Robotica (2026), **44**, pp. 343–385
doi:10.1017/S0263574725102518

CAMBRIDGE UNIVERSITY PRESS

REVIEW ARTICLE

Global localization for mobile robots in symmetrical indoor environments: a review, practical challenges, and experimental validation

Shiron Anto Melvin, Hai Wang and Amirmehdi Yazdani

School of Engineering and Energy, Murdoch University, Perth, WA, Australia
Corresponding author: Shiron Anto Melvin; Email: shironjaison19@gmail.com

Received: 12 November 2025; **Revised:** 5 July 2025; **Accepted:** 16 August 2025; **First published online:** 8 January 2026

Keywords: mobile robots; indoor global localization; symmetrical environment; probabilistic-based approaches; learning-based approaches; SLAM-based approaches; optimization-based approaches

Abstract

Efficient global localization of mobile robots in symmetrical indoor environments remains a formidable challenge, given the inherent complexities arising from uniform structures and a dearth of distinctive features. This review paper conducts an in-depth investigation into the nuances of global localization strategies, focusing on symmetrical environments, such as extended corridors, symmetrical rooms, tunnels, and industrial warehouses. The study comprehensively reviews and categorizes key techniques employed in this context, encompassing probabilistic-based approaches, learning-based approaches, Simultaneous Localization and Mapping (SLAM)-based approaches, and optimization-based approaches. The primary goal is to provide a contemporary and thorough literature review, offering insights into existing global localization solutions, followed by extant methods tailored for symmetrical indoor spaces. Also, the paper addresses practical challenges associated with implementing various global localization techniques, contributing to a holistic understanding of their real-world applicability. Comparative experimental results demonstrate that hybrid approaches achieve superior localization accuracy in symmetrical environments compared to any single method alone. These experiments, conducted in indoor settings with different symmetry levels, highlight the hybrid approach's robustness and precision in resolving symmetry-induced ambiguities. This work signifies a significant step forward in mobile robot global localization, which addresses symmetrical environments' complexities by leveraging the strengths of hybrid methodologies.

1. Introduction

Over the past few decades, wheeled autonomous mobile robots have become widely utilized in factories, shopping malls, and homes to perform various service tasks due to their ability to navigate independently in indoor environments. A key component of this autonomous navigation capability is accurate localization, which enables the robot to determine its position and provides essential input for planning and control processes. Localization is particularly critical among the various capabilities required for effective robot operation. Inaccurate or unreliable localization can lead to navigation failures or even collisions, underscoring its importance in ensuring safe and efficient robot performance.

Global localization is a fundamental problem in mobile robotics that involves determining a robot's position and orientation within a known environment without prior knowledge of its initial location. This challenge is critical for autonomous navigation, especially in indoor environments where global positioning system (GPS) signals are often unavailable [1]. Successful global localization enables a robot to navigate, map, and interact with its surroundings autonomously. It is essential for various applications,

© The Author(s), 2026. Published by Cambridge University Press. This is an Open Access article, distributed under the terms of the Creative Commons Attribution-NonCommercial-ShareAlike licence (http://creativecommons.org/licenses/by-nc-sa/4.0/), which permits non-commercial re-use, distribution, and reproduction in any medium, provided the same Creative Commons licence is used to distribute the re-used or adapted article and the original article is properly cited. The written permission of Cambridge University Press or the rights holder(s) must be obtained prior to any commercial use.

https://doi.org/10.1017/S0263574725102518 Published online by Cambridge University Press

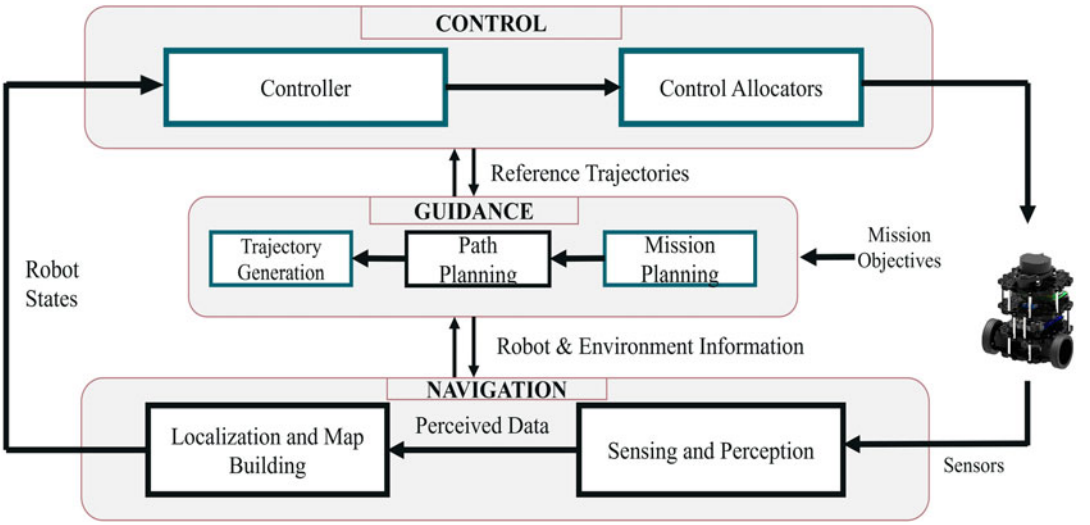


Figure 1. *Overview of an autonomous mobile robot system.*

from service robots in homes and offices to automated guided vehicles in warehouses. Figure 1 shows how global localization is intricately connected to mobile robots' navigation, guidance, and control, forming a fundamental aspect of their autonomous operation. Global localization gives the robot an accurate understanding of its position within a global coordinate system, essential for effective navigation and precise waypoint following. It ensures accurate guidance by continuously updating the robot's position, maintaining the desired trajectory, and avoiding deviations. Additionally, it supplies crucial feedback for control systems, enabling precise adjustments to the robot's movements based on its current state.

The paper broadly classifies the overview of existing methods for global localization into probabilistic-based, learning-based, Simultaneous Localization and Mapping (SLAM)-based, and optimization-based approaches.

(i) *Probabilistic-based approaches*: Bayesian filters, like the extended Kalman filter (EKF) and the particle filter, fall under this category. They use probabilistic models to estimate the robot's global pose by incorporating sensor measurements and motion models. (ii) *Learning-based approaches*: Learning-based approaches aim to capture complex relationships between sensor inputs and global poses, reducing reliance on explicit models. (iii) *SLAM-based approaches*: SLAM methods involve building a map of the environment while simultaneously localizing the robot within the map. SLAM methods can be broadly classified based on the primary sensors they use into three categories: Light Detection and Ranging (LiDAR)-based SLAM, Vision-based SLAM (VSLAM), and Hybrid SLAM. Each of these methods leverages different types of sensor data to build maps and localize the robot within an environment. (iv) *Optimization-based approaches*: Optimization-based approaches for mobile robot localization focus on formulating the localization problem as an optimization problem. The objective is to find the robot's pose (position and orientation) that minimizes (or maximizes) a certain cost (or reward) function.

Indoor environments often pose unique challenges for global localization, particularly when they are symmetrical or lack distinctive features. These symmetrical environments, as shown in Figure 2, can adversely affect the global localization of mobile robots due to the challenge of observing a large quantity of similar structures with few distinct features in these settings [2, 3]. These scenarios intensify the problem of global localization, requiring more sophisticated algorithms to determine the robot's position reliably. Current solutions depend on approaches such as artificial landmarks [2], QR codes [4], etc. Furthermore, achieving global localization usually requires more than just a few sensor scans to succeed [5, 6], especially in the case of symmetric environments.

https://doi.org/10.1017/S0263574725102518 Published online by Cambridge University Press



Figure 2. *Symmetrical scenarios: (a) office environment, (b) tunnel, (c) industrial warehouse, and (d) long corridors.*

Despite significant advancements, several gaps and challenges remain in the field of global localization:

- *Ambiguity in symmetrical environments*: Numerous existing methods [5–10] struggles with accurately localizing the robot in environments with repetitive patterns or symmetrical structures. In these environments, the robot may encounter difficulties distinguishing one location from another due to the repetitive nature of the surroundings. For instance, if a corridor looks identical from one end to the other, the robot's sensors may provide similar readings in multiple locations, leading to ambiguity about its exact position.
- *Featureless regions*: Regions with few or no distinctive features can lead to poor localization performance due to the lack of reference points.

Examples include empty hallways, large open spaces, or plain walls. In these regions, sensors like cameras or LiDAR may struggle to find enough data points or features to determine the robot's position accurately. Without adequate reference points, the robot may have trouble accurately understanding its location, leading to potential errors in navigation.

- *Sensor noise and inaccuracies*: Sensor data is often noisy and may lead to incorrect localization, especially in dynamic or cluttered environments.

For instance, a LiDAR sensor might pick up reflections from shiny surfaces or moving objects, leading to false readings. Similarly, cameras can be affected by varying lighting conditions, and inertial measurement units (IMUs) can drift over time.

- *Computational complexity*: The high computational demands of some localization methods limit their real-time applicability on resource-constrained robots.

Global localization algorithms often require complex computations to process sensor data, map the environment, and estimate the robot's position. The algorithms, such as particle filters, graph-based

https://doi.org/10.1017/S0263574725102518 Published online by Cambridge University Press

optimization, or deep learning models, are computationally intensive. Over high computational load can cause delays, reducing the robot's ability to respond effectively to its environment.

- *Adaptability*: Many algorithms are environment-specific and struggle to generalize across different indoor settings.

Researchers design existing localization algorithms with specific environments in mind. They rely on particular features, sensor configurations, or assumptions about the environment that do not generalize well to different settings. For example, an algorithm optimized for a well-lit office environment might perform poorly in a dimly lit warehouse.
Therefore, this review paper aims to address these gaps and challenges by

1. *Analyzing state-of-the-art methods in global localization*: Providing a comprehensive overview of current global localization techniques like adaptive Monte Carlo localization (AMCL), Hector SLAM, cartographer, and hybrid SLAM, highlighting their strengths and weaknesses in symmetrical environments. The importance of this analysis lies in understanding how these techniques perform under specific conditions of symmetry and what limitations they may have. By critically evaluating these methods, the review highlights which approaches are most effective and where improvements are needed.
2. *Addressing critical challenges in global localization imposed by symmetrical environment*: Discussing the challenges posed by symmetrical indoor environments, such as symmetrical features and a lack of distinct landmarks, which can complicate localization tasks. Understanding these critical challenges is crucial for improving the robustness and reliability of localization systems. Pinpointing where the current methods fall short and exploring possible solutions aims to provide insights into how to address these challenges effectively.
3. *Benchmarking performance*: Establishing a benchmark scenario by constructing a symmetrical environment to evaluate localization algorithms and presenting a comparative analysis based on these benchmark scenarios. Comparative analysis based on this benchmark scenario clearly shows how different methods perform in controlled conditions. The rationale behind benchmarking is to provide a standardized framework for assessing algorithm performance, which helps identify best practices and improvement areas.

This paper offers several significant contributions to global localization in symmetrical environments. First, it presents a comprehensive review and critical analysis of prominent localization techniques including AMCL, Hector SLAM, Cartographer, and hybrid SLAM highlighting their advantages and limitations when applied in highly symmetrical settings. Second, it systematically identifies the core challenges posed by symmetrical environments, such as the scarcity of distinctive landmarks and the presence of repetitive structural features, and provides informed insights into potential strategies for improving localization robustness under such conditions. Third, the paper introduces a benchmarking framework that involves the design of a controlled, standardized symmetrical environment, enabling consistent evaluation and comparison of localization algorithms. This framework facilitates the identification of best-performing methods and highlights areas requiring further improvement. Collectively, these contributions aim to deepen the understanding of localization in geometrically ambiguous spaces and support the development of more resilient and accurate systems.

While review papers [11–13] explore localization in robotics, they primarily emphasize theoretical frameworks and general algorithmic categories. They often overlook the practical and critical challenges posed by global localization in symmetrical environments, where repetitive spatial structures introduce pose ambiguity, aliasing, and belief collapse issues that can drastically degrade localization accuracy. Furthermore, existing surveys rarely systematically examine how different algorithms cope with environmental symmetry in real-world scenarios. They also lack benchmark-driven evaluations that explore

https://doi.org/10.1017/S0263574725102518 Published online by Cambridge University Press

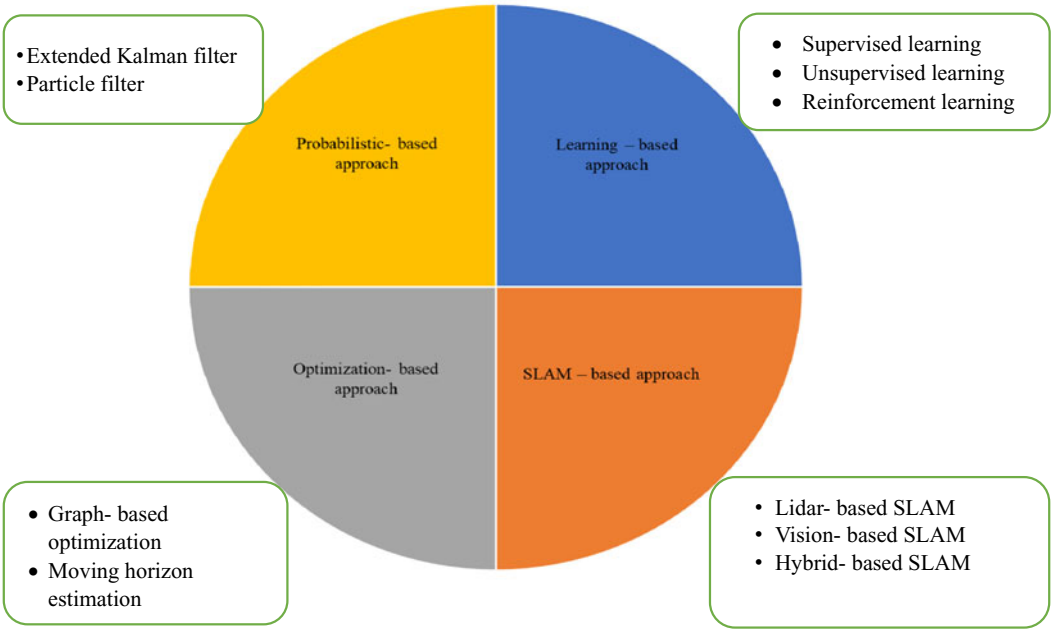


Figure 3. *Classification of indoor global localization approaches.*

the computational burden and robustness of these methods under controlled, variable-symmetry conditions. This paper addresses these limitations by narrowing its focus to the global localization problem under symmetry-induced uncertainty. It reviews state-of-the-art techniques including probabilistic-based, learning-based, SLAM-based, and optimization-based approaches. To ground the discussion, the paper introduces two experimental test cases (low and high symmetry) that help reveal practical performance boundaries. These gaps and limitations form the core motivation of this review to provide a practically guided and symmetry-aware perspective that advances the understanding and application of global localization techniques in complex indoor environments.

This review paper begins by categorizing the fundamental challenges of global localization for indoor mobile robots into probabilistic, learning-based, SLAM-based, and optimization-based approaches based on their underlying methodologies and techniques to estimate a robot's position within an environment. Figure 3 shows a general classification of such methods:

Section 2 presents a detailed examination of the four primary categories for global localization. The discussion thoroughly explores each category, highlighting the specific methodologies and their respective applications in estimating a robot's position within an environment. This comprehensive analysis aims to elucidate each approach's distinct characteristics and advantages, offering a nuanced understanding of their roles in advancing localization accuracy and efficiency.

Section 3 delves into the critical issue of global localization when encountering a symmetrical indoor space, a significant challenge in robotics due to these environments' inherent ambiguities. It also evaluates existing solutions, including integrating visual features with traditional LiDAR-based methods and combining vision sensors with probabilistic localization techniques, to enhance localization accuracy in these environments.

In Section 4, this paper presents a comparative analysis of four algorithms (AMCL, Hector SLAM, hybrid SLAM, and Cartographer) operating in low- and high-symmetry scenarios. It provides corresponding experimental findings and insights. Section 5 presents a comprehensive discussion of our findings, interpreting the results within the context of existing literature and analyzing their implications for the field. Section 6 gives conclusions, emphasizing current challenges and outlining potential directions for future research endeavors.

https://doi.org/10.1017/S0263574725102518 Published online by Cambridge University Press

2. Review of global indoor localization of mobile robots

2.1. Localization principle

The mobile robot utilizes odometry to track its movement while traversing in a familiar environment. However, the inherent uncertainty in odometry measurements can lead to confusion regarding the robot's precise position. The robot employs localization techniques to mitigate this uncertainty and accurately understand its location relative to the environment. The robot collects observations about its surroundings by incorporating information from its external sensors, including laser, vision, and ultrasonic sensors. These sensory inputs offer valuable data for a more robust understanding of the environment and the robot's position. The robot enhances its localization capabilities by integrating these sensory observations with the data obtained from odometry. This fusion of information allows the robot to refine its estimate of its current position, effectively reducing the potential for position uncertainty to escalate uncontrollably.

In summary, by leveraging odometry data and information gathered from its exteroceptive sensors, the mobile robot can localize itself within its environment, thereby mitigating the effects of odometry uncertainty and maintaining a reliable understanding of its position.

2.2. Localization approaches

Global localization for mobile robots is classified into four approaches based on underlying methods and techniques used to determine the robot's location: *probabilistic-based, learning-based, SLAM-based, and optimization-based approaches.*

2.2.1. Probabilistic-based approaches

Probabilistic approaches in robotics employ probability theory to manage uncertainties inherent in sensor data and robot motion. Two prominent methods within this framework are Kalman filters and particle filters.

Kalman filters: These are optimal estimators for linear systems with Gaussian noise. They maintain a belief about the robot's state as a Gaussian distribution, updating this belief through a prediction–correction cycle.

Particle filters (Monte Carlo Localization – MCL): These are nonparametric approaches suitable for nonlinear and non-Gaussian systems. They represent the robot's belief as a set of weighted samples (particles), each hypothesizing a possible state.

Kalman filter. Kalman filter [11] is widely used for localizing mobile robots because it estimates the state of a dynamic system from a series of noisy measurements. It is one of the most commonly used methods for sensor fusion in mobile robotics applications [12]. This filter frequently combines data from sensors to create a statistically optimal estimate. If a system describes a linear model and models the uncertainties of the sensors and the system as white Gaussian noise, the Kalman filter provides an optimal statistical estimate for the observed data.

Figure 4 represents the schematic diagram of applying the Kalman filter to localize the mobile robot. The algorithm works by a two-phase process with prediction and update phases. (i) In the prediction step, the Kalman filter predicts the current state of the robot based on the mathematical model of its motion (e.g., how it should move given the control input like velocity or steering angle) and the previous state estimate. The filter also predicts the uncertainty associated with its state estimate, which grows over time due to uncertainties in the model and the control inputs. (ii) In the update step, the Kalman filter updates its state estimate once it gets new sensor data. It compares the predicted state to the actual sensor measurements and uses this "innovation" to correct the state estimate. The filter then reduces the uncertainty in the state estimate based on how reliable the sensor data is. Thus, Kalman filter-based localization efficiently and accurately addresses the challenge of tracking a robot's position. This optimal sensor fusion method is particularly effective when monitoring the robot's movement from an initially known location. Typically employed in continuous world representation, this localization method encounters limitations when the robot's uncertainty becomes substantial, such as when the

https://doi.org/10.1017/S0263574725102518 Published online by Cambridge University Press

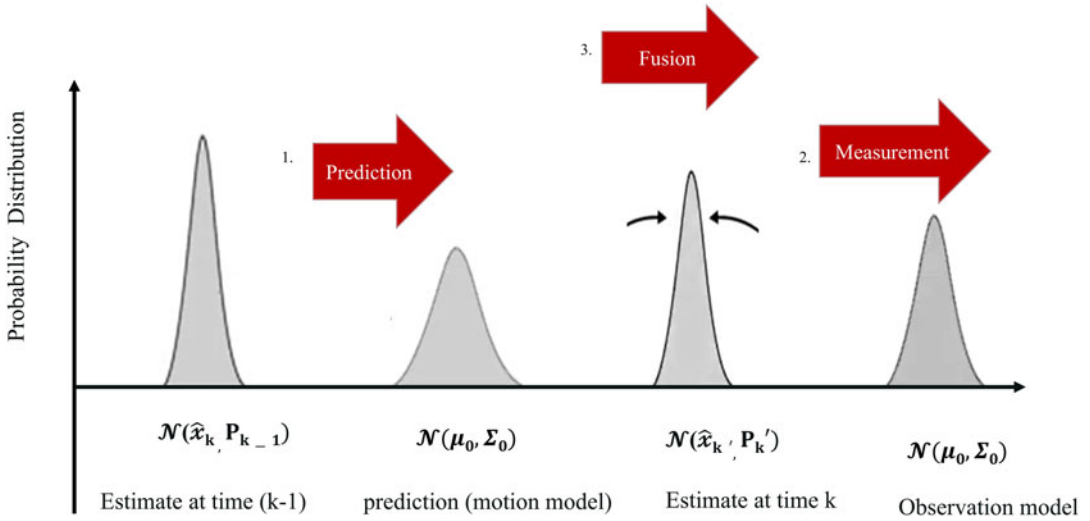


Figure 4. *Schematic diagram of Kalman filter-based mobile robot localization.*

robot's initial pose is unknown. In such cases, the localization approach fails to capture potential robot positions accurately and irreversibly loses track.

In the realm of Kalman filter theory, the assumption is that the system is linear and characterized by Gaussian noise. Nevertheless, numerous robotic systems exhibit nonlinear characteristics. This necessity has led to the development of an alternative localization method called the EKF. The EKF is an extension of the Kalman filter tailored for nonlinear systems. In the EKF, the initial step involves linearizing the system and applying the Kalman filter. In [13], the localization of a mobile robot using EKF was presented, which used a nonlinear model to predict the coordinates of a wheeled mobile robot. To improve the localization performance and to reduce accumulated errors while using dead reckoning, a method that uses EKF with an infrared sensor was proposed [14]. Studies reported in ref. [15] and [16] employed various sensors like sonar, IMU, and encoders along with EKF for the localization of a mobile robot. Despite the performance improvements in these methods, EKF faced limitations when applied to mobile robot localization, particularly due to its reliance on linearizing nonlinear systems [17], which led to significant errors. EKF's performance was affected by inaccurate covariance propagation and the complexity of computing Jacobians, which were often error-prone [18]. Additionally, it assumes Gaussian noise, which may not hold in real-world scenarios.

In ref. [19], the authors addressed the above issues with EKF and introduced the Unscented Kalman Filter (UKF), which used the unscented transform to propagate sigma points through the nonlinear system without linearization. UKF resulted in higher accuracy, better handling of nonlinearities, and robustness to non-Gaussian noise, making UKF more suitable for complex mobile robot localization tasks. UKF overcomes EKF linearization errors using a deterministic sampling approach, representing the state distribution as a set of sample points, and achieves better accuracy [20]. The study based on real data shows that the performances of the EKF and UKF are nearly identical to those in ref. [21]. In the Kalman filter approach, the system requires knowledge of the robot's initial position with some approximations. As a result, it does not solve the global localization problem. Also, Kalman filters represent uncertainty as a single Gaussian distribution, which is limited when the robot's belief about its position could have multiple possible solutions (e.g., in environments with symmetry or ambiguity).

Particle filter. EKF estimation assumes that noise from sensors and the environment follows a Gaussian probability density function. In practice, this assumption may not always hold. Additionally, some sensors have strongly nonlinear measurement models, which leads to increased estimation errors when linearizing the nonlinear equations. When these assumptions are not suitable, the particle filter is a more

https://doi.org/10.1017/S0263574725102518 Published online by Cambridge University Press

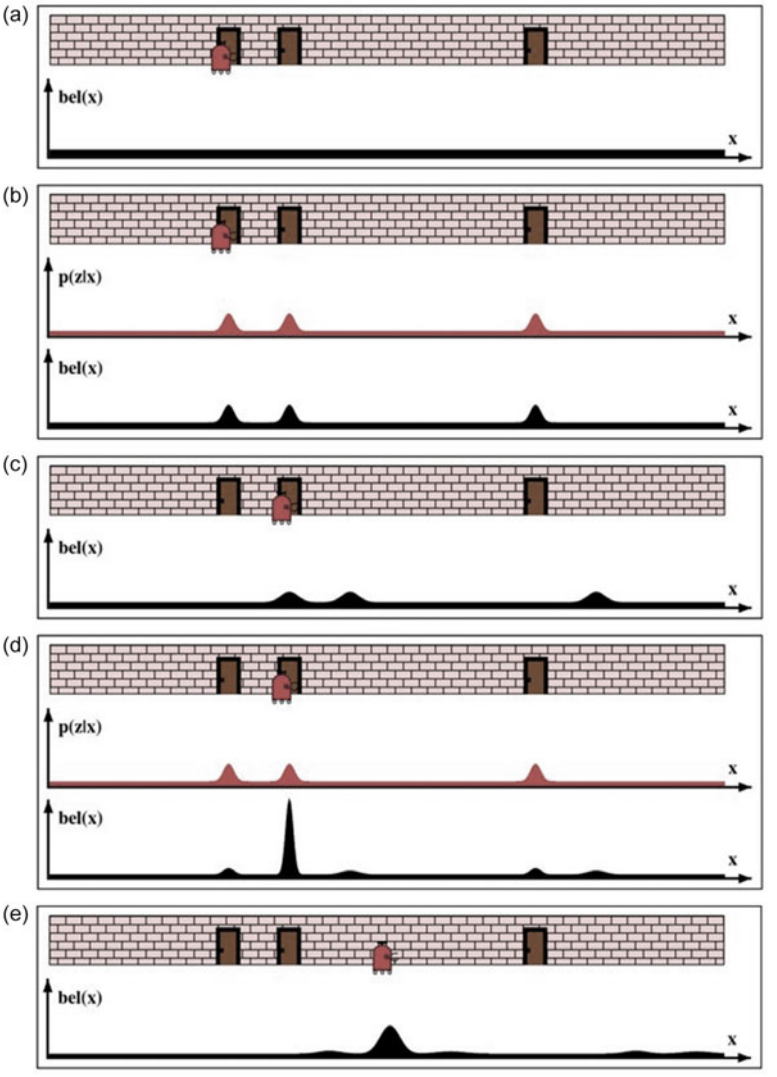


Figure 5. *Monte Carlo localization for mobile robot localization [22].*

appropriate algorithm for global localization of mobile robots. It processes raw sensor data directly, eliminating the need to extract features from sensor values. Unlike the EKF, which is bound to a unimodal distribution, particle filters are nonparametric and do not rely on a Gaussian density assumption. The particle filter algorithm also uses increasingly sophisticated methods to represent uncertainty, surpassing the limitations of the standard Kalman filter. We now turn our attention to a popular localization algorithm that describes the belief $bel(x_t)$ by particles. The algorithm is called MCL. The MCL has emerged as one of the most widely adopted localization algorithms in robotics due to its popularity and effectiveness. Its straightforward implementation and versatility make it suitable for addressing various robot localization challenges with generally favorable outcomes.

Figure 5(a–e) illustrates MCL using the one-dimensional hallway example [22]. Initially, the robot does not know exactly where it is. Therefore, it randomly guesses by placing several possible positions (represented as particles) all over the map. As the robot senses its surroundings, it can refine its position estimate by continuously updating and resampling particles based on sensor data and movement. As the robot gathers more information, it increasingly focuses on the most likely locations, improving its

https://doi.org/10.1017/S0263574725102518 Published online by Cambridge University Press

Table I. *Monte Carlo localization (MCL) algorithm based on particle filters.*

1. Algorithm MCL $(\mathcal{X}_{t-1}, u_t, z_t, m)$:
2. $\overline{\mathcal{X}}_t = \mathcal{X}_t = \varnothing$
3. for $m = 1$ to M do
4. $x_t^{[m]} =$ sample_motion_model $\left(u_t, x_{t-1}^{[m]}\right)$
5. $w_t^{[m]} =$ measurement_model $(z_t, x_t^{[m]}, m)$
6. $\overline{\mathcal{X}}_t = \mathcal{X}_t + \langle x_t^{[m]}, w_t^{[m]} \rangle$
7. endfor
8. $m = 1$ to M do
9. draw i with probability $\propto w_t^{[i]}$
10. add $x_t^{[i]}$ to $\mathcal{X}_t$
11. endfor
12. return $\mathcal{X}_t$

Table II. *The particle filter algorithm.*

1. Algorithm Particle_filter $(\mathcal{X}_{t-1}, u_t, z_t)$:
2. $\overline{\mathcal{X}}_t = \mathcal{X}_t = \varnothing$
3. for $m = 1$ to M do
4. sample $x_t^{[m]} \sim p(x_t \mid u_t, x_{t-1}^{[m]})$
5. $w_t^{[m]} = p(z_t \mid x_t^{[m]})$
6. $\overline{\mathcal{X}}_t = \overline{\mathcal{X}}_t + \langle x_t^{[m]}, w_t^{[m]} \rangle$
7. endfor
8. for $m = 1$ to M do
9. draw i with probability $\propto w_t^{[i]}$
10. add $x_t^{[i]}$ to $\mathcal{X}_t$
11. Endfor
12. return $\mathcal{X}_t$

accuracy of localization. This process ensures that the robot's position estimate converges towards the proper location over time.

Table I shows the MCL algorithm based on particle filter, obtained by substituting the appropriate probabilistic motion and perceptual models into the particle filter algorithm shown in Table II. The basic MCL algorithm represents the belief bel(x_t) by a set of M particles:

$$\mathcal{X}_t = \left\{x_t^{[1]}, x_t^{[2]}, \ldots, x_t^{[M]}\right\} \tag{1}$$

Line 4 in the algorithm (Table I) samples from the motion model, using particles from the present belief as starting points. The measurement model is then applied in line 5 to determine the important weight of that particle. The initial belief bel(x_0) is obtained by randomly generating M such particles from the prior distribution $p(x_0)$ and assigning the uniform importance factor M^{-1} to each particle. The function's motion model predicts the next state of particles based on robot dynamics, while the measurement model updates the particle weights by predicted sensor observation with actual sensor data.

In early works, the MCL algorithm relies on particle filter and uses a large number of weighted particles from the proposed distribution of robot pose and then updates them based on the measurement of sensors. The MCL algorithm always converges to the global optimum, but in large-scale scenes, it may take a lot of particles and a long time to converge [23]. To make sampling more efficient, many researchers have improved the MCL algorithm. For instance, the mixture algorithm combines the MCL algorithm with the dual MCL, using the probabilistic mixing ratio. Sometimes, it reduces the error

https://doi.org/10.1017/S0263574725102518 Published online by Cambridge University Press

by more than an order of magnitude compared to the original MCL. Self-adaptive Monte Carlo localization (SAMCL) distributes global samples in a similar energy region instead of distributing them randomly on the map, which results in better localization performance [24]. However, they depend on the specific environment, which makes them unreliable, and they are still unable to solve the slow convergence problem in large-scale environments. The MCL method is still the gold standard for robot localization and often uses LiDARs [25–27], cameras [28], or Wi-Fi [29]. Such MCL methods often use a 2D LiDAR sensor and an occupancy grid map to estimate the robot's pose, and they are quite robust.

2.2.2. Learning-based approaches

This paper section focuses on learning-based methods for achieving global localization for indoor mobile robots. These methods leverage machine learning algorithms to interpret sensor data and predict the robot's location. The system can recognize patterns and improve localization accuracy over time by training on large datasets. Learning-based approaches for global localization can be classified into three main categories: supervised learning, unsupervised learning, and reinforcement learning (RL) based on the learning process involved.

Supervised learning approaches involve training a model on a labeled dataset, where the input data (e.g., LiDAR scans) are paired with the corresponding output labels (e.g., specific locations or poses). The model learns to map inputs to outputs based on the provided labels.

Unsupervised learning approaches do not rely on labeled data. Instead, they aim to find patterns or structures in the input data without explicit output labels.

RL involves training an agent to make a sequence of decisions by rewarding it for good actions and penalizing it for bad ones. In the context of localization, RL can be used to optimize the robot's navigation and localization strategies based on its interactions with the environment.

Researchers have recently achieved global localization of mobile robots by incorporating environmental information about the robot's location, which can be accomplished through semantic place classification. It involves assigning semantic labels or categories to different locations or areas within an environment based on their characteristics or attributes. This classification provides contextual information about the environment, which can be valuable for place recognition. By associating semantic labels with specific places, the robot can better understand the environment and distinguish between locations based on their semantic meaning rather than just raw sensor data. This approach enhances the reliability and accuracy of place recognition, particularly in complex indoor environments where traditional methods may struggle due to similarities in sensor data. Presented in ref. [30], the semantic model of the indoor environment provides the robot with a representation closer to human perception, thereby enhancing the navigation task. Also, incorporating semantic information into the map improves the accuracy of robot navigation, as in ref. [31]. The semantic model of the indoor environment provides the robot with a representation closer to human perception, thereby enhancing the navigation task.

"Semantic place labelling" or "semantic place classification" can be performed by making use of data acquired from its sensors [32]. Classification methods based on visual data captured with a 2D camera [33] or fused data [34], including the visual and range data acquired from a 2D laser range finder, have been studied previously. In these studies, researchers generally extracted features such as Scale-Invariant Feature Transform (SIFT) [35] and Speeded Up Robust Features (SURF) [36] from images or calculated hand-crafted features from laser range data. They then processed these features to classify a robot location. The main drawback of the studies using visual data is the performance dependency on illumination variations corresponding to different lighting conditions.

The paper categorizes the related work on semantic place classification into supervised and unsupervised learning methods based on how the semantic information is learned. This semantic information might sometimes be uncertain, especially in environments with symmetrical or repetitive structures. In such cases, RL-based algorithms can incorporate uncertainty into their decision-making processes, allowing robots to weigh different possibilities and choose actions that reduce ambiguity. Also, RL

https://doi.org/10.1017/S0263574725102518 Published online by Cambridge University Press

can learn robust policies for localization in the presence of sensor error, noise, or unexpected environmental changes. Based on this, learning-based approaches are classified into supervised, unsupervised, and RL.

Supervised learning. In recent years, the application of supervised machine learning techniques has significantly advanced the field of robotic place recognition and semantic mapping to improve global localization. Researchers have developed various models and algorithms to enhance a robot's ability to understand the environment and navigate complex environments.

For instance, a bubble space model presented in ref. [37] for place representation contained two new variables: varying robot pose and multiple features. These variables were transformed into bubble descriptors and classified by a multi-class support vector machine (SVM) classifier to learn specific places. A Voronoi Random Field (VRF) algorithm [38] was proposed, based on Conditional Random Field, and an SVM classifier was used to achieve semantic labeling of Generalized Voronoi Graphs nodes. A low-level image descriptor for addressing the room classification problem was proposed by refs. [39] and [40] using an SVM classifier. A semantic map construction method, which combined place and object recognition using supervised learning, was presented in ref. [41]. High-dimensional histograms were used to represent an observed scene and achieved the classification of visual places through SVM [42]. Performance was evaluated using the Freiburg University, Buildings 79 and 52 dataset [43].

Unsupervised learning. In global localization, supervised learning approaches rely on labeled datasets, requiring human intervention to annotate data for training. In contrast, unsupervised learning methods offer several advantages. They eliminate the dependency on labeled data, allowing models to discover new patterns and relationships in raw, unlabeled data. Unsupervised models can autonomously extract features and find patterns in the data, enabling the robot to learn, classify the places, and adapt to new environments without explicit human intervention. Additionally, these methods can incrementally learn and organize environmental data, allowing robots to update their understanding of surroundings in real time.

Recent advancements in semantic place classification using unsupervised learning have significantly improved global localization by leveraging various sensor modalities and machine learning techniques. Traditional approaches utilized visual cues from video and image streams to detect and categorize "unknown" place labels [44]. Additionally, a Gaussian mixture model was used to form a probability distribution of features extracted by image-preprocessing, while a place detector grouped features into places using odometry data and a hidden Markov model [45]. The unsupervised method proposed in ref. [46], based on the Single-Linkage (SLINK) agglomerative algorithm [47] and an incremental approach, allows the robot to learn about organizing the observed environment and localizing within it. Finally, the authors in ref. [48] used CENsus TRansform hISTogram (CENTRIST) and GIST descriptors, applying an unsupervised approach with a Self-Organizing Map for classification.

Incorporating semantic place recognition to enhance global localization using unsupervised learning was applied to various sensors, including 2D and 3D LiDARs, red, green, blue (RGB), and red, green, blue-depth (RGB-D) cameras. In ref. [49], AdaBoost features from 2D LiDAR scans were used to infer semantic labels such as office, corridor, and kitchen. They combined the semantic information with an occupancy grid map in an MCL framework. This method required a detailed map and a manually assigned semantic label to every grid cell. According to ref. [50], researchers constructed semantic maps from cameras by assigning a place category to each occupancy grid cell. They used the Places205 ConvNet to recognize places and relied on a LiDAR-based SLAM for building the occupancy grid map. In ref. [51], available building information models (BIMs) were used to extract and localize geometric and semantic information by matching 2D LiDAR-based features corresponding to walls, corners, and columns. While the automatic extraction of semantic and geometric maps from a BIM was promising, it was not suitable for global localization due to its inability to overcome the challenges of a repetitively structured environment.

Recent works in extracting semantic information with deep learning models showed significant improvement in the performance of mobile robot localization using text spotting. Using textual cues

https://doi.org/10.1017/S0263574725102518 Published online by Cambridge University Press

for localization was surprisingly uncommon, with notable works presented in refs. [52] and [53]. The study in ref. [52] presented a novel underground parking localization method using text recognition, employing optical character recognition (OCR) to detect parking slot numbers and combining OCR with a particle filter for localization. In ref. [53], an indoor localization system for dynamic environments was presented, using text spotting integrated into an MCL framework, which operated on 2D LiDAR scans and camera data.

Reinforcement learning. Building upon exploring supervised and unsupervised learning approaches in global localization, RL emerges as a pivotal technique in enabling mobile robots to navigate and understand complex environments autonomously. Unlike traditional methods, RL allows an agent to learn optimal behaviors through trial-and-error interactions within its environment, receiving feedback as rewards or penalties [54]. The key idea of RL is to determine the mapping between actions and states to maximize a numerical reward signal. The learner agent interacts with the environment and receives rewards or penalties for the performed actions. Therefore, RL uses rewards and punishments as signals for positive and negative behaviors to optimize agent behavior continuously.

Recent studies have demonstrated the efficacy of RL in addressing challenges associated with continuous state and action spaces noise in reward functions, as well as the integration of particle filters for improved state estimation. In ref. [47], a novel framework that utilizes a particle filter for learning in real-world situations was presented. This approach dealt with the challenge of expressing continuous states and actions by dividing the continuous state and action space, allowing a more efficient search for solutions with fewer memories and parameters. The study in ref. [55] addressed deep RL agent performance degradation due to noise in reward functions and proposed a novel filter to mitigate its effect. The proposed algorithm combined optimal filtering and RL concepts to improve the RL policy in the presence of skewed probabilistic distributions in reward functions. The method in [56] combined the ideas of particle filters, which are used for state estimation in nonlinear and non-Gaussian systems, with RL. This approach assigned randomness in the system and measurement models to the RL policy, allowing for optimization of various objective functions, such as estimation accuracy, particle variance, likelihood of parameters, and regularization term. This interpretation enabled the particle filter design process to benefit from the RL's learning capabilities, improving the particle filter's overall performance and accuracy in estimating a system's state. These advancements underscore the potential of RL to significantly enhance the performance and robustness of global localization systems for indoor mobile robots.

Recently, deep learning models showed notable advancements in global localization. PoseNet [57], based on GoogLeNet, was the first work to utilize deep learning for this purpose, demonstrating the effectiveness of convolutional neural networks (CNNs) for complex regression tasks. Bayesian PoseNet [58] extended this approach by leveraging uncertainty measures for estimating re-localization error and scene detection. LSTMNet [59] uses LSTM units to reduce the dimensionality of CNN outputs, while Vidloc [60] employs video clips as input to improve pose estimation accuracy. HourglassNet [61] introduced an encoder–decoder CNN architecture with up-convolution layers to preserve fine-grained information. Branch Net [62] was a multi-task CNN addressing the coupling of orientation and translation. PoseNet2 [63] introduced a novel loss function for optimal learning of position and orientation, although its localization precision might decrease with scene scale.

A new approach recently emerged where deep learning was employed for robot localization. In refs. [64] and [65], deep CNNs were utilized for camera pose regression and initializing particles in MCL. In refs. [66], a collection of 2D laser data was converted into an RGB image and an occupancy grid map, which were combined into a multi-channel image. A neural network architecture was employed to predict the 3-DOF robot pose, with experiments conducted on an actual mobile robot to validate the efficacy of this approach. The methodology in ref. [67] uses a deep hierarchical framework for robot global localization, involving two main components: coarse localization and fine localization. Coarse localization used RGB images and a CNN-LSTM network, while fine localization combines fast branch-and-bound scan matching (FBBS) and MCL using LiDAR point cloud data. The framework achieved faster convergence and more accurate estimation compared to other methods. Deep learning-based object detection

https://doi.org/10.1017/S0263574725102518 Published online by Cambridge University Press

has gained the attention of researchers for robot semantic navigation. In ref. [68], a real-time semantic map production system for indoor robot navigation was proposed, utilizing LiDAR and vision-based systems, achieving 78.86% average map construction accuracy.

2.2.3. SLAM-based approaches

Over the past decades, significant progress has been made in advancing autonomous navigation for mobile robots. The development of SLAM technology is crucial in enhancing robots' ability to localize themselves and construct detailed maps of their surroundings without prior knowledge of the environment. SLAM technology is the foundation for autonomous decision-making, planning, and control of robots, enabling them to explore unknown environments effectively.

This paper categorizes SLAM approaches based on the sensors employed during the localization process, focusing on three primary categories: LiDAR-SLAM, visual SLAM, and Hybrid SLAM.

LiDAR-SLAM: This approach utilizes LiDAR sensors to generate precise representations of the environment. It is particularly effective in scenarios where accurate distance measurements are crucial, such as autonomous driving and robotics operating in complex terrains.

Visual SLAM: Leveraging cameras, visual SLAM systems capture images to interpret and map surroundings. These systems are advantageous due to their cost-effectiveness and the rich contextual information they provide. However, its performance can be affected by lighting conditions and textured environments.

Hybrid SLAM: To harness the strengths of multiple sensors, Hybrid SLAM integrates data from various sources, such as LiDAR, cameras, and IMUs. This fusion enhances robustness and accuracy, compensating for the limitations inherent in individual sensor modalities. For instance, combining LiDAR's precise distance measurements with the rich visual context from cameras can improve performance in diverse environmental conditions.

The subsequent sections provide an in-depth review of studies within each of these categories, analyzing their methodologies, applications, and contributions to the advancement of SLAM technologies.

LIDAR-based SLAM. LiDAR-SLAM uses LiDAR sensor data as the primary source of information, enabling it to achieve localization and mapping under diverse environmental conditions. Object information from LiDAR presents a series of scattered points with accurate angular and distance information, known as a point cloud. Generally, LiDAR-SLAM extracts and matches the features of two consecutive point cloud graphs taken at different times, calculates the relative motion, distance, and pose changes through LiDAR odometry, and completes the robot's self-positioning.

The significant advantage of the LiDAR sensor is its relatively accurate range measurement, simple error model, stable operation in environments without direct glare from intense light, and relatively easy point cloud processing. LiDAR-SLAM is categorized into two types based on the LiDAR sensor: 2D LiDAR-SLAM and 3D LiDAR-SLAM. 2D LiDAR-SLAM is lower in cost and suitable for simple indoor environments. At the same time, 3D LiDAR-SLAM, though more expensive, is ideal for outdoor surveying and mapping and handles larger amounts of information.

A complete 2D LiDAR-SLAM involves a front-end incremental estimation part and a back-end mapping part. The front-end consists of two components: obtaining sensor data and scan matching. The back-end's main task is to solve a nonlinear optimization problem. The first mainstream approach for the back-end is based on filtering, like the Bayesian estimation process. A representative algorithm was Gmapping [69], which only estimated the robot's position at the current moment, resulting in low computational requirements. However, this method had the disadvantage that errors generated at a previous moment could not be corrected, potentially leading to suboptimal mapping, especially in large environments. The second method was primarily based on nonlinear optimization, with graph optimization widely used in LiDAR-SLAM. In graph-based SLAM, the robot's pose was represented as a node or vertex, and the relationships between poses formed the edges. Errors accumulated during map building were optimized using the nonlinear least squares method to produce the final map. A typical algorithm employing this method was Cartographer [70]. Fast SLAM [71], introduced in 2002, was the first system

https://doi.org/10.1017/S0263574725102518 Published online by Cambridge University Press

capable of generating real-time, faster maps, but it suffered from high memory usage and significant particle dissipation issues. The Optimal RBPF [72] method was proposed to address these shortcomings, which optimized the Gmapping algorithm to mitigate particle degradation problems. Karto-SLAM [72], an early system, introduced the concept of sparsity in graph-based optimization, laying the groundwork for Cartographer. Cartographer improved accuracy and robustness by leveraging graph optimization techniques while reducing computational requirements, displacing several early filter-based methods.

Over the past several decades, 2D SLAM has been well developed. Still, certain challenging environments, such as fog, smoke, and dust, significantly impact 2D LiDAR SLAM as they cause the laser beams to scatter and reduce the accuracy of scans. With the introduction of Lidar Odometry and Mapping (LOAM), 3D SLAM addressed the influence of many environmental factors. 3D SLAM typically uses 3D LiDAR and odometer data as input data, and 3D point cloud maps, robot trajectory, or pose graph as output. Section 2.2.3(C) will summarize work related to the hybrid SLAM based on 3D LiDAR.

Vision-based SLAM. VSLAM [73] is a technique that enables a robot to build a map of its environment while simultaneously keeping track of its location within that environment, using visual information from one or more cameras. Unlike LiDAR-based SLAM, which relies on laser sensors, VSLAM primarily uses cameras to capture images of the surroundings. VSLAM can be divided into feature-based and direct-based methods, each differing in processing and utilizing visual data to estimate the camera's pose and build a map of the environment.

Feature-based VSLAM methods: These methods detect and extract distinct features, such as corners, edges, or blobs, from images. Once extracted features are matched across consecutive frames to estimate the camera's motion and incrementally build a map of the environment. Feature-based methods are known for their robustness in environments with rich textures and are less sensitive to changes in lighting conditions. However, they may struggle in scenarios with low-texture regions or repetitive patterns, where distinct features are scarce or ambiguous.

Direct-based VSLAM methods: In contrast, direct methods utilize the intensity values of all pixels in the image without prior feature extraction. They aim to minimize the photometric error between consecutive frames by directly aligning images based on pixel intensities. This approach allows for using all available image information, making it effective in environments with low texture where feature-based methods might fail.

Feature-based VSLAM methods

In the realm of VSLAM, algorithms that rely on feature detection and matching have been instrumental in enabling robots to navigate and map their environments effectively. Early stages of feature-based VSLAM, feature extraction primarily relied on corner detection methods such as Harris [74], Features from Accelerated Segment Test (FAST) [75], and Good Features to Track (GFTT) [76]. However, their reliability is often insufficient. Consequently, researchers now prefer more stable local image features. Modern VSLAM methods based on point features typically employ feature detection algorithms like SIFT [35], SURF [77], and Oriented FAST and Rotated BRIEF (ORB) [78] to extract key points for matching images and then determined the pose by minimizing reprojection error.

MonoSLAM [79], introduced in 2007, was recognized as the inaugural real-time monocular VSLAM algorithm. It was the first successful application of the SLAM methodology from mobile robotics to the 'pure vision' domain of a single uncontrolled camera, achieving real-time but drift-free performance. The key novel contributions included an active approach to mapping and measurement, using a general motion model for smooth camera movement, and solutions for monocular feature initialization and feature orientation estimation. In the same year, Parallel Tracking and Mapping (PTAM) [80] improved over MonoSLAM, addressing its instability over extended periods and in diverse environments. It was the first SLAM algorithm to employ nonlinear optimization in the back end, tackling the issue of rapid data growth in filter-based methods.

Additionally, PTAM introduced the separation of tracking and mapping into distinct threads. The front end utilized FAST corner detection to extract features and estimate camera motion, while the

https://doi.org/10.1017/S0263574725102518 Published online by Cambridge University Press

back end handled nonlinear optimization and mapping. This approach ensured real-time performance in camera pose calculation and maintained overall SLAM accuracy. However, due to the absence of a loopback detection module, PTAM could accumulate errors during prolonged operation.

In 2015, ORB SLAM [81] was proposed and considered a notable advancement over PTAM, as it integrated a loop closure detection module, effectively mitigating cumulative errors. In real time, this monocular visual SLAM system used ORB features to perform SLAM. The loop closure detection thread also used the word bag model named Directed Bag of Words (DBoW) [82] for loop closure. The loop closure method based on the BoW model could detect the loop closure quickly by detecting the image similarity and achieve good results in the processing speed and the accuracy of map construction. In subsequent years, ORB-SLAM2 [83] and ORB-SLAM3 [84] were introduced and became widely utilized visual SLAM solutions due to their real-time CPU performance and robustness. However, the ORB-SLAM series heavily depended on environmental features, which could pose challenges in environments lacking sufficient texture features to extract enough feature points.

The reliance of point feature-based SLAM systems on the quality and quantity of point features presents challenges, particularly in scenes with weak textures like corridors, windows, and white walls, affecting system robustness and accuracy, potentially resulting in tracking failures. Additionally, rapid camera movement, changes in illumination, and other factors can significantly degrade the matching quality and quantity of point features. Growing interest was in integrating line features into SLAM systems to enhance feature-based SLAM algorithms. Two endpoints, represented by lines and line features, were used in the SLAM system to detect and track the two endpoints of lines in small scenes. The system could use line features alone or combined with point-line features, which was groundbreaking in VSLAM research. In ref. [85], a line-based monocular SLAM algorithm is proposed, which uses lines as basic features, and the UKF as a tracking algorithm. Due to the absence of loop closure and the representation of line segments as infinite instead of finite, the practical application of the scheme becomes challenging. Both [86], which used a monocular camera, and [87], which used a stereo camera, adopted the line segment detection (LSD) algorithm to detect line features and then combined the point-line features in SLAM. This approach performed well, even when most point features disappeared. Furthermore, it improved the accuracy, robustness, and stability of the SLAM system, though its real-time performance was lacking.

Direct-based VSLAM methods

In the field of VSLAM, direct methods have emerged as a significant advancement by utilizing pixel intensity information directly, thereby preserving all image details and eliminating the need for feature extraction and matching. This approach enhances computational efficiency and demonstrates strong adaptability to environments with intricate textures.

The Dense Tracking and Mapping (DTAM) algorithm [88] was proposed as the first practical direct method of VSLAM. DTAM allowed tracking by comparing the input images with those generated by reconstructed maps, enabling precise and detailed reconstruction of the environment. However, it increased the computational cost of storing and processing the data, making it feasible to run in real time on a GPU. Large-Scale Direct (LSD)-SLAM [89] introduced a direct monocular SLAM algorithm that enabled large-scale, consistent maps construction with accurate pose estimation and real-time 3D environment reconstruction. The algorithm included a novel direct tracking method that detected scale-drift and a probabilistic solution to incorporate noisy depth values into tracking, enabling real-time operation on a CPU. In ref. [90], a semi-direct monocular visual odometry algorithm was introduced, offering precision, robustness, and faster performance compared to current state-of-the-art methods. Semi-Direct Visual Odometry (SVO) combines the advantages of the feature point and direct methods. The algorithm is divided into two main threads: motion estimation and mapping. Motion estimation was carried out by feature point matching, but mapping was performed through a direct method. The research work in ref. [91] proposed Direct Sparse Odometry (DSO), which effectively utilized image pixels, making it robust even in featureless regions and capable of achieving more accurate results than

https://doi.org/10.1017/S0263574725102518 Published online by Cambridge University Press

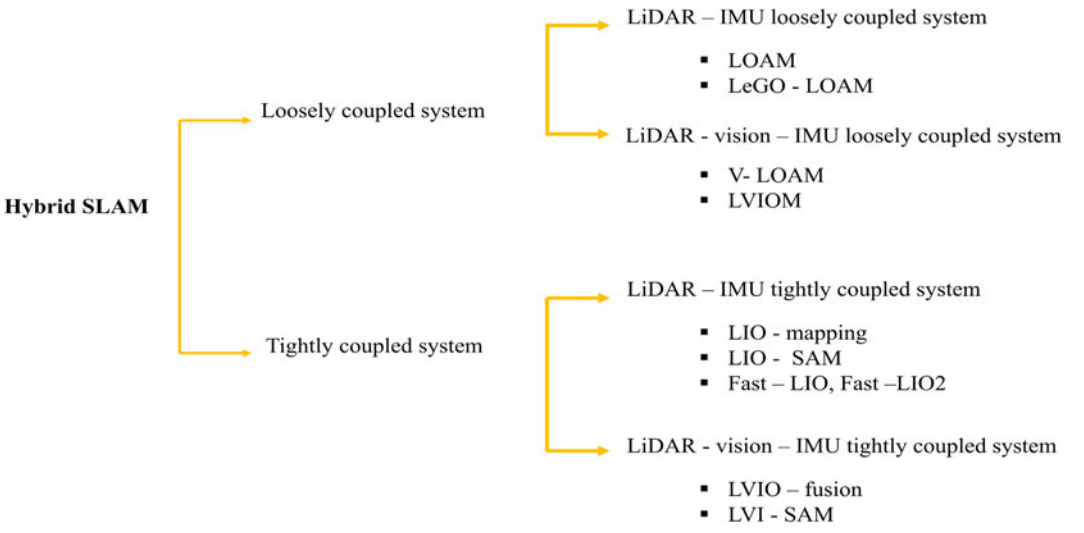


Figure 6. *Classification of hybrid SLAM based on LiDAR sensor.*

SVO. DSO calculated accurate camera orientation even when feature point detector performance was poor, improving the robustness in low-texture areas or with blurred images. DSO also employed geometric and photometric camera calibration for high-accuracy estimation. However, DSO only considered local geometric consistency, leading to inevitable cumulative errors, and it was not a complete SLAM system as it lacked loop closure and map reuse.

Both feature-based and direct-based VSLAM methods have their advantages and limitations. Feature-based methods are generally more robust against varying lighting conditions and can handle larger inter-frame motions due to their reliance on distinct features. However, they may require significant computational resources for feature extraction and matching. By leveraging raw pixel intensities, direct methods can perform well in low-texture environments and offer more accurate pose estimation in specific scenarios. Nevertheless, they can be sensitive to changes in lighting and may face challenges with large inter-frame motions.

Hybrid SLAM. The development of single-sensor SLAM systems is relatively mature, with LIDAR, cameras, and IMU being the standard sensors in SLAM systems. 3D LIDAR provides the system with rich structural information about the environment, but the data are often discrete and numerous. Cameras capture color and texture in the environment at high speed but cannot directly perceive depth, and they are often perturbed by light. IMU sensitively perceives weak changes in the system within a very short time, but long-term drift is inevitable. Each of these sensors has distinct characteristics with clear advantages and disadvantages.

Single-sensor SLAM systems are fragile and uncertain, and they cannot handle multiple complex environments, such as high-speed scenes, small spaces, and open and large scenes simultaneously. As a result, hybrid SLAM has become a new trend in SLAM system development. Hybrid SLAM systems use a combination of LIDAR, cameras, and IMUs, which can be categorized into loosely coupled or tightly coupled modes as shown in Figure 6.

Loosely coupled systems: In loosely coupled systems, each sensor's data is processed independently, and the results are later integrated to estimate the robot's state. This approach simplifies the system architecture and reduces computational load. However, it may not fully exploit the complementary strengths of each sensor, potentially limiting accuracy in complex environments.

Tightly coupled systems: Conversely, tightly coupled systems integrate data from all sensors simultaneously, considering their interdependencies during processing. This method allows for more accurate state

https://doi.org/10.1017/S0263574725102518 Published online by Cambridge University Press

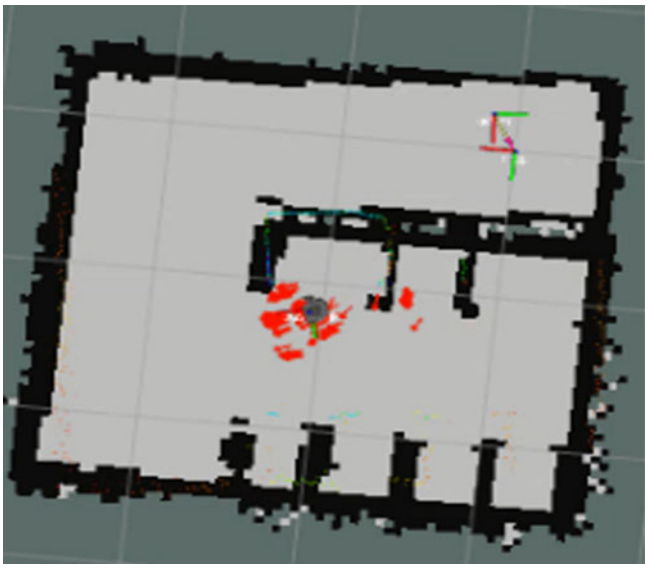


Figure 7. *An occupancy grid map which shows multiple hypotheses about the robot's pose due to symmetry in the environment and similar lidar scans.*

estimation, especially in challenging environments. However, it requires more computational resources and a more complex system design.

Recent advancements in hybrid SLAM have led to the development of systems combining LiDAR, cameras, and IMUs in loosely and tightly coupled configurations. In this paper, we categorize these works into LIDAR-IMU loosely coupled systems, Visual-LIDAR-IMU loosely coupled systems, LIDAR-IMU tightly coupled systems, and Visual-LIDAR-IMU tightly coupled systems, based on the coupling method of the system and the types of sensors being fused. The development of SLAM reflects a transition from loose coupling to tight coupling. First, we discuss Hybrid SLAM based on LIDAR, followed by visual SLAM.

Hybrid SLAM loosely coupled system

The emergence of loosely coupled systems has opened up a new stage in developing Hybrid SLAM systems. Until now, its application in low-cost platforms with limited computational power is still widespread.

- LiDAR-IMU loosely coupled system

The majority of loosely coupled systems were found in earlier works. The research work on the LOAM [92] algorithm was the main reason why LIDAR-based 3D SLAM technology became popular and well-known. The LOAM method enabled real-time 3D mapping by combining high-frequency odometry with precise point cloud registration. LOAM aided in global localization by utilizing odometry algorithms to estimate the velocity of the lidar and correct distortion in the point cloud. By matching and registering point clouds, LOAM created a map that helped determine the position of the lidar in a global coordinate system. This process ensured accurate mapping and localization in real time, contributing to the overall success of the global localization task. Compared to the state-of-the-art LOAM, Lightweight and Ground-Optimized Lidar Odometry and Mapping (LeGO-LOAM) [93] is designed to be lightweight, allowing it to run efficiently on low-power embedded systems. The design minimizes the computational resources required for processing LiDAR data and performing pose estimation. It leveraged the presence of a ground plane in its segmentation and optimization steps. This ground-optimized strategy helped improve pose estimation accuracy by utilizing planar features extracted from the ground. LeGO-LOAM

https://doi.org/10.1017/S0263574725102518 Published online by Cambridge University Press

aided global localization by integrating loop closure detection and correction mechanisms within its SLAM framework.

- LiDAR-vision-IMU loosely coupled system

LIDAR odometer degradation occurs in unstructured and repetitive environments. Even with the assistance of an IMU for positioning, it still cannot work correctly for long periods. In contrast, vision sensors that do not require specific structural features, such as edges and planes, but rely on sufficient texture and color information to accomplish localization. However, vision sensors cannot obtain in-depth information intuitively. Therefore, combining a camera with LIDAR provides a complementary solution. The authors of LOAM extended their algorithms by combining feature tracking of monocular cameras with IMU in V-LOAM [94]. Subsequently, the authors of V-LOAM released its iterative version, Lidar-Visual-Inertial Odometry and Mapping (LVIOM) [95]. The improved method presented a data processing pipeline to estimate ego-motion online and build a map of the traversed environment, leveraging data from a 3D laser scanner, a camera, and an IMU. Unlike traditional methods that used a sequential, multilayer processing pipeline, solving for motion from coarse to fine, the system used visual–inertial coupling methods to estimate motion and perform scan matching to refine motion estimation and mapping further. The resulting system enabled high-frequency, low-latency motion estimation and dense, accurate 3D map registration.

Hybrid SLAM tightly coupled system

- LiDAR-IMU tightly coupled system

The study in [96] proposed a tightly coupled LIDAR inertial localization and mapping (LIOM) framework, which jointly optimized measurements from LIDAR and IMU. The accuracy of LIOM is better than that of LOAM. However, the proposed method has a limitation as it requires initialization. Also, slow-motion scenarios could lead to drift due to the relatively sparse local map during their runtime. The study in ref. [86] released a follow-up work, lidar inertial odometry (LIO) via smoothing and mapping (LIO-SAM) [97], which utilized a factor graph for multi-sensor fusion and global optimization, and employed local sliding window scan matching for real-time performance enhancement. However, since feature extraction relied on geometric environments, this method still could not work for a long time in open scenes, and LIO degradation persisted during extended movements in such environments. FAST-LIO [98] proposed an efficient and robust LIO framework based on tightly coupled iterative Kalman filters for UAV systems to address this. The method fused more than 1200 feature points in real time and achieved reliable navigation results in a fast-motion, noisy, cluttered environment. To prevent the growing time of building a k-d tree of the map, the system could only work in small environments (e.g., hundreds of meters). FAST-LIO2 [99], built on FAST-LIO, addresses the growing computation issue by introducing a new data structure, ikd-Tree, which supports incremental map updates at every step and efficient kNN inquiries. Benefiting from the drastically decreased computation load, the odometry was performed by directly registering raw LiDAR points to the map, improving the accuracy and robustness of odometry and mapping, especially when a new scan contained no prominent features.

- LIDAR-Visual-IMU (LVI) tightly coupled system

The advanced framework of the LIO system has facilitated the development of the tightly integrated LVI system by leveraging the optimized LVI technology over the last couple of years. LiDAR and Visual Inertial Odometry Fusion (LVIO-Fusion) [100] was a robust SLAM framework that combined stereo camera, Lidar, IMU, and odometry data through graph-based optimization for accurate localization in urban traffic. The framework utilized a self-adaptive actor-critic method in RL to dynamically adjust sensor weights, enhancing system performance across various environments. Lidar-visual-inertial odometry via smoothing and mapping (LVI-SAM) [101] consisted of a Visual-Inertial System (VIS) and

https://doi.org/10.1017/S0263574725102518 Published online by Cambridge University Press

LIDAR-Inertial System (LIS). VIS can provide a pose prior for LIS, which provides pose estimation and accurate feature point depth for VIS initialization.

In addition, researchers developed a SLAM-based localization and navigation system for social robots, particularly the Pepper robot, to address the challenges of navigating dynamic indoor environments [102]. This system enhanced Pepper's ability to autonomously navigate and interact with humans using advanced algorithms and sensor integration. Furthermore, researchers applied SLAM using an improved wall-follower approach to an autonomous maze-solving robotic system [103]. They also advanced the SLAM technique by fusing it with visible light positioning to improve robot localization and navigation, as demonstrated in ref. [104].

In summary, researchers took a major step forward in hybrid SLAM by developing systems that integrate Lidar, vision, and IMU to work together smoothly, along with additional sensors like wheel/leg odometers and GNSS. Similarly, the increased computational complexity is one of the most challenging problems. In addition, some details need to be optimized, such as dynamic environments, unstructured environments, rain, and snow weather. Table III shows the strengths and weaknesses of each method of hybrid SLAM. The strengths and weaknesses of existing SLAM work are compared in Table III.

2.2.4. Optimization-based approaches

Regarding the final classification, optimization-based approaches formulate localization as an optimization problem, seeking to minimize the discrepancy between predicted and actual sensor measurements. It specifically focuses on pose graph optimization and moving horizon estimation, which optimize over a finite horizon. Optimization-based approaches are divided into post graph-based optimization and moving horizon-based pose estimation for robots.

In graph-based optimization, pose graph techniques use optimization algorithms to refine the robot's pose estimate by considering constraints and relationships between poses in a graph structure. Moving Horizon Estimation (MHE) is an advanced state estimation technique for localizing mobile robots. MHE solves an optimization problem over a finite time horizon, considering a limited set of past measurements and states, and optimizing within this window to estimate the current state. This approach helps smooth out noise and correct errors. Table IV discusses the advantages and disadvantages of both optimization approaches.

Pose graph-based optimization. Pose graph-based optimization is a method used in robotics to estimate the robot's trajectory and pose (position and orientation) within an environment based on sensor measurements and constraints. In this, the environment and the robot's trajectory are represented as a graph, where each node represents a pose of the robot at a specific point in time, and each edge represents a constraint between two poses. The graph is typically called a "pose graph." Optimization algorithms, such as Gauss–Newton or Levenberg–Marquardt, adjust the poses in the graph to satisfy all constraints while minimizing overall error. They find the poses that best explain the sensor measurements and constraints.

In ref. [105], a graph-based SLAM algorithm was proposed for indoor dynamic scenarios, consisting of local scan matching and global graph-based optimization. Local scan matching involved preprocessing the LIDAR data, detecting line features, and estimating the rotational and translational values. In contrast, global optimization constructed a topological graph using motion constraints and performed batch optimization to assess the global robot trajectory. A novel method of grid map segmentation for mobile robot global localization using a clustering algorithm and graph representation of clusters was presented in ref. [106]. The technique recovered from incorrect global localization without prior pose information, addressing the "kidnapped robot problem." An improved method in ref. [107] enhanced mobile robot localization accuracy by combining visual sensors, an IMU, and an odometer through tight coupling and nonlinear optimization. A monocular visual odometer that combined local and global optimization of point-line features to improve self-positioning accuracy in mobile robots was proposed in ref. [108]. The limitations of SLAM using a single sensor were addressed in ref. [109] by combining 2D LiDAR and RGB-D camera data for joint optimization. A grid-based visual localization system

https://doi.org/10.1017/S0263574725102518 Published online by Cambridge University Press

https://doi.org/10.1017/S0263574725102518 Published online by Cambridge University Press

Table III. *Comparison of existing related work on hybrid SLAM.*

Hybrid SLAM	Method	Advantages	Disadvantages
LiDAR-IMU loosely coupled system	LOAM [92]	• Low-drift mapping in real time. • Low computational complexity	• Lack of closed-loop and backend optimization.
	LeGO-LOAM [93]	• Capture fine environmental details. • Two-stage optimization.	• Closed-loop detection accuracy is low. • Complex terrain failure.
LIDAR-Visual-IMU Loosely Coupled System	V-LOAM [94]	• High performance, low drift, and robustness to motion.	• Visual features lose tracking during fast motion due to a lack of correlation between vision and lidar.
	LVIOM [95]	• Functionality in texture-less and feature-less environments.	• Inertial system state stops updating when vision fails.
LiDAR- IMU tightly coupled system	LIOM [96]	• Handles sensor degradation by automatic reconfiguration. • Robustness in dynamic motion, a dark, textureless environment.	• Time-consuming scan matching, without GPU utilization
	LIO-SAM [97]	• Uses selective keyframes and a sliding window approach for efficiency. • Eliminates drift issues with loop closure factors.	• Degradation in open scenes.
	FAST-LIO [98]	• A new formula was used to compute the Kalman gain efficiently. • Fast and efficient.	• Cumulative error.
	FAST-LIO2 [99]	• Directly registers raw points to the map for increased accuracy.	• Struggles in rapidly changing scenes.
LIDAR-Visual-IMU Tightly Coupled System	LVIO-Fusion [100]	• Graph optimization with GPS and loop closure. • Self-adaptive algorithm for adjusting sensors' weight.	• High computational cost. • Difficult to deploy.
	LVI-SAM [101]	• VIS and LIS complement each other. • Optimize depth information.	• Poor closed-loop performance

https://doi.org/10.1017/S0263574725102518 Published online by Cambridge University Press

Table IV. *Summary of existing methods based on optimization-based approaches.*

Optimization-based approaches	Ref.	Contributions	Performance analysis	Advantages	Disadvantages
Pose graph-based optimization	[105]	Local scan matching and global graph-based optimization.	• Effectiveness is tested both locally and globally.	• Robust to disturbances from moving objects.	• An offline approach requires the robot to sample all sensor data.
	[106]	Graph representation of clusters.	• The Voronoi graph-based segmentation method is considered the most stable.	• Address the "kidnapped robot problem."	• Challenging to determine the optimal grid resolution.
	[107]	Combines visual sensors, IMU, and odometer.	• Outperforms the VINSMONO algorithm	• Odometer information enhances positioning results.	• Limited scalability.
	[108]	Combines local and global optimization of point-line features.	• TUM standard dataset	• A combination of point and line features allows for a better environmental description.	• RGBD visual odometer is affected by depth error and effective distance limitations.
	[109]	2D LiDAR and RGB-D camera data for joint optimization.	• Compares the positioning accuracy and mapping effect of pure vision SLAM and fusion SLAM	• Joint optimization of LiDAR and camera data improves localization accuracy	• Computational complexity increases.
	[110]	A grid-based visual localization system is based on artificial markers.	• Tested in both laboratory and factory environments. • Tested using 856 images. • High marker detection, 1188 out of 1216.	• Improved visual localization solution.	• Marker occlusion and inferences.

https://doi.org/10.1017/S0263574725102518 Published online by Cambridge University Press

Table IV. *Continued*

Optimization-based approaches	Ref.	Contributions	Performance analysis	Advantages	Disadvantages
	[111]	Compares particle filters and graph-based localization.	• Compare the accuracy and runtime behavior of both localization approaches.	• Graph-based localization approach is superior in terms of accuracy.	• Graph-based systems require libraries such as g2o or GTSAM.
Moving horizon estimation (MHE)	[112]	MHE integrated with EKF for 3D CL using a single beacon.	• Outperformed EKF in two simulation scenarios: constraints on states and constraints on noise.	• MHE ensures estimation accuracy and prevents sudden interference.	• Requires a relatively long time to solve the nonlinear optimization compared to EKF.
	[113]	UWB-based localization.	• Estimation error is evaluated using mean square error (MSE) and root mean square error (RMSE).	• It offers accurate positioning, low cost, and strong interference tolerance, making it suitable for a dynamic complex environment.	• The influence of non-Gaussian noise on UWB measurements can affect the system's performance.
	[114]	Adaptive vision-based sensor fusion.	• Promising results compared to a commercial multi-camera system.	• Adaptive Moving Horizon Estimator (MHE) improved state estimation using ArUco markers.	• MHE has limitations in adapting to sensor uncertainties.

https://doi.org/10.1017/S0263574725102518 Published online by Cambridge University Press

Table IV. Continued

Optimization-based approaches	Ref.	Contributions	Performance analysis	Advantages	Disadvantages
	[115]	Distributed multi-robot SLAM.	• Effectiveness of the method using actual experimental data. • Collected the measurement data using three mobile robots and three landmarks.	• Improved estimation accuracy with reasonable computational effort.	• Communication among robots is solely through sensor measurements and wireless communication. • Restrict the applicability of the algorithm.
	[116]	Multi-sensor fusion using MHE.	• MHE estimation output is compared with that of a UKF. • Validation includes autonomous driving sequences from the EU long-term dataset.	• Integration of the CGMRES method enhances estimation accuracy and reduces computational cost.	• Higher processing requirements can potentially impact real-time performance.
		Compare MHE and EKF effectiveness in crowded areas.	• Experimented in a crowded environment, operating the electric wheelchair with LiDAR.	• Reduced computational complexity by 88% compared to the trust region method.	• LiDAR-equipped robots face localization failure in crowded environments.

was presented in ref. [110], which relied on artificial markers arranged on the ceiling, with the mobile robot capturing images using an up-facing monocular camera. Comparison of two state estimation techniques, particle filters and graph-based localization, was evaluated in the context of automated vehicles [111]. The study discussed the advantages of both algorithms and their ability to adapt to available computational resources at runtime.

Moving horizon estimation. MHE is mathematically formulated as an optimization problem that minimizes the discrepancy between the predicted state (based on the dynamic model) and the measured state (based on sensor data) over a moving time horizon. By integrating sensor measurements with dynamic modeling, MHE enables the robot to determine its global position within an indoor environment, even starting from an unknown location. It does this by continuously updating the localization estimate with new sensor data, allowing the robot to gradually refine its understanding of the environment and accurately locate itself within the global coordinate system.

In ref. [112], the three-dimensional multi-robot cooperative localization problem was addressed by introducing a novel approach based on MHE integrated with EKF for improved localization accuracy. The proposed method bounded localization error, imposed constraints on states and noises, and utilized previous range measurements for current estimation, outperforming traditional EKF-based methods. An ultra-wideband (UWB) localization algorithm for mobile robots presented in ref. [113], which employed a moving horizon optimization, integrating a kinematic model, reference poses, and handling heavy-tailed noise in UWB ranging. Indoor UAV localization is challenging due to the absence of reliable Global Navigation Satellite Systems (GNSS) signals, requiring the development of robust sensor fusion algorithms for state estimation. An adaptive MHE implemented in ref. [114] for indoor UAV localization using ArUco markers, showing improved state estimation performance compared to a high-accuracy motion tracking camera system. According to ref. [115], the focus was on distributed multi-robot SLAM using moving horizon estimation. The algorithm involved sharing sensor measurements and past estimates among robots, and it also utilized the Conjugate/Gradient minimal residual (C/GMRES) method for optimization to enhance computational efficiency. An efficient and versatile multi-sensor fusion framework based on MHE was proposed in ref. [116], emphasizing its capability to handle different sensor characteristics, missing measurements, outliers, and real-time applications. The proposed scheme aimed to improve localization accuracy and robustness, with experimental results demonstrating superior performance compared to the UKF. The accuracy of SLAM using MHE in crowded indoor environments, where measured data may be insufficient, is discussed in ref. [117]. Researchers applied MHE to SLAM to improve motion model evaluation in scenarios with limited data. They conducted experiments comparing MHE-based SLAM with EKF-based SLAM and demonstrated that Sequential Quadratic Programming (SQP) reduced computational complexity.

Symmetrical indoor global localization. This section delves into the problem and key challenge of global localization when a mobile robot encounters a symmetrical indoor environment. Global localization in mobile robotics refers to the process by which a robot determines its position and orientation within an environment without prior knowledge of its starting location. This task becomes particularly challenging in indoor settings due to the absence of GPS signals and the prevalence of symmetrical structures. Symmetrical environments can lead to perceptual aliasing, where different locations appear identical to the robot's sensors, causing ambiguity in position estimation. For instance, long corridors or repetitive architectural features can mislead the robot into believing it is in multiple possible locations simultaneously. This ambiguity complicates the localization process, as traditional sensors like LiDAR or cameras may not provide distinctive information to disambiguate these positions.

Addressing these challenges is crucial for enhancing the autonomy and efficiency of mobile robots across various applications, such as automated delivery systems, indoor surveillance, and service robotics. This section provides an overview of the complexities involved and outlines the existing efforts aimed at overcoming those obstacles.

https://doi.org/10.1017/S0263574725102518 Published online by Cambridge University Press



Figure 8. *Turtlebot 3 Burger robot platform.*

3.1. Problem formulation

This paper focuses on the 2-D localization problem, meaning that the robot pose x_t at time step $t \in \mathbb{Z}_{\geq 0}$ is a random variable that satisfies $\mathrm{x}_t = [x, y, \theta]^T \in \mathbb{R}^2 \times \mathbb{S}$.

Problem 1: Given sensor observations from 2-D laser rangefinder $\mathrm{z}^{\mathbf{L}}$, control inputs u, and a pre-built occupancy grid map (OGM) m^{L}, the problem of localization in ordinary environments can be formulated as the estimation of robot pose x_t :

$$p\left(\mathbf{x}_t \mid \mathbf{z}^{\mathrm{L}}_{1:t}, \mathbf{u}_{1:t}, \mathbf{m}^{\mathbf{L}}\right) \tag{2}$$

However, in symmetrical environments, it is tough to match the observations correctly $\mathrm{z}^{\mathbf{L}}$ with OGM m^{L} due to the high degree of symmetry in geometric features and low similarity to actual laser measurements. As a result, the robot fails to localize itself in the environment, leading to multiple hypotheses that need to be resolved correctly, as shown in Figure 8. This challenge arises because of symmetry if the environment needs proper attention to improve the localization accuracy of a mobile robot in a symmetrical scenario.

3.2. Key challenges

Global localization for indoor mobile robots can pose a considerable challenge in a symmetrical environment, where the surroundings exhibit significant similarities or mirror-like features, similar rooms, and long corridors. One key challenge is the ambiguity in distinguishing between such similar or symmetrically arranged features. Here are some specific aspects of this challenge:

- *Limited discriminative features:* Symmetrical structures may result in a scarcity of discriminative features that can be used for accurate localization. Without unique and easily distinguishable features, the robot may rely on less reliable cues, increasing the likelihood of misjudging its position.
- *Vulnerable to local minima:* Symmetrical environments may increase the likelihood of the robot getting stuck in local minima during the global localization process. Ambiguous features may lead the robot to converge on an incorrect position, thinking it has reached the correct location when it has not.
- *Sensor noisy measurements:* Sensor noise and inaccuracies can be exacerbated in symmetrical environments, where sensors may provide ambiguous readings due to the presence of similar structures. This noise can lead to inaccuracies in the robot's perception of its surroundings and contribute to localization errors.

This review paper aims to shed light on the challenges with indoor global localization of mobile robots operating in symmetrical environments. Addressing these key challenges helps to develop an adaptable and sophisticated robotics system capable of effectively navigating through a symmetrical indoor space.

https://doi.org/10.1017/S0263574725102518 Published online by Cambridge University Press

3.3. Review of existing solutions

A review of existing solutions for indoor mobile robot global localization highlights various approaches developed to overcome the challenges of symmetrical environments. Researchers have explored a spectrum of techniques, ranging from traditional methods to state-of-the-art algorithms, to enhance the accuracy and efficiency of localization systems. Table V summarizes existing global localization work in symmetrical indoor environments.

From the studies listed in Table V, indoor global localization in symmetrical environments presents a complex yet critical challenge for various applications. Despite the inherent difficulties posed by symmetrical features, recent advancements in sensor technologies, machine learning algorithms, and probabilistic methods have shown promising results in improving localization accuracy. However, further research and development are necessary to comprehensively address the unique challenges of symmetrical environments. By leveraging emerging technologies, such as advanced sensor fusion techniques and deep learning algorithms, and exploring innovative hybrid approaches, we can enhance the robustness and accuracy of indoor localization systems in symmetrical environments.

4. Experimental results and discussions

This section presents the evaluation of various global localization algorithms using a benchmarking framework designed for controlled symmetrical environments. The experiments are conducted in two distinct test cases, one with low symmetry and the other with high symmetry, to analyze how different approaches perform under varying levels of structural ambiguity. Comparisons of probabilistic-based, learning-based, SLAM-based, and optimization-based localization methods are undertaken to assess their accuracy, robustness, and computational efficiency. The results highlight the strengths and limitations of each approach, providing insights into their suitability for real-world deployment in symmetrical indoor spaces. Additionally, we discuss the impact of repetitive structures on localization performance and explore potential strategies to enhance robustness in such challenging environments. The findings from this study aim to guide the selection and development of effective localization techniques for applications where symmetry-induced ambiguities pose significant challenges.

4.1. Setup and initialization

To assess the effectiveness of each method in environments with varying levels of symmetry, we utilize the Turtlebot3 Burger robot platform as shown in Figure 8. This platform features a maximum translational velocity of 0.22 m/s and a maximum rotational velocity of 2.84 rad/s. It is equipped with an LDS-01 Laser Distance Sensor, which provides a 360-degree field of view and an IMU. The Turtlebot3 Burger robot is powered by a 32-bit ARM Cortex®-M7 microcontroller unit (MCU) with FPU, running at 216 MHz and delivering 462 DMIPS. Additionally, it incorporates a Single Board Computer (SBC), specifically the Raspberry Pi 3 Model B+.

4.2. Operating scenario

Figure 9 shows the two representative examples of symmetrical environments in experiments, including one low symmetrical square room and one high symmetrical environment with symmetrical rooms

The Hector SLAM algorithm, relying on 2-D LiDAR data, is employed to construct occupancy grid maps of low and high symmetrical environments as shown in Figure 10. Figures 10(a-b) represent low- and high-symmetry environments. Mapping sessions are conducted during the day, with multiple runs performed to select the most accurate representation. Four distinct ground truth poses are manually determined in each scenario using a tape ruler method.

https://doi.org/10.1017/S0263574725102518 Published online by Cambridge University Press

https://doi.org/10.1017/S0263574725102518 Published online by Cambridge University Press

Table V. *Summary of global localization in a symmetrical indoor environment.*

Ref.	Approaches	Performance analysis	Advantages	Disadvantages
[9]	Adaptive dynamic clustered particle filtering method.	• Experiments were conducted offline. • Fair comparison with standard MCL, Cluster-MCL, and ADC-MCL algorithms.	• Improves computational efficiency without sacrificing estimation accuracy.	Verified by simulation
[2]	Laser-based probabilistic method.	• Experiments were done using the Pioneer P3-DX robot. • Evaluated both on real data and simulations.	• Reduce the ambiguity in a structurally symmetrical environment.	Careful deployment of artificial landmarks
[3]	Learning policy for autonomous landmark deployment	• Data association performance of the approach is significantly higher compared to other strategies.	• Learned policies for landmark deployment result in significantly more accurate pose estimates. • Generalizable to a new environment.	Careful deployment of artificial landmarks
[10]	Difference evaluation and scan matching	• Reduce translation and rotation absolute pose errors by an average of 27.29% and 25.82%.	• Higher adaptability and positioning accuracy in narrow, long, straight environments.	Does not consider the situation where the mobile robot is inside a specific room
[118]	Cluster algorithm to group hypotheses	• Active localization system decreases both ground truth error and pose uncertainty.	• Improves robot localization in symmetrical and open environments.	Performs only rotational movement
[119]	Text information and laser scan data	• Improves the global localization speed and success rate to 96.2% with a few particles. • Mobile robot can recover from robot kidnapping after a short movement of 2.72 m.	• Combining text information and laser scan data provides a robust and accurate localization solution.	Limited generalizability of the approach to a complex environment.

https://doi.org/10.1017/S0263574725102518 Published online by Cambridge University Press

Table V. *Continued*

Ref.	Approaches	Performance analysis	Advantages	Disadvantages
[7]	Object detection and textual with MCL	• Experiments were executed with 10,000 particles in the filter. • Tested: office environment. • Absolute trajectory error for global localization of algorithm over 11 sequence is 0.079/0.23 m.	• Robust to structural changes in the environment. • Reducing the need for detailed geometric maps.	Performance of the framework may also be influenced by the sparsity of the CAD map and accuracy of object detection.
[120]	Magnetic field and 3D lidar	• Tested: Rectangular corridor, polygonal corridor, semi-indoor corridor. • Achieves over 95% localization success rate and takes less than 2.0 m average traveling distance	• Framework does not rely on pre-installed infrastructures. • Flexible and cost-effective.	Can be affected by external magnetic interference
[121]	Integrates information from RGB-D camera, lidar, and motor encoder	• Evaluated in two real-world buildings with highly symmetric environments.	• Reducing ambiguity as the agent moved along its trajectory. • Accurate estimation even in visually and geographically similar scenarios.	Estimation of objects on each edge is only limited by the maximum distance range of the RGB-D camera model and its field of view.
[122]	Deep-learning-based strategy for solving the kidnapped robot problem.	• Pose accuracy: 2 m and 5°scene classification accuracy: up to 98%.	• Increase the number of features input to a neural network.	2D lidar, which has a limitation in range. Generalizability of method.

https://doi.org/10.1017/S0263574725102518 Published online by Cambridge University Press

Table V. *Continued*

Ref.	Approaches	Performance analysis	Advantages	Disadvantages
[123]	Probabilistic approach for collaborative multi-robot localization	• Experiment was conducted with two robots. • Tested: symmetric hallway, hexagonal environment with edge length of 8 m.	• Faster and more accurate localization. • Amortization of high-cost sensors.	Identification of negative detections would increase computational overhead.
[8]	Active localization strategy for indoor position tracking-clustering informative points.	• Tested: open and symmetric environment. • Experiments indicate that Spectral Clustering and K-means agents perform better. • Experiments indicate that Spectral Clustering and K-means agents perform better.	• Ability to prune a significant number of wrong hypotheses.	Decreases both ground truth error and pose uncertainty. Overall, the mission completion time has increased.
[124]	Combining visual recognition of objects with topological graphs.	• Tested: third floor of the Polytechnic School of the University of Alcalá.	• Depth information from the RGB-D camera improved the estimation of the relative positions of objects.	Multiple detections can be generated for the same object if the distance is over 15 m from the camera.
[125]	Visual SLAM, text detection, and recognition for mobile robot localization.	• Tested: corridor • Higher success rates than AMCL, ORB-SLAM, and text-based methods, with a localization success time of 19.	• Improves the robustness and precision of global localization in corridor environments.	Does not consider the situation where the mobile robot is inside a specific room.
[126]	Integrating multiple measurements with low accuracy into MCL	• Pioneer 3-DX platform in a long vertical corridor.	• Multiple weak cues can provide robust position estimates in ambiguous environments.	Increase in computational cost.

https://doi.org/10.1017/S0263574725102518 Published online by Cambridge University Press

Table V. *Continued*

Ref.	Approaches	Performance analysis	Advantages	Disadvantages
[127]	Landmark extraction and matching method	• Indoor environment with uncontrolled illumination.	• Natural, salient image patches as landmarks provide additional highly discriminative information for localization.	The quality of the results depends on the interest point detection.
[128]	Combining map-based and appearance-based global localization with the particle filter.	• Experiments were conducted in a corridor with strong perceptual aliasing.	• Use of reflectance images makes the technique suitable for environments with changing lighting conditions.	Robustness of the approach to other situations is not mentioned.
[129]	Combining vision and sonar sensors	•Experiments were conducted in the hallway. • Combining sonar and vision-based, the average error in x and y direction is small, 1050 mm.	Sensor fusion improves localization	Both vision-based and sonar-based approaches have specific weaknesses when used alone.
[130]	Coarse-to-fine paradigm that uses visual features	• Corridor environment • Localization success rate of 98.8% and an average moving distance of only 0.31 m	• A monocular camera provides dense information.	

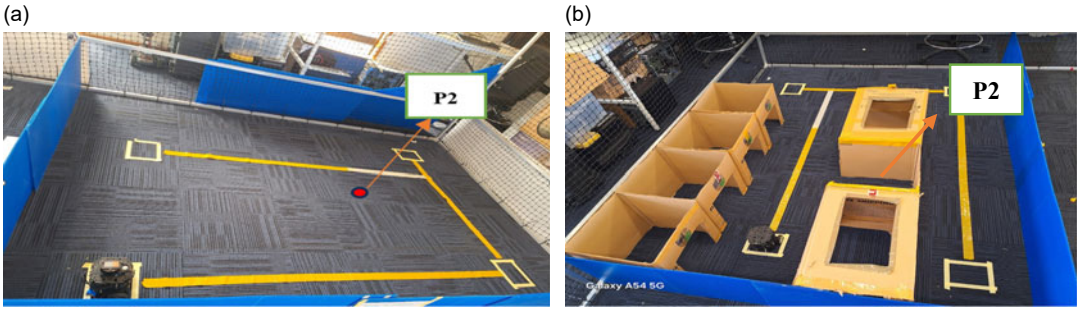


Figure 9. *Operating scenarios: (a) low-symmetry square area and (b) high-symmetry area with several symmetrical rooms.*

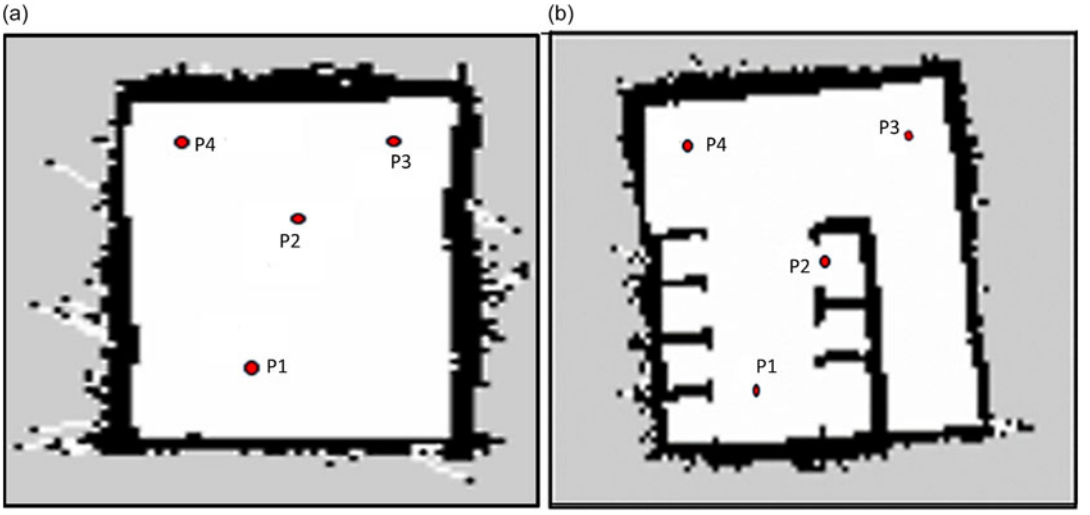


Figure 10. *Occupancy grid map for both the cases: (a) low-symmetry environment and (b) high-symmetry environment.*

4.3. Comparative analysis

4.3.1. Algorithms for comparison

AMCL, Hector SLAM, Hybrid SLAM, and Cartographer SLAM are commonly used in robotics to test the global localization of mobile robots due to their distinct characteristics and capabilities. Each method has strengths and weaknesses, making it suitable for different scenarios and environments.

AMCL (adaptive Monte Carlo localization). AMCL uses a particle filter to represent the belief state of a robot's pose. Particle filters provide a nonparametric representation of the belief or probability distribution over the robot's pose. This flexibility allows them to handle complex and nonlinear relationships between the robot's motion and the sensory measurements. Global localization often involves dealing with multimodal distributions, where the true position of the robot is associated with multiple possible poses. Particle filters are well suited for representing and tracking such multimodal distributions, allowing them to capture the uncertainty associated with various possible locations. In Figure 11, AMCL is demonstrated as the robot operating in a symmetrical environment. In Figure 11(a), the robot is initially uncertain about its position, causing the particles to spread throughout a large part of the free space. In Figure 11(b), as the robot moves and approaches a corner of the map, the particles start to converge. Finally, in Figure 11(c), the ambiguity is resolved when the robot is close to its goal, and the robot accurately identifies its position.

https://doi.org/10.1017/S0263574725102518 Published online by Cambridge University Press

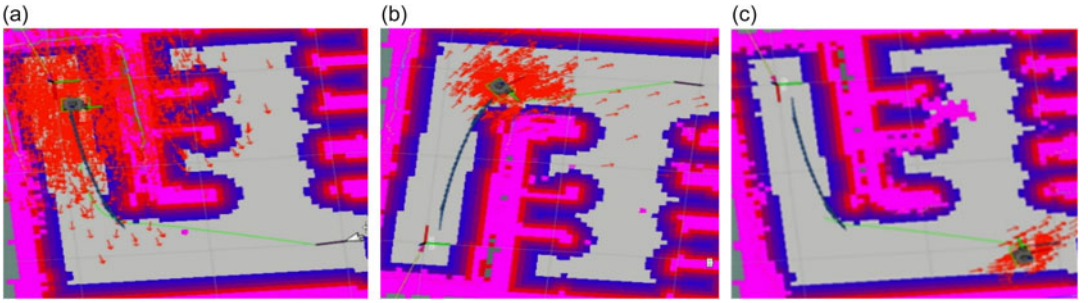


Figure 11. *Mobile robot localization using AMCL.*

Hector SLAM. Hector SLAM relies on scan matching techniques to align successive sensor measurements and build a map of the environment. This method benefits scenarios with low computational resources or limited sensor capabilities. Hector SLAM is known for its real-time performance, making it suitable for applications where low latency is crucial.

Hybrid SLAM. Hybrid SLAM refers to a fusion approach in SLAM that combines different sensor modalities or algorithms to overcome limitations and improve the overall performance of SLAM systems. In a hybrid SLAM system, multiple sensors such as LiDAR, cameras, inertial sensors, and odometry may be integrated, along with various SLAM algorithms, to leverage the strengths of each component. This integration enables more robust and accurate localization and mapping results. This paper uses Hector SLAM with EKF to evaluate the localization performance in both operating scenarios. The EKF provides a mathematical framework incorporating sensor measurements and motion models to estimate the robot's pose. EKF effectively combines odometry information with sensor measurements (in this paper it is IMU) and Hector SLAM to improve localization accuracy.

Cartographer SLAM. In contrast to Hector SLAM, which is a 2D grid-based SLAM method primarily utilizing LiDAR sensors and recognized for its simplicity, Cartographer employs advanced sensor fusion techniques and global optimization to enhance localization accuracy. While Hector SLAM is appreciated for its user-friendly nature, Cartographer is a more versatile and customizable SLAM library, providing extensive features and configuration options. This increased flexibility often translates into superior localization precision, particularly in environments with complex features and varying lighting conditions.

The rqt_tf_tree of all four algorithms (AMCL, Hector SLAM, Hybrid-SLAM, and Cartographer) used in this paper for comparison are shown in Figures 12–15. The rqt_tf_tree is a visualization tool in the Robot Operating System (ROS) that helps understand a robotic system's transform tree (TF) hierarchy. TF is a mechanism in ROS that maintains the relationship between different coordinate frames in the robot's environment, allowing for seamless coordination between sensors, actuators, and other components. Hence, using rqt_tf_tree helps to understand how sensor data is transformed into a common reference frame, how different robot components are spatially related, and how transformations are propagated through the system. It also helps to identify missing or incorrect TF transforms and debug issues with coordinate frame transformations.

4.3.2. Performance analysis

When comparing these algorithms as baselines, factors such as accuracy, robustness, computational efficiency, and performance in different environments must be considered. Evaluating these algorithms can help to gain insights into their strengths and limitations, helping to choose the most suitable algorithm for a given robotic application in a symmetrical environment. In this paper, the performance analysis of the above algorithms has been evaluated in the context of global localization of an indoor mobile robot in a symmetrical environment in two cases: (a) low-symmetry environment and (b) high-symmetry environment.

https://doi.org/10.1017/S0263574725102518 Published online by Cambridge University Press

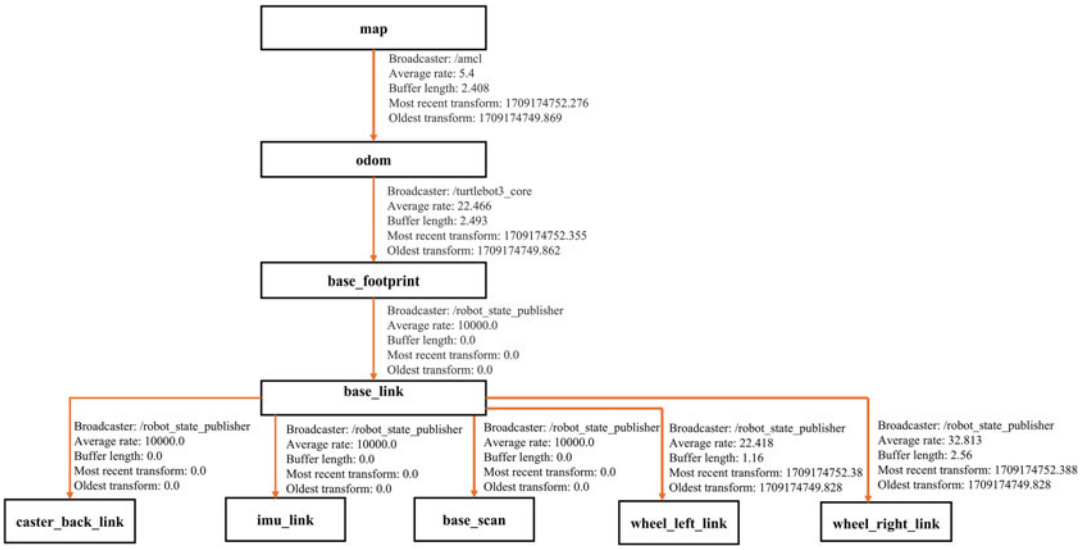


Figure 12. *AMCL frame transform hierarchy visualized with rqt_tf_tree.*

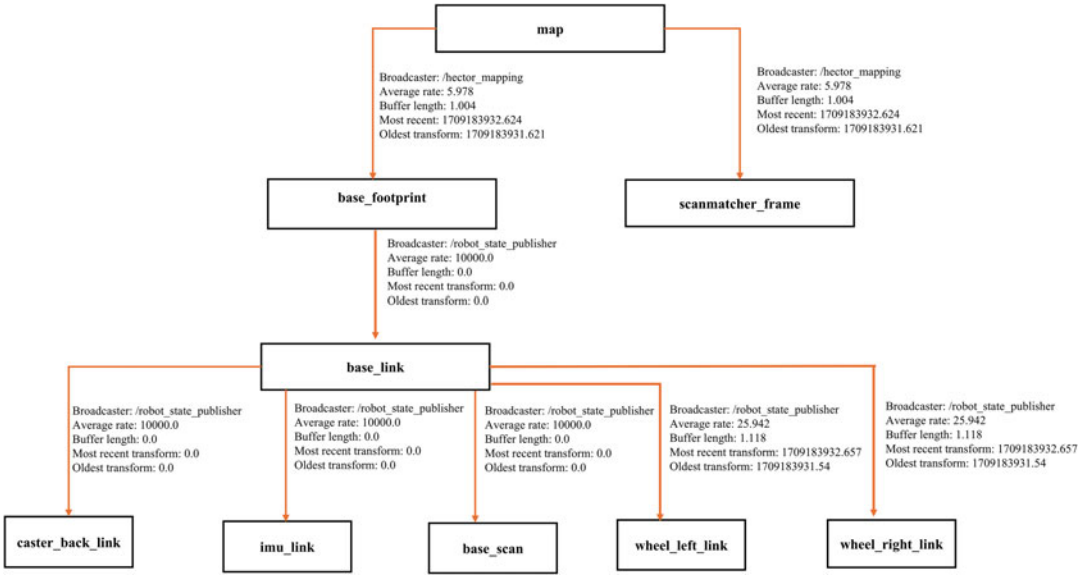


Figure 13. *Hector SLAM frame transform hierarchy visualized with rqt_tf_tree.*

Four distinct locations have been identified within the occupancy grid map illustrated in Figure 11 for assessment purposes. Points P1 and P2 correspond to poses in a lengthy corridor and within one of the symmetrical rooms. Emphasis has been placed on the higher importance of points P1 and P2, as they serve as critical benchmarks for evaluating algorithmic performance in symmetrical environments. However, poses P3 and P4 also hold significance, contributing to the overall global localization performance of the mobile robot. The precise ground truth values for each of the four poses have been determined using the tape and ruler method, and these measurements are detailed in Table VI. For each evaluated approach at each pose, 10 individual random tests were conducted.

Accuracy analysis. The goal of assessing accuracy in robot localization is to evaluate how accurately the robot can estimate its pose. Root mean square error (RMSE) calculates the error between each

https://doi.org/10.1017/S0263574725102518 Published online by Cambridge University Press

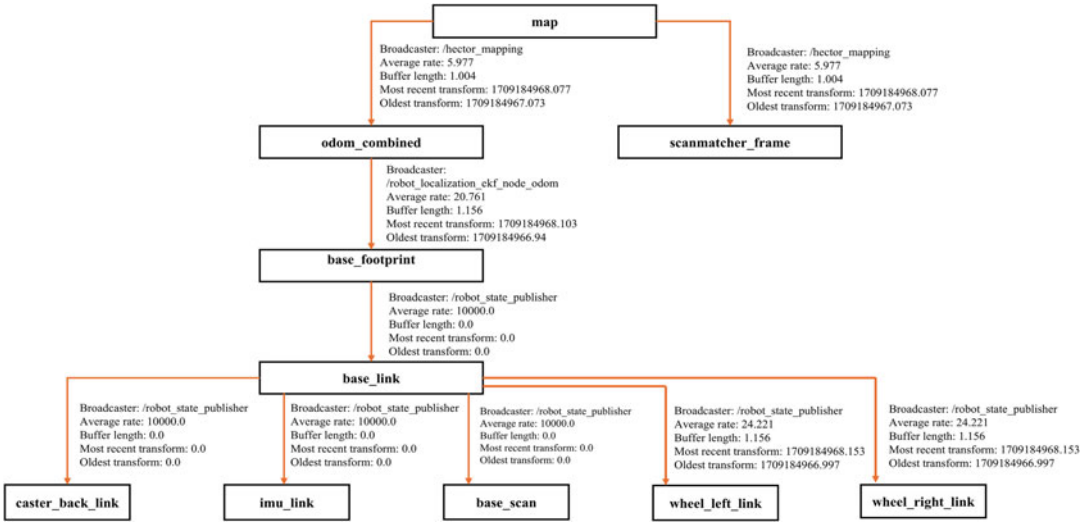


Figure 14. *Hybrid SLAM frame transform hierarchy visualized with rqt_tf_tree.*

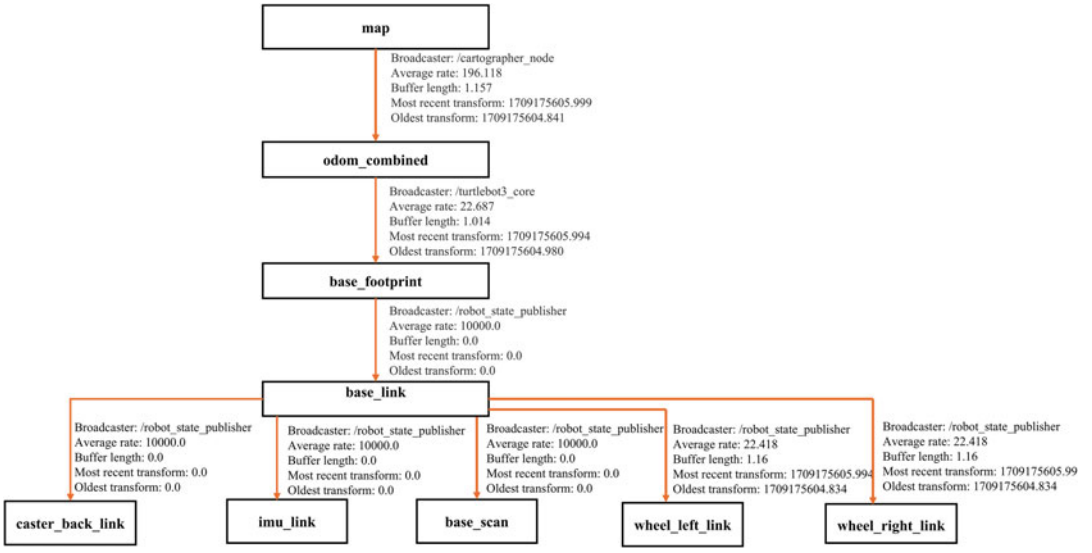


Figure 15. *Cartographer frame transform hierarchy visualized with rqt_tf_tree.*

estimated pose and its corresponding ground truth pose, which is given below:

$$\mathrm{RMSE} = \sqrt{\frac{\sum_{i=1}^{N} \left(x_i - \hat{x}_i\right)^2}{N}} \tag{3}$$

where (x_i-x $\hat{o}_i$) is the error between the estimated pose and the ground truth pose, and n is the total number of measurements. The localization accuracy is shown in Figure 16(a-(b) for both low- and high-symmetry cases. As seen from Figure 16(a), the accuracy of AMCL in determining the poses P1 and P2 is 90% and 89%. The situation worsens as the complexity increases, as in Figure 16(b). Degradation of accuracy happens as AMCL struggles to maintain a diverse set of particles to represent the robot pose accurately. In a symmetrical square environment, the particle diversity is compromised. From Figure 16(b), the AMCL struggles to converge to the correct pose; as a result, accuracy is reduced further to 84%. Meanwhile, utilizing scan matching techniques, Hector SLAM enhances accuracy to 94%

https://doi.org/10.1017/S0263574725102518 Published online by Cambridge University Press

Table VI. *Ground truth value of each pose (m, m, degree).*

Chosen poses	***Ground truth value***
Pose 1	[0.287, 1.424, 174.59]
Pose 2	[1.09, 0.861, −92.544]
Pose 3	[2.13, −0.114, −5.23]
Pose 4	[2.204, 1.764, 92.20]

in low symmetrical environments 16(a) for poses P1 and P2 by using distinctive features in lidar sensor data to differentiate between various locations. This attribute makes Hector SLAM advantageous in symmetrical structures, enabling the algorithm to adjust to sensor readings' fluctuations promptly. However, when examining poses P3 and P4 in a high symmetrical environment in Figure 16(b), the accuracy of Hector SLAM is reduced to 89%–90% since the walls in these poses are plain and lack sufficient distinctive features for effective differentiation. Hybrid SLAM, which uses Hector SLAM and EKF for sensor fusion, can potentially handle symmetrical environments more effectively by integrating information from odometry and IMU. This helps to obtain a better pose estimate from both sensor measurements, which improves the localization accuracy of poses to 98%–99%, as depicted in Figure 16(a-b). Combining odometry and IMU in Hybrid SLAM can contribute to more accurate pose estimation, which is beneficial in an environment where symmetrical feature pose challenges other algorithms. Adding to it, the cartographer shows the same range of improvement in localization accuracy for three poses P1, P2, and P3 in both cases as in Figure 16(a-b) due to its advanced sensor fusion and global optimization, making it well suited for improving performance in symmetrical indoor environment.

Robustness analysis. The robustness analysis of the comparison baseline algorithm is evaluated using the correct localization rate (CLR), mismatched localization rate (MLR), and localization failure rate (LFR). These three metrics denote the ratio of successful cases, the ratio of mismatched/misplaced cases, and the ratio of instances where the algorithm fails to localize the robot, over the total number of localization cases, respectively. The sum of CLR, MLR, and LFR must be 100%.

The localization robustness is shown in a stacked histogram in Figure 17, where both low- and high-symmetry cases are represented by Figure 17(a-b), respectively. For low- to high-symmetry environments, the robustness of AMCL degraded to 23.5%. At the same time, other methods like Hector SLAM, Hybrid SLAM, and Cartographer show a decrement in their CLR of 32.52%, 37%, and 42.50%, respectively. The relatively low success rate of all localization methods is attributed to the increased number of similar environments. Therefore, it is evident that the appearance of a symmetrical structure in a complex environment will further lead to deterioration in the performance of AMCL, Hector SLAM, Hybrid SLAM, and Cartographer. This performance degradation shows that the presence of a symmetrical environment is still a challenging problem requiring more research.

Computational complexity analysis. The complexity analysis of algorithms is assessed based on CPU usage and processing time. This paper calculates the computational cost of each algorithm's average processing time to localize the poses. Figure 18(a-b) show the CPU usage of each comparison baseline algorithm.

The symmetrical environment was created at Murdoch University's Advanced Robotics laboratory with dimensions of 2 m by 3 m. The heightened computational complexity of AMCL in highly symmetrical environments, increasing from 31.5% to 33.5%, is attributed to multiple factors, including the number of particles used to represent potential robot poses (which remains constant at 5000 particles in both environments) and the dimensionality of the state space. On the other hand, the Hybrid SLAM

https://doi.org/10.1017/S0263574725102518 Published online by Cambridge University Press

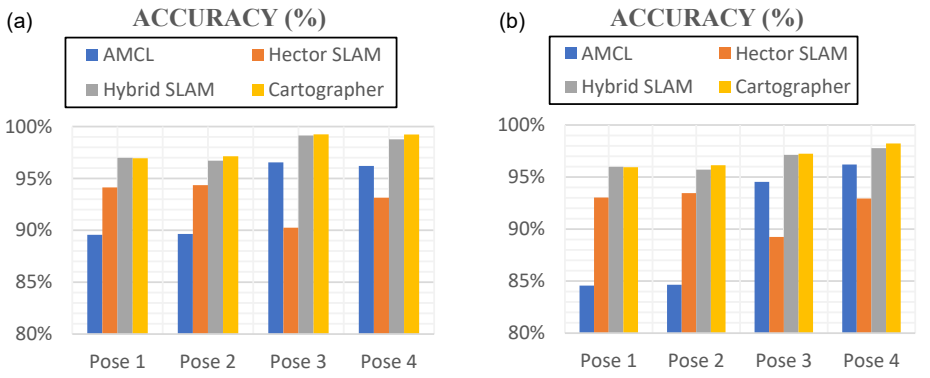


Figure 16. *Accuracy analysis for: (a) low-symmetry environment and (b) high-symmetry environment.*

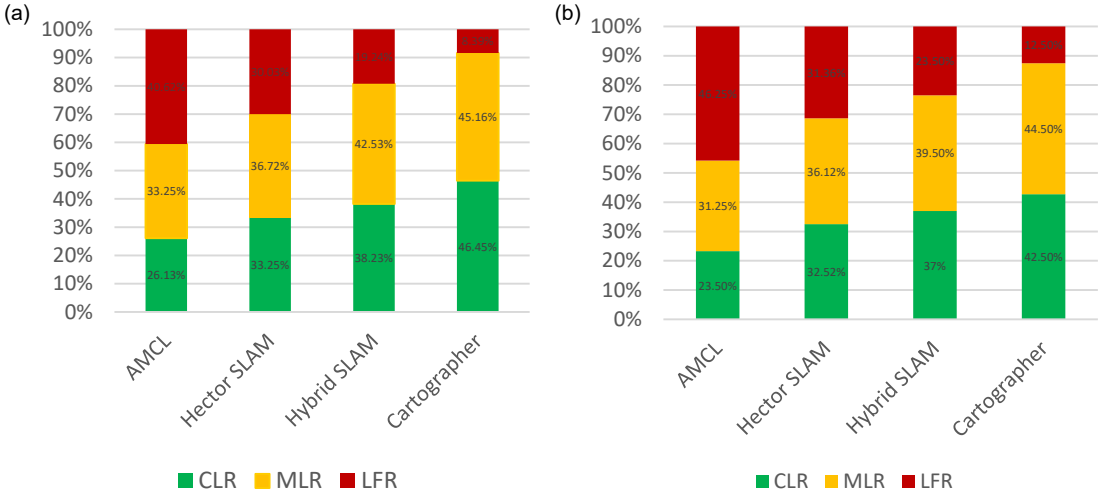


Figure 17. *Robustness analysis of each algorithm in: (a) low symmetrical and (b) high symmetrical environment.*

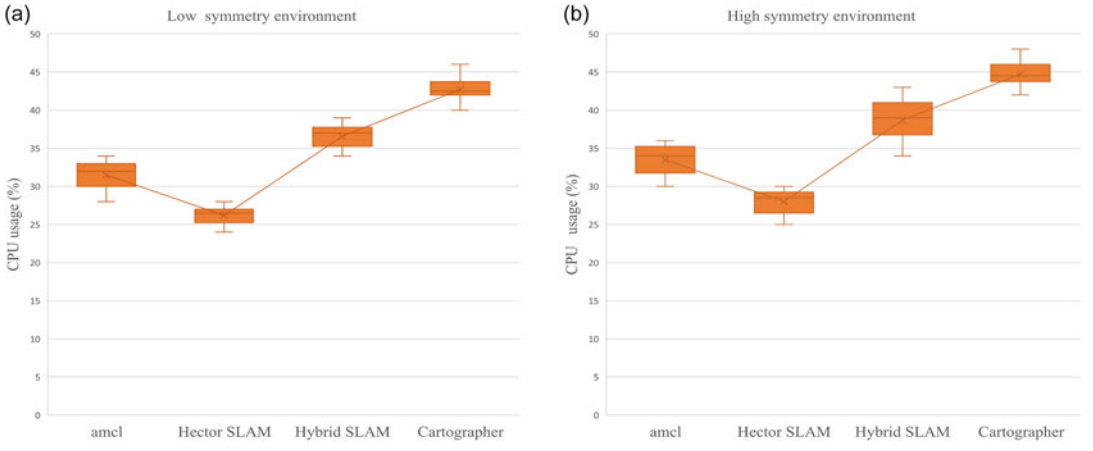


Figure 18. *CPU usage of each algorithm for both (a) low and (b) high symmetrical environments.*

https://doi.org/10.1017/S0263574725102518 Published online by Cambridge University Press

and Cartographer algorithm integrates data from diverse sensors, such as odometry and IMU, and utilizes an EKF. The rise in computational cost for Hybrid SLAM, which is 38.71%, and Cartographer of 44.7% is driven by the intricacy of the sensor fusion process. In comparison, Hector SLAM offers a lower computational complexity of 28% compared to other alternatives.

Therefore, finding a harmonious trade-off between computational cost and the desired accuracy and real-time performance is crucial in the final selection of a localization algorithm for a symmetrical environment robotic application. While learning-based methods offer the potential for high accuracy and adaptability, their computational requirements and the need for extensive training data must also be considered in this balance. For this reason, we do not include learning-based methods in comparison with other algorithms in this paper. However, in future work, we will explore using learning-based methods to address the challenges of symmetrical environments. Additionally, when incorporating learning-based methods, the algorithm evaluation needs to be approached differently to adequately assess their performance and suitability for application in a symmetrical environment, which will be evaluated in the future work of this paper.

5. Discussion and key insights

In the context of global localization for mobile robots operating within symmetrical indoor environments, our study underscores the importance of selecting appropriate localization algorithms to ensure both accuracy and robustness. Traditional methods, such as AMCL, demonstrate adequate performance in environments with low symmetry but encounter significant challenges as environmental complexity and symmetry increase. This decline in performance is primarily due to difficulties in maintaining particle diversity, leading to reduced localization accuracy. These challenges are consistent with findings in recent research, such as [131, 132], where particle filters faced difficulties in environments with high symmetry, particularly in preserving particle diversity.

Conversely, approaches like Hector SLAM, which utilize scan matching techniques, offer improved accuracy by leveraging distinctive features in LiDAR sensor data. However, their effectiveness diminishes in highly symmetrical settings where such distinctive features are scarce. Research in ref. [133] demonstrated that Hector SLAM excels in environments with rich structural features but struggles in scenarios where such features are less pronounced, which aligns with our findings in highly symmetrical environments.

Our findings indicate that Hybrid SLAM and Cartographer, which leverage advanced sensor fusion and global optimization strategies, deliver superior localization performance in symmetrical environments. These approaches integrate data from multiple sensors – such as LiDAR, IMU, and odometry – to enhance pose estimation accuracy. Previous studies [134, 135] have demonstrated that such multi-sensor fusion significantly improves the robustness and precision of localization systems, particularly in complex and highly symmetrical settings. Furthermore, Cartographer, with its use of globally optimized maps, has been shown to perform exceptionally well in environments with intricate geometric structures [136], further emphasizing the effectiveness of global optimization in addressing the challenges posed by symmetry.

Robustness analysis further reveals that, although all evaluated methods exhibit some degradation in performance within highly symmetrical environments, Hybrid SLAM and Cartographer maintain higher levels of robustness compared to AMCL and Hector SLAM. This is attributed to their comprehensive sensor fusion capabilities. As noted in ref. [137], systems relying solely on single-sensor modalities (such as LiDAR or vision) often struggle in environments with repetitive features or minimal texture, highlighting the importance of integrating multiple sensor sources.

In terms of computational complexity, Hybrid SLAM and Cartographer incur higher computational costs due to their intricate sensor fusion mechanisms. While methods like Hector SLAM offer a more computationally efficient alternative, they may not provide the same level of accuracy in highly symmetrical environments. This reinforces the idea with the support of research in ref. [138, 139] that there

https://doi.org/10.1017/S0263574725102518 Published online by Cambridge University Press

exists a balance between computational complexity with performance needs, particularly in complex, feature-scarce environments

In conclusion, this study underscores the critical need for advanced sensor fusion and optimization techniques to achieve effective global localization in symmetrical indoor environments. While traditional methods may suffice in less complex settings, the challenges posed by symmetry and environmental complexity necessitate more sophisticated approaches. Future research should focus on refining such methods to balance computational efficiency with localization accuracy and robustness, ensuring reliable performance across a diverse range of indoor environments. Moreover, incorporating learning-based methods into localization strategies presents a promising avenue for addressing the challenges posed by symmetrical environments. For instance, the study in ref. [140] demonstrated that integrating machine learning with probabilistic localization significantly improved performance in environments with repetitive patterns, such as symmetrical indoor spaces.

6. Conclusion

Global localization of mobile robots in symmetrical indoor environments presents significant challenges due to the lack of distinctive features and the inherent ambiguities introduced by uniform structures. This review has comprehensively analyzed various global localization strategies, categorizing them into probabilistic-based, learning-based, SLAM-based, and optimization-based approaches. We have highlighted the strengths, limitations, and practical applicability of these methodologies in real-world scenarios with critical evaluations.

Our findings have indicated that hybrid approaches, which integrate multiple localization techniques, consistently outperform standalone methods in symmetrical environments. Findings from the reviewed literature indicate that combining probabilistic models, learning-based techniques, and SLAM enhances localization accuracy and robustness. Such hybrid methodologies effectively mitigate the challenges posed by symmetry, enabling reliable robot localization in complex indoor settings such as corridors, tunnels, and industrial warehouses.

This review underscores the importance of continued research in hybrid global localization techniques, particularly in developing adaptive frameworks that dynamically switch between strategies based on environmental conditions. Future advancements should focus on enhancing computational efficiency, integrating additional sensor modalities, and addressing real-time implementation constraints. By leveraging the strengths of hybrid approaches, researchers and practitioners can further refine localization systems, paving the way for more autonomous and intelligent robotic navigation in challenging indoor environments.

Data availability statement. Data sharing is not applicable to this article as no new data were created or analyzed in this study.

Acknowledgements. I would like to express my deepest gratitude to all those who supported me throughout this research specially my supervisors, Dr. Hai Wang and Dr. Amir Mehdi Yazdani.

Authors contributions. Author Shiron Anto Melvin developed the research design and objectives, conducted a thorough review of existing literature review, and identified gaps the research aim. Shiron conducted data collection and analysis and interpreted findings in relation to research objectives. Co-authors Dr. Hai Wang and Dr. Amir Mehdi Yazdani assisted in formulating research question and objectives, provided their inputs on experimental design, and reviewed and revised the manuscripts by providing critical feedback.

Financial support. This research is supported by Murdoch University.

Competing interests. The authors hereby declare that there are no conflicts of interest.

Ethical standards. NA.

https://doi.org/10.1017/S0263574725102518 Published online by Cambridge University Press

References

[1] A. A. S. Alarifi, A. M. Alsaleh, M. Alnafessah, A. Al-Hadhrami, S. Al-Ammar, M. A. Al-Khalifa and S. Hend, "Ultra wideband indoor positioning technologies: Analysis and recent advances," *Sensors* **16**(5), 1424–8220 (2016).

[2] D. Amarasinghe, G. K. I. Mann and R. G. Gosine, "Landmark detection and localization for mobile robot applications: A multisensor approach," *Robotica* **28**(5), 663–673 (2010).

[3] H. K. M. Beinhofer and W. Burgard, "Deploying Artificial Landmarks to Foster Data Association in Simultaneous Localization and Mapping," **In:** *IEEE International Conference on Robotics and Automation*, (2013) pp. 5235–5240.

[4] P. Nazemzadeh, D. F. D. Macii and L. Palopoli, "Indoor localization of mobile robots through QR code detection and dead reckoning data fusion," *IEEE/ASME Trans. Mechatron.* **22**(6), 2588–2599 (2017).

[5] D. Amarasinghe, G. K. I. Mann and R. G. Gosine, "Landmark detection and localization for mobile robot applications: A multisensor approach," *Robotica.* **28**(5), 663–673, (2010).

[6] Z. Wu, M. W. G. Peng, X. Tang and D. Wang, "Magnetic-Assisted Initialization for Infrastructure-Free Mobile Robot Localization," **In:** *IEEE International Conference on Cybernetics and Intelligent Systems (CIS) and IEEE Conference on Robotics*, (2019) pp. 518–523.

[7] N. Zimmerman, T. G. X. Chen, J. Behley and C. Stachniss, "Long-term localization using semantic cues in floor plan maps," *IEEE Robot. Autom. Lett.* **8**(1), 176–183 (2023).

[8] F. Andrade, M. Llofriu, M. Marzoa Tanco, G. Trinidad Barnech and G. Tejera, "Active localization strategy for hypotheses pruning in challenging environments," *J. Intell. Robot. Syst.* **106**(2), 47 (2022).

[9] L. Zhibin, S. Zongying, Z. Mingguo and X. Wenli, "Mobile Robots Global Localization Using Adaptive Dynamic Clustered Particle Filters," **In:** *IEEE/RSJ International Conference on Intelligent Robots and Systems*, (2007) pp. 1059–1064.

[10] L. Zeng, S. Guo, Z. Xu and M. Zhu, "An indoor global localization technique for mobile robots in long straight environments," *IEEE Access* **8**, 209644–209656 (2020).

[11] A. Abdelgawad, "Auto-localization system for indoor mobile robot using RFID fusion," *Robotica* **33**(9), 1899–1908 (2015).

[12] J. Park and J. Lee, "Beacon selection and calibration for the efficient localization of a mobile robot," *Robotica* **32**(1), 115–131 (2014).

[13] A. Eman and H. Ramdane, "Mobile Robot Localization Using Extended Kalman Filter," **In:** *2020 3rd International Conference on Computer Applications & Information Security (ICCAIS)*, (2020) pp. 1–5.

[14] M. A. Mohammed Faisal, R. Hedjar, H. Mathkour, M. Zuair, H. Altaheri, M. Zakariah, M. A. Bencherif and M. A. Mekhtiche, "Enhancement of mobile robot localization using extended kalman filter," *Adv. Mech. Eng.* **8**(11), (2016). doi:10.1177/1687814016680142.

[15] H. J. Lee, M. S. Kim and M. C. Lee, "Technique to Correct the Localization Error of the Mobile Robot Positioning System Using an RFID," **In:** *SICE Annual Conference*, (2007) pp. 1506–1511.

[16] M. Brossard and S. Bonnabel, "Learning Wheel Odometry and IMU Errors for Localization," **In:** *2019 International Conference on Robotics and Automation (ICRA)*, (2019) pp. 291–297.

[17] Q. Xu, C. R. H. Yan and J. Ji, "Laser Sensor-Based Localization of Mobile Robot Using Unscented Kalman Filter," **In:** *2016 IEEE International Conference on Mechatronics and Automation*, (2016) pp. 1726–1731.

[18] J. P. Muhammad Latif Anjum, W. Hwang, H. I. Kwon, J. H. Kim, C. Lee, K. S. Kim and D. I. D. Cho, "Sensor Data Fusion Using Unscented Kalman Filter for Accurate Localization of Mobile Robots," **In:** *International Conference on Control, Automation and Systems*, (2010) pp. 947–952.

[19] I. Ullah, Y. Shen, X. Su, C. Esposito and C. Choi, "A localization based on unscented kalman filter and particle filter localization algorithms," *IEEE Access* **8**, 2233–2246 (2020).

[20] C. Yang, W. Shi and W. Chen, "Comparison of unscented and extended kalman filters with application in vehicle navigation," *J. Navig.* **70**(2), 411–431 (2017).

[21] L. D'Alfonso, W. Lucia, P. Muraca and P. Pugliese, "Mobile robot localization via EKF and UKF: A comparison based on real data," *Robot. Auton. Syst.* **74**(Part A), 122–127 (2015).

[22] W. B. Sebastian Thrun and D. Fox, *Probabilistic Robotics* (MIT Press, 2008) pp. 227–229.

[23] T. P. Kucner, M. Magnusson and A. J. Lilienthal, "Where am I? An NDT-based prior for MCL," **In:** *European Conference on Mobile Robots (ECMR)*, (2015) pp.1–6.

[24] L. Zhang, R. Zapata and P. Lepinay, "Self-adaptive Monte Carlo localization for mobile robots using range sensors," **In:** *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*, (2009) pp. 1541–1546.

[25] S. Du, T. Chen, Z. Lou and Y. Wu, "A 2D-LiDAR-based localization method for indoor mobile robots using correlative scan matching," *Robotica* **43**(2), 514–541 (2025).

[26] Q. Liu, X. Di and B. Xu, "Autonomous vehicle self-localization in urban environments based on 3D curvature feature points – Monte Carlo localization," *Robotica* **40**(3), 817–833 (2022).

[27] S. Julius Fusic and R. Sitharthan, "Self-adaptive learning particle swarm optimization-based path planning of mobile robot using 2D Lidar environment," *Robotica* 977–1000 (2024).

[28] C. Hao, L. Chengju and C. Qijun, "Self-localization in highly dynamicenvironments based on dual-channel unscented particle filter," *Robotica* **39**(7), 1216–1229 (2021).

[29] V. M. Olivera, J. M. C. Plaza and O. S. Serrano, "WiFi localization methods for autonomous robots," *Robotica* **24**(4), 455–461 (2006).

[30] R. Alqobali, M. Alshmrani, R. Alnasser, A. Rashidi, T. Alhmiedat and OMd Alia, "A survey on robot semantic navigation systems for indoor environments," *Appl. Sci.* **14**(1), 89 (2024).

[31] R. Alqobali, R. Alnasser, A. Rashidi, M. Alshmrani and T. Alhmiedat, "A real-time semantic map production system for indoor robot navigation," *Sensors* **24**(20), 6691 (2024).

https://doi.org/10.1017/S0263574725102518 Published online by Cambridge University Press

[32] O. Mozos, *Semantic Labeling of Places with Mobile Robots* (Springer Tracts in Advanced Robotics, 2008).
[33] E. Fazl-Ersi and J. K. Tsotsos, "Energy Minimization via Graph Cuts for Semantic Place Labeling," **In:** *IEEE/RSJ International Conference on Intelligent Robots and Systems*, (2010) pp. 5947–5952.
[34] A. Rottmann, Ó. M. Mozos, C. Stachniss and W. Burgard, "Semantic Place Classification of Indoor Environments with Mobile Robots Using Boosting," **In:** *Proceedings of the 20th National Conference on Artificial Intelligence*, vol. 3 (2005) pp. 1306–1311.
[35] W. Burger and M. J. Burge, "Scale-Invariant Feature Transform (SIFT)," **In:** *Digital Image Processing: An Algorithmic Introduction Using Java*, (2016) pp. 609–664.
[36] H. Bay, A. Ess, T. Tuytelaars and L. Van Gool, "Speeded-up robust features (SURF)," *Comput. Vis. Image. Und.* **110**(3), 346–359 (2008).
[37] Ö E. and I. Bozma, "Place Representation in Topological Maps Based on Bubble Space," **In:** *IEEE International Conference on Robotics and Automation*, (2012) pp. 3497–3502.
[38] L. Shi, S. Kodagoda and G. Dissanayake, "Application of Semi-Supervised Learning with Voronoi Graph for Place Classification, " **In:** *IEEE/RSJ International Conference on Intelligent Robots and Systems*, (2012) pp. 2991–2996.
[39] P. Ursic, M. Kristan, D. Skocaj and A. Leonardis, "Room Classification Using a Hierarchical Representation of Space," **In:** *IEEE/RSJ International Conference on Intelligent Robots and Systems*, (2012) pp. 1371–1378.
[40] T. Yeh and T. Darrell, "Dynamic Visual Category Learning," **In:** *IEEE Conference on Computer Vision and Pattern Recognition*, (2008) pp. 1–8.
[41] I. Kostavelis and A. Gasteratos, "Semantic maps from multiple visual cues," *Expert. Syst. Appl.* **68**, 45–57 (2017).
[42] A. Pronobis and P. Jensfelt, "Large-Scale Semantic Mapping and Reasoning with Heterogeneous Modalities," **In:** *IEEE International Conference on Robotics and Automation*, (2012) pp. 3515–3522.
[43] B. Kaleci, K. Turgut and H. Dutagaci, "2DLaserNet: A deep learning architecture on 2D laser scans for semantic classification of mobile robot locations," *Eng Sci. Technol. Int. J.* **28**, 101027 (2022).
[44] A. Ranganathan, "PLISS: Labeling places using online changepoint detection," *Auton. Robot.* **32**(4), 351–368 (2012).
[45] A. Chella, I. Macaluso and L. Riano, "Automatic Place Detection and Localization in Autonomous Robotics," **In:** *IEEE/RSJ International Conference on Intelligent Robots and Systems*, (2007) pp. 741–746.
[46] O. Erkent, K. Hakanand H. I. Bozma, "Hierarchically self-organizing visual place memory," *Adv. Robotics.* **31**(16), 865–879 (2017).
[47] R. Sibson, "SLINK: An optimally efficient algorithm for the single-link cluster method," *Comput. J.* **16**(1), 30–34 (1973).
[48] H. Guillaume, M. Dubois, E. Frenoux and T. Philippe, "Temporal Bag-of-Words - A Generative Model for Visual Place Recognition Using Temporal Integration," **In:** *Proceedings of the Sixth International Conference on Computer Vision Theory and Applications*, (2011) pp. 286–295.
[49] A. Rottmann, Ó.M. Mozos, C. Stachniss and W. Burgard, "Semantic Place Classification of Indoor Environments with Mobile Robots Using Boosting," **In:** *Proceedings of the 20th national conference on Artificial intelligence* - Volume 3, (2005) pp. 1306–1311.
[50] N. Sünderhauf, F. Dayoub, S. McMahon, B. Talbot, R. Schulz, P. Corke, G. Wyeth, B. Upcroft and M. Milford, "Place Categorization and Semantic Mapping on a Mobile Robot," **In:** *IEEE international conference on robotics and automation (ICRA)*, (2016). 5729–5736.
[51] R. W. M. Hendrikx, P. P. E.Torta, H. P. J. Bruyninckx and M. J. G. van de Molengraft, "Connecting Semantic Building Information Models and Robotics," **In:** *an application to 2D LiDAR-based localization. 2021 IEEE International Conference on Robotics and Automation (ICRA)*, (2021) pp. 11654–11660.
[52] L. Cui, C. chunyanJ. Huang, A. Rosendo and L. Kneip, "Monte-Carlo Localization in Underground Parking Lots Using Parking Slot Numbers," **In:** *IEEE/RSJ International Conference on Intelligent Robots and Systems*, (2021) pp. 2267–2274.
[53] N. Zimmerman, L. WiesmannT. Guadagnino, T. Läbe, J. Behley and C. Stachniss, "Robust Onboard Localization in Changing Environments Exploiting Text Spotting," **In:** *IEEE/RSJ International Conference on Intelligent Robots and Systems*, (2022) pp. 23–27.
[54] M. Khelifi and S. Moussaoui, "ReLM: An Efficient Reinforcement Learning-Based Localization Algorithm for Mobile Wireless Sensor Networks," **In:** *2021 International Conference on INnovations in Intelligent SysTems and Applications*, (2021) pp. 1–6.
[55] O. Dogru and R. C. B., "Huang, reinforcement learning with constrained uncertain reward function through particle filtering," *IEEE T. Ind. Electron.* **69**(7), 7491–7499 (2022).
[56] R. Yoshimura, I. Maruta, K. Fujimoto, K. Sato and Y. Kobayashi, "Particle filter design based on reinforcement learning and its application to mobile robot localization," *IEICE Trans. Inf. Syst.* **E105.D**(5), 1010–1023 (2022).
[57] A. Kendall and M. G. R. Cipolla, " PoseNet: A Convolutional Network for Real-Time 6-Dof Camera Relocalization," **In:** *IEEE International Conference on Computer Vision (ICCV)*, (2015) pp. 2938–2946.
[58] A. Kendall and R. Cipolla, "Modelling Uncertainty in Deep Learning for Camera Relocalization," **In:** *IEEE International Conference on Robotics and Automation (ICRA)*, (2016) pp. 4762–4769.
[59] F. Walch, C. Hazirbas, L. Leal-Taixé, T. Sattler, S. Hilsenbeck and D. Cremers. Image-based localization using LSTMs for structured feature correlation, (2016).
[60] R. Clark, S. W. A. Markham, N. Trigoni and H. Wen, "VidLoc: A Deep Spatio-Temporal Model for 6-Dof Video-Clip Relocalization," **In:** *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, (2017) pp. 2652–2660.
[61] I. Melekhov, J. Y. J. Kannala and E. Rahtu, "Image-Based Localization Using Hourglass Networks," **In:** *IEEE International Conference on Computer Vision Workshops (ICCVW)*, (2017) pp. 870–877.

https://doi.org/10.1017/S0263574725102518 Published online by Cambridge University Press

[62] J. Wu, L. Ma and X. Hu, "Delving Deeper into Convolutional Neural Networks for Camera Relocalization," **In:** *IEEE International Conference on Robotics and Automation (ICRA)*, (2017) pp. 5644–5651.
[63] R. C. A. Kendall, "Geometric Loss Functions for Camera Pose Regression with Deep Learning," **In:** *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, (2017) pp. 6555–6564.
[64] H. Jo and E. Kim, "New Monte Carlo localization using deep initialization: A three-dimensional liDAR and a camera fusion approach, (2020) 74485–74496.
[65] S. Xu, W. Chou and H. Dong, "A robust indoor localization system integrating visual localization aided by CNN-based image retrieval with Monte Carlo localization," *Sensors* **19**(2), 249 (2019).
[66] S. Yu, F. Yan, Y. Zhuang, D. Gu, S. Xu, W. Chou and H. Dong, "A robust indoor localization system integrating visual localization aided by CNN-based, image retrieval with Monte Carlo localization," *Sensors* **19**(2), 249 (2019).
[67] Z. Wang, F. Gao, Y. Zhao, Y. Yin and L. Wang, "A deep hierarchical framework for robot global localization," *J. Intell. Robot. Syst.* **106**(2), 46 (2022).
[68] A. Alotaibi, H. Alatawi, A. Binnouh, L. Duwayriat, T. Alhmiedat and O. M. Alia, "Deep learning-based vision systems for robot semantic navigation: An experimental study," *Technologies* **12**(9), 157 (2024).
[69] M. Kok, A. Solin and T. B. Schon, "Rao-Blackwellized particle smoothing for simultaneous localization and mapping," *Data-Centric Eng.* **5**, e15 (2024).
[70] W. Hess, D. Kohler, H. Rapp and D. Andor, "Real-Time Loop Closure in 2D LIDAR SLAM," **In:** *IEEE International Conference on Robotics and Automation (ICRA)*, (2016) pp. 1271–1278.
[71] W. Shiguang and W. Chengdong, "An Improved FastSLAM2.0 Algorithm Using Kullback-Leibler Divergence," **In:** *4th International Conference on Systems and Informatics (ICSAI)*, (2017) pp. 225–228.
[72] K. Konolige, G. Grisetti, R. Kümmerle, W. Burgard, B. Limketkai and R. Vincent, "Efficient Sparse Pose Adjustment for 2D Mapping," **In:** *IEEE/RSJ International Conference on Intelligent Robots and Systems*, (2010) pp. 22–29.
[73] C. Cadena, L. Carlone, H. Carrillo, Y. Latif, D. Scaramuzza, J. Neira, I. Reid and J. J. Leonard, "Past, present, and future of simultaneous localization and mapping: Toward the robust-perception age," *IEEE T. Robot.* **32**(6), 1309–1332 (2016).
[74] S. Kim, "Robust corner detection by image-based direct curvature field estimation for mobile robot navigation," *Int. J. Adv. Robot. Syst.* **9**(5), 187 (2012).
[75] M. Liu, W. T. Kao and T. Delbruck, "Live Demonstration: A Real-Time Event-Based Fast Corner Detection Demo Based on FPGA," **In:** *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, (2019) pp. 1678–1679.
[76] T. Meng, C. Fu, M. Huang, X. Wang, J. He, T. Huang and W. Shi, "Localization-Guided Track: A Deep Association Multi-Object Tracking Framework Based On Localization Confidence of Detections," **In:** *Computer Vision and Pattern Recognition*, (2023).
[77] H. Bay, T. Tuytelaars and L. Van Gool, "SURF: Speeded Up Robust Features," **In:** *Computer Vision – ECCV*, (2006) pp. 404–417.
[78] E. Rublee, V. Rabaud, K. Konolige and G. Bradski, "ORB: An Efficient Alternative to SIFT or SURF," **In:** *2011 International Conference on Computer Vision*, (2011) pp. 2564–2571.
[79] A. J. Davison, I. D. Reid, N. D. Molton and O. Stasse, "MonoSLAM: Real-time single camera SLAM," *IEEE T. Pattern. Anal.* **29**(6), 1052–1067 (2007).
[80] G. Klein and D. Murray,"Parallel Tracking and Mapping for Small AR Workspaces," **In:** *6th IEEE and ACM International Symposium on Mixed and Augmented Reality*, (2007) pp. 225–234.
[81] R. Mur-Artal, J. M. M. Montiel and J. D. Tardós, "ORB-SLAM: A versatile and accurate monocular SLAM system," *IEEE Trans. Robot.* **31**(5), 1147–1163 (2015).
[82] D. Galvez-López and J. D. Tardos, "Bags of binary words for fast place recognition in image sequences," *IEEE Trans. Robot.* **28**(5), 1188–1197 (2012).
[83] R. Mur-Artal and J. Tardos, "ORB-SLAM2: An open-source SLAM system for monocular, stereo and RGB-D cameras," *IEEE Tans. Robot.* **33**(5), 1255–1262 (2017).
[84] C. Campos, R. Elvira, J. J. G. Rodríguez, J. M. M. Montiel and J. D. Tardós, "ORB-SLAM3: An accurate open-source library for visual, visual–inertial, and multimap SLAM," *IEEE Trans. Robot.* **37**(6), 1874–1890 (2021).
[85] E. Perdices, L. M. López and J. M. Cañas. LineSLAM: Visual Real Time Localization Using Lines and UKF. **In:** *ROBOT2013: First Iberian Robotics Conference: Advances in Robotics*, Vol. 1, (2014) pp. 663–678.
[86] R. Gomez-Ojeda, F. A. Moreno, D. Zuñiga-Noël, D. Scaramuzza and J. Gonzalez-Jimenez, "PL-SLAM: A stereo slam system through the combination of points and line segments," *IEEE Trans. Robot.* **35**(3), 734–746 (2019).
[87] A. Pumarola, A. Vakhitov, A. Agudo, A. Sanfeliu and F. Moreno-Noguer, "PL-SLAM: Real-Time Monocular Visual SLAM with Points and Lines," **In:** *IEEE International Conference on Robotics and Automation (ICRA)*, (2017) pp. 4503–4508.
[88] R. A. Newcombe, S. J. Lovegrove and A. J. Davison, "DTAM: Dense Tracking and Mapping in Real-Time," **In:** *International Conference on Computer Vision*, (2011) pp. 2320–2327.
[89] J. Engel, T. Schöps and D. Cremers, "LSD-SLAM: Large-Scale Direct Monocular SLAM," **In:** *Computer Vision – ECCV. 2014. ECCV. 2014. Lecture Notes in Computer Science*, (2014) pp. 834–849.
[90] C. Forster, M. Pizzoli and D. Scaramuzza, "SVO: Fast Semi-Direct Monocular Visual Odometry," **In:** *IEEE International Conference on Robotics and Automation (ICRA)*, (2014) pp. 15–22.
[91] J. Engel, V. Koltun and D. Cremers, "Direct sparse odometry," *IEEE Trans. Pattern. Anal. Mach. Intell.* **40**(3), 611–625 (2018).
[92] J. Zhang and S. Singh, "LOAM : Lidar Odometry and Mapping in Real-Time," **In:** *Robotics: Science and Systems Conference (RSS)*, (2014) pp. 109–111.

https://doi.org/10.1017/S0263574725102518 Published online by Cambridge University Press

[93] T. Shan and B. Englot, LeGO-LOAM: Lightweight and ground-optimized lidar odometry and mapping on variable terrain, (2018) 4758–4765.
[94] J. Zhang and S. Singh, "Visual-Lidar Odometry and Mapping: Low-Drift, Robust, and Fast," **In:** *IEEE International Conference on Robotics and Automation (ICRA)*, (2015) pp. 2174–2181.
[95] J. Zhang and S. Singh, "Laser-visual-inertial odometry and mapping with high robustness and low drift," *J. Field. Robot.* **35**(8), 1–23 (2018).
[96] H. Ye, Y. Chen and M. Liu, "Tightly Coupled 3d Lidar Inertial Odometry and Mapping," **In:** *International Conference on Robotics and Automation (ICRA)*, (2019) pp. 3144–3150.
[97] T. Shan, B. Englot, D. Meyers, W. Wang, C. Ratti and D. Rus, "LIO-SAM: Tightly-Coupled Lidar Inertial Odometry via Smoothing and Mapping," **In:** *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, (2020) pp. 5135–5142.
[98] W. Xu and F. Zhang, "FAST-LIO: A fast, robust lidar-inertial odometry package by tightly-coupled iterated Kalman filter," *IEEE Robot. Autom. Lett.* **6**(2), 3317–3324 (2021).
[99] W. Xu, Y. Cai, D. He, J. Lin and F. Zhang, "FAST-LIO2: Fast direct lidar-inertial odometry," *IEEE Trans. Robot.* **38**(4), 2053–2073 (2022).
[100] Y. Jia, H. Luo, F. Zhao, G. Jiang, Y. Li, J. Yan, Z. Jiang and Z. Wang, "Lvio-Fusion: A Self-Adaptive Multi-Sensor Fusion Slam Framework Using Actor-Critic Method," **In:** *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, (2021) pp. 286–293.
[101] T. Shan, B. Englot, C. Ratti and D. Rus, LVI-SAM: Tightly-coupled lidar-visual-inertial odometry via smoothing and mapping, (2021).
[102] T. Alhmiedat, A. M. Marei, W. Messoudi, S. Albelwi, A. Bushnag, Z. Bassfar, F. Alnajjar and A. O. Elfaki, "A SLAM-based localization and navigation system for social robots: The pepper robot case," *Machines* **11**(2), 158 (2023).
[103] S. Alamri, H. Alamri, W. Alshehri, S. Alshehri, A. Alaklabi and T. Alhmiedat, "An autonomous maze-solving robotic system based on an enhanced wall-follower approach," *Machines* **11**(2), 249 (2023).
[104] W. Guan, L. Huang, S. Wen, Z. Yan, W. Liang, C. Yang and Z. Liu, "Robot localization and navigation using visible light positioning and SLAM fusion," *J. Lightwave Technol.* **39**(22), 7040–7051 (2021).
[105] J. Yin, L. Carlone, S. Rosa and B. Bona, "Graph-Based Robust Localization and Mapping for Autonomous Mobile Robotic Navigation, " **In:** *IEEE International Conference on Mechatronics and Automation*, (2014) pp. 1680–1685.
[106] B. Liu, L. Zuo, C. H. Zhang and Y. Liu, "An Approach to Graph-Based Grid Map Segmentation for Robot Global Localization," **In:** *2018 IEEE International Conference on Mechatronics and Automation (ICMA)*, (2018) pp. 1812–1817.
[107] L. Yao and F. Li, "Mobile robot localization based on vision and multisensor," *J. Robot.* **2020**(1), 8701619 (2020).
[108] Z. Jia and J. Leng. Mobile Robot Vision Odometer Based on Point- Line Features and Graph Optimization," **In:** *Chinese Control And Decision Conference (CCDC)*, (2018) pp. 3398–3403.
[109] Y. Li, S. Li, Y. Tan and J. Hu, "SLAM of Mobile Robot Based on The Joint Optimization of LiDAR and Camera," **In:** *7th International Conference on Control and Robotics Engineering (ICCRE)*, (2022) pp. 21–25.
[110] Y. Li, S. Zhu, Y. Yu and Z. Wang, "An improved graph-based visual localization system for indoor mobile robot using newly designed markers," *Int. J. Adv. Robot. Syst.* **15**(2), 172988141876919 (2018).
[111] D. Wilbers, C. Merfels and C. Stachniss, "A Comparison of Particle Filter and Graph-Based Optimization for Localization with Landmarks in Automated Vehicles," **In:** *Third IEEE International Conference on Robotic Computing (IRC)*, (2019) pp. 220–225.
[112] S. Wang, L. Chen, D. Gu and H. Hu, "Single Beacon Based Multi-Robot Cooperative Localization Using Moving Horizon Estimation," **In:** *IEEE International Conference on Robotics and Automation (ICRA)*, (2014) pp. 1625–1630.
[113] W. He, Y. Sun, H. Zhu and A. Liu, "Ultra-Wideband Localization of Mobile Robots Based on Moving Horizon Optimization," **In:** *International Conference on Advanced Robotics and Mechatronics (ICARM)*, (2023) pp. 457–463.
[114] S. Sajjadi, J. Bittick, F. Janabi-Sharifi and I. Mantegh, "A Robust and Adaptive Sensor Fusion Approach for Indoor UAV Localization," **In:** *International Conference on Unmanned Aircraft Systems (ICUAS)*, (2023) pp. 441–447.
[115] Y. Kishimoto, K. Takaba and A. Ohashi, "Moving Horizon Multi-Robot SLAM Based on C/GMRES Method," **In:** *International Conference on Advanced Mechatronic Systems (ICAMechS)*, (2019) pp. 22–27.
[116] M. Osman, M. W. Mehrez, M. A. Daoud, A. Hussein, S. Jeon and W. Melek, "A generic multi-sensor fusion scheme for localization of autonomous platforms using moving horizon estimation," *Trans. Inst. Meas. Control.* **43**(15), 3413–3427 (2021).
[117] T. Kasahara, K. Tsuno, K. Nonaka and K. Sekiguchi, "Comparative Experiments of Moving Horizon Estimation Based SLAM in Indoor Environment," **In:** *12th Asian Control Conference (ASCC)*, (2019) pp. 1131–1136.
[118] F. Andrade, M. LLofriu, M. M. Tanco, G. T. Barnech and G. Tejera, "Active Localization for Mobile Service Robots in Symmetrical and Open Environments," **In:** *Latin American Robotics Symposium (LARS), 2021 Brazilian Symposium on Robotics (SBR), and 2021 Workshop on Robotics in Education (WRE)*, (2021) pp. 270–275.
[119] G. Ge, Y. Zhang, W. Wang, Q. Jiang, L. Hu and Y. Wang, "Text-MCL: Autonomous mobile robot localization in similar environment using text-level semantic information," *Machines* **10**(3), 169 (2022).
[120] Z. Wu, W. Wang, H. Zhang, J. Zhang, Q. Lyu and D. Wang, "Magnetic Field-Aided Global Localization in Repetitive Environments," **In:** *IEEE International Conference on Cybernetics and Intelligent Systems (CIS) and IEEE Conference on Robotics, Automation and Mechatronics (RAM)*, (2023) pp. 1–6.
[121] S. Lafuente-Arroyo, S. Maldonado-Bascón, D. Delgado-Mena, C. Gutiérrez-Álvarez and F. J. Acevedo-Rodríguez, "Multisensory integration for topological indoor localization of mobile robots in complex symmetrical environments," *Expert. Syst. Appl.* **241**, 122561 (2023).

https://doi.org/10.1017/S0263574725102518 Published online by Cambridge University Press

[122] S. Yu, F. Yan, Y. Zhuang and D. Gu, "A deep-learning-based strategy for kidnapped robot problem in similar indoor environment," *J. Intell. Robot. Syst.* **100**(3-4), 765–775 (2020).
[123] D. Fox, W. Burgard, H. Kruppa and S. Thrun, "A probabilistic approach to collaborative multi-robot localization," *Auton. Robot.* **8**(3), 325–344 (2000).
[124] D. Delgado-Mena, S. Lafuente-Arroyo, S. Maldonado-Bascón and J. A. Castro-Garcıa,"An Approach for Global Localization in Complex Indoor Environments Based on Topological Maps," **In:** *Proceedings of V XoveTIC Conference*, XoveTIC, (2022) pp. 27–30.
[125] G. Ge, Z. Qin and L. Fan, "An improved VSLAM for mobile robot localization in corridor environment," *Adv. Multimed.* **2022**, 1–10 (2022).
[126] J. Laaksonen and V. Kyrki, "Localization in ambiguous environments using multiple weak cues," *Intel. Serv. Robot.* **1**(4), 281–288 (2008).
[127] F. Fraundorfer, H. Bischof and S. Ober, "Natural, salient image patches for robot localization," Pattern Recognition. 881–884 (2004).
[128] D. Zhang, R. Kurazume, Y. Iwashita and T. Hasegawa, "Robot Localization Under Perceptual Aliasing Conditions Based on Laser Reflectivity Using Particle Filter," **In:** *IEEE/SICE International Symposium on System Integration (SII)*, (2011) pp.90–95.
[129] P. Zingaretti and E. Frontoni. Vision and Sonar Sensor Fusion for Mobile Robot Localization in Aliased Environments," **In:** *2nd IEEE/ASME International Conference on Mechatronics and Embedded Systems and Applications*, (2006) pp. 1–6.
[130] G. Ge, Y. Zhang, W. Wang, L. Hu, Y. Wang and Q. Jiang, "Visual-feature-assisted mobile robot localization in a long corridor environment," *Front. Inf. Technol. Electron. Eng.* **24**(6), 876–889 (2023).
[131] G. Peng, W. Zheng, Z. Lu, J. Liao, L. Hu, G. Zhang, D. He and J. Zhong, "An improved AMCL algorithm based on laser scanning match in a complex and unstructured environment," *Complexity*, **2018**(1), 1–11 (2018).
[132] L. Ronghua, H. Bingrong and L. Maohai, "Coevolution-based adaptive Monte Carlo localization," Cutting Edge Robotics (2005).279.
[133] A. Hossain and R. Chowdhury, "Implementation of Hector SLAM Algorithm for Mapping Indoor Environments with Obstacles," **In:** *IEEE 3rd International Conference on Robotics, Automation, Artificial-Intelligence and Internet-of-Things*, (2024) pp. 218–223.
[134] E. H. C. Harik and A. Korsaeth, "Combining Hector SLAM and artificial potential field for autonomous navigation inside a greenhouse," *Robotics* **7**(2), 22 (2018).
[135] J. Xue, Z. Cheng, H. Li, P. Yang, J. Wei and C. Chen, "Sensor Fusion SLAM: An Efficient and Robust Slam System for Dynamic Environments," **In:** *IEEE 25th China Conference on System Simulation Technology and its Application (CCSSTA)*, (2024) pp. 358–365.
[136] Z. Zheng, M. Xie, C. Jiang, H. Bian and W. Wang,"Indoor Mobile Robot Map Construction Based on Improved Cartographer Algorithm," **In:** *WRC Symposium on Advanced Robotics and Automation (WRC SARA)*, (2024) pp. 470–475.
[137] P. Chen, X. Zhao, L. Zeng, L. Liu, S. Liu, L. Sun, Z. Li, H. Chen, G. Liu, Z. Qiao, Y. Qu, D. Xu, L. Li and L. Li, "A Review of research on slam technology based on the fusion of LiDAR and vision," *Sensors* 1447 (2025).
[138] X. Wu, P. Li, Q. Li and Z. Li, "Two-dimensional-simultaneous localisation and mapping study based on factor graph elimination optimisation,"*Sustainability* **15**(2), 1172 (2023).
[139] B. Bodin, L. Nardi, H. Wagstaff, P. H. J. Kelly and M. O. Boyle, "Algorithmic Performance-Accuracy Trade-Off in 3D Vision Applications," **In:** *IEEE International Symposium on Performance Analysis of Systems and Software (ISPASS)*, (2018) pp. 123–124.
[140] P. Roy and C. Chowdhury, "A survey of machine learning techniques for indoor localization and navigation systems," *J. Intell. Robot. Syst.* **101**(3), 63 (2021).

Cite this article: S. A. Melvin, H. Wang and A. Yazdani (2026). "Global localization for mobile robots in symmetrical indoor environments: a review, practical challenges, and experimental validation", Robotica **44**, 343–385. https://doi.org/10.1017/S0263574725102518

https://doi.org/10.1017/S0263574725102518 Published online by Cambridge University Press