the learned depth prior in textureless areas. MonoRec [355] proposes a self-supervised approach that jointly estimates camera motion, depth, and dynamic object motion, effectively disentangling ego-motion from scene motion. This demonstrates a holistic learned approach to disentangling camera motion from object motion. VLocNet++ [356] is a deep multitask learning architecture that jointly learns semantic segmentation, global pose regression, and visual odometry. The network uses a shared encoder to extract common features and then branches into separate decoders for each specific task. The total loss is a weighted combination of the individual task losses, for example, the global pose loss $\mathcal{L}_p$ and the semantic segmentation loss $\mathcal{L}_s$:

$$L_{total} = \alpha_1 \mathcal{L}_p + \alpha_2 \mathcal{L}_s. \tag{56}$$

Several studies [35,37,38] utilize DNNs to predict robot poses directly from semantic-rich inputs, allowing the system to learn robust semantic-geometric relationships from data implicitly. In addition, a recent trend is to move from modular pipelines to end-to-end learned frameworks that jointly estimate geometry and semantics. SimVODIS [357] and SimVODIS++ [358] propose a single neural architecture that simultaneously performs visual odometry, object detection, and instance segmentation. This multi-task learning allows the network to leverage shared representations. The pose and depth branches are trained in a self-supervised manner using photometric consistency, while the semantic branches are supervised. SimVODIS++ further introduces an attentive pose estimation module that learns to focus on salient and likely static regions, effectively ignoring dynamic objects without explicit masking. BEV-Locator [39] proposes an end-to-end network that transforms multi-view images into a Bird's-Eye-View (BEV) representation. This visual BEV feature map is then directly matched against a pre-existing semantic BEV map to regress the camera pose. Although promising, these systems are often limited to the training distribution.

In summary, these methods are fundamentally limited by the supervised learning paradigm. The scale and diversity of the annotated training data constrain their performance. A model trained on an indoor dataset may perform poorly in an outdoor driving scenario. This dependency on labeled data is a significant bottleneck for creating truly general-purpose SLAM systems and motivates the recent shift towards foundation models, which learn from much broader, web-scale data.

8.2. Foundation model

The most recent and disruptive trend in AI is the rise of large-scale, pre-trained Foundation Models, particularly Vision-Language Models (VLMs) like CLIP, which are trained on web-scale datasets of image-text pairs. Their emergence is a paradigm shift for Semantic vSLAM, moving the field from a "closed-world" assumption (with a small, predefined set of object classes) to an "open-world" or "open-vocabulary" setting. Instead of relying on a detector trained on a fixed set of classes, a VLM can be prompted with arbitrary natural language text to detect virtually any object. This is achieved by comparing the embedding of an image region with the text embedding of the query. The similarity R between an image patch I and a text query T is typically calculated as the cosine similarity of their respective embeddings, E_I and E_T, produced by the foundation model:

$$R(I,T) = \frac{E_I \cdot E_T}{\|E_I\| \cdot \|E_T\| + \epsilon}. \tag{57}$$

The primary impact of foundation models is the ability to build dense, queryable 3D maps that are not limited to a predefined set of classes. ConceptFusion [71] was a pioneering work that demonstrated how to fuse open-set CLIP features from multiple views into a globally consistent 3D map, creating a rich, multi-modal feature field that can be queried with text or images. FindAnything [359] and LOSS-SLAM [74] build on this by creating lightweight, open-vocabulary, and object-centric mapping frameworks. They use a Segment Anything Model (SAM) to generate class-agnostic segments and then fuse the associated VLM features of these segments into volumetric submaps, creating maps that are both dense and queryable. These methods typically fuse high-dimensional pixel-aligned features with 3D point clouds or meshes using volumetric fusion or learned neural fields:

$$\mathbf{f}p = \frac{1}{N}\sum_{i=1}^{N} \mathcal{P}(g_\phi(I_i))[u_i], \tag{58}$$

where $\mathcal{P}$ projects image features to the 3D map, g_ϕ is the feature encoder (*e.g.*, CLIP), and u_i is the pixel coordinate corresponding to 3D point p in image I_i. However, CLIP and SAM require significant GPU memory and latency, limiting onboard deployment. This approach has also been applied to real-time panoptic reconstruction, as demonstrated by PanoRecon [281], which fuses such open-set semantic features into a panoptic 3D map.

Foundation models enable the construction of more abstract and structured representations like 3D Scene Graphs with open-ended semantic nodes. While earlier works like Wald et al. [311] and SceneGraphFusion [292] laid the groundwork for incrementally building 3D scene graphs from RGB-D data with a fixed set of classes, foundation models have enabled a shift to open-vocabulary graphs. ConceptGraphs [73] builds a 3D scene graph where object nodes are associated not with a fixed class label, but with open-vocabulary "concepts" derived from VLM features. QueSTMaps [255] focuses on creating queryable semantic topological maps where each node in the topological map is associated with a local scene graph. While these methods achieve efficient performance on 3D representations, high-dimensional features lack precise spatial anchoring, which complicates drift correction and map updates. This hierarchical structure allows for efficient, language-based queries about both the high-level structure of the environment and the detailed object relationships within a specific place. Clio [313] proposes a task-driven approach in which the robot uses a VLM to determine which semantic information to include in its scene graph based on a list of tasks provided in natural language, addressing the key question of the required semantic granularity. In addition, Hier-SLAM [90] and Hier-SLAM++[76] use an LLM to generate a hierarchical tree of semantic concepts (*e.g.*, chair → seating furniture → furnishings) to support coarse-to-fine reasoning and memory-efficient learning. This symbolic knowledge graph is then used to produce compact, hierarchical semantic embeddings for each 3D Gaussian primitive. A novel semantic loss function is used during optimization to ensure that the learned semantic features are consistent both within and across the levels of this hierarchy, creating a highly structured and efficient semantic map. The semantic class can be expressed hierarchically as:

$$s_i^c = \{v_i^l, e_i^m | \quad l = 0, \ldots, L; m = 0, \ldots, L-1\}, \tag{59}$$

where v_i^l is the node at level l and e_i^m represents hierarchical semantic links. Li et al. [360] used VLMs to resolve the perceptual aliasing that plagues loop closure in repetitive environments. By querying the model with images of two visually similar locations, the VLM can provide fine-grained textual descriptions that highlight subtle differences (*e.g.*, "the desk with a blue pen"), enabling the system to reject false loop closures correctly. However, assigning per-point features leads to large memory footprints, limiting scalability to large scenes. SENT-Map [297] exploits LLMs to create topological or task-driven maps stored in structured formats and used for symbolic task planning. These maps encode not only geometry and semantics but also ownership, affordances, and scene-level context. LEXIS [75] leverages VLMs to allow for loop closures to be initiated via language queries. For example, a user could command the robot to "return to the room with the red chair". The system then uses a Vision-Language Model (VLM) to ground this query, identifying past keyframes that match the description and using them as high-confidence loop closure candidates. However, it is not a fully autonomous loop closure system and relies on a human operator to provide the language-based cues for initiating the search.

In summary, foundation models are revolutionizing the perception and reasoning capabilities of SLAM systems, pushing them from simple geometric mappers with a fixed set of classes towards genuine open-world, language-grounded scene understanding systems. While open-set in capability, many models still fail in ambiguous, repetitive, or occluded environments without further fine-tuning. To bridge language and geometry effectively, future directions include neuro-symbolic fusion, scene graphs, and compact embeddings. Combining CLIP features with learned scene embeddings [292] or using memory-efficient retrieval architectures [72] offers promising directions.

9. Continuous representation

Traditional SLAM systems have relied on discrete map representations like point clouds, voxel grids, or meshes. While effective, these representations can be memory-intensive and may struggle to represent complex geometry and appearance in a continuous and photorealistic manner. As detailed in the recent survey by Tosi et al. [361], a paradigm shift has occurred towards continuous representations, where the scene is modeled not as a discrete set of elements, but as a continuous function. These learned functions, often parameterized by neural networks or other primitives, can store geometry and appearance at arbitrary resolutions, enabling high-fidelity rendering and novel view synthesis. This section reviews the main categories of continuous representations emerging in the SLAM field: neural implicit representations, 3D Gaussian splatting, and hybrid approaches.

9.1. Neural implicit representation

Neural implicit representations model a scene as a continuous function f that maps a 3D coordinate (and potentially a viewing direction) to a physical property, such as occupancy, signed distance, or color and density based on Neural Radiance Fields (NeRFs) [362]. Neural Radiance Fields represent scenes as continuous volumetric functions parameterized by neural networks, typically multi-layer perceptions (MLPs), that map 3D spatial coordinates and viewing directions to color and volume density values. Novel views are rendered through differentiable volumetric ray marching, enabling photorealistic image synthesis from arbitrary viewpoints. A NeRF learns a mapping $f_\theta : \mathbb{R}^3 \times \mathbb{R}^d \rightarrow \mathbb{R}^C \times \mathbb{R}$, with $\mathbf{d}$ the viewing direction, and σ the volume density:

$$(\mathbf{X}, \mathbf{d}) \rightarrow (\sigma, \mathbf{s}^c). \tag{60}$$

This mapping is parameterized by the weight θ of a Multi-Layer Perceptron (MLP). To render a pixel, volumetric rendering is used to integrate the semantic color and density predictions along a camera ray $\mathbf{r}(t) = \mathbf{o} + t\mathbf{d}$ passing through that pixel:

$$\begin{aligned} C(\mathbf{r}) &= \int_{t_n}^{t_f} T(t)\sigma(\mathbf{r}(t))\mathbf{s}^c(\mathbf{r}(t), \mathbf{d})dt, \\ T(t) &= \exp\Big(-\int_{t_n}^{t_f} \sigma(\mathbf{r}(s))ds\Big). \end{aligned} \tag{61}$$

By optimizing the network weights θ to minimize photometric error between rendered and ground-truth images, the MLP learns a complete, continuous representation of the scene. An earlier, related concept was presented by Czarnowski et al. [363], which introduced the idea of using a learned, compact code vector to represent the geometry of a local keyframe. This "deep factor" was integrated into a factor graph for probabilistic dense monocular SLAM, showcasing the potential of learned compact codes. However, this method did not produce a continuous map of the entire scene, instead representing geometry on a pre-keyframe basis. iMAP [59] was a landmark achievement, demonstrating the first real-time SLAM system using a pure implicit representation. Instead of a large MLP, it uses a single, small MLP and a set of active keyframes to represent the scene. The use of low-capacity MLP meant that it could only map very small-scale scenes and was prone to catastrophic forgetting. Several works [364,365] also integrate NeRF-based objects with dedicated MLP models into a lightweight SLAM framework, achieving real-time object-level mapping from monocular RGB input. In addition, to overcome the scalability and efficiency limitations of the initial real-time methods, MeSLAM [366] addresses them by using multiple lightweight MLPs, each small MLP

representing a local region of the scene, allowing the total memory footprint to scale more gracefully with the environment size. However, managing the consistency and overlap between multiple MLPs can be complex. Co-SLAM [367] introduced a hybrid implicit representation that combines a sparse set of feature grids with a coordinate-based MLP.

ESLAM [368] and NISB-Map [369] propose submap-based approaches to enable large-scale mapping, which divide the environment into a series of local submaps and represent them by their own independent neural implicit model. While it enables theoretically unbounded scalability, because memory and computational load scale with submap size, stitching submaps together seamlessly and handling loop closures between them remain significant challenges. Point-SLAM [61] anchors the neural radiance field to an explicit point cloud to regularize the neural field, yielding sharper reconstructions. NeRF-SLAM [370] was one of the first works to tackle this for dense monocular SLAM, but it was limited to small workspaces and not real-time. To address the speed issue, NICE-SLAM [60], IMODE [371], and NICER-SLAM [372] extend NeRF-based SLAM with hierarchical neural fields, supporting efficient optimization and scene decomposition. SNI-SLAM [63] employs a neural implicit representation, hierarchical semantic encoding for multilevel scene comprehension, and cross-attention mechanisms for collaborative integration of appearance, geometry, and semantic features. Loopy-SLAM [373] addressed a major limitation of the lack of loop closure by introducing a pose graph optimization back-end to an implicit SLAM system. When a loop is detected, a pose graph is optimized, and the resulting correction is used to deform the neural implicit map, ensuring global consistency. DynaMoN [65] incorporates semantic segmentation to distinguish static and dynamic regions, enabling motion-aware localization and reconstruction. By focusing the NeRF training on the static background while accounting for dynamic elements, it achieves robust camera tracking and high-quality novel-view synthesis even in the presence of motion.

9.2. 3D Gaussian splatting

In contrast to implicit methods that encode the entire scene into the weights of a coordinate-based neural network, another powerful approach is to use an explicit continuous representation. In this paradigm, the scene is still modeled as a continuous function, but it is composed of a discrete set of primitives, each with its own explicit and optimizable properties. This avoids the need for expensive, repetitive neural network queries during rendering, which is the primary bottleneck for pure implicit methods. The state-of-the-art explicit representation that has revolutionized the field is 3D Gaussian Splatting. A scene is represented as a collection of 3D Gaussians, where each Gaussian G is a continuous volumetric function defined by its mean μ, covariance matrix $\sum$, semantic color $\mathbf{s}^c$, and opacity α:

$$G(X) = \alpha \cdot \exp\left(-\frac{1}{2}(\mathbf{X}-\mu)^T \sum^{-1} (\mathbf{X}-\mu)\right). \tag{62}$$

Rendering is achieved by "splatting" these 3D Gaussians onto the 2D image plane and blending them in depth order. Because this process is fully differentiable and leverages fast GPU rasterization, the parameters of all Gaussians can be optimized directly by minimizing the error between the rendered and ground-truth images at very high speed. The first wave of Gaussian Splatting SLAM systems demonstrated its viability for real-time, dense mapping. SplaTam [66] was a pioneering work that used RGB-D data to incrementally build a map of 3D Gaussians while simultaneously tracking the camera pose against the map. It established a core tracking framework by rendering the map and minimizing photometric and geometric errors, followed by a mapping step that adds and optimizes Gaussians from new keyframes. Similarly, GS-SLAM [68] and Gaussian Splatting SLAM [89] demonstrated comparable capabilities, solidifying Gaussian Splatting as a leading representation for dense visual SLAM. However, the number of Gaussians required to represent a scene grows linearly with the size and complexity of the environment, making it difficult to scale to very large areas without submapping strategies, which introduce their own complexities like inter-map loop closures. SGS-SLAM [69], SemGauss-SLAM [70], NEDS-SLAM [67], and GS^{3}LAM [374] all augment each 3D Gaussian with a learned semantic feature vector. This allows the system to render not only color and depth but also dense, pixel-aligned semantic maps from novel viewpoints. The optimization objective in these systems is typically extended to include a semantic loss (*e.g.*, cross-entropy) between the rendered semantic labels and the ground truth segmentation from a 2D perception network. Hier-SLAM [90] extends this concept to Semantic vSLAM. It augments each 3D Gaussian with a learned semantic feature vector. This allows the system to render not only color and depth but also semantic maps. Hier-SLAM++ [76] further uses an LLM to create a hierarchical semantic structure, which is then encoded into the Gaussians, enabling a highly efficient and structured semantic map. SDD-SLAM [375] assigns semantic labels to distinguish static background Gaussians from dynamic object Gaussians, allowing for independent tracking and rendering. To address scalability, Xin et al. [376] propose a submap-based approach, where the global map is composed of multiple local Gaussian Splatting submaps that are optimized and stitched together, addressing the memory challenges of a single monolithic representation. Yugay et al. [377] tackles the problem of long-term dynamics by introducing a keyframe management system that can discard outdated observations, allowing the Gaussian map to adapt to changes in the environment over time.

9.3. Hybrid representation

Hybrid representation aims to combine the advantages of both implicit and explicit models to achieve both high quality and real-time performance. It typically uses an explicit, sparse data structure (like a feature grid, an octree, or a point cloud) to partition the scene and provide a geometric scaffold. This structure is then coupled with a lightweight, implicit neural network that learns to represent fine details and appearance. NICE-SLAM [60] was a foundation work that introduced a hierarchical, multi-resolution feature grid, enabling high-quality surface reconstruction in real-time. It demonstrated that this hybrid approach could overcome the speed limitation of pure NeRF-based SLAM. Subsequent works such as Co-SLAM [367], H^2-Mapping [62], and H^3-Mapping [64] further refined this by incorporating explicit geometric priors to guide training of the implicit feature grids, thereby accelerating convergence and improving reconstruction quality. Point-SLAM [61] anchors the neural radiance field to an explicit point cloud generated by a standard SLAM front-end. This provides a strong geometric prior that regularizes the implicit field. Similarly, Neural Surfel Reconstruction [378] uses explicit surfels to represent the scene's geometry, where each surfel is associated with a learned neural descriptor that captures local appearance. A higher-level hybrid approach is taken by Neural Topological SLAM [379], which combines an explicit topological graph of the environment with local implicit models at each node capable of reconstructing the appearance of a specific place. While hybrid methods are powerful, they represent a compromise. The resolution of the underlying explicit grid structure often constrains the quality of the final reconstruction. While they are much faster than pure implicit methods, they can be more complex to implement and may not achieve the same rendering speed as pure explicit methods like Gaussian Splatting.

In summary, the move towards continuous representations marks a significant evolution in SLAM, enabling levels of photorealism and detail previously unattainable with discrete maps. A clear trade-off exists between the different approaches. Implicit representations such as NeRFs offer state-of-the-art performance for novel view synthesis but have historically been too slow for real-time SLAM. Explicit representations, dominated by 3D Gaussian Splatting, provide a compelling alternative with breakneck rendering speeds, making them

highly suitable for real-time SLAM applications, though they can be memory-intensive. Hybrid methods have emerged as a powerful middle ground, combining the strengths of both to achieve real-time performance with high-quality reconstruction. The current trajectory of the field shows immense momentum behind explicit Gaussian Splatting and sophisticated hybrid models as the most promising avenues for building the next generation of dense, photorealistic, and semantically-aware SLAM systems.

10. Multi-robot semantic vSLAM

Deploying a team of robots to map an environment offers significant advantages over a single-robot system, including increased speed, efficiency, and robustness to individual robot failures. Multi-robot SLAM, however, introduces a new layer of complexity: in addition to solving the standard SLAM problem for itself, each robot must also be able to share information, understand the perspective of its teammates, and fuse its local map into a globally consistent world model. When semantics are involved, these challenges extend to the semantic domain, requiring robots to build a shared, coherent understanding of the environment's meaning. This section reviews the key challenges and approaches in multi-robot Semantic vSLAM, covering system architectures, map fusion, and collaborative active mapping.

10.1. System architectures: Centralized vs. Decentralized

Multi-robot SLAM can be formulated as a joint optimization problem. Given a team of N robots, the goal is to estimate the set of all trajectories $\mathcal{X}^N = \{\mathcal{X}, \dots, \mathcal{X}_N\}$ and the global semantic map $\mathcal{M}$ by maximizing the posterior probability given all inter- and intra-robot measurements $\mathcal{Z}$:

$$(\hat{\mathcal{X}}^N, \hat{\mathcal{M}}) = \underset{\mathcal{X}^N, \mathcal{M}}{\operatorname{argmax}} P(\mathcal{X}^N, \mathcal{M} \mid \mathcal{Z}). \tag{63}$$

The architecture of the system – centralized or decentralized – determines how this joint problem is solved. In a centralized architecture, all robots send their raw or partially processed data (*e.g.*, keyframes, local submaps, detected objects) to a single, powerful base station or server. This central server is responsible for performing the most computationally intensive tasks: jointly optimizing all robot trajectories, detecting inter-robot loop closures, and fusing all data into a single, global map. Systems like HD-CCSOM [307] and the framework by Yue et al. [306] use this approach. A central server receives semantic occupancy submaps or object lists from multiple robots. It then solves the large-scale data association problem to find the correct relative poses between all the submaps and fuses them into a unified global representation. Hammer [79] demonstrates this for modern representations, where multiple robots, even heterogeneous ones, send their individually constructed Gaussian Splatting maps to a central server for alignment and fusion. However, this approach requires high bandwidth and reliable communication with the central server. It has a single point of failure, if the server goes down, the entire collaborative system fails. In a decentralized architecture, there is no central server. Robots communicate directly with each other (peer-to-peer) to share information and build a consistent map. Each robot is responsible for maintaining its own estimate of the global map. Kimera-Multi [77] is a state-of-the-art decentralized system that builds a distributed, dense metric-semantic mesh map. When two robots are within communication range, they exchange their local trajectory and map data, perform a relative pose estimation to "dock" their maps, and then communicate updates to maintain consistency. SlideSLAM [78] and the work by Tchuiev and Indelman [380] also propose robust decentralized frameworks that allow for consistent, distributed semantic mapping and localization without a central server.

10.2. Map fusion and inter-robot data association

The core technical challenge in multi-robot SLAM is fusing the maps from different robots. This requires finding the rigid body transformation ${}_W T_{R_j \to R_i}$ that aligns the coordinate frame of robot j with that of robot i in the world frame W. The most common method is to perform place recognition between robots. When robot i observes a place that robot j has seen before, an inter-robot loop closure constraint is created. The optimization problem is to find the transformation that best aligns the matched landmarks from the two maps. Given a set of matched landmark pairs $(l_{i,k}, l_{j,k})$, the transformation is found by minimizing a geometric error:

$${}_W T^*_{R_j \to R_i} = \underset{T}{\operatorname{argmin}} \sum_K \| m_{i,k} = T \cdot m_{j,k} \|^2. \tag{64}$$

Semantics play a crucial role in making this inter-robot association more robust. The association can be based on matching object constellations or semantic graph structures between the two maps. The hierarchical framework by Yue et al. [306] determines inter-robot object association using both geometric consistency and semantic similarity, measured by the KL divergence between the objects' semantic label distributions. A low KL divergence indicates high semantic similarity, increasing the confidence in the match. CoSAR [381] explicitly addresses the limitation of network bandwidth in realistic scenarios by designing a system that can adapt to varying communication quality. It prioritizes the transmission of compact semantic information (like object detections) over dense geometric data when the bandwidth is low, ensuring that the most critical information for alignment can still be shared.

10.3. Collaborative active mapping and exploration

Given a team of robots, the goal of active mapping is to coordinate their movements to explore and map an unknown environment as efficiently as possible. The problem is often framed as a decentralized decision-making process where each robot k chooses its next action a_k to contribute to a global utility function f_U. A typical formulation is to maximize the information gain about the map:

$$\mathcal{A}^* = \underset{\mathcal{A}}{\operatorname{argmax}} f_U(\mathcal{A}) = I(\mathcal{M}, \mathcal{Z} \mid \mathcal{A}), \tag{65}$$

where $\mathcal{A} = \{a_1, \dots, a_N\}$ is the set of joint actions for all robots, and $I(\cdot)$ is the mutual information between the map $\mathcal{M}$ and future expected measurements $\mathcal{Z}$ given those actions. The problem is often framed as finding a set of next-best-view waypoints for the team that maximizes a global utility function. This utility function can be designed to encourage rapid exploration (*e.g.*, maximizing frontier coverage) or to maximize the quality of the semantic map. Liu et al. [382] define the utility as the expected reduction in the uncertainty of the metric-semantic map. This directs robots towards regions where the semantic classification of objects is still uncertain, thereby improving the quality of the final semantic map. Asgharivaskasi et al. [383] use Riemannian optimization to plan paths for robot teams that maximize map quality. Aguilar et al. [384] present a system with a heterogeneous team of ground robots and a drone, where the drone provides a high-level overview to guide the ground robots' exploration and build a detailed semantic map.

In summary, multi-robot Semantic vSLAM extends the single-robot problem with significant challenges in communication, data fusion, and coordination. The choice between centralized and decentralized architectures presents a fundamental trade-off between optimality and robustness. Semantics provides a crucial common language for robots to understand each other's maps and to coordinate their actions, enabling teams of robots to build rich, globally consistent world models more efficiently than any single robot could alone.

11. Open research questions

Despite the significant progress in Semantic vSLAM over the past decade, several fundamental challenges remain. The transition from controlled laboratory settings to complex, dynamic, and long-term real-world deployments requires further innovation. This section highlights key open research questions and promising future directions that will shape the next generation of Semantic vSLAM systems, drawing upon the insights from several recent surveys and forward-looking research papers.

11.1. Semantic benchmarks and evaluation methods

The progress of any data-driven field is intrinsically linked to the quality of its benchmarks and the rigor of its evaluation metrics. For Semantic vSLAM, evaluation must go beyond geometric accuracy to assess the quality of the semantic understanding itself. This requires both suitable datasets with rich ground truth and standardized metrics that can capture semantic correctness, as consistently highlighted in major surveys of the field.

Evaluating semantic mapping requires datasets that provide not only sensor data and ground truth trajectories but also dense, per-pixel or per-point semantic and instance-level annotations. For indoor scenes, datasets like ScanNet [385], Matterport3D [386], and the Active Vision Dataset [387] have become standards, offering dense semantic and instance-level labels, with some providing higher-level annotations for rooms and object relationships, crucial for evaluating place categorization as demonstrated by Sünderhauf et al. [181]. For outdoor and autonomous driving contexts, KITTI Vision [388,389], KITTI-360 [390], and nuScenes [391] are the primary benchmarks, providing dense per-point labels, 3D object bounding boxes, and 3D semantic segmentation in complex urban environments. Most existing benchmarks, while useful, were not designed for the primary purpose of evaluating semantic, long-term, or dynamic mapping. The majority of popular indoor datasets are captured in static environments over very short trajectories. This makes them unsuitable for developing or fairly evaluating systems designed to handle dynamic objects or environmental changes. There is a critical need for large-scale, long-term datasets that capture the same environment over weeks, months, or seasons. These datasets would be essential for developing and evaluating lifelong SLAM systems capable of handling appearance changes, evolving object layouts, and seasonal variations. While datasets like ScanNet provide instance-level semantic labels, the annotations are typically "flat". They lack the hierarchical structure, functional affordances, and inter-object relationships that are necessary to train and evaluate advanced scene understanding systems that build representations like 3D scene graphs.

Table 8 summarizes the key metrics used for evaluating Semantic vSLAM systems, organized by evaluation category, in which for each metric, we provide its definition, mathematical formulation (where applicable), typical application scenarios, and its specific role in system assessment. While the geometric accuracy of a system is still evaluated using standard metrics like Absolute Trajectory Error (ATE) and Relative Pose Error (RPE), the semantic quality of the map is assessed through a different set of criteria. For dense semantic maps, the most common metric is the mean Intersection-over-Union (mIoU), calculated as the average IoU across all classes. For panoptic systems that distinguish object instances, the Panoptic Quality (PQ) is used, which combines Segmentation Quality (SQ) and Recognition Quality (RQ) in the formula $PQ = SQ \times RQ$. Beyond pixel-level accuracy, a crucial aspect is evaluating the system's utility for downstream tasks. This is often done by measuring performance on high-level tasks such as place categorization, using standard classification metrics like precision, recall, and F1-score. For the critical sub-task of loop closure detection, a precision–recall curve is the most informative metric, as it captures the essential trade-off between correctly identifying re-observed locations and rejecting false positives, a key focus in the work by Garg et al. [182]. However, a significant open challenge is the lack of quantitative semantic evaluation for the most abstract and open-ended representations. The evaluation of the most abstract representations remains an open challenge, often relying on computationally expensive metrics like graph edit distance. For systems producing complex 3D scene graphs or leveraging open-vocabulary foundation models, objective metrics are difficult to define. Metrics like graph edit distance are computationally expensive and not widely adopted. As a result, most recent studies in these advanced areas focus on qualitative evaluation, showcasing compelling examples of reconstructed scenes, correct query responses, or visually plausible maps, rather than reporting standardized, quantitative scores. This gap highlights a critical need for the community to develop new, rigorous metrics to benchmark the true semantic and reasoning capabilities of these state-of-the-art systems.

11.2. From semantic vSLAM to active semantic vSLAM

Most current SLAM systems are passive, they build a map based on a pre-defined or human-controlled trajectory. This is analogous to a passenger in a car who builds a mental map of a city but has no control over the route. The next frontier, and a significant open research challenge, is Active SLAM, where the robot intelligently and autonomously decides where to move next in order to build the most useful map as efficiently as possible. This requires a tight coupling between perception, mapping, and decision-making, transforming the robot from a passive observer into an active agent. This problem is typically formulated as a Partially Observable Markov Decision Process (POMDP), where the robot chooses an action a^* from a set of possible actions $\mathcal{A}$ that maximizes a long-term utility function $U(\cdot)$ representing the expected information gain about the map. While traditional active SLAM focuses on geometric exploration by directing the robot towards frontiers between known free space and unknown territory, Active Semantic vSLAM reframes this objective in semantic terms, making the process significantly more intelligent.

In this paradigm, the utility function is designed to maximize semantic understanding. For instance, the core objective can be to reduce the uncertainty of object classifications in the map, a principle explored in the work of Liu et al. [392] and Tao et al. [393]. The objective can also be driven by a high-level task, such as finding a specific object category (*e.g.*, "find a chair"), where the system learns to map the environment in a way that is explicitly useful for achieving this semantic goal, as demonstrated by Georgakis et al. [394]. To develop these intelligent exploration policies, recent approaches have turned to advanced decision-making techniques. Zhang et al. [211] propose a method based on spectral analysis of the semantic pose graph to identify the most informative exploration candidates, while Tian et al. [395] use deep reinforcement learning to train an agent that can leverage high-level layout semantics to navigate more efficiently than methods based on geometric frontiers alone. Ravichandran et al. [396] demonstrate how a robot can learn an effective navigation policy by using a Graph Neural Network (GNN) that operates directly on a 3D scene graph. A foundational concept in this area, explored in the thesis by Baxter [397], is the idea of goal-oriented exploration, where an agent learns to build a map specifically to help it find a target object, demonstrating the tight coupling between perception and action. Despite its promise, active Semantic vSLAM remains a challenging research problem. The computational cost of evaluating the potential information gain from many possible future viewpoints can be prohibitive for real-time operation. Furthermore, designing utility functions that perfectly capture complex, task-driven semantic goals is difficult and an area of ongoing research. Finally, long-horizon planning in large, unknown environments remains a significant challenge for any autonomous system.

Table 8
Semantic evaluation metrics quick-reference.

Metric	Definition/Formula	Applicable scenario	Role in semantic vSLAM evaluation
	Trajectory and Localization Accuracy		
Absolute Trajectory Error (ATE)	$\text{ATE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N} \|\mathbf{p}_i - \mathbf{p}_i^{gt}\|^2}$ after $\mathcal{SE}(3)$ alignment	Global trajectory evaluation, full SLAM systems	Measures overall localization accuracy and global consistency, essential for evaluating map quality and long-term drift correction
Relative Pose Error (RPE)	$\text{RPE} = \sqrt{\frac{1}{M}\sum_{i=1}^{M} \|(\mathbf{T}_i^{-1}\mathbf{T}_{i+\Delta})^{-1}(\mathbf{T}_i^{gt^{-1}}\mathbf{T}_{i+\Delta}^{gt})\|^2}$	Odometry evaluation, local tracking accuracy	Evaluates local consistency and drift rate, indicates visual odometry and short-term tracking performance
Absolute Translation Error	$\text{Trans}_{\text{err}} = \|\mathbf{p} - \mathbf{p}^{gt}\|_2$	Position accuracy assessment	Measures positional accuracy in meters, critical for navigation and manipulation tasks
Absolute Rotation Error	$\text{Rot}_{\text{err}} = \arccos\left(\frac{\text{tr}(\mathbf{R}^T\mathbf{R}^{gt})-1}{2}\right)$	Orientation accuracy assessment	Measures orientational accuracy in degrees, important for applications sensitive to viewing direction
	Semantic Segmentation Quality		
Mean Intersection over Union (mIoU)	$\text{mIoU} = \frac{1}{C}\sum_{c=1}^{C} \frac{TP_c}{TP_c+FP_c+FN_c}$	Dense semantic segmentation, semantic map labeling	Standard metric for pixel-wise semantic accuracy, directly reflects the quality of semantic extraction and label consistency
Pixel Accuracy (PA)	$\text{PA} = \frac{\sum_{c=1}^{C} TP_c}{\sum_{c=1}^{C}(TP_c+FP_c)}$	Overall segmentation assessment	Measures proportion of correctly labeled pixels, intuitive but can be biased toward dominant classes
Mean Pixel Accuracy (MPA)	$\text{MPA} = \frac{1}{C}\sum_{c=1}^{C} \frac{TP_c}{TP_c+FN_c}$	Class-balanced evaluation	Provides class-balanced accuracy by averaging per-class accuracy, useful for imbalanced class distributions
Frequency-Weighted IoU (FW-IoU)	$\text{FW-IoU} = \frac{1}{\sum_c n_c}\sum_{c=1}^{C} \frac{n_c \cdot TP_c}{TP_c+FP_c+FN_c}$	Real-world scene evaluation	Weights class IoU by pixel frequency, balances importance of common and rare classes
	Instance and Panoptic Segmentation Quality		
Panoptic Quality (PQ)	$\text{PQ} = \underbrace{\frac{\sum_{(p,g)\in TP}\text{IoU}(p,g)}{\lvert TP\rvert}}_{\text{SQ}} \times \underbrace{\frac{\lvert TP\rvert}{\lvert TP\rvert + \frac{1}{2}\lvert FP\rvert + \frac{1}{2}\lvert FN\rvert}}_{\text{RQ}}$	Panoptic semantic mapping, stuff and things evaluation	Jointly evaluates recognition and segmentation, decomposes into SQ and RQ for detailed analysis
Segmentation Quality (SQ)	$\text{SQ} = \frac{\sum_{(p,g)\in TP}\text{IoU}(p,g)}{\lvert TP\rvert}$	Matched segment quality	Measures average IoU of successfully matched segments, indicates boundary precision for detected instances
Recognition Quality (RQ)	$\text{RQ} = \frac{\lvert TP\rvert}{\lvert TP\rvert+\frac{1}{2}\lvert FP\rvert+\frac{1}{2}\lvert FN\rvert}$	Detection performance	Measures F1-score of segment matching, indicates detection and association performance
Average Precision (AP)	$\text{AP} = \int_0^1 p(r)\,dr$ at IoU thresholds	Instance segmentation, object detection	Evaluates precision–recall trade-off, AP_{50}, AP_{75}, $\text{AP}_{50:95}$ provide granular assessment
	Map Quality and 3D Reconstruction		
Accuracy (Acc)	$\text{Acc} = \frac{1}{\lvert\mathcal{P}\rvert}\sum_{\mathbf{p}\in\mathcal{P}} \min_{\mathbf{q}\in\mathcal{Q}} \|\mathbf{p}-\mathbf{q}\|$	Dense reconstruction evaluation	Measures distance from reconstructed points to ground truth, indicates precision and absence of spurious geometry
Completion (Comp)	$\text{Comp} = \frac{1}{\lvert\mathcal{Q}\rvert}\sum_{\mathbf{q}\in\mathcal{Q}} \min_{\mathbf{p}\in\mathcal{P}} \|\mathbf{q}-\mathbf{p}\|$	Dense reconstruction evaluation	Measures distance from ground truth to reconstruction, indicates coverage and completeness
Completion Ratio (CR)	$\text{CR} = \frac{\lvert\{\mathbf{q}\in\mathcal{Q} : \min_{\mathbf{p}} \|\mathbf{q}-\mathbf{p}\| < \tau\}\rvert}{\lvert\mathcal{Q}\rvert} \times 100\%$	Coverage assessment	Percentage of ground truth covered within threshold τ, interpretable completeness measure
Chamfer Distance (CD)	$\text{CD} = \text{Acc} + \text{Comp}$	Overall reconstruction quality	Combined metric balancing precision and recall of geometric reconstruction
F1-Score	$\text{F1} = \frac{2\times\text{Precision}\times\text{Recall}}{\text{Precision}+\text{Recall}}$	Loop closure detection, place recognition	Evaluates loop closure performance, balances precision and recall for revisited location detection
	Object-Level and Scene Graph Evaluation		
Object Localization Error (OLE)	$\text{OLE} = \|\mathbf{c} - \mathbf{c}^{gt}\|_2$	Object-centric SLAM	Measures 3D centroid error for detected objects, essential for object-level mapping and manipulation
3D Intersection over Union (3D IoU)	$\text{IoU}_{3D} = \frac{\lvert B\cap B^{gt}\rvert}{\lvert B\cup B^{gt}\rvert}$	3D object detection, bounding box evaluation	Evaluates 3D bounding box accuracy, critical for object-based SLAM using cuboids or quadrics
Scene Graph Edit Distance	Minimum graph edits (node/edge insertions, deletions, substitutions) to transform predicted graph to ground truth	Scene graph evaluation, relationship assessment	Measures structural accuracy of predicted scene graphs, evaluates both node and edge correctness

Notation: N = number of poses; M = number of pose pairs; C = number of semantic classes; TP, FP, FN = true positives, false positives, false negatives; $\mathbf{T}$, $\mathbf{R}$, $\mathbf{p}$ = transformation matrix, rotation matrix, position vector; $\mathcal{P}$, $\mathcal{Q}$ = reconstructed and ground truth point sets; τ = distance threshold; n_c = number of pixels of class c; B = 3D bounding box.

11.3. Lifelong semantic vSLAM

Real-world environments are not static, they are in a constant state of flux. Furniture is rearranged, seasons alter the appearance of outdoor scenes, new objects are introduced, and old ones are removed. A truly autonomous robot must be able to operate for weeks, months, or years within such evolving spaces, continuously updating its map to reflect these changes. This is the grand challenge of Lifelong SLAM. It requires a fundamental shift from the traditional goal of building a single, static, and complete map to the much more difficult task of maintaining a dynamic, ever-evolving world model that can learn and adapt over time without human intervention. This long-term adaptation is intrinsically a continual learning problem. The robot is constantly presented with a non-stationary stream of data from its environment, and it must learn from this new information without catastrophically forgetting the stable, foundational knowledge it has learned in the past. As framed by Vödisch et al. [398], a lifelong system must achieve a delicate balance between plasticity (the ability to learn new information and changes rapidly) and stability (the ability to retain and preserve old, still-valid knowledge). Li et al. [399] formalize this as the need for the system to both "learn to memorize and to forget". The system must robustly memorize the permanent or semi-permanent structure of the environment while actively identifying and forgetting transient elements (like a delivery box that is only present for a day) or outdated information (like the position of a desk that has been moved) to prevent map corruption and unbounded growth. Early works like CoVIO [400] are beginning to develop online continual learning frameworks for core SLAM components like visual-inertial odometry, which is a foundational step towards building fully adaptive systems.

A key technical innovation for enabling lifelong operation is the move towards spatio-temporal mapping, where the map explicitly represents the temporal dimension. A standard 3D map is atemporal, it cannot distinguish between a chair that was present yesterday and one that is present today. Systems like Khronos [123] pioneer a solution by proposing a unified spatio-temporal Metric-Semantic vSLAM system. It builds a 4D (3D space + time) graph representation of the world. In this graph, object observations are associated over time to create a "world-tube" that represents the object's lifecycle—when it appeared, if it moved, and when it disappeared. This allows the system to explicitly reason about object permanence and dynamics, a crucial capability for understanding a changing world. Since manually annotating data for every new environment a robot might encounter over its lifetime is impossible, lifelong systems must also adapt in a self-supervised manner. BYE (Build Your Encoder) by Huang et al. [401] presents a novel framework for this, allowing a robot to build its own customized perception encoder from a single unlabeled exploration sequence in a new environment. By leveraging temporal consistency and multiview geometry as a powerful self-supervisory signal, the robot can fine-tune its perception model to the specific objects and appearance of its current surroundings, enabling robust long-term understanding without any human annotation. Ultimately, the purpose of a lifelong map is to serve as a persistent, queryable knowledge base—a long-term memory for the robot. This map should be able to answer questions that involve both space and time, such as "Where did I last see my keys yesterday?". Systems like QueSTMaps [255], which focus on creating queryable semantic topological maps, represent a significant step in this direction. A true lifelong system would extend this queryable memory with a temporal dimension, allowing the robot to reason about the history of its environment. Despite this progress, Lifelong SLAM remains one of the most significant open challenges, primarily due to the immense computational and memory costs of storing and reasoning about a 4D world model and the difficulty of developing robust methods for change detection, forgetting, and true continual learning without failure.

11.4. Generalization & robustness

A major open challenge is achieving true generalization and robustness, two deeply intertwined concepts that are paramount for real-world deployment and are consistently highlighted as critical research frontiers in recent surveys. Generalization refers to the ability of a system to perform well in new, unseen environments, a task made difficult by the heavy reliance of modern Semantic vSLAM on deep learning models. This creates a "domain gap," where a perception model trained on one type of data (*e.g.*, indoor offices) fails to generalize to another (*e.g.*, a residential home), representing a primary bottleneck. The shift towards foundation models that learn from web-scale data and the development of online adaptation methods are promising avenues for mitigating this issue. Robustness, on the other hand, is the ability of a SLAM system to maintain consistent and bounded error performance even when faced with challenging conditions that violate its core assumptions, such as aggressive motion, sensor noise, or visual degradation. As formalized in studies on measuring SLAM robustness, this can be quantified by observing a system's failure rate and accuracy degradation under controlled "perturbations". Many recent systems are explicitly designed with this in mind, for example, RS-SLAM [267] is intended to be a "Robust Semantic vSLAM" system for dynamic environments, while Kimera-Multi [77] is engineered to be a "Robust, Distributed" system for multi-robot teams that can handle intermittent communication. Ultimately, a truly robust system must also be capable of graceful failure. This requires the system to quantify its own uncertainty and, upon detecting a high likelihood of failure, enter a safe mode (*e.g.*, stop or request assistance) rather than producing a confident but catastrophically incorrect result. This ability to reason about its own limitations is a hallmark of a truly mature and deployable autonomous system.

11.5. Reproducible research

The increasing complexity of modern Semantic vSLAM systems, which often integrate multiple deep learning models, complex data fusion pipelines, and sophisticated back-end optimizers, makes reproducibility a significant and persistent challenge. This has long been an issue in the robotics community, where real-life experiments are often difficult to replicate across different research groups due to variations in hardware, software environments, and physical conditions. While static datasets are commonly used for benchmarking the perception and reconstruction components of an SLAM system, they cannot fully capture the nuances of a complete, end-to-end system's performance, which is highly sensitive to implementation details and parameter tuning. As a result, even when using the same dataset, it is often difficult to make a fair, one-to-one comparison between different published methods, a challenge consistently highlighted in recent surveys. To address this, there is a dire need for a continued and expanded commitment within the community to open-sourcing code and models. Open-source frameworks, such as the libraries provided for Kimera [57] and Hydra [58], are invaluable assets, they not only allow other researchers to verify reported results but also provide a common baseline upon which new algorithms can be built and fairly compared. By allowing others to modify specific modules (*e.g.*, only the data association or loop closure component), these frameworks make it much more straightforward to isolate and evaluate the impact of a novel contribution. In summary, for the field of Semantic vSLAM to mature and achieve real-world impact, improving the reproducibility of its research is essential, requiring a concerted effort towards open-sourcing systems, establishing more comprehensive benchmarks, and agreeing upon a unified set of evaluation metrics that capture both geometric accuracy and semantic quality.

11.6. Practical applications

Ultimately, the goal of Semantic vSLAM research is to enable autonomous systems to perform practical tasks in the real world. For the field to have a broad impact, research must be increasingly driven by practical application demands, moving beyond benchmark trajectory accuracy to task-based performance in complex, human-centric environments. The rich, structured world models produced by Semantic vSLAM are a crucial enabling technology for a wide range of applications. In service robotics and assisted living, a semantic map allows a robot to understand and execute natural language commands like "bring me the cup from the kitchen table," a capability that systems like QueSTMaps [255] and LEXIS [75] are designed to support. For autonomous driving, vehicles require rich, semantic maps of road networks that include lane markings, traffic signs, and dynamic agents, a challenge addressed by robust systems like DynaSLAM II [239] and RS-SLAM [267]. In augmented and virtual reality, the high-fidelity, photorealistic maps created by modern continuous representation SLAM (*e.g.*, Gaussian Splatting SLAM [89]) are essential for seamlessly blending virtual content with the real world, allowing virtual objects to interact realistically with physical surfaces. Furthermore, in aerial robotics and inspection, semantic maps enable drones to understand and perform complex tasks, such as inspecting specific windows on a building or monitoring particular crops, a use case targeted by systems like VPS-SLAM [212]. Finally, in domains like search and rescue, a semantic map provides invaluable situational awareness, allowing first responders to quickly understand the layout of an unknown building and locate people or objects of interest. Addressing the specific challenges of these practical applications will continue to be a primary driver of innovation in the Semantic vSLAM field.

12. Conclusion

This survey has explored the rapidly advancing field of semantic visual SLAM, a critical evolution that enriches traditional geometric mapping with a high-level, human-like understanding of the environment. We began by analyzing the limitations of conventional SLAM systems in complex, dynamic scenes, establishing the need for a paradigm shift from purely geometric representations to context-aware world models. Our review systematically deconstructed the modern Semantic vSLAM pipeline, analyzing its core components from semantic extraction and localization to the intricacies of semantic mapping, data association, fusion, and globally consistent loop closure optimization. Beyond summarizing the current state of these modular components, this survey has highlighted the transformative impact of two recent technological shifts. First, the move towards continuous representations – from implicit Neural Radiance Fields to the explicit, real-time capabilities of 3D Gaussian Splatting – is fundamentally reshaping the fidelity and realism of the mapping process. Second, the advent of Foundation Models is revolutionizing the perception and reasoning layers, moving the field from a "closed-world" assumption with a fixed vocabulary to a truly "open-world" paradigm, enabling zero-shot object recognition and language-grounded scene understanding. Despite this remarkable progress, we have also identified key open research questions that must be addressed for the field to mature. The need for more comprehensive benchmarks that capture long-term dynamics, the transition from passive mapping to active SLAM for task-driven exploration, and the grand challenge of achieving true lifelong operation in ever-changing environments remain significant frontiers. Furthermore, issues of generalization, robustness, and reproducible research are essential to transitioning these advanced algorithms from the laboratory to real-world applications.

In conclusion, this survey consolidates existing knowledge, provides a structured overview of the state-of-the-art, and identifies the pivotal challenges and future directions that will inspire the next wave of innovation. By advancing Semantic vSLAM, we aim not only to build more accurate maps but to drive the development of truly intelligent systems capable of perceiving, understanding, and meaningfully interacting with the complexity of the physical world.

CRediT authorship contribution statement

Thanh Nguyen Canh: Writing – review & editing, Writing – original draft, Investigation, Formal analysis, Conceptualization. **Haolan Zhang:** Writing – original draft, Investigation, Conceptualization. **Xiem HoangVan:** Writing – review & editing, Supervision, Conceptualization. **Nak Young Chong:** Writing – review & editing, Writing – original draft, Supervision, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

The authors are unable or have chosen not to specify which data has been used.

References

[1] R.C. Smith, P. Cheeseman, On the representation and estimation of spatial uncertainty, Int. J. Robot. Res. 5 (4) (1986) 56–68.
[2] J.A. Placed, J. Strader, H. Carrillo, N. Atanasov, V. Indelman, L. Carlone, J.A. Castellanos, A survey on active simultaneous localization and mapping: State of the art and new frontiers, IEEE Trans. Robot. (2023).
[3] S. Zhang, S. Zhao, D. An, J. Liu, H. Wang, Y. Feng, D. Li, R. Zhao, Visual SLAM for underwater vehicles: A survey, Comput. Sci. Rev. 46 (2022) 100510.
[4] T.N. Canh, V.-T. Nguyen, X. HoangVan, A. Elibol, N.Y. Chong, S3M: Semantic segmentation sparse mapping for UAVs with RGB-d camera, in: 2024 IEEE/SICE International Symposium on System Integration, SII, IEEE, 2024, pp. 899–905.
[5] J. Cheng, L. Zhang, Q. Chen, X. Hu, J. Cai, A review of visual SLAM methods for autonomous driving vehicles, Eng. Appl. Artif. Intell. 114 (2022) 104992.
[6] T.-j. Lee, C.-h. Kim, D.-i.D. Cho, A monocular vision sensor-based efficient SLAM method for indoor service robots, IEEE Trans. Ind. Electron. 66 (1) (2018) 318–328.
[7] F. Munoz-Montoya, M.-C. Juan, M. Mendez-Lopez, C. Fidalgo, Augmented reality based on SLAM to assess spatial short-term memory, IEEE Access 7 (2018) 2453–2466.
[8] X. Jiang, L. Zhu, J. Liu, A. Song, A SLAM-based 6dof controller with smooth auto-calibration for virtual reality, Vis. Comput. 39 (9) (2023) 3873–3886.
[9] W. Hess, D. Kohler, H. Rapp, D. Andor, Real-time loop closure in 2D LIDAR SLAM, in: 2016 IEEE International Conference on Robotics and Automation, ICRA, IEEE, 2016, pp. 1271–1278.
[10] Q. Zou, Q. Sun, L. Chen, B. Nie, Q. Li, A comparative analysis of LiDAR SLAM-based indoor navigation for autonomous vehicles, IEEE Trans. Intell. Transp. Syst. 23 (7) (2021) 6907–6921.
[11] I.A. Kazerouni, L. Fitzgerald, G. Dooly, D. Toal, A survey of state-of-the-art on visual SLAM, Expert Syst. Appl. 205 (2022) 117734.
[12] C. Campos, R. Elvira, J.J.G. Rodríguez, J.M. Montiel, J.D. Tardós, Orb-slam3: An accurate open-source library for visual, visual–inertial, and multimap slam, IEEE Trans. Robot. 37 (6) (2021) 1874–1890.
[13] F. Lu, E. Milios, Globally consistent range scan alignment for environment mapping, Auton. Robots 4 (1997) 333–349.
[14] L. Ma, J. Stückler, C. Kerl, D. Cremers, Multi-view deep learning for consistent semantic mapping with rgb-d cameras, in: 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS, IEEE, 2017, pp. 598–605.
[15] J. McCormac, A. Handa, A. Davison, S. Leutenegger, Semanticfusion: Dense 3d semantic mapping with convolutional neural networks, in: 2017 IEEE International Conference on Robotics and Automation, ICRA, IEEE, 2017, pp. 4628–4635.
[16] N. Sünderhauf, T.T. Pham, Y. Latif, M. Milford, I. Reid, Meaningful maps with object-oriented semantic mapping, in: 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS, IEEE, 2017, pp. 5079–5085.
[17] K. Chen, J. Xiao, J. Liu, Q. Tong, H. Zhang, R. Liu, J. Zhang, A. Ajoudani, S. Chen, Semantic visual simultaneous localization and mapping: A survey, IEEE Trans. Intell. Transp. Syst. (2025).
[18] T.N. Canh, T.S. Nguyen, C.H. Quach, X. HoangVan, M.D. Phung, Multisensor data fusion for reliable obstacle avoidance, in: 2022 11th International Conference on Control, Automation and Information Sciences, ICCAIS, IEEE, 2022, pp. 385–390.
[19] F. Dellaert, D.J. Bruemmer, A.C.C. Workspace, Semantic SLAM for collaborative cognitive workspaces., in: AAAI Technical Report (5), 2004, pp. 85–86.

[20] Z. Zhao, W. Zhang, J. Gu, J. Yang, K. Huang, Lidar mapping optimization based on lightweight semantic segmentation, IEEE Trans. Intell. Veh. 4 (3) (2019) 353–362.

[21] C. Jin, A. Elibol, P. Zhu, N.Y. Chong, Semantic mapping based on image feature fusion in indoor environments, in: 2021 21st International Conference on Control, Automation and Systems, ICCAS, IEEE, 2021, pp. 693–698.

[22] T.N. Canh, A. Elibol, N.Y. Chong, X. HoangVan, Object-oriented semantic mapping for reliable uavs navigation, in: 2023 12th International Conference on Control, Automation and Information Sciences, ICCAIS, IEEE, 2023, pp. 139–144.

[23] J. McCormac, R. Clark, M. Bloesch, A. Davison, S. Leutenegger, Fusion++: Volumetric object-level slam, in: 2018 International Conference on 3D Vision (3DV), IEEE, 2018, pp. 32–41.

[24] T. Hempel, A. Al-Hamadi, An online semantic mapping system for extending and enhancing visual SLAM, Eng. Appl. Artif. Intell. 111 (2022) 104830.

[25] X. Ruan, P. Guo, J. Huang, A semantic octomap mapping method based on cbam-pspnet, J. Web Eng. 21 (3) (2022) 879–910.

[26] Z. Liu, P. van Oosterom, J. Balado, A. Swart, B. Beers, Data frame aware optimized octomap-based dynamic object detection and removal in mobile laser scanning data, Alex. Eng. J. 74 (2023) 327–344.

[27] H. Li, X. Yang, H. Zhai, Y. Liu, H. Bao, G. Zhang, Vox-surf: Voxel-based implicit surface representation, IEEE Trans. Vis. Comput. Graphics 30 (3) (2022) 1743–1755.

[28] J.-L. Matez-Bandera, P. Ojeda, J. Monroy, J. Gonzalez-Jimenez, J.-R. Ruiz-Sarmiento, Voxeland: Probabilistic instance-aware semantic mapping with evidence-based uncertainty quantification, 2024, arXiv preprint arXiv:2411.08727.

[29] A. Zaganidis, L. Sun, T. Duckett, G. Cielniak, Integrating deep semantic segmentation into 3-d point cloud registration, IEEE Robot. Autom. Lett. 3 (4) (2018) 2942–2949.

[30] X. Chen, A. Milioto, E. Palazzolo, P. Giguere, J. Behley, C. Stachniss, Suma++: Efficient lidar-based semantic slam, in: 2019 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS, IEEE, 2019, pp. 4530–4537.

[31] C. Yi, I.H. Suh, G.H. Lim, B.-U. Choi, Active-semantic localization with a single consumer-grade camera, in: 2009 IEEE International Conference on Systems, Man and Cybernetics, IEEE, 2009, pp. 2161–2166.

[32] N. Atanasov, M. Zhu, K. Daniilidis, G.J. Pappas, Localization from semantic observations via the matrix permanent, Int. J. Robot. Res. 35 (1–3) (2016) 73–99.

[33] N. Akai, T. Hirayama, H. Murase, Semantic localization considering uncertainty of object recognition, IEEE Robot. Autom. Lett. 5 (3) (2020) 4384–4391.

[34] N. Akai, Mobile robot localization considering uncertainty of depth regression from camera images, IEEE Robot. Autom. Lett. 7 (2) (2022) 1431–1438.

[35] C. Toft, E. Stenborg, L. Hammarstrand, L. Brynte, M. Pollefeys, T. Sattler, F. Kahl, Semantic match consistency for long-term visual localization, in: Proceedings of the European Conference on Computer Vision, ECCV, 2018, pp. 383–399.

[36] J.L. Schönberger, M. Pollefeys, A. Geiger, T. Sattler, Semantic visual localization, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 6896–6906.

[37] L. Xiao, J. Wang, X. Qiu, Z. Rong, X. Zou, Dynamic-SLAM: Semantic monocular visual localization and mapping based on deep learning in dynamic environment, Robot. Auton. Syst. 117 (2019) 1–16.

[38] J. Wu, Q. Shi, Q. Lu, X. Liu, X. Zhu, Z. Lin, Learning invariant semantic representation for long-term robust visual localization, Eng. Appl. Artif. Intell. 111 (2022) 104793.

[39] Z. Zhang, M. Xu, W. Zhou, T. Peng, L. Li, S. Poslad, Bev-locator: An end-to-end visual semantic localization network using multi-view images, Sci. China Inf. Sci. 68 (2) (2025) 122106.

[40] S. Yang, S. Scherer, Cubeslam: Monocular 3-d object slam, IEEE Trans. Robot. 35 (4) (2019) 925–938.

[41] S.L. Bowman, N. Atanasov, K. Daniilidis, G.J. Pappas, Probabilistic data association for semantic slam, in: 2017 IEEE International Conference on Robotics and Automation, ICRA, IEEE, 2017, pp. 1722–1729.

[42] A. Asgharivaskasi, N. Atanasov, Active bayesian multi-class mapping from range and semantic segmentation observations, in: 2021 IEEE International Conference on Robotics and Automation, ICRA, IEEE, 2021, pp. 1–7.

[43] S.W. Chen, G.V. Nardari, E.S. Lee, C. Qu, X. Liu, R.A.F. Romero, V. Kumar, Sloam: Semantic lidar odometry and mapping for forest inventory, IEEE Robot. Autom. Lett. 5 (2) (2020) 612–619.

[44] I.D. Miller, F. Cladera, T. Smith, C.J. Taylor, V. Kumar, Stronger together: Air-ground robotic collaboration using semantics, IEEE Robot. Autom. Lett. 7 (4) (2022) 9643–9650.

[45] R. Kümmerle, G. Grisetti, H. Strasdat, K. Konolige, W. Burgard, G 2 o: A general framework for graph optimization, in: 2011 IEEE International Conference on Robotics and Automation, IEEE, 2011, pp. 3607–3613.

[46] S. Agarwal, K. Mierle, T.C.S. Team, Ceres Solver, 2023.

[47] F. Dellaert, G. Contributors, Borglab/gtsam, 2022, http://dx.doi.org/10.5281/zenodo.5794541.

[48] J. Ansel, E. Yang, H. He, N. Gimelshein, A. Jain, M. Voznesensky, B. Bao, P. Bell, D. Berard, E. Burovski, G. Chauhan, A. Chourdia, W. Constable, A. Desmaison, Z. DeVito, E. Ellison, W. Feng, J. Gong, M. Gschwind, B. Hirsh, S. Huang, K. Kalambarkar, L. Kirsch, M. Lazos, M. Lezcano, Y. Liang, J. Liang, Y. Lu, C. Luk, B. Maher, Y. Pan, C. Puhrsch, M. Reso, M. Saroufim, M.Y. Siraichi, H. Suk, M. Suo, P. Tillet, E. Wang, X. Wang, W. Wen, S. Zhang, X. Zhao, K. Zhou, R. Zou, A. Mathews, G. Chanan, P. Wu, S. Chintala, PyTorch 2: Faster Machine Learning Through Dynamic Python Bytecode Transformation and Graph Compilation, in: 29th ACM International Conference on Architectural Support for Programming Languages and Operating Systems, Volume 2, ASPLOS'24, ACM, 2024.

[49] K. Tateno, F. Tombari, I. Laina, N. Navab, Cnn-slam: Real-time dense monocular slam with learned depth prediction, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 6243–6252.

[50] M. Rünz, L. Agapito, Co-fusion: Real-time segmentation, tracking and fusion of multiple objects, in: 2017 IEEE International Conference on Robotics and Automation, ICRA, IEEE, 2017, pp. 4471–4478.

[51] K.-N. Lianos, J.L. Schonberger, M. Pollefeys, T. Sattler, Vso: Visual semantic odometry, in: Proceedings of the European Conference on Computer Vision, ECCV, 2018, pp. 234–250.

[52] C. Yu, Z. Liu, X.-J. Liu, F. Xie, Y. Yang, Q. Wei, Q. Fei, DS-SLAM: A semantic visual SLAM towards dynamic environments, in: 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS, IEEE, 2018, pp. 1168–1174.

[53] M. Runz, M. Buffier, L. Agapito, Maskfusion: Real-time recognition, tracking and reconstruction of multiple moving objects, in: 2018 IEEE International Symposium on Mixed and Augmented Reality, ISMAR, IEEE, 2018, pp. 10–20.

[54] L. Nicholson, M. Milford, N. Sünderhauf, Quadricslam: Dual quadrics from object detections as landmarks in object-oriented slam, IEEE Robot. Autom. Lett. 4 (1) (2018) 1–8.

[55] Y. Wu, Y. Zhang, D. Zhu, Y. Feng, S. Coleman, D. Kerr, EAO-SLAM: Monocular semi-dense object SLAM based on ensemble data association, in: 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS, IEEE, 2020, pp. 4966–4973.

[56] G. Narita, T. Seno, T. Ishikawa, Y. Kaji, Panopticfusion: Online volumetric semantic mapping at the level of stuff and things, in: 2019 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS, IEEE, 2019, pp. 4205–4212.

[57] A. Rosinol, M. Abate, Y. Chang, L. Carlone, Kimera: an open-source library for real-time metric-semantic localization and mapping, in: 2020 IEEE International Conference on Robotics and Automation, ICRA, IEEE, 2020, pp. 1689–1696.

[58] N. Hughes, Y. Chang, L. Carlone, Hydra: A real-time spatial perception system for 3D scene graph construction and optimization, 2022, arXiv preprint arXiv:2201.13360.

[59] E. Sucar, S. Liu, J. Ortiz, A.J. Davison, Imap: Implicit mapping and positioning in real-time, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021, pp. 6229–6238.

[60] Z. Zhu, S. Peng, V. Larsson, W. Xu, H. Bao, Z. Cui, M.R. Oswald, M. Pollefeys, Nice-slam: Neural implicit scalable encoding for slam, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 12786–12796.

[61] E. Sandström, Y. Li, L. Van Gool, M.R. Oswald, Point-slam: Dense neural point cloud-based slam, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2023, pp. 18433–18444.

[62] C. Jiang, H. Zhang, P. Liu, Z. Yu, H. Cheng, B. Zhou, S. Shen, H _ {2} -mapping: Real-time dense mapping using hierarchical hybrid representation, IEEE Robot. Autom. Lett. 8 (10) (2023) 6787–6794.

[63] S. Zhu, G. Wang, H. Blum, J. Liu, L. Song, M. Pollefeys, H. Wang, Sni-slam: Semantic neural implicit slam, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2024, pp. 21167–21177.

[64] C. Jiang, Y. Luo, B. Zhou, S. Shen, H3-mapping: Quasi-heterogeneous feature grids for real-time dense mapping using hierarchical hybrid representation, IEEE Robot. Autom. Lett. (2024).

[65] N. Schischka, H. Schieber, M.A. Karaoglu, M. Gorgulu, F. Grötzner, A. Ladikos, N. Navab, D. Roth, B. Busam, Dynamon: Motion-aware fast and robust camera localization for dynamic neural radiance fields, IEEE Robot. Autom. Lett. (2024).

[66] N. Keetha, J. Karhade, K.M. Jatavallabhula, G. Yang, S. Scherer, D. Ramanan, J. Luiten, Splatam: Splat track & map 3d gaussians for dense rgb-d slam, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2024, pp. 21357–21366.

[67] Y. Ji, Y. Liu, G. Xie, B. Ma, Z. Xie, H. Liu, Neds-slam: A neural explicit dense semantic slam framework using 3d gaussian splatting, IEEE Robot. Autom. Lett. (2024).

[68] C. Yan, D. Qu, D. Xu, B. Zhao, Z. Wang, D. Wang, X. Li, Gs-slam: Dense visual slam with 3d gaussian splatting, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2024, pp. 19595–19604.

[69] M. Li, S. Liu, H. Zhou, G. Zhu, N. Cheng, T. Deng, H. Wang, Sgs-slam: Semantic gaussian splatting for neural dense slam, in: European Conference on Computer Vision, Springer, 2024, pp. 163–179.

[70] S. Zhu, R. Qin, G. Wang, J. Liu, H. Wang, Semgauss-slam: Dense semantic gaussian splatting slam, 2025, arXiv preprint arXiv:2503.07494.

[71] K.M. Jatavallabhula, A. Kuwajerwala, Q. Gu, M. Omama, T. Chen, A. Maalouf, S. Li, G. Iyer, S. Saryazdi, N. Keetha, et al., Conceptfusion: Open-set multimodal 3d mapping, 2023, arXiv preprint arXiv:2302.07241.

[72] C. Liu, K. Wang, J. Shi, Z. Qiao, S. Shen, Fm-fusion: Instance-aware semantic mapping boosted by vision-language foundation models, IEEE Robot. Autom. Lett. 9 (3) (2024) 2232–2239.

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

[74] K. Singh, T. Magoun, J.J. Leonard, LOSS-slam: Lightweight open-set semantic simultaneous localization and mapping, 2024, arXiv preprint arXiv:2404.04377.

[75] C. Kassab, M. Mattamala, L. Zhang, M. Fallon, Language-extended indoor slam (lexis): A versatile system for real-time visual scene understanding, in: 2024 IEEE International Conference on Robotics and Automation, ICRA, IEEE, 2024, pp. 15988–15994.

[76] B. Li, V.C. Hao, P.J. Stuckey, I. Reid, H. Rezatofighi, Hier-slam++: Neuro-symbolic semantic slam with a hierarchically categorical gaussian splatting, 2025, arXiv preprint arXiv:2502.14931.

[77] Y. Tian, Y. Chang, F.H. Arias, C. Nieto-Granda, J.P. How, L. Carlone, Kimera-multi: Robust, distributed, dense metric-semantic slam for multi-robot systems, IEEE Trans. Robot. 38 (4) (2022).

[78] X. Liu, J. Lei, A. Prabhu, Y. Tao, I. Spasojevic, P. Chaudhari, N. Atanasov, V. Kumar, Slideslam: Sparse, lightweight, decentralized metric-semantic slam for multi-robot navigation, 2024, arXiv preprint arXiv:2406.17249.

[79] J. Yu, T. Chen, M. Schwager, HAMMER: Heterogeneous, multi-robot semantic Gaussian splatting, IEEE Robot. Autom. Lett. (2025).

[80] C. Galindo, A. Saffiotti, S. Coradeschi, P. Buschka, J.-A. Fernandez-Madrigal, J. González, Multi-hierarchical semantic maps for mobile robotics, in: 2005 IEEE/RSJ International Conference on Intelligent Robots and Systems, IEEE, 2005, pp. 2278–2283.

[81] M. Hosseinzadeh, Y. Latif, T. Pham, N. Suenderhauf, I. Reid, Structure aware slam using quadrics and planes, in: Computer Vision–ACCV 2018: 14th Asian Conference on Computer Vision, Perth, Australia, December 2–6, 2018, Revised Selected Papers, Part III 14, Springer, 2019, pp. 410–426.

[82] Y. Wang, K. Xu, Y. Tian, X. Ding, DRG-SLAM: a semantic RGB-d SLAM using geometric features for indoor dynamic scene, in: 2022 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS, IEEE, 2022, pp. 1352–1359.

[83] K. Doherty, D. Fourie, J. Leonard, Multimodal semantic slam with probabilistic data association, in: 2019 International Conference on Robotics and Automation, ICRA, IEEE, 2019, pp. 2419–2425.

[84] K.J. Doherty, D.P. Baxter, E. Schneeweiss, J.J. Leonard, Probabilistic data association via mixture models for robust semantic SLAM, in: 2020 IEEE International Conference on Robotics and Automation, ICRA, IEEE, 2020, pp. 1098–1104.

[85] E. Michael, T. Summers, T.A. Wood, C. Manzie, I. Shames, Probabilistic data association for semantic slam at scale, in: 2022 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS, IEEE, 2022, pp. 4359–4364.

[86] A. Hornung, K.M. Wurm, M. Bennewitz, C. Stachniss, W. Burgard, OctoMap: An efficient probabilistic 3D mapping framework based on octrees, Auton. Robots 34 (2013) 189–206.

[87] D. Pagliari, F. Menna, R. Roncella, F. Remondino, L. Pinto, Kinect fusion improvement using depth camera calibration, Int. Arch. Photogramm. Remote. Sens. Spat. Inf. Sci. 40 (2014) 479–485.

[88] X. Yang, H. Li, H. Zhai, Y. Ming, Y. Liu, G. Zhang, Vox-fusion: Dense tracking and mapping with voxel-based neural implicit representation, in: 2022 IEEE International Symposium on Mixed and Augmented Reality, ISMAR, IEEE, 2022, pp. 499–507.

[89] H. Matsuki, R. Murai, P.H. Kelly, A.J. Davison, Gaussian splatting slam, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2024, pp. 18039–18048.

[90] B. Li, Z. Cai, Y.-F. Li, I. Reid, H. Rezatofighi, Hier-SLAM: Scaling-up semantics in SLAM with a hierarchically categorical Gaussian splatting, 2024, arXiv preprint arXiv:2409.12518.

[91] I. Kostavelis, A. Gasteratos, Semantic mapping for mobile robotics tasks: A survey, Robot. Auton. Syst. 66 (2015) 86–103.

[92] M. Sualeh, G.-W. Kim, Simultaneous localization and mapping in the epoch of semantics: a survey, Int. J. Control. Autom. Syst. 17 (3) (2019) 729–742.

[93] K. Chen, J. Zhang, J. Liu, Q. Tong, R. Liu, S. Chen, Semantic visual simultaneous localization and mapping: A survey, 2022, arXiv preprint arXiv:2209.06428.

[94] W. Chen, G. Shang, A. Ji, C. Zhou, X. Wang, C. Xu, Z. Li, K. Hu, An overview on visual slam: From tradition to semantic, Remote. Sens. 14 (13) (2022) 3010.

[95] H. Pu, J. Luo, G. Wang, T. Huang, H. Liu, Visual SLAM integration with semantic segmentation and deep learning: A review, IEEE Sensors J. 23 (19) (2023) 22119–22138.

[96] Y. Wang, Y. Tian, J. Chen, K. Xu, X. Ding, A survey of visual SLAM in dynamic environment: The evolution from geometric to semantic approaches, IEEE Trans. Instrum. Meas. (2024).

[97] S. Garg, N. Sünderhauf, F. Dayoub, D. Morrison, A. Cosgun, G. Carneiro, Q. Wu, T.-J. Chin, I. Reid, S. Gould, et al., Semantics for robotic mapping, perception and interaction: A survey, Found. Trends® Robot. 8 (1–2) (2020) 1–224.

[98] S. Wen, P. Li, Y. Zhao, H. Zhang, F. Sun, Z. Wang, Semantic visual SLAM in dynamic environment, Auton. Robots 45 (4) (2021) 493–504.

[99] T. Lai, A review on visual-slam: Advancements from geometric modelling to learning-based semantic scene understanding using multi-modal sensor fusion, Sensors 22 (19) (2022) 7265.

[100] H. Bavle, J.L. Sanchez-Lopez, C. Cimarelli, A. Tourani, H. Voos, From slam to situational awareness: Challenges and survey, Sensors 23 (10) (2023) 4849.

[101] B. Georgevich Ferreira, A.J. Miranda de Sousa, L. Reis, Semantic mapping for robotics: Survey, trends and challenges, Trends Challenges (2025).

[102] X. Song, X. Liang, Z. Huaidong, Semantic mapping techniques for indoor mobile robots: Review and prospect, Meas. Control. 58 (3) (2025) 377–393.

[103] B. Georgevich Ferreira, A.J. Miranda De Sousa, L. Reis, Semantic Mapping for Robotics: Survey, Trends and Challenges, 2025.

[104] S. Thrun, Probabilistic robotics, Commun. ACM 45 (3) (2002) 52–57.

[105] S. Thrun, W. Burgard, D. Fox, Probabilistic roboticsv2, MIT Press, 2005.

[106] O. Sigaud, O. Buffet, Markov decision processes in artificial intelligence, John Wiley & Sons, 2013.

[107] J. Long, E. Shelhamer, T. Darrell, Fully convolutional networks for semantic segmentation, Proc. the IEEE Conf. Comput. Vis. Pattern Recognit. (2015) 3431–3440.

[108] K. He, G. Gkioxari, P. Doll'ar, R. Girshick, Mask R-CNN, Proc. the IEEE Int. Conf. Comput. Vis. (2017) 2961–2969.

[109] V. Indelman, L. Carlone, F. Dellaert, Planning in the continuous domain: A generalized belief space approach for autonomous navigation in unknown environments, Int. J. Robot. Res. 34 (7) (2015) 849–882.

[110] N. Sünderhauf, M. Milford, Dual quadrics from object detection boundingboxes as landmark representations in slam, 2017, arXiv preprint arXiv:1708.00965.

[111] R.F. Salas-Moreno, R.A. Newcombe, H. Strasdat, P.H. Kelly, A.J. Davison, Slam++: Simultaneous localisation and mapping at the level of objects, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2013, pp. 1352–1359.

[112] L.-C. Chen, G. Papandreou, F. Schroff, H. Adam, Rethinking atrous convolution for semantic image segmentation, 2017, arXiv preprint arXiv:1706.05587.

[113] B. Bescos, J.M. Fácil, J. Civera, J. Neira, Dynaslam: Tracking, mapping, and inpainting in dynamic scenes, IEEE Robot. Autom. Lett. 3 (4) (2018) 4076–4083.

[114] Y. Fan, Q. Zhang, Y. Tang, S. Liu, H. Han, Blitz-SLAM: A semantic SLAM in dynamic environments, Pattern Recognit. 121 (2022) 108225.

[115] Q. Ji, Z. Zhang, Y. Chen, E. Zheng, Drv-slam: An adaptive real-time semantic visual slam based on instance segmentation toward dynamic environments, IEEE Access 12 (2024) 43827–43837.

[116] J. Stückler, N. Biresev, S. Behnke, Semantic mapping using object-class segmentation of RGB-d images, in: 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems, IEEE, 2012, pp. 3005–3010.

[117] Y. Nakajima, K. Tateno, F. Tombari, H. Saito, Fast and accurate semantic mapping through geometric-based incremental segmentation, in: 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS, IEEE, 2018, pp. 385–392.

[118] J. Yu, S. Shen, SemanticLoop: Loop closure with 3D semantic graph matching, IEEE Robot. Autom. Lett. 8 (2) (2023) 568–575.

[119] Z. Cao, Q. Zhang, J. Guang, S. Wu, Z. Hu, J. Liu, SemanticTopoLoop: Semantic loop closure with 3D topological graph based on quadric-level object map, IEEE Robot. Autom. Lett. (2024).

[120] S. Arshad, G.-W. Kim, Semantics aware loop closure detection in visual SLAM, in: 2021 21st International Conference on Control, Automation and Systems, ICCAS, IEEE, 2021, pp. 21–24.

[121] Z. Qian, J. Fu, J. Xiao, Towards accurate loop closure detection in semantic SLAM with 3D semantic covisibility graphs, IEEE Robot. Autom. Lett. 7 (2) (2022) 2455–2462.

[122] S. Lin, J. Wang, M. Xu, H. Zhao, Z. Chen, Topology aware object-level semantic mapping towards more robust loop closure, IEEE Robot. Autom. Lett. 6 (4) (2021) 7041–7048.

[123] L. Schmid, M. Abate, Y. Chang, L. Carlone, Khronos: A unified approach for spatio-temporal metric-semantic slamin dynamic environments, in: Proc. of Robotics: Science and Systems, 2024.

[124] J.C.V. Soares, M. Gattass, M.A. Meggiolaro, Visual SLAM in human populated environments: exploring the trade-off between accuracy and speed of YOLO and mask R-CNN, in: 2019 19th International Conference on Advanced Robotics, ICAR, IEEE, 2019, pp. 135–140.

[125] Y. Liu, J. Miura, RDS-SLAM: Real-time dynamic SLAM using semantic segmentation methods, IEEE Access 9 (2021) 23772–23785.

[126] J.C.V. Soares, M. Gattass, M.A. Meggiolaro, Crowd-SLAM: visual SLAM towards crowded environments using object detection, J. Intell. Robot. Syst. 102 (2) (2021) 50.

[127] Y. Liu, J. Miura, RDMO-slam: Real-time visual SLAM for dynamic environments using semantic label prediction with optical flow, IEEE Access 9 (2021) 106981–106997.

[128] J. Redmon, S. Divvala, R. Girshick, A. Farhadi, You only look once: Unified, real-time object detection, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 779–788.
[129] S. Ren, K. He, R. Girshick, J. Sun, Faster r-cnn: Towards real-time object detection with region proposal networks, Adv. Neural Inf. Process. Syst. 28 (2015).
[130] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, A.C. Berg, Ssd: Single shot multibox detector, in: Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, the Netherlands, October 11–14, 2016, Proceedings, Part I 14, Springer, 2016, pp. 21–37.
[131] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, A.L. Yuille, Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs, IEEE Trans. Pattern Anal. Mach. Intell. 40 (4) (2017) 834–848.
[132] V. Badrinarayanan, A. Kendall, R. Cipolla, Segnet: A deep convolutional encoder-decoder architecture for image segmentation, IEEE Trans. Pattern Anal. Mach. Intell. 39 (12) (2017) 2481–2495.
[133] X. Wang, T. Kong, C. Shen, Y. Jiang, L. Li, Solo: Segmenting objects by locations, in: Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XVIII 16, Springer, 2020, pp. 649–665.
[134] A. Kirillov, Y. Wu, K. He, R. Girshick, Pointrend: Image segmentation as rendering, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 9799–9808.
[135] A. Kirillov, K. He, R. Girshick, C. Rother, P. Dollár, Panoptic segmentation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 9404–9413.
[136] B. Cheng, M.D. Collins, Y. Zhu, T. Liu, T.S. Huang, H. Adam, L.-C. Chen, Panoptic-deeplab: A simple, strong, and fast baseline for bottom-up panoptic segmentation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 12475–12485.
[137] J. Redmon, A. Farhadi, YOLO9000: better, faster, stronger, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 7263–7271.
[138] J. Redmon, A. Farhadi, Yolov3: An incremental improvement, 2018, arXiv preprint arXiv:1804.02767.
[139] A. Bochkovskiy, C.-Y. Wang, H.-Y.M. Liao, Yolov4: Optimal speed and accuracy of object detection, 2020, arXiv preprint arXiv:2004.10934.
[140] G. Jocher, Ultralytics YOLOv5, 2020, http://dx.doi.org/10.5281/zenodo.3908559.
[141] C. Li, L. Li, H. Jiang, K. Weng, Y. Geng, L. Li, Z. Ke, Q. Li, M. Cheng, W. Nie, et al., YOLOv6: A single-stage object detection framework for industrial applications, 2022, arXiv preprint arXiv:2209.02976.
[142] C.-Y. Wang, A. Bochkovskiy, H.-Y.M. Liao, YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2023, pp. 7464–7475.
[143] G. Jocher, J. Qiu, A. Chaurasia, Ultralytics YOLO, 2023.
[144] C.-Y. Wang, I.-H. Yeh, H.-Y. Mark Liao, Yolov9: Learning what you want to learn using programmable gradient information, in: European Conference on Computer Vision, Springer, 2024, pp. 1–21.
[145] R. Girshick, J. Donahue, T. Darrell, J. Malik, Rich feature hierarchies for accurate object detection and semantic segmentation, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2014, pp. 580–587.
[146] R. Girshick, Fast r-cnn, in: Proceedings of the IEEE International Conference on Computer Vision, 2015, pp. 1440–1448.
[147] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, S. Zagoruyko, End-to-end object detection with transformers, in: European Conference on Computer Vision, Springer, 2020, pp. 213–229.
[148] H. Zhang, F. Li, S. Liu, L. Zhang, H. Su, J. Zhu, L.M. Ni, H.-Y. Shum, Dino: Detr with improved denoising anchor boxes for end-to-end object detection, 2022, arXiv preprint arXiv:2203.03605.
[149] Y. Zhou, O. Tuzel, Voxelnet: End-to-end learning for point cloud based 3d object detection, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 4490–4499.
[150] C.R. Qi, W. Liu, C. Wu, H. Su, L.J. Guibas, Frustum pointnets for 3d object detection from rgb-d data, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 918–927.
[151] C. Wang, D. Xu, Y. Zhu, R. Martín-Martín, C. Lu, L. Fei-Fei, S. Savarese, Densefusion: 6d object pose estimation by iterative dense fusion, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 3343–3352.
[152] W. Wu, L. Guo, H. Gao, Z. You, Y. Liu, Z. Chen, YOLO-slam: A semantic SLAM system towards dynamic environment with geometric constraint, Neural Comput. Appl. (2022) 1–16.
[153] X. Xia, P. Zhang, J. Sun, YOLO-based semantic segmentation for dynamic removal in visual-inertial SLAM, in: Chinese Intelligent Systems Conference, Springer, 2023, pp. 377–389.
[154] C. Gong, Y. Sun, C. Zou, B. Tao, L. Huang, Z. Fang, D. Tang, Real-time visual SLAM based YOLO-fastest for dynamic scenes, Meas. Sci. Technol. 35 (5) (2024) 056305.
[155] C. Rui, Y. Liu, J. Shen, Z. Li, Z. 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, IEEE, 2021, pp. 409–414.
[156] Z. Xie, Z. Li, Y. Zhang, J. Zhang, F. Liu, W. Chen, A multi-sensory guidance system for the visually impaired using yolo and ORB-SLAM, Information 13 (7) (2022) 343.
[157] G. Li, H. Fan, G. Jiang, D. Jiang, Y. Liu, B. Tao, J. Yun, RGBD-slam based on object detection with two-stream YOLOv4-MobileNetv3 in autonomous driving, IEEE Trans. Intell. Transp. Syst. 25 (3) (2023) 2847–2857.
[158] C. Shao, L. Zhang, W. Pan, Faster R-CNN learning-based semantic filter for geometry estimation and its application in vSLAM systems, IEEE Trans. Intell. Transp. Syst. 23 (6) (2021) 5257–5266.
[159] X. Zhang, X. Wang, R. Zhang, Dynamic semantics SLAM based on improved mask R-CNN, IEEE Access 10 (2022) 126525–126535.
[160] L. Zhang, L. Wei, P. Shen, W. Wei, G. Zhu, J. Song, Semantic SLAM based on object detection and improved octomap, IEEE Access 6 (2018) 75545–75559.
[161] Q.U. Islam, F. Khozaei, I. Baig, D. Ignatyev, et al., Advancing autonomous SLAM systems: Integrating YOLO object detection and enhanced loop closure techniques for robust environment mapping, Robot. Auton. Syst. 185 (2025) 104871.
[162] P. Wu, P. Tong, X. Zhou, X. Yang, Dyn-darkslam: YOLO-based visual SLAM in low-light conditions, in: 2024 IEEE 25th China Conference on System Simulation Technology and Its Application, CCSSTA, IEEE, 2024, pp. 346–351.
[163] O. Ronneberger, P. Fischer, T. Brox, U-net: Convolutional networks for biomedical image segmentation, in: Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18, Springer, 2015, pp. 234–241.
[164] A. Kendall, V. Badrinarayanan, R. Cipolla, Bayesian segnet: Model uncertainty in deep convolutional encoder-decoder architectures for scene understanding, 2015, arXiv preprint arXiv:1511.02680.
[165] H. Zhao, J. Shi, X. Qi, X. Wang, J. Jia, Pyramid scene parsing network, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 2881–2890.
[166] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, H. Adam, Encoder-decoder with atrous separable convolution for semantic image segmentation, in: Proceedings of the European Conference on Computer Vision, ECCV, 2018, pp. 801–818.
[167] D. Bolya, C. Zhou, F. Xiao, Y.J. Lee, Yolact: Real-time instance segmentation, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2019, pp. 9157–9166.
[168] S. Liu, L. Qi, H. Qin, J. Shi, J. Jia, Path aggregation network for instance segmentation, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 8759–8768.
[169] K. Chen, J. Pang, J. Wang, Y. Xiong, X. Li, S. Sun, W. Feng, Z. Liu, J. Shi, W. Ouyang, et al., Hybrid task cascade for instance segmentation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 4974–4983.
[170] B. Cheng, I. Misra, A.G. Schwing, A. Kirillov, R. Girdhar, Masked-attention mask transformer for universal image segmentation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 1290–1299.
[171] Y. Fang, S. Yang, X. Wang, Y. Li, C. Fang, Y. Shan, B. Feng, W. Liu, Instances as queries, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021, pp. 6910–6919.
[172] B. Xu, W. Li, D. Tzoumanikas, M. Bloesch, A. Davison, S. Leutenegger, Mid-fusion: Octree-based object-level multi-instance dynamic slam, in: 2019 International Conference on Robotics and Automation, ICRA, IEEE, 2019, pp. 5231–5237.
[173] H. Oleynikova, Z. Taylor, M. Fehr, R. Siegwart, J. Nieto, Voxblox: Incremental 3d euclidean signed distance fields for on-board mav planning, in: 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS, IEEE, 2017, pp. 1366–1373.
[174] L. Jiang, H. Zhao, S. Shi, S. Liu, C.-W. Fu, J. Jia, Pointgroup: Dual-set point grouping for 3d instance segmentation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 4867–4876.
[175] J. Hou, A. Dai, M. Nießner, 3D-sis: 3d semantic instance segmentation of rgb-d scans, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 4421–4430.
[176] W. Shi, J. Xu, D. Zhu, G. Zhang, X. Wang, J. Li, X. Zhang, RGB-d semantic segmentation and label-oriented voxelgrid fusion for accurate 3D semantic mapping, IEEE Trans. Circuits Syst. Video Technol. 32 (1) (2021) 183–197.
[177] R. Mohan, A. Valada, Efficientps: Efficient panoptic segmentation, Int. J. Comput. Vis. 129 (5) (2021) 1551–1579.
[178] D. De Geus, P. Meletis, G. Dubbelman, Fast panoptic segmentation network, IEEE Robot. Autom. Lett. 5 (2) (2020) 1742–1749.
[179] G.F. Abati, J.C.V. Soares, V.S. Medeiros, M.A. Meggiolaro, C. Semini, Panoptic-SLAM: Visual SLAM in dynamic environments using panoptic segmentation, in: 2024 21st International Conference on Ubiquitous Robots, UR, IEEE, 2024, pp. 01–08.

[180] G. Li, J. Cai, C. Huang, H. Luo, J. Yu, PS-SLAM: A visual SLAM for semantic mapping in dynamic outdoor environment using panoptic segmentation, IEEE Access (2025).
[181] N. Sünderhauf, F. Dayoub, S. McMahon, B. Talbot, R. Schulz, P. Corke, G. Wyeth, B. Upcroft, M. Milford, Place categorization and semantic mapping on a mobile robot, in: 2016 IEEE International Conference on Robotics and Automation, ICRA, IEEE, 2016, pp. 5729–5736.
[182] S. Garg, A. Jacobson, S. Kumar, M. Milford, Improving condition-and environment-invariant place recognition with semantic place categorization, in: 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS, IEEE, 2017, pp. 6863–6870.
[183] S. Arshad, T.-H. Park, Visem: A visual and semantic information fusion based place recognition for long term autonomous navigation, in: 2023 IEEE 26th International Conference on Intelligent Transportation Systems, ITSC, IEEE, 2023, pp. 2151–2156.
[184] S. Woo, S.-W. Kim, Context-based visual-language place recognition, 2024, arXiv preprint arXiv:2410.19341.
[185] A. Krizhevsky, I. Sutskever, G.E. Hinton, Imagenet classification with deep convolutional neural networks, Adv. Neural Inf. Process. Syst. 25 (2012).
[186] K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, 2014, arXiv preprint arXiv:1409.1556.
[187] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, A. Rabinovich, Going deeper with convolutions, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 1–9.
[188] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 770–778.
[189] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, L. Fei-Fei, Imagenet: A large-scale hierarchical image database, in: 2009 IEEE Conference on Computer Vision and Pattern Recognition, Ieee, 2009, pp. 248–255.
[190] B. Zhou, A. Lapedriza, A. Khosla, A. Oliva, A. Torralba, Places: A 10 million image database for scene recognition, IEEE Trans. Pattern Anal. Mach. Intell. 40 (6) (2017) 1452–1464.
[191] G. Patterson, C. Xu, H. Su, J. Hays, The sun attribute database: Beyond categories for deeper scene understanding, Int. J. Comput. Vis. 108 (2014) 59–81.
[192] R. Arandjelovic, P. Gronat, A. Torii, T. Pajdla, J. Sivic, Netvlad: CNN architecture for weakly supervised place recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 5297–5307.
[193] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, et al., An image is worth 16x16 words: Transformers for image recognition at scale, 2020, arXiv preprint arXiv:2010.11929.
[194] R. Wang, Y. Shen, W. Zuo, S. Zhou, N. Zheng, Transvpr: Transformer-based place recognition with multi-level attention aggregation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 13648–13657.
[195] I. Kostavelis, A. Gasteratos, Semantic maps from multiple visual cues, Expert Syst. Appl. 68 (2017) 45–57.
[196] H.-Y. Lin, C.-W. Yao, K.-S. Cheng, V.L. Tran, Topological map construction and scene recognition for vehicle localization, Auton. Robots 42 (2018) 65–81.
[197] N. Zimmerman, T. Guadagnino, X. Chen, J. Behley, C. Stachniss, Long-term localization using semantic cues in floor plan maps, IEEE Robot. Autom. Lett. 8 (1) (2022) 176–183.
[198] L. Suomela, J. Kalliola, H. Edelman, J.-K. Kämäräinen, Placenav: Topological navigation through place recognition, in: 2024 IEEE International Conference on Robotics and Automation, ICRA, IEEE, 2024, pp. 5205–5213.
[199] K. Muravyev, A. Melekhin, D. Yudin, K. Yakovlev, PRISM-TopoMap: online topological mapping with place recognition and scan matching, IEEE Robot. Autom. Lett. (2025).
[200] X. Long, W. Zhang, B. Zhao, Pspnet-SLAM: A semantic SLAM detect dynamic object by pyramid scene parsing network, IEEE Access 8 (2020) 214685–214695.
[201] H. Zhou, J. Fang, L. Zhang, Y. Xu, Y. Liu, Semantic scene completion from a single depth image, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2019, pp. 1746–1754.
[202] J. Civera, D. Gálvez-López, L. Riazuelo, J.D. Tardós, J.M.M. Montiel, Towards semantic SLAM using a monocular camera, in: 2011 IEEE/RSJ International Conference on Intelligent Robots and Systems, IEEE, 2011, pp. 1277–1284.
[203] S. Vasudevan, R. Siegwart, Bayesian space conceptualization and place classification for semantic maps in mobile robotics, Robot. Auton. Syst. 56 (6) (2008) 522–537.
[204] X. Kong, X. Yang, G. Zhai, X. Zhao, X. Zeng, M. Wang, Y. Liu, W. Li, F. Wen, Semantic graph based place recognition for 3d point clouds, in: 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS, IEEE, 2020, pp. 8216–8223.
[205] K. Singh, J.J. Leonard, Open-set semantic uncertainty aware metric-semantic graph matching, 2024, arXiv preprint arXiv:2409.11555.
[206] K.B. Tennakoon, O. De Silva, A. Jayasiri, G.K. Mann, R.G. Gosine, Factor graph localization for mobile robots using google indoor street view and CNN-based place recognition, Drone Syst. Appl. 11 (2023) 1–19.
[207] N. Patel, P. Krishnamurthy, F. Khorrami, Semantic segmentation guided slam using vision and lidar, in: ISR 2018; 50th International Symposium on Robotics, VDE, 2018, pp. 1–7.
[208] Z.-J. Du, S.-S. Huang, T.-J. Mu, Q. Zhao, R.R. Martin, K. Xu, Accurate dynamic SLAM using CRF-based long-term consistency, IEEE Trans. Vis. Comput. Graphics 28 (4) (2020) 1745–1757.
[209] H. Jeon, C. Han, D. You, J. Oh, RGB-d visual SLAM algorithm using scene flow and conditional random field in dynamic environments, in: 2022 22nd International Conference on Control, Automation and Systems, ICCAS, IEEE, 2022, pp. 129–134.
[210] A. Pronobis, R.P. Rao, Learning deep generative spatial models for mobile robots, in: 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS, IEEE, 2017, pp. 755–762.
[211] R. Zhang, H.M. Bong, G. Beltrame, Active semantic mapping and pose graph spectral analysis for robot exploration, in: 2024 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS, IEEE, 2024, pp. 13787–13794.
[212] H. Bavle, P. De La Puente, J.P. How, P. Campoy, VPS-SLAM: Visual planar semantic SLAM for aerial robotic systems, IEEE Access 8 (2020) 60704–60718.
[213] T. Qin, T. Chen, Y. Chen, Q. Su, Avp-slam: Semantic visual mapping and localization for autonomous vehicles in the parking lot, in: 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS, IEEE, 2020, pp. 5939–5945.
[214] I.D. Miller, R. Soussan, B. Coltin, T. Smith, V. Kumar, Robust semantic mapping and localization on a free-flying robot in microgravity, in: 2022 International Conference on Robotics and Automation, ICRA, IEEE, 2022, pp. 4121–4127.
[215] H. Liu, H. Ma, L. Zhang, Visual odometry based on semantic supervision, in: 2019 IEEE International Conference on Image Processing, ICIP, IEEE, 2019, pp. 2566–2570.
[216] L. Cui, C. Ma, SOF-SLAM: A semantic visual SLAM for dynamic environments, IEEE Access 7 (2019) 166528–166539.
[217] L. An, X. Zhang, H. Gao, Y. Liu, Semantic segmentation–aided visual odometry for urban autonomous driving, Int. J. Adv. Robot. Syst. 14 (5) (2017) 1729881417735667.
[218] X. Yuan, S. Chen, Sad-slam: A visual slam based on semantic and depth information, in: 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS, IEEE, 2020, pp. 4930–4935.
[219] L. Yan, X. Hu, L. Zhao, Y. Chen, P. Wei, H. Xie, Dgs-slam: A fast and robust rgbd slam in dynamic environments combined by geometric and semantic information, Remote. Sens. 14 (3) (2022) 795.
[220] C. Sheng, S. Pan, W. Gao, Y. Tan, T. Zhao, Dynamic-DSO: direct sparse odometry using objects semantic information for dynamic environments, Appl. Sci. 10 (4) (2020) 1467.
[221] X. Li, Y. Shen, J. Lu, Q. Jiang, O. Xie, Y. Yang, Q. Zhu, Dystslam: an efficient stereo vision SLAM system in dynamic environment, Meas. Sci. Technol. 34 (2) (2022) 025105.
[222] Y. Zhong, S. Hu, G. Huang, L. Bai, Q. Li, WF-SLAM: A robust VSLAM for dynamic scenarios via weighted features, IEEE Sensors J. 22 (11) (2022) 10818–10827.
[223] C. Zhang, T. Huang, R. Zhang, X. Yi, PLD-SLAM: A new RGB-d SLAM method with point and line features for indoor dynamic scene, ISPRS Int. J. Geo-Information 10 (3) (2021) 163.
[224] Y. Wang, H. Bu, X. Zhang, J. Cheng, YPD-SLAM: A real-time VSLAM system for handling dynamic indoor environments, Sensors 22 (21) (2022) 8561.
[225] C. Yuan, Y. Xu, Q. Zhou, PLDS-SLAM: point and line features SLAM in dynamic environment, Remote. Sens. 15 (7) (2023) 1893.
[226] X. Lu, H. Wang, S. Tang, H. Huang, C. Li, DM-SLAM: Monocular SLAM in dynamic environments, Appl. Sci. 10 (12) (2020) 4252.
[227] T. Ji, C. Wang, L. Xie, Towards real-time semantic rgb-d slam in dynamic environments, in: 2021 IEEE International Conference on Robotics and Automation, ICRA, IEEE, 2021, pp. 11175–11181.
[228] X. Hu, Y. Zhang, Z. Cao, R. Ma, Y. Wu, Z. Deng, W. Sun, CFP-SLAM: A real-time visual SLAM based on coarse-to-fine probability in dynamic environments, in: 2022 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS, IEEE, 2022, pp. 4399–4406.
[229] Y. Wei, B. Zhou, Y. Duan, J. Liu, D. An, DO-SLAM: research and application of semantic SLAM system towards dynamic environments based on object detection, Appl. Intell. 53 (24) (2023) 30009–30026.
[230] J. He, M. Li, Y. Wang, H. Wang, OVD-SLAM: An online visual SLAM for dynamic environments, IEEE Sensors J. 23 (12) (2023) 13210–13219.
[231] Q.U. Islam, H. Ibrahim, P.K. Chin, K. Lim, M.Z. Abdullah, MVS-SLAM: Enhanced multiview geometry for improved semantic RGBD slam in dynamic environment, J. Field Robot. 41 (1) (2024) 109–130.
[232] F. Zhong, S. Wang, Z. Zhang, Y. Wang, Detect-SLAM: Making object detection and SLAM mutually beneficial, in: 2018 IEEE Winter Conference on Applications of Computer Vision, WACV, IEEE, 2018, pp. 1001–1010.

[233] M. Contreras, N.P. Bhatt, E. Hashemi, DynaNav-SVO: Dynamic stereo visual odometry with semantic-aware perception for autonomous navigation, IEEE Trans. Intell. Veh. (2024).
[234] S. Cheng, C. Sun, S. Zhang, D. Zhang, SG-SLAM: A real-time RGB-d visual SLAM toward dynamic scenes with semantic and geometric information, IEEE Trans. Instrum. Meas. 72 (2022) 1–12.
[235] H. Zhang, T.N. Canh, C. Li, N.Y. Chong, Adaptive prior scene-object SLAM for dynamic environments, in: 2025 IEEE International Conference on Real-Time Computing and Robotics, RCAR, IEEE, 2025, pp. 1–6.
[236] R. Qian, H. Guo, M. Chen, G. Gong, H. Cheng, A visual SLAM algorithm based on instance segmentation and background inpainting in dynamic scenes, in: 2023 38th Youth Academic Annual Conference of Chinese Association of Automation, YAC, IEEE, 2023, pp. 132–136.
[237] B. Bescos, C. Cadena, J. Neira, Empty cities: A dynamic-object-invariant space for visual SLAM, IEEE Trans. Robot. 37 (2) (2020) 433–451.
[238] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, Y. Bengio, Generative adversarial networks, Commun. ACM 63 (11) (2020) 139–144.
[239] B. Bescos, C. Campos, J.D. Tardós, J. Neira, Dynaslam II: Tightly-coupled multi-object tracking and SLAM, IEEE Robot. Autom. Lett. 6 (3) (2021) 5191–5198.
[240] I. Ballester, A. Fontán, J. Civera, K.H. Strobl, R. Triebel, DOT: Dynamic object tracking for visual SLAM, in: 2021 IEEE International Conference on Robotics and Automation, ICRA, IEEE, 2021, pp. 11705–11711.
[241] P. Zhou, Y. Liu, Z. Meng, Pointslot: Real-time simultaneous localization and object tracking for dynamic environment, IEEE Robot. Autom. Lett. 8 (5) (2023) 2645–2652.
[242] J. Zhang, M. Henein, R. Mahony, V. Ila, Robust ego and object 6-dof motion estimation and tracking, in: 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS, IEEE, 2020, pp. 5017–5023.
[243] H. Zhang, H. Uchiyama, S. Ono, H. Kawasaki, MOTSLAM: MOT-assisted monocular dynamic SLAM using single-view depth estimation, in: 2022 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS, IEEE, 2022, pp. 4865–4872.
[244] Z. Xing, X. Zhu, D. Dong, DE-SLAM: SLAM for highly dynamic environment, J. Field Robot. 39 (5) (2022) 528–542.
[245] M. Gonzalez, E. Marchand, A. Kacete, J. Royan, Twistslam: Constrained slam in dynamic environment, IEEE Robot. Autom. Lett. 7 (3) (2022) 6846–6853.
[246] M. Schorghuber, D. Steininger, Y. Cabon, M. Humenberger, M. Gelautz, SLAMANTIC-leveraging semantics to improve VSLAM in dynamic environments, in: Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops, 2019.
[247] J. Zhang, M. Henein, R. Mahony, V. Ila, VDO-SLAM: A visual dynamic object-aware SLAM system, 2020, arXiv preprint arXiv:2005.11052.
[248] J. Huang, S. Yang, T.-J. Mu, S.-M. Hu, Clustervo: Clustering moving instances and estimating visual odometry for self and surroundings, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 2168–2177.
[249] Y. Chang, J. Hu, S. Xu, OTE-SLAM: An object tracking enhanced visual SLAM system for dynamic environments, Sensors 23 (18) (2023) 7921.
[250] Y. Zhang, P. Sun, Y. Jiang, D. Yu, F. Weng, Z. Yuan, P. Luo, W. Liu, X. Wang, Bytetrack: Multi-object tracking by associating every detection box, in: European Conference on Computer Vision, Springer, 2022, pp. 1–21.
[251] I.A. Bârsan, P. Liu, M. Pollefeys, A. Geiger, Robust dense mapping for large-scale dynamic environments, in: 2018 IEEE International Conference on Robotics and Automation, ICRA, IEEE, 2018, pp. 7510–7517.
[252] T. Whelan, R.F. Salas-Moreno, B. Glocker, A.J. Davison, S. Leutenegger, Elastic-Fusion: Real-time dense SLAM and light source estimation, Int. J. Robot. Res. 35 (14) (2016) 1697–1716.
[253] M. Grinvald, F. Furrer, T. Novkovic, J.J. Chung, C. Cadena, R. Siegwart, J. Nieto, Volumetric instance-aware semantic mapping and 3D object discovery, IEEE Robot. Autom. Lett. 4 (3) (2019) 3037–3044.
[254] T. Schöps, T. Sattler, M. Pollefeys, Surfelmeshing: Online surfel-based mesh reconstruction, IEEE Trans. Pattern Anal. Mach. Intell. 42 (10) (2019) 2494–2507.
[255] Y. Mehan, K. Gupta, R. Jayanti, A. Govil, S. Garg, M. Krishna, QueSTMaps: Queryable semantic topological maps for 3D scene understanding, in: 2024 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS, IEEE, 2024, pp. 13311–13317.
[256] R. Mascaro, M. Chli, Scene representations for robotic spatial perception, Annu. Rev. Control. Robot. Auton. Syst. 8 (2024).
[257] A. Sharma, W. Dong, M. Kaess, Compositional and scalable object slam, in: 2021 IEEE International Conference on Robotics and Automation, ICRA, IEEE, 2021, pp. 11626–11632.
[258] M. Hosseinzadeh, K. Li, Y. Latif, I. Reid, Real-time monocular object-model aware sparse SLAM, in: 2019 International Conference on Robotics and Automation, ICRA, IEEE, 2019, pp. 7123–7129.
[259] K. Chen, J. Liu, Q. Chen, Z. Wang, J. Zhang, Accurate object association and pose updating for semantic SLAM, IEEE Trans. Intell. Transp. Syst. 23 (12) (2022) 25169–25179.
[260] D. Kochanov, A. Ošep, J. Stückler, B. Leibe, Scene flow propagation for semantic mapping and object discovery in dynamic street scenes, in: 2016 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS, IEEE, 2016, pp. 1785–1792.
[261] R. Mascaro, L. Teixeira, M. Chli, Volumetric instance-level semantic mapping via multi-view 2D-to-3D label diffusion, IEEE Robot. Autom. Lett. 7 (2) (2022) 3531–3538.
[262] X. Hu, Multi-level map construction for dynamic scenes, 2023, arXiv preprint arXiv:2308.04000.
[263] B. Li, D. Zou, D. Sartori, L. Pei, W. Yu, Textslam: Visual slam with planar text features, in: 2020 IEEE International Conference on Robotics and Automation, ICRA, IEEE, 2020, pp. 2102–2108.
[264] M. Grinvald, F. Tombari, R. Siegwart, J. Nieto, Tsdf++: A multi-object formulation for dynamic object tracking and reconstruction, in: 2021 IEEE International Conference on Robotics and Automation, ICRA, IEEE, 2021, pp. 14192–14198.
[265] A. Asgharivaskasi, N. Atanasov, Semantic octree mapping and shannon mutual information computation for robot exploration, IEEE Trans. Robot. 39 (3) (2023) 1910–1928.
[266] G. Sun, X. Zhang, Y. Chu, Y. Liu, X. Zhang, Y. Zhuang, Volumetric instance-level semantic mapping via blendmask, in: 2022 IEEE/ASME International Conference on Advanced Intelligent Mechatronics, AIM, IEEE, 2022, pp. 374–379.
[267] T. Ran, L. Yuan, J. Zhang, D. Tang, L. He, RS-SLAM: A robust semantic SLAM in dynamic environments based on RGB-d sensor, IEEE Sensors J. 21 (18) (2021) 20657–20664.
[268] L. Li, Z. Liu, Ü. Özgïner, J. Lian, Y. Zhou, Y. Zhao, Dense 3D semantic SLAM of traffic environment based on stereo vision, in: 2018 IEEE Intelligent Vehicles Symposium, IV, IEEE, 2018, pp. 965–970.
[269] J. Qian, V. Chatrath, J. Yang, J. Servos, A.P. Schoellig, S.L. Waslander, Pocd: Probabilistic object-level change detection and volumetric mapping in semi-static scenes, 2022, arXiv preprint arXiv:2205.01202.
[270] J. Stückler, B. Waldvogel, H. Schulz, S. Behnke, Dense real-time mapping of object-class semantics from RGB-d video, J. Real-Time Image Process. 10 (2015) 599–609.
[271] N.D. Reddy, P. Singhal, V. Chari, K.M. Krishna, Dynamic body vslam with semantic constraints, in: 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS, IEEE, 2015, pp. 1897–1904.
[272] L. Morreale, A. Romanoni, M. Matteucci, P. di Milano, Dense 3D visual mapping via semantic simplification, in: 2019 International Conference on Robotics and Automation, ICRA, IEEE, 2019, pp. 6891–6897.
[273] D. Seichter, S.B. Fischedick, M. Köhler, H.-M. Groß, Efficient multi-task rgb-d scene analysis for indoor environments, in: 2022 International Joint Conference on Neural Networks, IJCNN, IEEE, 2022, pp. 1–10.
[274] V.A. Prisacariu, O. Kähler, S. Golodetz, M. Sapienza, T. Cavallari, P.H. Torr, D.W. Murray, Infinitam v3: A framework for large-scale 3d reconstruction with loop closure, 2017, arXiv preprint arXiv:1708.00783.
[275] Q.-H. Pham, B.-S. Hua, T. Nguyen, S.-K. Yeung, Real-time progressive 3D semantic segmentation for indoor scenes, in: 2019 IEEE Winter Conference on Applications of Computer Vision, WACV, IEEE, 2019, pp. 1089–1098.
[276] Z. Yang, C. Liu, Tupper-map: Temporal and unified panoptic perception for 3d metric-semantic mapping, in: 2021 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS, IEEE, 2021, pp. 1094–1101.
[277] Y. Xiong, R. Liao, H. Zhao, R. Hu, M. Bai, E. Yumer, R. Urtasun, Upsnet: A unified panoptic segmentation network, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 8818–8826.
[278] D. Kim, S. Woo, J.-Y. Lee, I.S. Kweon, Video panoptic segmentation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 9859–9868.
[279] L. Schmid, J. Delmerico, J.L. Schönberger, J. Nieto, M. Pollefeys, R. Siegwart, C. Cadena, Panoptic multi-tsdfs: a flexible representation for online multi-resolution volumetric mapping and long-term dynamic scene consistency, in: 2022 International Conference on Robotics and Automation, ICRA, IEEE, 2022, pp. 8018–8024.
[280] D. Seichter, B. Stephan, S.B. Fischedick, S. Mueller, L. Rabes, H.-M. Gross, Panopticndt: Efficient and robust panoptic mapping, in: 2023 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS, IEEE, 2023, pp. 7233–7240.
[281] D. Wu, Z. Yan, H. Zha, Panorecon: Real-time panoptic 3d reconstruction from monocular video, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2024, pp. 21507–21518.
[282] Z. Zhou, Y. Ma, J. Fan, S. Zhang, F. Jing, M. Tan, Eprecon: An efficient framework for real-time panoptic 3D reconstruction from monocular video, in: 2022 International Conference on Robotics and Automation, ICRA, IEEE, 2024.
[283] F. Bernuy, J. Ruiz del Solar, Semantic mapping of large-scale outdoor scenes for autonomous off-road driving, in: Proceedings of the IEEE International Conference on Computer Vision Workshops, 2015, pp. 35–41.
[284] F. Bernuy, J. Ruiz-del Solar, Topological semantic mapping and localization in urban road scenarios, J. Intell. Robot. Syst. 92 (2018) 19–32.
[285] W. Tian, X. Ren, X. Yu, M. Wu, W. Zhao, Q. Li, Vision-based mapping of lane semantics and topology for intelligent vehicles, Int. J. Appl. Earth Obs. Geoinf. 111 (2022) 102851.

[286] L. Zhao, P. Luo, Z. Zhao, L. Dong, Indoor environment semantic topological mapping based on deep learning, in: 2018 IEEE International Conference on Real-Time Computing and Robotics, RCAR, IEEE, 2018, pp. 520–525.
[287] F. Blochliger, M. Fehr, M. Dymczyk, T. Schneider, R. Siegwart, Topomap: Topological mapping and navigation based on visual slam maps, in: 2018 IEEE International Conference on Robotics and Automation, ICRA, IEEE, 2018, pp. 3818–3825.
[288] Y. Chen, J. Zhang, Y. Lou, Topological and semantic map generation for mobile robot indoor navigation, in: International Conference on Intelligent Robotics and Applications, Springer, 2021, pp. 337–347.
[289] N. Kim, O. Kwon, H. Yoo, Y. Choi, J. Park, S. Oh, Topological semantic graph memory for image-goal navigation, in: Conference on Robot Learning, PMLR, 2023, pp. 393–402.
[290] I. Armeni, Z.-Y. He, J. Gwak, A.R. Zamir, M. Fischer, J. Malik, S. Savarese, 3D scene graph: A structure for unified semantics, 3d space, and camera, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2019, pp. 5664–5673.
[291] U.-H. Kim, J.-M. Park, T.-J. Song, J.-H. Kim, 3-d scene graph: A sparse and semantic representation of physical environments for intelligent agents, IEEE Trans. Cybern. 50 (12) (2019) 4921–4933.
[292] S.-C. Wu, J. Wald, K. Tateno, N. Navab, F. Tombari, Scenegraphfusion: Incremental 3d scene graph prediction from rgb-d sequences, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 7515–7525.
[293] Y.C. Sousa, H.F. Bassani, Topological semantic mapping by consolidation of deep visual features, IEEE Robot. Autom. Lett. 7 (2) (2022) 4110–4117.
[294] B. Yang, T. Jiang, W. Wu, Y. Zhou, L. Dai, Automated semantics and topology representation of residential-building space using floor-plan raster maps, IEEE J. Sel. Top. Appl. Earth Obs. Remote. Sens. 15 (2022) 7809–7825.
[295] Z. Cao, Y. Sun, Z. Ma, M. Zhou, A context-enhanced full-resolution floor plan segmentation network for topological semantic mapping, in: 2024 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS, IEEE, 2024, pp. 9761–9768.
[296] S. Fredriksson, A. Saradagi, G. Nikolakopoulos, Semantic and topological mapping using intersection identification, IFAC-PapersOnLine 56 (2) (2023) 9251–9256.
[297] R.S.R. Kathirvel, Z.A. Chavis, S.J. Guy, K. Desingh, SENT map-semantically enhanced topological maps with foundation models, in: ICRA 2025 Workshop on Foundation Models and Neuro-Symbolic AI for Robotics, IEEE, 2025.
[298] N. Tomatis, I. Nourbakhsh, R. Siegwart, Hybrid simultaneous localization and map building: a natural integration of topological and metric, Robot. Auton. Syst. 44 (1) (2003) 3–14.
[299] B. Kuipers, J. Modayil, P. Beeson, M. MacMahon, F. Savelli, Local metrical and global topological maps in the hybrid spatial semantic hierarchy, in: IEEE International Conference on Robotics and Automation, 2004. Proceedings. ICRA'04. 2004, vol. 5, IEEE, 2004, pp. 4845–4851.
[300] R. Drouilly, P. Rives, B. Morisset, Hybrid metric-topological-semantic mapping in dynamic environments, in: 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS, IEEE, 2015, pp. 5109–5114.
[301] S. Yang, Y. Huang, S. Scherer, Semantic 3D occupancy mapping through efficient high order CRFs, in: 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS, IEEE, 2017, pp. 590–597.
[302] P. Kohli, M.P. Kumar, P.H. Torr, P3 & beyond: Solving energies with higher order cliques, in: 2007 IEEE Conference on Computer Vision and Pattern Recognition, IEEE, 2007, pp. 1–8.
[303] C. Malleson, J.-Y. Guillemaut, A. Hilton, Hybrid modeling of non-rigid scenes from RGBD cameras, IEEE Trans. Circuits Syst. Video Technol. 29 (8) (2018) 2391–2404.
[304] R.C. Luo, M. Chiou, Hierarchical semantic mapping using convolutional neural networks for intelligent service robotics, IEEE Access 6 (2018) 61287–61294.
[305] S. Wen, Y. Zhao, X. Liu, F. Sun, H. Lu, Z. Wang, Hybrid semi-dense 3D semantic-topological mapping from stereo visual-inertial odometry SLAM with loop closure detection, IEEE Trans. Veh. Technol. 69 (12) (2020) 16057–16066.
[306] Y. Yue, C. Zhao, R. Li, C. Yang, J. Zhang, M. Wen, Y. Wang, D. Wang, A hierarchical framework for collaborative probabilistic semantic mapping, in: 2020 IEEE International Conference on Robotics and Automation, ICRA, IEEE, 2020, pp. 9659–9665.
[307] Y. Deng, M. Wang, Y. Yang, Y. Yue, Hd-ccsom: Hierarchical and dense collaborative continuous semantic occupancy mapping through label diffusion, in: 2022 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS, IEEE, 2022, pp. 2417–2422.
[308] A. Rosinol, A. Violette, M. Abate, N. Hughes, Y. Chang, J. Shi, A. Gupta, L. Carlone, Kimera: From SLAM to spatial perception with 3D dynamic scene graphs, Int. J. Robot. Res. 40 (12–14) (2021) 1510–1546.
[309] Y. Zhang, G. Tian, X. Shao, S. Liu, M. Zhang, P. Duan, Building metric-topological map to efficient object search for mobile robot, IEEE Trans. Ind. Electron. 69 (7) (2021) 7076–7087.
[310] N. Hughes, Y. Chang, S. Hu, R. Talak, R. Abdulhai, J. Strader, L. Carlone, Foundations of spatial perception for robotics: Hierarchical representations and real-time systems, Int. J. Robot. Res. 43 (10) (2024) 1457–1505.
[311] J. Wald, H. Dhamo, N. Navab, F. Tombari, Learning 3d semantic scene graphs from 3d indoor reconstructions, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 3961–3970.
[312] H. Yang, P. Antonante, V. Tzoumas, L. Carlone, Graduated non-convexity for robust spatial perception: From non-minimal solvers to global outlier rejection, IEEE Robot. Autom. Lett. 5 (2) (2020) 1127–1134.
[313] D. Maggio, Y. Chang, N. Hughes, M. Trang, D. Griffith, C. Dougherty, E. Cristofalo, L. Schmid, L. Carlone, Clio: Real-time task-driven open-set 3d scene graphs, IEEE Robot. Autom. Lett. (2024).
[314] X. Zhao, W. Ding, Y. An, Y. Du, T. Yu, M. Li, M. Tang, J. Wang, Fast segment anything, 2023, arXiv preprint arXiv:2306.12156.
[315] A. Radford, J.W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, et al., Learning transferable visual models from natural language supervision, in: International Conference on Machine Learning, PmLR, 2021, pp. 8748–8763.
[316] D.-C. Hoang, A.J. Lilienthal, T. Stoyanov, Object-RPE: Dense 3D reconstruction and pose estimation with convolutional neural networks, Robot. Auton. Syst. 133 (2020) 103632.
[317] J. Kabalar, S.-C. Wu, J. Wald, K. Tateno, N. Navab, F. Tombari, Towards long-term retrieval-based visual localization in indoor environments with changes, IEEE Robot. Autom. Lett. 8 (4) (2023) 1975–1982.
[318] E. Bochinski, V. Eiselein, T. Sikora, High-speed tracking-by-detection without using image information, in: 2017 14th IEEE International Conference on Advanced Video and Signal Based Surveillance, AVSS, IEEE, 2017, pp. 1–6.
[319] M. Antonello, D. Wolf, J. Prankl, S. Ghidoni, E. Menegatti, M. Vincze, Multi-view 3D entangled forest for semantic segmentation and mapping, in: 2018 IEEE International Conference on Robotics and Automation, ICRA, IEEE, 2018, pp. 1855–1862.
[320] Y. Nakajima, K. Tateno, F. Tombari, H. Saito, Fast and Accurate Semantic Mapping through Geometric-based Incremental Segmentation, in: 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2018, pp. 385–392, ISSN: 2153-0866. URL https://ieeexplore.ieee.org/document/8593993/.
[321] L. Lu, Y. Zhang, P. Zhou, J. Qi, Y. Pan, C. Fu, J. Pan, Semantics-aware receding horizon planner for object-centric active mapping, IEEE Robot. Autom. Lett. 9 (4) (2024) 3838–3845.
[322] Y. Bao, Z. Yang, Y. Pan, R. Huan, Semantic-direct visual odometry, IEEE Robot. Autom. Lett. 7 (3) (2022) 6718–6725.
[323] J. Engel, T. Schöps, D. Cremers, LSD-SLAM: Large-scale direct monocular SLAM, in: European Conference on Computer Vision, Springer, 2014, pp. 834–849.
[324] B. Li, D. Zou, Y. Huang, X. Niu, L. Pei, W. Yu, Textslam: Visual SLAM with semantic planar text features, IEEE Trans. Pattern Anal. Mach. Intell. 46 (1) (2023) 593–610.
[325] R.A. Newcombe, S. Izadi, O. Hilliges, D. Molyneaux, D. Kim, A.J. Davison, P. Kohi, J. Shotton, S. Hodges, A. Fitzgibbon, Kinectfusion: Real-time dense surface mapping and tracking, in: 2011 10th IEEE International Symposium on Mixed and Augmented Reality, Ieee, 2011, pp. 127–136.
[326] J. Li, D. Meger, G. Dudek, Semantic mapping for view-invariant relocalization, in: 2019 International Conference on Robotics and Automation, ICRA, IEEE, 2019, pp. 7108–7115.
[327] H.W. Kuhn, The hungarian method for the assignment problem, Nav. Res. Logist. Q. 2 (1–2) (1955) 83–97.
[328] B. Mu, S.-Y. Liu, L. Paull, J. Leonard, J.P. How, SLAM with objects using a nonparametric pose graph, in: 2016 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS, IEEE, 2016, pp. 4602–4609.
[329] Y. Zhang, C. Wang, X. Wang, W. Zeng, W. Liu, Fairmot: On the fairness of detection and re-identification in multiple object tracking, Int. J. Comput. Vis. 129 (11) (2021) 3069–3087.
[330] D. Reynolds, Gaussian mixture models, in: Encyclopedia of Biometrics, Springer, 2015, pp. 827–832.
[331] J. Wang, M. Rünz, L. Agapito, DSP-SLAM: Object oriented SLAM with deep shape priors, in: 2021 International Conference on 3D Vision (3DV), IEEE, 2021, pp. 1362–1371.
[332] Y. Wu, Y. Zhang, D. Zhu, Z. Deng, W. Sun, X. Chen, J. Zhang, An object slam framework for association, mapping, and high-level tasks, IEEE Trans. Robot. 39 (4) (2023) 2912–2932.
[333] M. Strecke, J. Stuckler, Em-fusion: Dynamic object-level slam with probabilistic data association, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2019, pp. 5865–5874.
[334] S.A. Parkison, L. Gan, M.G. Jadidi, R.M. Eustice, Semantic iterative closest point through expectation-maximization., in: BMVC, vol. 1, 2018, p. 2.
[335] J. Zhang, M. Gui, Q. Wang, R. Liu, J. Xu, S. Chen, Hierarchical topic model based object association for semantic SLAM, IEEE Trans. Vis. Comput. Graphics 25 (11) (2019) 3052–3062.
[336] A. Iqbal, N.R. Gans, Localization of classified objects in slam using nonparametric statistics and clustering, in: 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS, IEEE, 2018, pp. 161–168.
[337] R. Shekhar, C. Jawahar, Word image retrieval using bag of visual words, in: 2012 10th IAPR International Workshop on Document Analysis Systems, IEEE, 2012, pp. 297–301.

[338] G. Berton, C. Masone, B. Caputo, Rethinking visual geo-localization for large-scale applications, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 4878–4888.
[339] A. Gawel, C. Del Don, R. Siegwart, J. Nieto, C. Cadena, X-view: Graph-based semantic multi-view localization, IEEE Robot. Autom. Lett. 3 (3) (2018) 1687–1694.
[340] T. Qin, P. Li, S. Shen, Vins-mono: A robust and versatile monocular visual-inertial state estimator, IEEE Trans. Robot. 34 (4) (2018) 1004–1020.
[341] S. Agarwal, K. Mierle, et al., Ceres solver: Tutorial & reference, Google Inc 2 (72) (2012) 8.
[342] M. Kaess, A. Ranganathan, F. Dellaert, iSAM2: Incremental smoothing and mapping using the Bayes tree, in: The International Journal of Robotics Research, vol. 31, (2) SAGE Publications, 2012, pp. 216–235.
[343] N. Merrill, G. Huang, CALC2. 0: Combining appearance, semantic and geometric information for robust and efficient visual loop closure, in: 2019 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS, IEEE, 2019, pp. 4554–4561.
[344] M. Hu, S. Li, J. Wu, J. Guo, H. Li, X. Kang, Loop closure detection for visual SLAM fusing semantic information, in: 2019 Chinese Control Conference, CCC, IEEE, 2019, pp. 4136–4141.
[345] Z. Yuan, K. Xu, B. Deng, X. Zhou, P. Chen, Y. Ma, SV-loop: Semantic-visual loop closure detection with panoptic segmentation, in: 2021 IEEE 6th International Conference on Signal and Image Processing, ICSIP, IEEE, 2021, pp. 245–250.
[346] K.A. Tsintotas, L. Bampis, S. Rallis, A. Gasteratos, Seqslam with bag of visual words for appearance based loop closure detection, in: International Conference on Robotics in Alpe-Adria Danube Region, Springer, 2018, pp. 580–587.
[347] I.T. Papapetros, K.M. Oikonomou, I. Kansizoglou, K.A. Tsintotas, A. Gasteratos, Semantic-based visual vocabulary for loop closure detection, in: 2023 IEEE International Conference on Imaging Systems and Techniques, IST, IEEE, 2023, pp. 1–5.
[348] D. Xiao, S. Li, Z. Xuanyuan, Semantic loop closure detection for intelligent vehicles using panoramas, IEEE Trans. Intell. Veh. 8 (10) (2023) 4395–4405.
[349] J. Kim, J. Kim, Semantic loop closure for reducing false matches in SLAM, in: 2025 22nd International Conference on Ubiquitous Robots, UR, IEEE, 2025, pp. 34–39.
[350] G. Singh, M. Wu, S.-K. Lam, D. Van Minh, Hierarchical loop closure detection for long-term visual SLAM with semantic-geometric descriptors, in: 2021 IEEE International Intelligent Transportation Systems Conference, ITSC, IEEE, 2021, pp. 2909–2916.
[351] H. Chen, G. Zhang, Y. Ye, Semantic loop closure detection with instance-level inconsistency removal in dynamic industrial scenes, IEEE Trans. Ind. Inform. 17 (3) (2021) 2030–2040.
[352] Y. Wang, W. Bai, Z. Zhang, H. Wang, H. Sun, Q. Cao, PRIOR-SLAM: Enabling visual SLAM for loop closure under large viewpoint variations, IEEE Trans. Robot. (2025).
[353] A. Agrawal, D. Agarwal, M. Arora, R. Mahajan, S. Beohar, L. Kenye, R. Kala, SLAM and map learning using hybrid semantic graph optimization, in: 2022 30th Mediterranean Conference on Control and Automation, MED, IEEE, 2022, pp. 731–736.
[354] N. Yang, R. Wang, J. Stueckler, D. Cremers, D3VO: Deep depth, deep pose and deep uncertainty for monocular visual odometry, Proc. the IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (2020) 1281–1292.
[355] F. Wimbauer, N. Yang, L. Von Stumberg, N. Zeller, D. Cremers, MonoRec: Semi-supervised dense reconstruction in dynamic environments from a single moving camera, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 6112–6122.
[356] N. Radwan, A. Valada, W. Burgard, Vlocnet++: Deep multitask learning for semantic visual localization and odometry, IEEE Robot. Autom. Lett. 3 (4) (2018) 4407–4414.
[357] U.-H. Kim, S.-H. Kim, J.-H. Kim, Simvodis: Simultaneous visual odometry, object detection, and instance segmentation, IEEE Trans. Pattern Anal. Mach. Intell. 44 (1) (2020) 428–441.
[358] U.-H. Kim, S.-H. Kim, J.-H. Kim, Simvodis++: Neural semantic visual odometry in dynamic environments, IEEE Robot. Autom. Lett. 7 (2) (2022) 4244–4251.
[359] S.B. Laina, S. Boche, S. Papatheodorou, S. Schaefer, J. Jung, S. Leutenegger, FindAnything: Open-vocabulary and object-centric mapping for robot exploration in any environment, 2025, arXiv preprint arXiv:2504.08603.
[360] H. Li, S. Yu, S. Zhang, G. Tan, Resolving loop closure confusion in repetitive environments for visual slam through ai foundation models assistance, in: 2024 IEEE International Conference on Robotics and Automation, ICRA, IEEE, 2024, pp. 6657–6663.
[361] F. Tosi, Y. Zhang, Z. Gong, E. Sandström, S. Mattoccia, M.R. Oswald, M. Poggi, How nerfs and 3d gaussian splatting are reshaping slam: a survey, 4, 2024, p. 1, arXiv preprint arXiv:2402.13255.
[362] B. Mildenhall, P.P. Srinivasan, M. Tancik, J.T. Barron, R. Ramamoorthi, R. Ng, Nerf: Representing scenes as neural radiance fields for view synthesis, Commun. ACM 65 (1) (2021) 99–106.
[363] J. Czarnowski, T. Laidlow, R. Clark, A.J. Davison, Deepfactors: Real-time probabilistic dense monocular slam, IEEE Robot. Autom. Lett. 5 (2) (2020) 721–728.
[364] X. Kong, S. Liu, M. Taher, A.J. Davison, Vmap: Vectorised object mapping for neural field slam, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2023, pp. 952–961.
[365] X. Han, H. Liu, Y. Ding, L. Yang, Ro-map: Real-time multi-object mapping with neural radiance fields, IEEE Robot. Autom. Lett. 8 (9) (2023) 5950–5957.
[366] E. Kruzhkov, A. Savinykh, P. Karpyshev, M. Kurenkov, E. Yudin, A. Potapov, D. Tsetserukou, Meslam: Memory efficient slam based on neural fields, in: 2022 IEEE International Conference on Systems, Man, and Cybernetics, SMC, IEEE, 2022, pp. 430–435.
[367] H. Wang, J. Wang, L. Agapito, Co-slam: Joint coordinate and sparse parametric encodings for neural real-time slam, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2023, pp. 13293–13302.
[368] M.M. Johari, C. Carta, F. Fleuret, Eslam: Efficient dense slam system based on hybrid representation of signed distance fields, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2023, pp. 17408–17419.
[369] B. Xiang, Y. Sun, Z. Xie, X. Yang, Y. Wang, Nisb-map: Scalable mapping with neural implicit spatial block, IEEE Robot. Autom. Lett. 8 (8) (2023) 4761–4768.
[370] A. Rosinol, J.J. Leonard, L. Carlone, Nerf-slam: Real-time dense monocular slam with neural radiance fields, in: 2023 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS, IEEE, 2023, pp. 3437–3444.
[371] H. Matsuki, E. Sucar, T. Laidow, K. Wada, R. Scona, A.J. Davison, Imode: Real-time incremental monocular dense mapping using neural field, in: 2023 IEEE International Conference on Robotics and Automation, ICRA, IEEE, 2023, pp. 4171–4177.
[372] Z. Zhu, S. Peng, V. Larsson, Z. Cui, M.R. Oswald, A. Geiger, M. Pollefeys, Nicer-slam: Neural implicit scene encoding for rgb slam, in: 2024 International Conference on 3D Vision (3DV), IEEE, 2024, pp. 42–52.
[373] L. Liso, E. Sandström, V. Yugay, L. Van Gool, M.R. Oswald, Loopy-slam: Dense neural slam with loop closures, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2024, pp. 20363–20373.
[374] L. Li, L. Zhang, Z. Wang, Y. Shen, Gs3lam: Gaussian semantic splatting slam, in: Proceedings of the 32nd ACM International Conference on Multimedia, 2024, pp. 3019–3027.
[375] H. Liu, L. Wang, H. Luo, F. Zhao, R. Chen, Y. Chen, M. Xiao, J. Yan, D. Luo, SDD-SLAM: Semantic-driven dynamic SLAM with Gaussian splatting, IEEE Robot. Autom. Lett. (2025).
[376] Z. Xin, C. Wu, P. Huang, Y. Zhang, Y. Mao, G. Huang, Large-scale Gaussian splatting SLAM, 2025, arXiv preprint arXiv:2505.09915.
[377] V. Yugay, T. Kersten, L. Carlone, T. Gevers, M.R. Oswald, L. Schmid, Gaussian mapping for evolving scenes, 2025, arXiv preprint arXiv:2506.06909.
[378] J. Cui, J. Zhang, L. Kneip, S. Schwertfeger, Neural surfel reconstruction: Addressing loop closure challenges in large-scale 3D neural scene mapping, Sensors (Basel, Switzerland) 24 (21) (2024) 6919.
[379] D.S. Chaplot, R. Salakhutdinov, A. Gupta, S. Gupta, Neural topological slam for visual navigation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 12875–12884.
[380] V. Tchuiev, V. Indelman, Distributed consistent multi-robot semantic localization and mapping, IEEE Robot. Autom. Lett. 5 (3) (2020) 4649–4656.
[381] K. Hu, L. Zhan, L. Zou, Y. Han, T. Bi, G.-M. Muntean, Cosar: Multi-robot collaborative semantic mapping over wireless networks, in: 2023 IEEE International Symposium on Broadband Multimedia Systems and Broadcasting, BMSB, IEEE, 2023, pp. 1–6.
[382] X. Liu, A. Prabhu, F. Cladera, I.D. Miller, L. Zhou, C.J. Taylor, V. Kumar, Active metric-semantic mapping by multiple aerial robots, 2022, arXiv preprint arXiv:2209.08465.
[383] A. Asgharivaskasi, F. Girke, N. Atanasov, Riemannian Optimization for Active Mapping With Robot Teams, IEEE Trans. Robot. 41 (2025) 1077–1097.
[384] G. Aguilar, I. Becerra, R. Murrieta-Cid, Multi-robot exploration and semantic map building: Heterogeneous terrestrial robots and a drone, Intel. Artif. 28 (76) (2025) 166–185.
[385] A. Dai, A.X. Chang, M. Savva, M. Halber, T. Funkhouser, M. Nießner, Scannet: Richly-annotated 3d reconstructions of indoor scenes, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 5828–5839.
[386] A. Chang, A. Dai, T. Funkhouser, M. Halber, M. Niessner, M. Savva, S. Song, A. Zeng, Y. Zhang, Matterport3D: Learning from RGB-d data in indoor environments, Int. Conf. 3D Vis. (3DV) (2017).
[387] P. Ammirato, A.C. Berg, J. Košecká, Active vision dataset benchmark, in: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, CVPRW, 2018, pp. 2127–21273.
[388] A. Geiger, P. Lenz, R. Urtasun, Are we ready for Autonomous Driving? The KITTI Vision Benchmark Suite, in: Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition, CVPR, 2012, pp. 3354–3361.
[389] M. Menze, A. Geiger, Object scene flow for autonomous vehicles, in: Conference on Computer Vision and Pattern Recognition, CVPR, 2015.
[390] Y. Liao, J. Xie, A. Geiger, KITTI-360: A novel dataset and benchmarks for urban scene understanding in 2D and 3D, Pattern Anal. Mach. Intell. (PAMI) (2022).

[391] H. Caesar, V. Bankiti, A.H. Lang, S. Vora, V.E. Liong, Q. Xu, A. Krishnan, Y. Pan, G. Baldan, O. Beijbom, Nuscenes: A multimodal dataset for autonomous driving, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 11621–11631.
[392] X. Liu, A. Prabhu, F. Cladera, I.D. Miller, L. Zhou, C.J. Taylor, V. Kumar, Active metric-semantic mapping by multiple aerial robots, 2023, arXiv preprint arXiv:2209.08465.
[393] Y. Tao, X. Liu, I. Spasojevic, S. Agarwal, V. Kumar, 3D active metric-semantic slam, IEEE Robot. Autom. Lett. 9 (3) (2024) 2989–2996.
[394] G. Georgakis, B. Bucher, K. Schmeckpeper, S. Singh, K. Daniilidis, Learning to map for active semantic goal navigation, 2022, arXiv preprint arXiv:2106.15648.
[395] C. Tian, S. Tian, Y. Kang, H. Wang, J. Tie, S. Xu, RASLS: Reinforcement learning active SLAM approach with layout semantic, in: 2024 International Joint Conference on Neural Networks, IJCNN, IEEE, 2024, pp. 1–8.
[396] Z. Ravichandran, L. Peng, N. Hughes, J.D. Griffith, L. Carlone, Hierarchical representations and explicit memory: Learning effective navigation policies on 3d scene graphs using graph neural networks, in: 2022 International Conference on Robotics and Automation, ICRA, IEEE, 2022, pp. 9272–9279.
[397] D.P. Baxter, et al., Toward robust active semantic SLAM via Max-Mixtures (Ph.D. thesis), Massachusetts Institute of Technology, 2020.
[398] N. Vödisch, D. Cattaneo, W. Burgard, A. Valada, Continual slam: Beyond lifelong simultaneous localization and mapping through continual learning, in: The International Symposium of Robotics Research, Springer, 2022, pp. 19–35.
[399] B. Li, Z. Yan, D. Wu, H. Jiang, H. Zha, Learn to memorize and to forget: A continual learning perspective of dynamic slam, in: European Conference on Computer Vision, Springer, 2024, pp. 41–57.
[400] N. Vödisch, D. Cattaneo, W. Burgard, A. Valada, Covio: Online continual learning for visual-inertial odometry, in: Proceedings of the Ieee/Cvf Conference on Computer Vision and Pattern Recognition, 2023, pp. 2464–2473.
[401] C. Huang, S. Yan, W. Burgard, BYE: Build your encoder with one sequence of exploration data for long-term dynamic scene understanding, IEEE Robot. Autom. Lett. (2025).

Thanh Nguyen Canh received his Engineering Degree in Robotics Engineering at the University of Engineering and Technology, Viet Nam National University (VNU) in 2022, and his M.Sc. degree from Japan Advanced Institute of Science and Technology (JAIST), Japan in 2024. Currently, he is pursuing a Ph.D. degree at JAIST. His research interests include Robotics, SLAM, AI, and robot vision.

Haolan Zhang received the B.Sc. degree in automation engineering from Dalian Polytechnic University, Dalian, China, in 2021 and his M.Sc. Degree from Gunma University, Kiryu, Japan in 2024. Currently, he is pursuing a Ph.D. degree at Japan Advanced Institute of Science and Technology (JAIST), Japan. His current research interests include SLAM, Computer Vision.

Xiem HoangVan is an Associate Professor at the Faculty of Electronics and Telecommunications, VNU-University of Engineering and Technology, Vietnam. He received his Ph.D. degree from Lisbon University, Portugal, in 2015, his M.Sc. degree from Sungkyunkwan University, South Korea, in 2011, all in Electrical and Computer Engineering. His research interests are machine learning, image, and video communications. Prof. Xiem has published about 100 papers on robotics, image, and video processing and regularly reviews for many renowned IEEE, IET, and EURASIP journals and serves as a technical committee member for international conferences and funding agencies worldwide. He has received several technical awards for his contributions to image and video coding, including 5 Best Paper awards, i.e., at Picture Coding Symposium 2015 (Australia), the International Workshop on Advanced Image Technology 2018 (Thailand), REC-ECIT 2022, IEEE-RIVE 2023, and ATC 2024. He is a recipient of the Fraunhofer Portugal award 2015, and Golden Globe Award for Young Scientists (under 35 years old) in Science and Technology 2019, and the VNU Top Young Scientist Award 2019.

Nak Young Chong received the B.S., M.S., and Ph.D. degrees in mechanical engineering from Hanyang University, Seoul, Korea, in 1987, 1989, and 1994, respectively. From 1994 to 2003, he was with Daewoo Heavy Industries, Geoje, Korea; Korea Institute of Science and Technology, Seoul, Korea; and Mechanical Engineering Laboratory and National Institute of Advanced Industrial Science and Technology, Tsukuba, Japan. In 2003, he joined a Faculty Member at Japan Advanced Institute of Science and Technology, Ishikawa, Japan, where he served as a Councilor, Director of the Center for Intelligent Robotics, and the Chair Professor of Intelligent Robotics Group, and is currently a Professor of information science. He was a Visiting Scholar with Northwestern University, Evanston, IL, USA; Georgia Institute of Technology, GA, USA; University of Genoa, Genoa, Italy; and Carnegie Mellon University, Pittsburgh, PA, USA, and serves/served as an Associate Faculty with the University of Nevada, Las Vegas, NV, USA; Kyung Hee University, Yongin, Korea; and Hanyang University, Ansan, Korea. He served as the Program (Co)-chair for JCK Robotics 2009, ICAM 2010, IEEE Ro-Man 2011/2013/2022, IEEE CASE 2012, URAI 2013/2014, DARS 2014, ICCAS 2016, and IEEE ARM 2019. He was the General (Co)-chair of URAI 2017 and UR 2020. He also served as the President of Korea Robotics Society, the Co-chair for IEEE RAS Networked Robots TC, and the Fujitsu Scientific System WG. He serves/served as a Senior Editor for *IEEE Robotics and Automation Letters*, *Intelligent Service Robotics*, and *International Journal of Advanced Robotic Systems*; and as an Associate Editor for IEEE Transactions on Robotics.