Transactions on Computational and Scientific Methods | Vo. 5, No. 11, 2025
ISSN: 2998-8780
https://pspress.org/index.php/tcsm
Pinnacle Science Press

Background, motivation, and the evolution of autonomous mobile robot systems

Ansel Rowden
Lakehead University, Thunder Bay, Canada
anselr983@lakehead.ca

Abstract: Autonomous mobile robots (AMRs) have emerged as a central technology in modern robotics, integrating perception, localization, mapping, planning, and control to enable robust autonomous navigation in complex and dynamic environments. Recent advances in multi-modal sensing, deep-learning-based perception, optimization-driven and learning-enhanced planning, and hierarchical control architectures have significantly expanded the capabilities and deployment scope of AMRs across industrial logistics, manufacturing, healthcare, agriculture, service robotics, and urban mobility. This survey provides a comprehensive review of the algorithmic foundations, system architectures, and practical applications that define contemporary AMR research. We examine the evolution of perception pipelines from geometric methods to multimodal and semantic understanding, analyze major developments in SLAM, including graph-based optimization and learning-augmented pipelines, and discuss global and local planning frameworks encompassing heuristic search, sampling-based algorithms, trajectory optimization, and reinforcement learning. We further investigate control techniques ranging from classical nonlinear control and MPC to safety-critical and hybrid learned controllers. In addition, we highlight integration challenges, real-world deployment issues, and emerging research directions such as lifelong autonomy, human-aware navigation, multi-robot collaboration, and the rise of foundation models for robotics. This survey aims to provide a unified perspective that can guide researchers and practitioners in advancing next-generation AMR systems capable of reliable long-term operation in real-world settings.

Keywords: Autonomous mobile robots, perception, SLAM, localization, motion planning, trajectory optimization, robot control, reinforcement learning, multi-sensor fusion, semantic mapping.

1. Introduction

Autonomous mobile robots (AMRs) have become one of the most influential research frontiers in robotics, integrating perception, localization, mapping, planning, and control into a unified autonomous system capable of operating without continuous human intervention. Over the past two decades, advances in sensor technology, deep learning, probabilistic robotics, and embedded computation have significantly changed the landscape of mobile robotics, enabling robots to navigate complex, dynamic, and unstructured environments with high degrees of reliability and adaptability. From industrial logistics and warehouse automation to autonomous driving, planetary exploration, household service robots, and emergency response applications, the capabilities of AMRs now extend across nearly every domain requiring real-world mobility and decision-making. As global demand for intelligent automation grows, understanding the algorithmic foundations, system architectures, and emerging developments in autonomous navigation is vital for robotics researchers and practitioners.

In contemporary AMR systems, perception serves as the foundational layer, enabling robots to capture, interpret, and reason about their surrounding environments through a combination of sensors such as LiDAR, stereo or monocular cameras, inertial measurement units (IMUs), radar, and ultrasonic modules. Classic methods for perception relied heavily on geometric modeling, feature extraction, and handcrafted algorithms, including Harris corners, SIFT, SURF, occupancy grid mapping, and scan matching. However, the past decade has seen a decisive shift toward deep-learning-based perception frameworks, particularly convolutional neural networks (CNNs), Transformer architectures, and multimodal fusion networks. These models demonstrate enhanced performance in depth estimation, semantic segmentation, object detection, motion prediction, and environment understanding, which in turn strengthen downstream modules such as simultaneous localization and mapping (SLAM) and motion planning. Meanwhile, multi-sensor fusion techniques, including extended Kalman filters (EKF), particle filters, and factor graph optimization, further improve the robustness and accuracy of perception by leveraging complementary modalities.

Following perception, AMRs rely on robust localization and mapping techniques to build internal representations of complex environments. SLAM has evolved from traditional LiDAR-based scan matching and graph optimization to visual SLAM, RGB-D SLAM, semantic SLAM, and deep-learning-enhanced SLAM methods. Classical systems such as EKF-SLAM and GraphSLAM laid the theoretical foundation, while modern solutions such as ORB-SLAM2, VINS-Mono, LOAM, LIO-SAM, DSO, and DeepFactors reflect advancements in both geometric and learned approaches. These systems enable AMRs to achieve real-time state estimation and map construction even under challenging conditions, such as low texture, motion blur, dynamic objects, and severe illumination variation. As AMRs expand into open and dynamic environments such as urban streets and crowded public spaces, semantic layers in mapping-identifying road features, obstacles, drivable regions, and interactive agents-become increasingly important for safe and predictable navigation.

Another essential component of autonomous mobility is path planning, which includes global planning, local planning, and motion generation. Classical global planning algorithms such as Dijkstra, A*, and their variants provide deterministic guarantees, while sampling-based methods such as Rapidly-Exploring Random Trees (RRT), RRT*, and Probabilistic Roadmaps (PRM) offer increased flexibility in high-dimensional configuration spaces. Local planning frameworks, including the Dynamic Window Approach (DWA), Timed Elastic Band (TEB), model predictive control (MPC), and optimization-based trajectory generation, enable robots to react quickly to dynamic obstacles and real-time sensor inputs. In recent years, learning-based planning methods such as reinforcement learning (RL), imitation learning (IL), and end-to-end neural motion generation have gained traction, allowing AMRs to adaptively learn navigation behaviors directly from data while incorporating safety and generalization constraints.

Finally, control systems provide the operational backbone that translates planned trajectories into executable motor commands. Traditional control approaches-including PID control, kinematic controllers for differential-drive robots, and linear/nonlinear MPC-remain foundational. However, with the rise of advanced actuators, highly dynamic platforms, and uncertain environments, AMRs increasingly adopt hybrid control architectures that combine classical control theory with machine-learning-based modules. These approaches include neural adaptive control, robust control under uncertainty, model-free reinforcement learning controllers, and hierarchical control structures that balance stability, responsiveness, and policy generalizability. As AMRs become more widely deployed in complex real-world environments, ensuring stability, interpretability, and safety in control policies is becoming a major research priority.

Despite significant progress, several challenges remain unresolved in AMR research. Robots must operate reliably under sensor degradation, dynamic obstacles, unpredictable human behavior, long-term autonomy requirements, and computational constraints. Real-world deployment also raises issues in scalability, multi-robot coordination, safety certification, environmental understanding, and generalization across environments. Emerging technologies-such as edge computing, large vision-language models, foundation models for robotics, diffusion-based motion planners, and energy-efficient computing-continue to reshape the research landscape and introduce new possibilities as well as challenges. With AMRs expected to play a central role in industrial automation, smart cities, intelligent transportation, and human − robot collaboration, a comprehensive synthesis of the current state of perception, planning, and control is both timely and essential.

In this survey, we provide an extensive and integrated overview of core algorithmic frameworks enabling autonomous mobile robots, spanning perception, SLAM, planning, and control. We trace their historical development, discuss the latest advances in deep-learning-enhanced methods, compare classical and modern solutions, and identify open challenges and future research directions. By integrating foundational theories with emerging breakthroughs, this review aims to offer a unified and systematic perspective that can guide researchers, engineers, and practitioners in advancing the design, deployment, and understanding of autonomous mobile robots.

2. Perception for Autonomous Mobile Robots

Perception forms the core of autonomous mobile robot (AMR) intelligence, enabling robots to interpret their surroundings, recognize obstacles, estimate motion, and extract actionable information essential for safe navigation. Modern perception systems integrate geometric modeling, probabilistic inference, and deep learning to transform raw sensor data into structured environmental understanding. At the hardware level, AMRs commonly deploy multimodal sensor suites combining LiDAR, cameras, IMUs, wheel odometry, radar, GPS, and ultrasonic modules, each offering complementary advantages. LiDAR sensors provide accurate range measurements and are robust to illumination changes, making them suitable for structured mapping and obstacle detection, particularly in self-driving systems and industrial robots. Cameras, on the other hand, offer rich semantic information at low cost, fueling the development of visual navigation and scene understanding. IMUs provide high-frequency motion cues for inertial navigation, while radar systems support reliable perception under adverse weather conditions. Fusing these heterogeneous signals through filtering, optimization, or deep-learning-based frameworks is essential for reducing noise, compensating for individual weaknesses, and enabling robust operation in unstructured or dynamic environments.

Traditional perception algorithms relied heavily on geometric and feature-based representations. Early mobile robots used edge detection, corner extraction, optical flow, and handcrafted 3D descriptors to detect landmarks and track motion. Methods such as Harris corners, SIFT, and SURF were widely used for robust feature matching, while occupancy grid mapping and scan matching enabled robots to construct consistent world models under uncertainty. Probabilistic frameworks like Bayesian filtering, Markov localization, extended Kalman filters (EKF), and particle filters provided robots with systematic approaches for handling noise and ambiguity in sensor data. These techniques formed the backbone of early SLAM systems and are still integral in low-power or real-time pipelines. However, purely geometric perception methods tend to degrade in environments with dynamic elements, weak textures, reflective surfaces, or extreme lighting conditions, creating a need for more adaptive and data-driven perception models.

With the rapid rise of deep learning, perception in AMRs has undergone a major transformation. Convolutional neural networks (CNNs), Transformer-based architectures, and multimodal fusion models now dominate the perception pipeline. State-of-the-art object detection frameworks such as Faster R-CNN [1], YOLOv7 [2], and DETR [3] have shown strong performance in real-time classification and localization of pedestrians, vehicles, and obstacles. Deep-learning-based semantic segmentation models-such as DeepLabv3+ [4], HRNet [5], and SegFormer [6]-enable AMRs to extract pixel-level understanding of drivable regions, sidewalks, curbs, and dynamic objects. Visual odometry and depth estimation have similarly benefitted from learning-based methods including Monodepth2 [7], RAFT-Stereo [8], and DROID-SLAM [9], which combine geometric constraints with learned feature representations. These advancements allow AMRs to achieve robust perception even in challenging scenes involving shadows, reflections, poor textures, crowded areas, or rapid motion.

A major trend in recent research is multimodal perception, where AMRs fuse LiDAR, vision, IMU, and radar signals to achieve more comprehensive environmental awareness. Classical fusion approaches include EKF-based fusion, loosely coupled LiDAR – camera systems, and tightly coupled visual – inertial odometry (VIO) frameworks. More advanced systems such as LIO-SAM [10], VINS-Mono [11], and FAST-LIO [12] integrate inertial preintegration, factor graph optimization, and incremental smoothing to improve long-term consistency. Deep-learning-based fusion frameworks-such as LiDAR-camera transformers, BEVFusion [13], and DeepFusion [14]-have further enhanced the ability of AMRs to reason about 3D structure and scene semantics by projecting sensor information into unified bird ’ s-eye-view (BEV) representations. These multimodal approaches significantly outperform single-sensor methods, especially in degraded visual environments, cluttered scenes, or high-speed motion.

Another emerging direction is semantic and instance-level perception, which extends beyond geometric understanding to incorporate high-level scene context. Semantic SLAM systems [15] integrate object detection and segmentation into the mapping process, allowing AMRs to identify meaningful structures such as walls, vehicles, furniture, and humans. The integration of semantics improves map readability, helps avoid dynamic obstacles, and reduces drift by introducing object-level constraints into optimization. Similarly, 4D perception-perceiving space and time jointly-enables robots to predict the motion of dynamic agents, estimate future trajectories, and plan safe actions in rapidly changing environments. Transformer-based temporal models, recurrent networks, and diffusion-based motion predictors are increasingly used to anticipate human motion patterns, crowd flows, and vehicle trajectories in dense urban environments.

Furthermore, AMR perception must be robust against real-world disturbances including sensor noise, calibration errors, occlusion, dynamic agents, and environmental variability. Techniques such as self-supervised learning, domain adaptation, and uncertainty modeling enable deep networks to generalize across lighting, weather, and seasonal changes. Continual learning and long-term autonomy frameworks allow AMRs to update their perception models over time while avoiding catastrophic forgetting. Safety-critical perception research focuses on designing fail-safe redundancy layers that guarantee minimal environmental understanding even in the presence of sensor degradation or partial system failures. These efforts are particularly important for autonomous vehicles, search-and-rescue robots, and service robots deployed in human-centered environments.

Looking forward, the convergence of large multimodal models, foundation models for robotics, and multi-sensor 4D scene representation is expected to redefine perception pipelines in AMRs. Vision – language models such as CLIP, BLIP-2, and recent robotics-oriented Transformer models promise to bring high-level semantic reasoning and open-vocabulary scene understanding into AMR platforms, enabling robots to interpret scenes in ways previously limited to human cognition. Meanwhile, energy-efficient perception algorithms, event-based vision sensors, high-resolution LiDAR, and neuromorphic hardware will continue to shape the evolution of mobile robot sensing. As AMRs become increasingly autonomous and operate in diverse real-world environments, perception will remain a central research challenge that deeply influences the performance of SLAM, planning, and control.

3. SLAM and Localization for Autonomous Mobile Robots

Simultaneous Localization and Mapping (SLAM) represents one of the most fundamental and extensively studied components within the architecture of autonomous mobile robots. SLAM enables robots to estimate their pose while simultaneously constructing a consistent representation of the environment, allowing long-term navigation in previously unknown spaces without reliance on external localization infrastructure. The SLAM problem lies at the intersection of probabilistic inference, geometric reconstruction, optimization theory, and real-time systems engineering. Over the past four decades, SLAM has evolved from classical filtering-based approaches to modern graph-based optimization, dense reconstruction, semantic mapping, and learning-enhanced pipelines. As AMRs expand into dynamic, unstructured, and large-scale environments, SLAM systems must meet increasingly demanding requirements for accuracy, robustness, computational efficiency, and semantic richness. These advancements reflect the continuous interplay between algorithmic theory, sensor evolution, and computational capabilities, which collectively shape the modern landscape of robot autonomy.

Early SLAM solutions were primarily rooted in probabilistic robotics, where the environment was represented using sparse landmarks and robot motion was modeled through Bayesian filtering frameworks. Extended Kalman Filter (EKF)-based SLAM was a dominant paradigm in the 1990s and early 2000s, where states-including robot pose and landmark positions-were embedded in a single state vector updated through nonlinear motion and measurement models [16]. EKF-SLAM provided real-time performance and theoretical grounding but suffered from well-known issues such as linearization errors, quadratic computational complexity with respect to map size, and limited scalability in complex environments. Particle filter approaches, including FastSLAM [17], attempted to alleviate some of these limitations by factorizing the SLAM posterior into robot trajectory particles and independent landmark estimators, enabling better scalability. However, particle depletion, high variance, and inefficiency in high-dimensional spaces restricted their applicability to large-scale deployment. Despite limitations, these filtering methods established the probabilistic foundation upon which subsequent SLAM solutions were built.

The modern era of SLAM was largely shaped by the introduction of graph-based SLAM and optimization-based formulations. In graph SLAM, robot poses are modeled as nodes within a graph, and constraints arising from odometry, loop closures, and landmark observations form edges. Solving SLAM becomes equivalent to finding the configuration of nodes that best satisfies all constraints, typically through nonlinear least-squares optimization. Pioneering frameworks such as g2o [18], TORO, and Ceres Solver enabled efficient sparse optimization through techniques such as incremental smoothing, sparse matrix factorization, and Gauss-Newton optimization. This representation offered superior scalability, accuracy, and robustness to drift compared to filtering approaches. The introduction of pose graph optimization allowed robots to incorporate loop closures-recognizing previously visited locations-to correct accumulated drift and maintain long-term consistency. This capability is essential in large indoor environments, urban streets, and multi-floor buildings where global consistency is critical.

Visual SLAM (V-SLAM) marked another transformative milestone, enabled by the proliferation of low-cost cameras and advances in computer vision. PTAM [19] demonstrated the feasibility of real-time parallel tracking and mapping on a single camera, leading to a wave of feature-based V-SLAM systems such as ORB-SLAM, ORB-SLAM2, and ORB-SLAM3 [20]. These systems rely on robust keypoint extraction, feature matching, epipolar geometry, and bundle adjustment to achieve accurate camera pose estimation. Visual SLAM offers high information density, enabling robots to navigate with consumer-grade hardware, making it widely adopted in drones, mobile robots, AR devices, and household service robots. However, purely visual systems remain sensitive to lighting, motion blur, textureless surfaces, and dynamic objects, motivating the development of fusion pipelines such as Visual – Inertial SLAM (VIO). Systems like VINS-Mono [11], OKVIS, and ROVIO integrate high-frequency IMU measurements with camera observations to improve robustness under aggressive motion and low-light conditions.

LiDAR-based SLAM continued to evolve in parallel, driven by its reliability and geometric precision. Classic LiDAR SLAM frameworks such as Hector SLAM and GMapping focused on 2D grid-based mapping for indoor robots, while modern 3D LiDAR SLAM systems-including LOAM [21], LIO-SAM [10], FAST-LIO [12], and Cartographer [22]-leveraged scan registration, feature extraction, and pose graph optimization to construct detailed 3D maps suitable for autonomous driving and UAVs. LiDAR SLAM excels in scenarios where environmental features are consistent across illumination changes and where precise depth measurements are required. The shift from point-to-point ICP to feature-based or surfel-based registration significantly improved real-time performance and accuracy in large-scale outdoor environments. The growing accessibility of solid-state LiDAR and mechanical spinning LiDAR further stimulated research in robust multi-layer and multi-return processing, enabling better interpretation of vegetation, reflective surfaces, and sloped terrains.

In recent years, SLAM research has undergone another major transition through the integration of deep learning. Learning-based SLAM approaches enhance traditional geometry pipelines by using neural networks to predict depth, optical flow, pose, or uncertainty directly from images. Systems such as DSO, DeepFactors, and DROID-SLAM [9] combine geometric optimization with learned feature encoders and motion priors, achieving improved performance in low-texture or repetitive environments. End-to-end SLAM methods push this trend further, attempting to jointly learn feature extraction, data association, and pose estimation. Although still challenged by generalization and interpretability issues, learning-based SLAM represents a significant shift toward flexible and semantically informed navigation. Semantic SLAM approaches explicitly incorporate instance segmentation, object detection, and scene semantics into the mapping process, producing rich 3D maps that identify drivable regions, walls, doors, vehicles, and humans [15]. These semantic-rich maps improve high-level reasoning, enable object-aware motion prediction, and support task-oriented navigation in human-centered environments.

Another significant research direction lies in multi-robot SLAM, where multiple robots collaboratively estimate their trajectories and construct shared maps. Collaborative SLAM systems explore decentralized optimization, communication-efficient map merging, and coordination strategies that enable large robot fleets to share the same representation of an environment. Such systems are essential for warehouse robotics, search-and-rescue missions, and persistent environmental monitoring. Techniques such as distributed pose graph optimization, consensus-based estimation, and inter-robot loop closure detection have shown promise, but challenges remain in communication bandwidth limitations, heterogeneous sensor configurations, and robustness to inconsistent data association.

SLAM in dynamic environments remains one of the most enduring challenges in robotics. Traditional SLAM algorithms typically assume static environments; however, real-world environments often contain humans, vehicles, animals, and moving obstacles. To address this, dynamic SLAM approaches attempt to model moving objects explicitly, reject dynamic features, or dynamically segment scenes into static and non-static components. Methods such as DynaSLAM, DS-SLAM, and motion segmentation networks leverage deep learning to filter out dynamic regions, while real-time multi-body SLAM attempts to track multiple moving entities simultaneously. These methods significantly improve localization accuracy and map stability in crowded environments such as streets, malls, and airports.

Looking ahead, SLAM is expected to move toward global consistency, long-term autonomy, and richer semantic understanding. Emerging research converges on spatiotemporal mapping, lifelong SLAM, large-scale learning-based priors, and integration of vision-language models for contextual reasoning. Additionally, new sensing modalities such as event cameras, radar SLAM, and LiDAR − camera BEV fusion will continue to influence the design of modern AMR localization pipelines. As AMRs become increasingly deployed in real-world settings, SLAM must satisfy not only accuracy and real-time performance but also reliability, safety, and resilience over extended operational lifetimes.

4. Path Planning for Autonomous Navigation

Path planning constitutes one of the central algorithmic pillars of autonomous mobile robots, bridging the gap between perception-derived environmental representations and the control commands that drive robot motion. As AMRs navigate through structured or unstructured environments, they must compute collision-free, dynamically feasible, and efficient trajectories that respect kinematic constraints, avoid obstacles, and account for uncertainties in perception and actuation. The path planning pipeline typically consists of global planning, which computes routes over a large spatial map, and local planning or trajectory generation, which refines these routes using real-time sensor inputs and dynamic obstacle reasoning. Modern planning frameworks integrate heuristic search, sampling-based exploration, nonlinear optimization, and increasingly, data-driven and learning-based models that enable robots to adapt to complex, dynamic, and partially observable environments. A simplified overview of the planning pipeline is illustrated in Figure 1, which shows the relationship between the global planner, local planner, environmental map, and the robot’s kinematic controller.

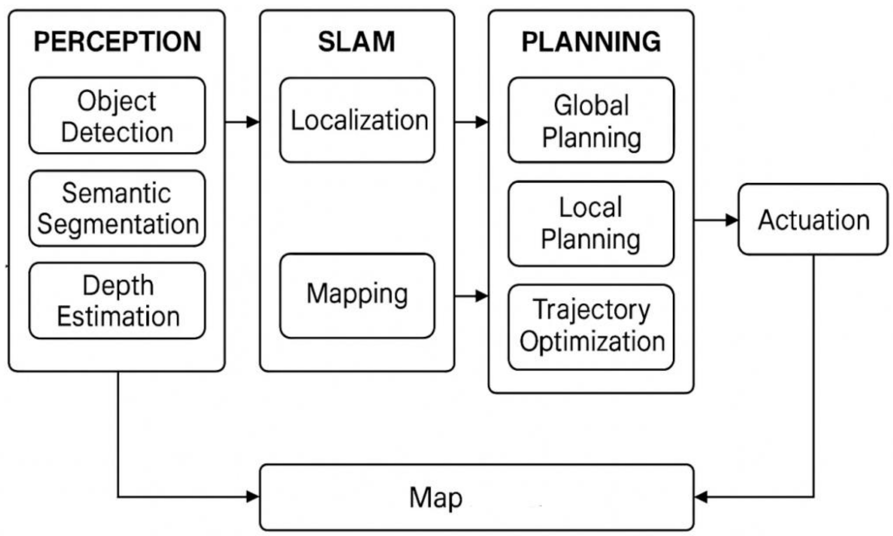

Figure 1. System architecture of an autonomous mobile robot.

Global path planning often operates on a topological or geometric map created by SLAM or prior environmental modeling. Classical planners such as Dijkstra’s algorithm and A* remain widely used due to their deterministic guarantees and simplicity, particularly when robot navigation requires finding the shortest feasible route on a grid or graph representation [23]. A* improves over Dijkstra by incorporating heuristics, enabling faster convergence in large maps. Variants such as Anytime Repairing A* (ARA*), Lifelong Planning A* (LPA*), and D* Lite enhance the adaptability of graph-based planning to real-time replanning and dynamic environments, making them suitable for field robots, planetary rovers, and autonomous ground vehicles operating in partially known terrains. These methods remain competitive in settings where accuracy and reliability take precedence over computational efficiency, especially when maps are relatively stable.

As navigation demands increased flexibility, sampling-based motion planners emerged as a powerful alternative for high-dimensional and continuous configuration spaces. Probabilistic Roadmaps (PRM) and Rapidly-Exploring Random Trees (RRT) revolutionized motion planning by offering probabilistic completeness and scalability to complex robots with nontrivial kinematics [24]. RRT* further introduced asymptotic optimality, enabling convergence toward the optimal solution under mild assumptions, while variants such as Informed RRT*, BIT*, and FMT* improved efficiency by guiding exploration toward promising regions of the search space [25]. These algorithms have been widely deployed in wheeled robots, aerial drones, underwater vehicles, and robotic manipulators due to their ability to handle non-Euclidean constraints, nonholonomic motion, and higher-dimensional state representations. Nonetheless, sampling-based planners often produce jagged, non-smooth paths that require post-processing, and their performance can degrade in cluttered or narrow passage environments where sampling becomes inefficient.

Optimization-based planning frameworks address several limitations of sampling techniques by formulating motion planning as a continuous optimization problem. These methods directly minimize cost functions that encode smoothness, safety margins, control effort, and kinematic feasibility. Model Predictive Control (MPC), CHOMP (Covariant Hamiltonian Optimization for Motion Planning), STOMP (Stochastic Trajectory Optimization for Motion Planning), and nonlinear trajectory optimization methods have demonstrated strong performance in generating smooth and dynamically consistent paths in real time [26]. Trajectory optimization is particularly well-suited for AMRs because it supports incremental replanning, incorporates dynamic obstacles, and naturally integrates robot dynamics into the planning process. In autonomous driving, optimization-based planners such as those used in Apollo, Autoware, and Baidu's navigation stack compute safe trajectories at high frequency while accounting for vehicle dynamics, road geometry, traffic rules, and interaction with human drivers.

Local planning is equally essential, as AMRs must respond to dynamic obstacles, sensor uncertainty, and unpredictable changes in the environment. The Dynamic Window Approach (DWA) remains one of the most widely used local planners due to its ability to efficiently explore admissible velocities of differential-drive robots under motion constraints [27]. The Timed Elastic Band (TEB) planner improves upon DWA by optimizing trajectories in the space-time domain, enabling robots to generate flexible and smooth paths around obstacles while balancing velocity, curvature, and time constraints [28]. These local planners interface with the global planner to refine navigation strategies and ensure safe robot behavior in cluttered or crowded environments. As AMRs increasingly operate among humans in indoor facilities, malls, airports, and urban streets, local planners must incorporate human motion prediction models to anticipate pedestrian trajectories and prevent intrusive or unsafe interactions.

Recent advances in machine learning have significantly altered the planning landscape, enabling AMRs to leverage data-driven models for navigation. Reinforcement learning (RL) frameworks allow robots to learn navigation policies directly through trial-and-error interactions with their environment, producing behaviors that generalize across complex and partially observable settings. Popular algorithms such as DDPG, PPO, SAC, and TD3 have been applied to differential-drive robots, UAVs, and more recently to multi-robot systems, allowing agents to learn collision avoidance strategies, human-aware navigation, or long-horizon decision policies [29]. Imitation learning (IL) approaches, including behavior cloning and inverse reinforcement learning, train navigation policies by observing expert demonstrations. These techniques enable robots to capture subtle navigation cues, social norms, and human-like motion patterns that are difficult to encode explicitly through optimization-based planners. End-to-end neural motion planners further aim to map raw sensor data-such as LiDAR scans or images-directly to control actions or trajectory commands, bypassing traditional modular pipelines. While promising, these methods still struggle with safety guarantees, interpretability challenges, and out-of-distribution generalization, making hybrid planners that combine learning with classical methods a more practical direction for real-world deployment.

Another important direction in planning research is multi-agent navigation and cooperative planning. Multi-robot coordination requires robots to share information, avoid collisions with each other, and execute complementary tasks simultaneously. Techniques such as decentralized MPC, reciprocal velocity obstacles

(RVO), optimal reciprocal collision avoidance (ORCA), and distributed consensus-based methods have been applied to robot fleets in warehouse automation, UAV swarms, and autonomous delivery systems [30]. Multi-robot planning introduces additional complexities, including communication constraints, distributed mapping, and task allocation-but also provides opportunities for collective intelligence and improved scalability in large operational environments.

As AMRs move toward widespread adoption, path planning must address several open challenges, including safety under uncertainty, scalability to large environments, interpretability of learned navigation policies, and the integration of semantic understanding into planning decisions. For example, high-level semantic cues-such as identifying doorways, walls, and human activity zones-can significantly improve planning efficiency and safety. Additionally, future AMR systems may utilize large-scale foundation models for environmental reasoning, enabling planners to incorporate contextual knowledge and generalize across unseen scenarios. The growing incorporation of semantic maps, BEV representations, multi-agent predictions, and multimodal sensor fusion suggests that planning will continue evolving toward richer and more holistic decision frameworks.

5. Control of Autonomous Mobile Robots

Control serves as the final operational layer of an autonomous mobile robot, translating high-level navigation decisions into dynamically feasible motor commands that govern the robot's motion in real-world environments. While perception, mapping, and planning contribute essential information about the robot's surroundings and navigation objectives, it is ultimately the control module that ensures accurate trajectory tracking, stability, safety, and responsiveness. Control algorithms for AMRs must address multiple challenges, including nonlinear dynamics, uncertainties in sensing and actuation, external disturbances, time delays, and the need for real-time performance. The diversity of AMR platforms-from differential-drive ground robots and car-like vehicles to legged robots and multi-rotor UAVs-introduces additional complexities related to kinematics, nonholonomic constraints, drift, and rapid motion changes. A general overview of the hierarchical control architecture commonly used in AMRs is illustrated in Figure 2, highlighting the interplay between high-level planning, mid-level trajectory tracking, and low-level actuator control.

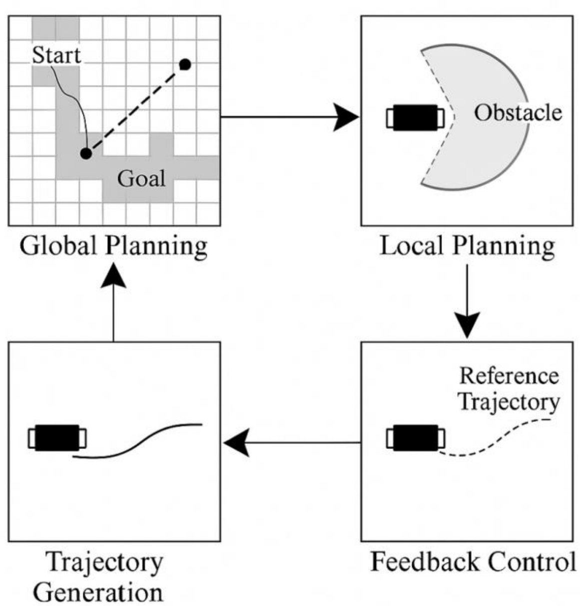

Figure 2. Motion planning and control for an autonomous mobile robot.

At the heart of AMR control lies the mathematical modeling of robot kinematics and dynamics. Many ground robots can be approximated using nonholonomic kinematic models, such as the unicycle or bicycle model, which constrain motion due to wheel configuration. For a differential-drive robot, the standard kinematic equations are given by:

$$\dot{x} = v\cos\theta, \quad \dot{y} = v\sin\theta, \quad \dot{\theta} = \omega$$

where x, y denote the robot position, θ is the orientation, v is the linear velocity, and ω is the angular velocity. These equations form the basis for many control strategies including feedback linearization, pure pursuit tracking, and nonlinear model-based controllers. For car-like robots or autonomous vehicles, the bicycle model is frequently used, where the steering angle and velocity determine the dynamics. More complex robots, such as UAVs or legged platforms, require full dynamic models including mass matrices, Coriolis forces, and gravity terms.

Classical control methods remain highly relevant due to their interpretability and robustness. Proportional – Integral – Derivative (PID) control continues to be widely used for low-level motor regulation, ensuring smooth velocity and heading tracking under minimal computational overhead [31]. For trajectory tracking, linear control methods such as Linear Quadratic Regulator (LQR) provide optimal feedback laws by minimizing a quadratic cost function, enabling stable responses for linearized dynamics. Nonlinear control strategies such as feedback linearization and backstepping are frequently employed to address the inherent nonlinearities in AMR motion models. For differential-drive robots, pure pursuit and Stanley controllers offer effective geometric tracking solutions and are widely used in autonomous driving due to their simplicity and reliability [32].

Model Predictive Control (MPC) has gained prominence in AMR control due to its ability to incorporate dynamic constraints, optimize over a prediction horizon, and respond adaptively to changes in the environment. MPC solves an optimization problem at each control step, minimizing a cost that typically includes trajectory tracking error, control effort, and smoothness terms subject to physical and environmental constraints. The general formulation of MPC can be expressed as:

$$\min_{\mathbf{u}} \sum_{k=0}^{N} \left\| \mathbf{x}_k - \mathbf{x}_k^{\mathrm{ref}} \right\|_Q^2 + \left\| \mathbf{u}_k \right\|_R^2$$

subject to the dynamic model

$$\mathbf{x}_{k+1} = f(\mathbf{x}_k, \mathbf{u}_k)$$

and system constraints such as actuator limits or obstacle-avoidance requirements. MPC has been widely integrated into industrial AMRs, autonomous driving stacks, aerial robots, and high-speed dynamic systems, where its ability to handle constraints and generate smooth trajectories is critical. Real-time variants such as Sequential Quadratic Programming (SQP)-based MPC, real-time Iterative Linear Quadratic Regulator (iLQR), and nonlinear MPC enable high-frequency control in demanding applications.

Robust control extends classical methods to account for uncertainties in modeling, sensing, and actuation. Techniques such as $H\infty$ control, sliding mode control, and adaptive control ensure stability even when the system model is partially inaccurate or subjected to external disturbances [33]. Sliding mode control, for example, introduces a discontinuous control law that forces the system state onto a predefined sliding manifold, ensuring robustness in the presence of bounded uncertainties. Adaptive control methods update controller parameters online, enabling AMRs to maintain stability despite payload variations, friction changes, or terrain irregularities. These techniques are especially valuable for service robots and agricultural robots that frequently encounter uncertain conditions in real-world deployments.

Recent advances in machine learning have significantly influenced AMR control architectures. Reinforcement learning (RL) offers a framework for learning control policies directly from interactions, bypassing the need for accurate modeling. Algorithms such as Deep Deterministic Policy Gradient (DDPG), Twin Delayed DDPG (TD3), Proximal Policy Optimization (PPO), and Soft Actor – Critic (SAC) have been applied to dynamic navigation, obstacle avoidance, and agile maneuvers in UAVs and wheeled robots [34]. These policies can capture complex nonlinear behaviors and adapt to variations in environment geometry or dynamics. However, RL controllers often suffer from limited interpretability, sensitivity to distribution shifts, and difficulty in guaranteeing safety-challenges that must be addressed before full deployment in safety-critical applications.

Hybrid control architectures that combine classical methods with learning-based components offer a promising middle ground. For instance, learning-based observers such as neural network – enhanced Kalman filters can estimate unmodeled disturbances, while classical controllers ensure baseline stability. Learning-based feedforward terms can compensate for nonlinearities or reduce control errors, while MPC remains responsible for constraint satisfaction. Recent developments such as neural MPC, differentiable control architectures, and end-to-end visuomotor policies demonstrate the growing convergence between optimal control theory and deep learning. In legged and aerial robots, differentiable dynamics models and policy-gradient-based optimization have enabled precise whole-body control and safe exploration of dynamic behavior spaces.

Safety remains a central concern in AMR control, particularly in human-centered environments. Control Barrier Functions (CBFs) have emerged as a powerful tool for enforcing safety constraints by shaping the control input to ensure forward invariance of safe sets [35]. CBFs can be paired with CLFs (Control Lyapunov Functions) to jointly ensure stability and safety. These methods allow robots to execute nominal control policies while guaranteeing collision avoidance or adherence to operational limits. The ability to embed formal guarantees into learned or optimization-based controllers is a critical trend for AMR deployment in public spaces, hospitals, and industrial facilities.

Future directions in AMR control point toward greater integration of data-driven dynamics models, semantics-aware control, physically grounded neural models, and large-scale foundation policies capable of generalizing across tasks, robots, and environments. Advances in sim-to-real transfer, differentiable physics, and model-based RL will continue to narrow the gap between learned controllers and classical methods, enabling more adaptive, efficient, and safe robot behavior. As AMRs become more ubiquitous, control systems must balance high adaptability with strong safety and explainability requirements, ensuring reliable long-term operation in unpredictable real-world scenarios.

6. Integration, System Architectures, and Practical Deployments

The integration of perception, mapping, planning, and control into a unified operational framework represents one of the most critical and challenging aspects of autonomous mobile robot (AMR) design. While each subsystem may achieve state-of-the-art performance independently, the overall success of an AMR depends on the robustness, real-time responsiveness, modularity, and reliability with which these components interact in real-world environments. System integration demands careful architectural design, efficient middleware communication, fault-tolerant mechanisms, and rigorous system-level testing. These considerations become particularly important as AMRs transition from laboratory prototypes to large-scale deployment in warehouses, hospitals, cities, and domestic environments, where robots must operate continuously, safely, and adaptively under uncertain and dynamic conditions. The increasing complexity of AMR software stacks and hardware platforms has led to the adoption of standardized frameworks, such as the Robot Operating System (ROS) and its successor ROS 2, which provide communication interfaces, middleware abstractions, and modular software components that simplify integration and deployment.

A typical AMR system architecture follows a multi-layer hierarchical structure composed of the perception layer, localization and mapping layer, planning and decision-making layer, and control and execution layer. These layers must exchange data continuously, often at different temporal resolutions and computational rates. For example, IMU and wheel-odometry signals may operate at hundreds of Hertz, while LiDAR scans arrive at tens of Hertz, camera frames at 30 – 60 Hz, and SLAM updates at lower but computationally heavy frequencies. Planning modules typically operate between 5 and 20 Hz, and low-level control commands must be generated at real-time rates exceeding 100 Hz to ensure smooth trajectory execution. Ensuring consistent synchronization across these asynchronous data streams remains a significant systems challenge. ROS 2, built on the DDS (Data Distribution Service) middleware, addresses many of these requirements by offering real-time QoS settings, distributed communication, standardized message types, and modular interfaces that allow developers to integrate heterogeneous sensing and computation pipelines effectively [36].

Hardware integration also plays a crucial role in AMR deployment. Modern robots incorporate increasingly sophisticated sensing arrays-such as 3D LiDARs, depth cameras, event cameras, ultrawideband (UWB) localization systems, and mmWave radars-which must be calibrated precisely and fused into a consistent representation. High-performance compute platforms, such as NVIDIA Jetson AGX, Intel NUC, ARM-based embedded systems, and automotive-grade ECUs, provide the computational backbone necessary for real-time SLAM, planning, and inference. Integration challenges include thermal management, power consumption, data bandwidth optimization, and ensuring that the robot can operate for extended periods without overheating or power failure. In industrial AMRs, redundant sensing paths and hardware watchdog systems are frequently used to guarantee fail-safe operation. The trend toward edge computing has further influenced AMR system design, enabling partial offloading of heavy computation (e.g., global mapping, semantic segmentation, or multi-agent coordination) to edge servers or cloud systems while retaining core navigation functions on the robot for safety and reliability.

Another major challenge in AMR integration is ensuring robust performance in real-world environments where uncertainty, dynamic elements, and unexpected events are commonplace. Real-world deployments require AMRs to maintain localization under sensor degradation, navigate around dynamic pedestrians, recover from temporary environmental occlusions, and handle ambiguous or incomplete sensor data. To address these issues, modern systems adopt multi-modal redundancy, where the robot automatically switches between localization sources-for example, transitioning from LiDAR-based localization to vision-inertial odometry during LiDAR occlusions or failures. Similarly, perception stacks may include fallback pipelines that use lightweight feature tracking or optical flow when deep-learning-based models fail under extreme motion blur or low-light conditions. Fault detection and recovery mechanisms, supported by system monitors such as ROS2's lifecycle nodes or industrial safety controllers, ensure that the robot can enter safe states when critical failures are detected.

Real-time constraints pose another layer of complexity. To meet the latency requirements of high-speed control and planning, modern AMRs increasingly rely on parallel processing, GPU acceleration, multi-threaded pipelines, and real-time operating systems (RTOS). For instance, time-critical tasks such as kinematic control and state estimation may run under RT kernels like PREEMPT_RT Linux or QNX, while higher-level perception and planning tasks run on general-purpose OS layers. Ensuring that these components communicate reliably without violating timing constraints involves careful scheduling, QoS tuning, and performance profiling. The integration of GPU-accelerated frameworks, such as CUDA-based point cloud registration or TensorRT-accelerated deep learning inference, has further improved the feasibility of running advanced perception algorithms concurrently with planning and control in real time.

Practical deployment of AMRs in industry introduces additional system-level challenges that extend beyond core robotics algorithms. These include fleet management for multi-robot systems, human – robot interaction, interoperability with existing infrastructure, and compliance with safety regulations. Warehouse robots, for example, must coordinate with other machines, shared network infrastructure, and human workers while ensuring that their navigation and task allocation remain globally optimal. Multi-robot management platforms use cloud-based coordination servers, global planners, and distributed communication to orchestrate hundreds of AMRs simultaneously. Healthcare robots deployed in hospitals require robust semantic understanding to recognize rooms, hallways, and patient-care areas, and must integrate into hospital networks for scheduling, access control, and task management. Outdoor AMRs used for delivery or urban navigation must comply with city-level traffic regulations, communication protocols, and safety certifications.

Robust integration also depends on exhaustive testing and validation. Before deployment, AMRs undergo simulation-based evaluation using platforms such as Gazebo, Webots, CARLA, or Isaac Sim, followed by controlled field tests and staged deployments. Simulation allows developers to test failure scenarios-such as sensor dropout, dynamic obstacles, or unexpected map inconsistencies-that would be unsafe or costly to reproduce in the physical world. Domain randomization, synthetic data augmentation, and physics-based simulation improve the generalization of perception and control models to real-world conditions. Once deployed, continuous monitoring, logging, and telemetry analysis help identify long-term drift, performance degradation, and rare failure events. Feedback loops between real-world operation and simulation-based regression testing help maintain system reliability over months or years of deployment.

Looking forward, AMR system integration is expected to evolve toward more modular, scalable, and intelligent architectures. The rise of large multi-modal foundation models suggests a future in which perception, planning, and control pipelines are more tightly intertwined through shared latent representations. Semantics-aware maps, cloud-assisted high-definition mapping, and collaborative multi-robot frameworks will support more sophisticated behaviors and enable the deployment of large robot fleets across factories, warehouses, and urban spaces. At the same time, ensuring safety, interpretability, and regulatory compliance will remain essential priorities as AMRs enter human-centered environments. Advances in real-time middleware, safety-certified software frameworks, hardware-accelerated perception, and adaptive resource allocation will continue shaping the future of AMR system deployments.

7. Applications of Autonomous Mobile Robots

Autonomous mobile robots (AMRs) have transitioned from research prototypes to essential operational assets across a wide spectrum of real-world domains, propelled by advances in perception, SLAM, planning, and control. The deployment of AMRs is motivated by their ability to perform tasks that are repetitive, hazardous, time-sensitive, or require precision beyond human capability. Modern AMR applications span industrial automation, logistics, manufacturing, healthcare, service robotics, agriculture, environmental monitoring, security, and emergency response. Each application domain imposes distinct requirements on navigation accuracy, robustness, semantic understanding, safety, communication, and integration with broader infrastructure. As AMRs increasingly operate in human-centered or dynamic environments, their practical utility depends on achieving high reliability under real-world uncertainties, compliance with regulatory frameworks, and seamless integration into existing workflows. The rapid evolution of AMR technologies continues to expand their practical relevance while simultaneously revealing new challenges associated with large-scale deployment, human – robot interaction, and long-term autonomy.

One of the most transformative AMR applications lies in industrial logistics and warehouse automation, where robot fleets perform material handling, pallet transport, inventory movement, and order picking. Companies such as Amazon Robotics, Fetch Robotics, GreyOrange, and Locus Robotics have demonstrated large-scale deployment of AMRs capable of navigating vast warehouses with thousands of moving agents [37]. In these environments, AMRs rely heavily on robust localization in repetitive layouts with low semantic variability and high-density shelving structures. They must interact with human workers while maintaining strict safety standards. Modern warehouse AMRs combine LiDAR-based localization, QR-code markers, visual fiducials, or UWB beacons to achieve centimeter-level accuracy. Fleet management systems coordinate hundreds of robots simultaneously, optimizing traffic flow, minimizing congestion, and performing real-time task allocation. The efficiency gains in warehousing-from reduced labor costs to increased throughput-remain among the most compelling commercial demonstrations of AMR capability.

Beyond warehousing, manufacturing facilities increasingly rely on AMRs to transport parts, tools, and subassemblies between production stations, replacing fixed conveyor systems with dynamic, reconfigurable material flow. In such environments, AMRs must maintain high reliability in the presence of heavy machinery, metallic structures, and electromagnetic noise, necessitating robust multi-sensor fusion and fault-tolerant navigation. Integration with manufacturing execution systems (MES) and enterprise resource planning (ERP) software enables dynamic task scheduling and automated workflow orchestration.

In the domain of healthcare, AMRs serve critical roles in hospitals by transporting medications, samples, medical equipment, linens, and sterilization supplies. Systems such as Aethon TUG, Relay, and autonomous UV-disinfection robots have shown that AMRs can improve hospital operational efficiency while reducing infection risks and relieving nurses from nonclinical tasks [38]. Healthcare environments pose unique challenges including narrow hallways, dynamic human traffic, irregular lighting, elevators, doorways, and the need for strict safety certification. Robots must integrate into hospital IT systems for access control, scheduling, and patient privacy compliance. The COVID-19 pandemic accelerated adoption of AMRs for disinfection, telepresence, and contactless delivery, highlighting the potential for broader integration into healthcare operations.

AMRs are also transforming the domain of outdoor navigation and urban mobility, especially through the rise of autonomous delivery robots and last-mile logistics systems. Sidewalk delivery robots-such as those developed by Starship, Nuro, and Coco-require high-level reasoning to navigate unpredictable environments involving pedestrians, pets, cyclists, and complex urban layouts. Outdoor conditions introduce challenges including GPS noise, occlusions, weather variability, and the need for long-range perception using multi-modal sensors and semantic scene understanding. Autonomous driving systems represent the most advanced form of AMR deployment, integrating high-definition maps, multi-sensor fusion, multi-agent prediction, and advanced decision-making under stringent safety constraints. Modern autonomous vehicles incorporate 360 ° sensing, real-time perception stacks, and hierarchical planning frameworks that reflect many aspects of AMR architecture at larger scale [39].

In agriculture, AMRs are used for autonomous harvesting, weeding, spraying, soil monitoring, and crop transportation. Agricultural environments present irregular terrain, vegetation occlusion, dynamic lighting, and limited structured features, making navigation substantially more challenging than indoor settings. Robots must adapt to deformable crop structures, uneven ground, and seasonal variability. Systems such as autonomous tractors, vineyard robots, and orchard harvesters leverage RTK-GPS, LiDAR, stereo vision, and machine learning for crop recognition and task execution. Precision agriculture applications benefit from long-range autonomous coverage and fine-grained environmental sensing, improving yield prediction and resource efficiency.

In security and surveillance, AMRs patrol facilities, campuses, and public areas to detect anomalies, monitor intrusions, and gather situational awareness. Robots such as SMP Robotics' security units employ multi-camera systems, thermal sensors, and cloud-connected analytics to autonomously survey vast areas. These robots must reason about open environments, dynamically track moving agents, and communicate with security centers. Reliability, robustness to communication loss, and safe operation around civilians are paramount concerns.

AMRs play crucial roles in emergency response, disaster relief, and hazardous environment inspection, where robots are deployed in environments that are unsafe or inaccessible to humans. Search-and-rescue robots operate in collapsed buildings, wildfire zones, radioactive or chemically contaminated sites, and offshore facilities. Navigation under such conditions demands resilient SLAM systems capable of handling dust, debris, degraded lighting, GPS-denied settings, and structural instability. Modular robot platforms, snake-like robots, and rugged ground vehicles equipped with thermal imaging, gas sensors, and multi-modal mapping tools can identify survivors, assess structural integrity, and support human responders. Reliability and resilience are key, as failures in dangerous environments can hinder rescue efforts.

Emerging applications include hospitality and service robots, such as restaurant runners, hotel delivery robots, autonomous shopping assistants, and cleaning robots. These robots must navigate crowded indoor environments with unpredictable human motion and must adhere to social navigation norms. Other expanding domains include environmental monitoring and conservation-where AMRs monitor wildlife, track environmental changes, and map forest or marine ecosystems. Long-duration autonomy remains a critical challenge in such domains, requiring energy-efficient hardware, adaptive perception pipelines, and self-recharging infrastructure.

Looking toward the future, AMR applications are expected to become more deeply integrated into smart cities, Industry 4.0 frameworks, healthcare automation ecosystems, and domestic robotics. Advances in large-scale multi-agent coordination, semantic mapping, human-aware navigation, and foundation models for perception and planning will expand AMR capabilities into tasks requiring complex reasoning and real-time collaboration. As regulatory frameworks evolve to address safety, privacy, and operational standards, AMRs will progressively transition from controlled environments into broader public deployment. Achieving long-term autonomy, robust safety guarantees, and scalable fleet management will be central to unlocking the next generation of AMR applications.

8. Challenges, Open Problems, and Future Directions

Despite the significant advances in perception, SLAM, planning, and control, autonomous mobile robots (AMRs) continue to face substantial challenges that impede reliable long-term deployment in complex, dynamic, and unstructured real-world environments. These challenges arise from limitations in sensing under adverse conditions, difficulties in achieving robust and drift-free localization, the unpredictability of human behaviors, computational constraints on embedded systems, safety requirements in human-centered spaces, and the need for scalable multi-robot coordination. As AMRs begin to operate in broader domains-including urban mobility, public environments, and autonomous logistics-researchers must address problems that span algorithmic foundations, system design, hardware robustness, and regulatory compliance. Understanding these challenges is essential for guiding future developments and establishing the next generation of intelligent, resilient, and adaptive AMR systems.

One of the most enduring challenges is achieving robust perception in highly variable or degraded environments. Although LiDAR, radar, and camera systems have improved in resolution and cost-efficiency, perception pipelines remain vulnerable under extreme lighting (glare, low-light, high dynamic range), environmental obscurants (fog, dust, snow), motion blur, and physical damage. Indoor AMRs must handle reflective floors and low-texture walls, while outdoor systems face weather-induced degradation and occlusion by dynamic crowds. Although deep-learning-based perception offers stronger generalization, models often struggle when operating out of distribution or under sensor failures. Self-supervised learning, domain adaptation, uncertainty modeling, and multi-modal redundancy can mitigate some of these issues, but perception robustness remains a key bottleneck for long-term autonomy in uncontrolled environments [40].

Another significant challenge lies in long-term, large-scale, and semantically meaningful SLAM. Existing SLAM systems can drift over time, break when encountering large dynamic objects, or degrade when environmental conditions change significantly over days, seasons, or years. Lifelong SLAM requires maps that evolve continuously, maintain consistency, and incorporate semantic information while avoiding catastrophic forgetting. Semantic SLAM pipelines still rely heavily on vision-based segmentation, which may suffer from label noise or temporal inconsistencies. Loop closures in dynamic scenes remain difficult, and global map optimization can be computationally prohibitive at scale. Furthermore, building shared maps for multi-robot systems introduces issues in data association, communication constraints, and heterogeneous sensor calibration. Developing SLAM systems that match human-like place recognition across long time horizons remains an open research frontier [41].

Path planning faces equally profound challenges, especially in dynamic, crowded, or uncertain environments involving humans. Classical global planners do not account for human social norms, and local planners often fail in dense environments where the robot must predict and negotiate human movement. While learning-based planners have shown promise in social navigation and reactive obstacle avoidance, guaranteeing safety, interpretability, and stability remains difficult. Multi-agent prediction models, particularly those based on Transformers and graph neural networks, offer stronger forecasting capabilities but require large annotated datasets and still struggle with rare or unexpected behaviors. Ensuring safe planning under uncertainty, particularly when perception is imperfect or delayed, remains one of the most difficult open problems in AMR navigation [42].

Control systems continue to encounter challenges related to model uncertainty, safety guarantees, real-time constraints, and sim-to-real transfer. Learned controllers often exhibit limited robustness when deployed outside their training distribution, while classical controllers struggle with complex, nonlinear dynamics or poorly modeled interactions. Model-based reinforcement learning and differentiable control architectures promise improved generalization, but verifying safety and stability remains difficult. Control barrier functions and formal verification tools provide mathematical guarantees, yet integrating them with learning-based policies is still an emerging research direction. Energy efficiency is another major challenge: achieving long-duration autonomy requires optimizing control policies for low power consumption, especially in mobile platforms with limited battery capacity [43].

On the system level, AMRs face substantial difficulties in scalability, real-time operation, and hardware reliability. Embedded compute platforms cannot always support heavy perception and planning workloads, especially when deep neural models are involved. GPU acceleration helps but increases power consumption and thermal load. Real-time scheduling becomes challenging when multiple high-frequency modules compete for compute resources. Additionally, hardware calibration drift, mechanical wear, and sensor degradation lead to performance drop-offs during long-term deployment, necessitating self-diagnosis and adaptive re-calibration pipelines. For large robot fleets, communication bandwidth limitations and network latency further complicate global coordination and cloud-assisted computation.

Human – robot interaction introduces additional complexities involving trust, predictability, transparency, and social compliance. Robots must behave in ways that humans intuitively understand, maintain appropriate distance, and avoid causing discomfort. Predicting human intentions remains difficult, and failures can lead to safety hazards. Social navigation datasets remain limited, and cultural differences influence human expectations of robot behavior. As AMRs become more common in public spaces, regulatory agencies will require standardized safety protocols, explainability of failures, and transparent behavior models.

Lastly, ethical, legal, and regulatory concerns play an increasingly important role in shaping the future of AMR deployment. Issues involving liability in autonomous navigation, data privacy in perception systems, certification standards for safety, and compliance with transportation regulations are all growing areas of focus. Establishing unified regulatory frameworks across industries and geographic regions will be essential for scaling AMR adoption. Ensuring transparency, auditability, and fairness in learned decision-making remains a pressing concern as robots increasingly rely on deep neural models whose internal workings are difficult to interpret.

Looking forward, several emerging research directions hold promise for overcoming these challenges. Large multi-modal foundation models, integrating vision, language, and action, could provide unified representations that improve generalization in perception, planning, and control. Advances in neuromorphic sensing, event cameras, radar imaging, and 4D LiDAR will increase robustness in challenging environments. Self-repairing SLAM, multi-agent collaborative mapping, and semantic world models may enable persistent long-term autonomy. Hierarchical planners that combine symbolic reasoning with learned low-level policies could bridge the gap between high-level task planning and reactive navigation. Meanwhile, the integration of cloud robotics, distributed sensing, and edge AI will enable large robot fleets to operate collaboratively across factories, hospitals, and cities. While substantial challenges remain, continued progress in algorithmic foundations, system integration, and safe deployment practices will drive AMRs toward becoming a ubiquitous and transformative technology across modern society.

9. Conclusion

Autonomous mobile robots (AMRs) have evolved into one of the most technologically rich and impactful domains in modern robotics, integrating advances across perception, localization, mapping, planning, control, and system-level engineering. Through decades of research, AMRs have progressed from simple line-following machines to sophisticated systems capable of navigating complex, dynamic, and uncertain environments with high levels of autonomy and reliability. The integration of multi-modal sensing, deep-learning-based perception, robust SLAM pipelines, optimization-driven and learning-enhanced planning frameworks, and hierarchical control architectures has enabled robots to operate across diverse domains including industrial logistics, autonomous driving, healthcare, agriculture, surveillance, and emergency response. Yet, despite these remarkable advances, numerous challenges remain before AMRs can achieve ubiquitous, safe, and long-term autonomy in real-world environments.

This survey has provided a comprehensive overview of the algorithmic and system-level foundations underpinning AMR technology. In perception, we examined the shift from geometry-driven pipelines to deep neural models and multi-sensor fusion frameworks capable of leveraging LiDAR, vision, radar, and IMU data. In SLAM and localization, we reviewed the transition from classical filtering-based methods toward graph optimization, visual – inertial odometry, LiDAR odometry, semantic mapping, and emerging learning-based SLAM paradigms. In planning, we outlined the spectrum from deterministic heuristic search and sampling-based exploration to optimization-based and data-driven navigation strategies capable of negotiating both structured and dynamic environments. In control, we highlighted the interplay between classical control theory, robust nonlinear methods, model predictive control, and reinforcement learning, emphasizing the growing trend toward hybrid and safety-critical control architectures. Finally, we discussed how successful AMR deployment depends not only on algorithmic breakthroughs but also on robust system integration across middleware, hardware, real-time computation, safety layers, and large-scale fleet coordination architectures.

Looking forward, the next generation of AMRs will demand even deeper integration of semantic understanding, predictive reasoning, and adaptive decision-making. The emergence of large multimodal foundation models promises to unify perception, language, planning, and control through shared latent representations, potentially enabling robots to interpret high-level instructions, understand context, and generalize across tasks and environments. Advances in lifelong SLAM, continual learning, multi-agent collaboration, and bio-inspired control strategies will further expand AMR capability in open-world settings. Moreover, hardware innovations such as event-based vision sensors, high-resolution 4D LiDAR, neuromorphic processors, and energy-efficient compute platforms will improve robustness and extend operational lifetimes. Ensuring safety, reliability, transparency, and ethical governance will be essential as AMRs transition into public spaces and human-centered environments.

While AMRs have already demonstrated transformative impact in logistics, manufacturing, healthcare, and many other fields, their full potential remains far from realized. Continued research across robotics, machine learning, computer vision, and systems engineering-combined with interdisciplinary efforts involving regulation, human – robot interaction, and large-scale system deployment-will shape the trajectory of AMR development over the coming decades. By addressing the challenges discussed in this survey and capitalizing on emerging technologies and methodologies, the robotics community can pave the way toward resilient, intelligent, and broadly deployable autonomous mobile systems that will play a central role in the future of automation and human society.

References

[1] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 39, no. 6, pp. 1137 – 1149, 2017.

[2] C.-Y. Wang et al., "YOLOv7: Trainable Bag-of-Freebies for Real-Time Object Detection," arXiv preprint arXiv:2207.02696, 2022.

[3] N. Carion et al., "End-to-End Object Detection with Transformers," in Proc. European Conference on Computer Vision (ECCV), 2020.

[4] L.-C. Chen et al., "Encoder-Decoder with Atrous Separable Convolution for Semantic Segmentation," in Proc. ECCV, 2018.

[5] J. Wang et al., "Deep High-Resolution Representation Learning for Visual Recognition," in Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2020.

[6] E. Xie et al., "SegFormer: Simple and Efficient Design for Semantic Segmentation with Transformers," in Proc. NeurIPS, 2021.

[7] C. Godard et al., "Digging into Self-Supervised Monocular Depth Estimation," in Proc. ICCV, 2019.

[8] Z. Teed and J. Deng, "RAFT-Stereo: Multilevel Recurrent Field Transforms for Stereo Matching," in Proc. CVPR, 2021.

[9] C. Teed and J. Deng, "DROID-SLAM: Deep Visual SLAM for Monocular, Stereo, and RGB-D Cameras," in Proc. NeurIPS, 2021.

[10] T. Shan et al., "LIO-SAM: Tightly-Coupled Lidar Inertial Odometry via Smoothing and Mapping," in Proc. IROS, 2020.

[11] T. Qin, P. Li, and S. Shen, "VINS-Mono: A Robust and Versatile Monocular Visual-Inertial State Estimator," IEEE Transactions on Robotics, vol. 34, no. 4, pp. 1004 – 1020, 2018.

[12] L. Xu, W. Xu, and F. Zhang, "FAST-LIO: A Fast, Robust LiDAR-Inertial Odometry Package," IEEE Robotics and Automation Letters, vol. 6, no. 2, pp. 3317 – 3324, 2021.

[13] Z. Liu et al., "BEVFusion: Multi-Task Multi-Sensor Fusion with Unified Bird's-Eye View Representation," in Proc. ICRA, 2023.

[14] Y. Chen et al., "DeepFusion: A Learning-Based Multi-Level Feature Fusion Framework," arXiv preprint, arXiv:2203.xxxx, 2022.

[15] J. McCormac et al., "SemanticFusion: Dense 3D Semantic Mapping with Convolutional Neural Networks," in Proc. ICRA, 2017.

[16] R. Smith, M. Self, and P. Cheeseman, "Estimating Uncertain Spatial Relationships in Robotics," in Autonomous Robot Vehicles, Springer, 1990.

[17]M. Montemerlo et al., “FastSLAM: A Factored Solution to the SLAM Problem,” in Proc. AAAI, 2002.

[18]R. Kümmerle et al., “g2o: A General Framework for Graph Optimization,” in Proc. ICRA, 2011.

[19]G. Klein and D. Murray, “Parallel Tracking and Mapping for Small AR Workspaces,” in Proc. ISMAR, 2007.

[20]R. Mur-Artal and J. D. Tardós, “ORB-SLAM2: An Open-Source SLAM System for Monocular, Stereo, and RGB-D Cameras,” IEEE Transactions on Robotics, vol. 33, no. 5, pp. 1255 – 1262, 2017.

[21]J. Zhang and S. Singh, “LOAM: Lidar Odometry and Mapping in Real Time,” in Proc. RSS, 2014.

[22]W. Hess et al., “Real-Time Loop Closure in 2D LIDAR SLAM,” in Proc. IROS, 2016.

[23]P. E. Hart, N. J. Nilsson, and B. Raphael, “A Formal Basis for the Heuristic Determination of Minimum Cost Paths,” IEEE Transactions on Systems Science and Cybernetics, vol. 4, no. 2, pp. 100 – 107, 1968.

[24]S. M. LaValle, “Rapidly-exploring Random Trees: A New Tool for Path Planning,” Technical Report, Iowa State University, 1998.

[25]J. J. Kuffner and S. M. LaValle, “RRT-Connect: An Efficient Approach to Single-Query Path Planning,” in Proc. ICRA, 2000.

[26]M. Zucker et al., “CHOMP: Covariant Hamiltonian Optimization for Motion Planning,” International Journal of Robotics Research, vol. 32, no. 9 – 10, pp. 1164 – 1193, 2013.

[27]D. Fox, W. Burgard, and S. Thrun, “The Dynamic Window Approach to Collision Avoidance,” Autonomous Robots, vol. 4, no. 1, pp. 33 – 52, 1997.

[28]C. Rösmann, W. Feiten, and T. Bertram, “Trajectory Modification Considering Dynamic Constraints of Autonomous Robots,” IEEE Robotics and Automation Letters, vol. 2, no. 1, pp. 340 – 347, 2017.

[29]Y. Zhu et al., “Reinforcement Learning for Navigation Using Learned Maps,” arXiv preprint arXiv:2004.04144, 2020.

[30]J. van den Berg, S. Guy, M. Lin, and D. Manocha, “Reciprocal n-Body Collision Avoidance,” in Proc. ISRR, 2011.

[31]K. J. Åström and R. M. Murray, Feedback Systems: An Introduction for Scientists and Engineers, Princeton University Press, 2010.

[32]S. Thrun et al., “Stanley: The Robot that Won the DARPA Grand Challenge,” Journal of Field Robotics, vol. 23, no. 9, pp. 661 – 692, 2006.

[33]J.-J. Slotine and W. Li, Applied Nonlinear Control, Prentice-Hall, 1991.

[34]Y. Tassa et al., “DeepMind Control Suite,” arXiv preprint arXiv:1801.00690, 2018.

[35]A. D. Ames et al., “Control Barrier Function Based Quadratic Programs for Safety Critical Systems,” in Proc. CDC, 2014.

[36]M. Macenski et al., “ROS 2: Design, Architecture, and Uses in the Wild,” Science Robotics, 2022.

[37]Amazon Robotics, “Automation and Robotics in Modern Fulfillment Centers,” Amazon Tech Blog, 2021.

[38]C. Gong et al., “Hospital Service Robots: A Survey of Deployment and Challenges,” IEEE Access, vol. 10, pp. 110672 – 110689, 2022.

[39]S. Kato et al., “Autoware on Board: Enabling Autonomous Vehicles with Open-Source Software,” IEEE Micro, vol. 35, no. 3, pp. 24 – 35, 2018.

[40]C. Cadena et al., “Past, Present, and Future of SLAM: Toward the Robust-Perception Age,” IEEE Transactions on Robotics, vol. 32, no. 6, pp. 1309 – 1332, 2016.

[41]T. Qiao, J. Engel, and D. Cremers, “Long-Term SLAM and Lifelong Navigation,” IEEE RA-L, vol. 6, no. 3, pp. 5764 – 5771, 2021.

[42]A. Rudenko et al., “Human Motion Trajectory Prediction: A Survey,” The International Journal of Robotics Research, vol. 39, no. 8, pp. 895 – 935, 2020.

[43]A. D. Ames et al., “Safety-Critical Control and Formal Methods in Robotics,” Annual Review of Control, Robotics, and Autonomous Systems, vol. 4, pp. 113 – 139, 2021.