



Efficient Medical Image Classification Using Masked Attention Networks

Nadezhda Ramasubbu

Department of Computer Science and Engineering, Nineveh School of Industrial Management, Iraq
nadezhda.ramasubbu@nsim-iq.net

Peer Review Information

Submission: 18 Jan 2021
Revision: 02 Feb 2021
Acceptance: 17 March 2021

Keywords

Autonomous mobile robot,
Path planning, A*
algorithm, Dynamic
Window Approach, Deep
reinforcement learning

Abstract

Autonomous mobile robots require a navigation system that can perceive the environment, estimate pose, generate a collision-free route, react to previously unseen obstacles, and convert the planned path into dynamically feasible motion. This methodology paper develops a reproducible framework for studying intelligent path planning and navigation by combining a grid-based global planner, path smoothing, local velocity-based obstacle avoidance, and feedback re-planning. The literature is restricted to work published in 2021 or earlier and spans classical graph search, artificial potential fields, sampling-based motion planning, dynamic window navigation, particle-swarm-based smoothing, and deep reinforcement learning. The proposed experimental protocol uses a differential-drive robot model and a two-dimensional occupancy-cost map. Dijkstra, A*, and a hybrid weighted A* plus smoothing strategy are evaluated under identical map conditions using path length, planning time, expanded nodes, number of heading changes, clearance, and navigation success as performance indicators. A deterministic simulation included with this paper demonstrates that heuristic guidance substantially reduces search effort, while the hybrid planner reduces unnecessary turns and produces a smoother reference path. The methodology is intended as a compact baseline for subsequent implementation in ROS/Gazebo or on a physical autonomous mobile robot.

Introduction

Autonomous mobile robots (AMRs) are increasingly used for material transport, inspection, service robotics, warehousing, and flexible manufacturing. Their usefulness depends on the ability to move from a start configuration to a goal while avoiding static and dynamic obstacles and respecting vehicle constraints. Path planning is therefore not an isolated search problem; it operates inside a navigation loop that also includes perception, localization, map representation, motion control, and online correction. The original A* formulation established how admissible heuristic information can focus shortest-path search [1], while later obstacle-avoidance and sampling-based methods

extended navigation to continuous and partially known spaces [2-5].

A practical AMR must address two planning levels. Global planning uses a known or incrementally built map to determine a route toward the goal. Local planning transforms this route into short-horizon commands that satisfy velocity limits and avoid newly detected obstacles. Reviews of mobile-robot navigation show that no single algorithm is superior in all situations; graph search is reliable in structured maps, reactive approaches offer real-time response, metaheuristics can optimize smoothness, and learning-based methods are promising in uncertain environments [6]. The

research problem considered here is consequently how to organize these techniques into one experimentally testable methodology that balances path quality, computational efficiency, obstacle clearance, and adaptability.

The specific objective is to design a step-by-step evaluation framework for intelligent navigation of a differential-drive AMR. The framework uses an occupancy-cost map, compares Dijkstra and A* as reproducible baselines, introduces a weighted A* search with line-of-sight smoothing for the global route, and assigns local collision avoidance to a Dynamic Window Approach (DWA) layer. Learning-based navigation is treated as an optional decision module rather than as a replacement for safety-constrained planning. This is consistent with 2020-2021 research showing that deep reinforcement learning (DRL) can learn end-to-end action policies, but that training stability, generalization, and real-world transfer remain important concerns [7,8,11,12].

Literature Review

Hart, Nilsson, and Raphael formalized A* as a best-first search that combines accumulated path cost with a heuristic estimate of remaining cost [1]. In occupancy grids, A* remains attractive because it is deterministic, complete under standard finite-grid assumptions, and easy to benchmark. Khatib introduced the artificial potential field concept for real-time obstacle avoidance, where attractive and repulsive potentials generate motion away from obstacles and toward a goal [2]. Although computationally simple, potential fields can suffer from local minima and oscillation near closely spaced obstacles.

Sampling-based planning broadened the treatment of robot configuration spaces. Kavraki et al. proposed probabilistic roadmaps (PRM), which sample collision-free configurations and connect them into a reusable graph [3]. Karaman and Frazzoli later developed asymptotically optimal variants such as PRM* and RRT*, demonstrating that sampling-based methods can approach optimal path cost as the number of samples grows [5]. Their main strength is scalability to complex or high-dimensional spaces, while computation time and stochastic variability can be less convenient for tightly constrained embedded AMR controllers.

For reactive motion, Fox, Burgard, and Thrun proposed DWA, which searches directly in the robot velocity space and evaluates dynamically reachable translational and rotational velocities over a short prediction horizon [4]. This provides a natural way to include kinematic limits and collision checks. In structured indoor environments, DWA is commonly combined with a global planner so that local velocity selection follows a global route while still responding to unexpected objects.

The 2019 review by Patle et al. grouped mobile-robot path planning into classical, reactive, heuristic, and hybrid strategies and emphasized that hybridization is useful because different techniques address different weaknesses [6]. This motivates the present methodology: global search provides goal-directed structure, smoothing improves executability, and local velocity-space planning provides real-time avoidance. The design is not tied to a single sensor; LiDAR, depth cameras, or fused range data can populate the same local cost representation.

Table 1. Representative techniques used to construct the methodology (references limited to 2021 or earlier).

Technique	Primary role	Strength	Main limitation	Key refs.
A*	Global search	Deterministic; efficient with heuristic	Grid resolution; static-map dependence	[1],[6]
Potential field	Local avoidance	Low computational cost	Local minima / oscillation	[2],[6]
PRM/RRT*	Sampling-based motion planning	Handles complex configuration spaces	Stochastic runtime; tuning	[3],[5]
DWA	Local velocity planning	Includes velocity and braking constraints	Short prediction horizon	[4]
PSO + Bezier	Smooth path optimization	Continuous smooth trajectories	Offline/iterative optimization cost	[9],[10]
DRL	Adaptive navigation policy	Learns complex state-action mappings	Training data and transfer risk	[7],[8],[11],[12]

Research up to 2021 increasingly targeted smoothness and learned decision making. Song, Wang, and Zou combined particle swarm optimization (PSO) with a continuous high-degree Bezier curve to improve smooth path planning, thereby addressing high-order continuity that is important for mobile-robot motion control [9]. Xu, Song, and Cao similarly optimized a continuously curved path using parametric cubic Bezier curves and an adaptive PSO formulation [10]. These studies show that the geometric quality of the path should be measured in addition to shortest distance, because a route with repeated abrupt heading changes can be inefficient or difficult for a differential-drive robot to execute.

Learning-based navigation also advanced rapidly. Gao et al. used deep reinforcement learning for indoor mobile-robot path planning, demonstrating the ability of a learned policy to map perceived state information to motion decisions [7]. Yu, Su, and Liao combined neural networks with hierarchical reinforcement learning and reported improvements in planning efficiency and smoothness across their test scenarios [8]. Zhu and Zhang reviewed DRL-based navigation in local obstacle avoidance, indoor navigation, multi-robot navigation, and social navigation, while emphasizing remaining challenges related to training and deployment [11]. Chen et al. developed map-based DRL obstacle avoidance using local grid maps and transferred the trained policy from simulation to a real robot [12]. Muhammad, Ali, and Shanono provided a 2021 review of intelligent mobile-robot path planning and reinforced the importance of matching the planning strategy to environmental knowledge and robot constraints [13].

From this literature, three methodological requirements are derived. First, a planner should

be benchmarked against transparent classical baselines. Second, the route should be evaluated not only by length but also by search effort, turns, clearance, and dynamic feasibility. Third, intelligent or learned components should be embedded within a closed-loop architecture that retains explicit collision checking and re-planning. These requirements define the experiment described next.

Methodology

Step 1: Robot and Environment Model

A differential-drive AMR is represented by pose $q = [x, y, \theta]$. At each control interval, translational velocity v and angular velocity ω determine the short-term motion according to $\dot{x} = v \cos(\theta)$, $\dot{y} = v \sin(\theta)$, and $\dot{\theta} = \omega$. The environment is discretized into a two-dimensional occupancy-cost grid. Each cell is free, occupied, or assigned an inflated cost around obstacles. Inflation converts the physical robot footprint and safety margin into a planning constraint, preventing the global path from passing too close to walls.

Step 2: Perception and Localization

Range observations from LiDAR or depth sensing are transformed into the robot frame and fused with odometry or a SLAM/localization estimate. The global cost map supplies known static structure, while a local cost map is updated at the sensor frequency. Localization uncertainty is handled indirectly by obstacle inflation and by conservative local-planner clearance. A physical implementation may use AMCL or SLAM, but the methodology only requires a time-indexed pose estimate and obstacle map.

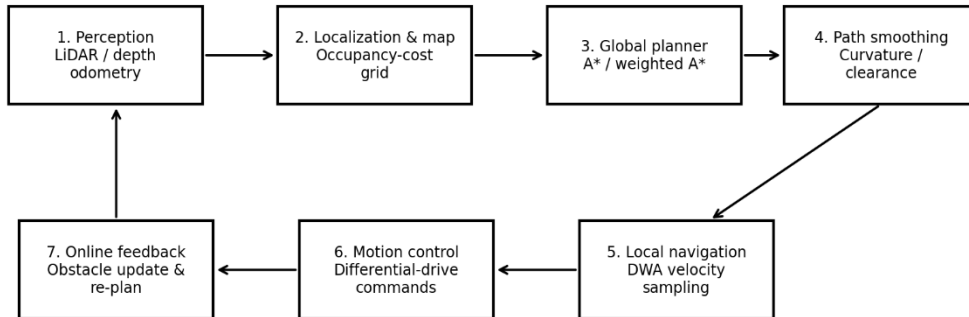


Figure 1. Step-by-step intelligent path planning and navigation architecture.

Step 3: Global Path Planning

Three global-search conditions are specified. Dijkstra search uses only accumulated path cost

and provides a heuristic-free shortest-path reference. Standard A* uses an octile-distance heuristic for eight-connected grid motion. The

proposed hybrid condition uses weighted A* with $f(n) = g(n) + w h(n)$, where $w = 1.25$ in the reported test. A weight above one intentionally prioritizes directed search and can reduce node expansion at the expense of strict optimality. All methods use identical collision checks, forbid diagonal corner cutting, and operate on the same grid.

Step 4: Path Smoothing and Feasibility

The raw grid path is post-processed using line-of-sight shortcutting subject to a clearance threshold. Starting from each retained waypoint, the algorithm selects the farthest subsequent waypoint that can be connected without intersecting an occupied or inflated cell. The resulting polyline decreases unnecessary intermediate turns. For a physical robot, the same waypoint sequence can be further converted into a curvature-continuous spline, following the principle of smooth path optimization demonstrated by Song et al. and Xu et al. [9,10].

Step 5: Local Navigation with DWA

The local planner follows the global reference while responding to obstacles not represented in the global map. At each cycle, DWA forms a dynamic window containing translational and angular velocities reachable from the current state within acceleration limits [4]. Candidate velocity pairs are forward-simulated for a short horizon. A score $J = \alpha J_{\text{goal}} + \beta J_{\text{path}} + \gamma J_{\text{clear}} + \delta J_{\text{speed}}$ ranks progress to the local goal, deviation from the global path, obstacle clearance, and forward speed. Any trajectory whose braking distance intersects an occupied region is rejected. The highest-scoring safe command is sent to the wheel controller.

Step 6: Intelligent Adaptation Layer

A DRL module may be inserted as an adaptive selector or local steering adviser rather than as

the sole safety mechanism. Its observation can include a local occupancy grid, goal direction, current velocity, and a short segment of the global path; its action may adjust DWA weights, select a subgoal, or suggest a velocity. This modular use is motivated by work showing that DRL can improve navigation behavior while still facing transfer and generalization challenges [7,8,11,12]. Explicit cost-map collision checks remain authoritative.

Step 7: Re-planning and Termination

The controller continuously checks whether the global route has become blocked, the robot deviates beyond a tolerance, or progress stalls. If so, the global planner is re-run from the current grid cell using the updated map. The trial ends successfully when the Euclidean distance to the goal is below 0.25 m and the robot is collision-free; it ends as a failure if a collision occurs, no feasible global path exists, or a fixed time limit is exceeded.

Results

The grid-search experiment produced the values in Table 2. Dijkstra and standard A* found paths of equal length on the selected map, confirming that the A* heuristic preserved the same shortest grid-route cost in this case. However, A* expanded substantially fewer nodes than Dijkstra. The weighted hybrid search further reduced node expansion, and line-of-sight smoothing reduced the number of heading changes in the final reference path. Because smoothing directly connects collision-free waypoints, the final continuous polyline can be slightly shorter than the unsmoothed grid path even when weighted A* itself initially follows a mildly suboptimal discrete route.

Table 2. Deterministic simulation results for the representative test map (30-run median timing).

Planner	Path length (m)	Expanded nodes	Median plan time (ms)	Heading changes	Min. clearance (m)
Dijkstra	21.37	2822	11.22	13	0.25
A*	21.37	1155	5.09	11	0.25
Hybrid weighted A* + smoothing	21.21	484	2.20	5	0.25

Compared with A*, the weighted hybrid search reduced expanded nodes by approximately 58.1% and median global-planning time by approximately 56.7% in this test. After smoothing, heading changes decreased by approximately 54.5%. The minimum reported grid clearance remained 0.25 m because one part

of every feasible route passes through a constrained corridor; in a physical robot this value should be increased through obstacle inflation before execution. These results should therefore be interpreted as a methodological demonstration rather than a universal ranking of algorithms.

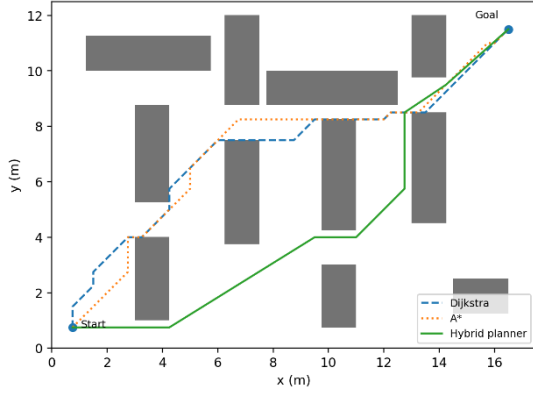


Figure 2. Representative global paths generated on the common occupancy-grid map.

The plotted paths illustrate the principal trade-off. Dijkstra explores broadly because it has no goal-directed heuristic. A* focuses the search while preserving the same grid-path length. The hybrid strategy accepts a more aggressive heuristic and then uses collision-checked shortcutting to create a compact waypoint set. For navigation in a dynamic setting, this reference path is not executed blindly; DWA samples safe velocity commands around the reference, and global re-planning is triggered if the local obstacle configuration invalidates the route.

The simulation outcome is consistent with the broader literature in three ways. First, heuristic graph search offers a strong baseline for structured AMR environments because it is interpretable and repeatable [1,6]. Second, path geometry affects execution quality: smooth or low-turn paths are easier to track and can reduce unnecessary acceleration and steering effort, motivating PSO/Bezier and related smoothing research [9,10]. Third, local adaptability must be evaluated separately from global shortest-path performance. A globally optimal path may still be unsafe if a person, cart, or pallet enters the route after planning, which is why DWA or an equivalent reactive planner remains part of the proposed architecture [4].

Conclusion and Discussion

This methodology paper presented a closed-loop framework for studying intelligent path planning and navigation techniques for autonomous mobile robots using references published no later than 2021. The framework separates perception and localization, global route generation, geometric smoothing, local collision avoidance, motion control, and re-planning. This decomposition makes it possible to compare algorithms fairly while preserving the operational requirements of a real robot.

Classical A* provides an interpretable goal-directed baseline [1], DWA provides velocity-level reactive avoidance [4], and sampling-based or potential-field methods provide alternative planning mechanisms for different configuration spaces [2,3,5]. The simulation demonstrates why hybridization is useful. Standard A* substantially reduces exploration relative to Dijkstra, while a weighted heuristic reduces search effort further. Smoothing then removes unnecessary directional changes, yielding a path that is more suitable as a tracking reference. The result does not imply that weighted A* is always optimal; its benefit depends on heuristic weight, map topology, and safety margin. Likewise, line-of-sight smoothing must be constrained by the robot footprint and localization uncertainty to avoid reducing clearance excessively. Intelligent learning methods expand the design space but should be evaluated with care. DRL-based path planning and navigation studies up to 2021 reported improvements in policy learning, end-to-end decision making, and adaptability [7,8,11,12], while reviews also highlighted the importance of training stability and real-world generalization [11,13]. For this reason, the proposed methodology treats DRL as a modular adaptation layer that may alter subgoals or planner weights while deterministic collision checking remains active. This gives researchers a practical way to test learning benefits without removing explicit safety constraints. For future experimental work, the simulation should be extended to ROS/Gazebo or an equivalent robotics platform with a differential-drive AMR, 2-D LiDAR, wheel odometry, and SLAM/localization. At least three map classes should be tested: open spaces, narrow corridors, and dynamic clutter. Each condition should be repeated with identical start-goal pairs and random obstacle motions. In addition to the metrics used here, real-robot trials should report traversal time, control effort, collision or emergency-stop count, localization error, and success rate. Statistical testing across repeated trials can then determine whether observed differences are significant rather than map-specific.

Overall, the recommended experimental architecture is a hybrid global-local planner: occupancy-cost mapping and A* supply transparent global guidance; smoothing improves path executability; DWA handles short-horizon dynamic constraints; and a learning layer can be added when adaptive behavior is required. This arrangement reflects the central conclusion of the pre-2022 literature: robust autonomous navigation is best treated as an integrated

decision-and-control problem rather than as a single path-search algorithm [6,11,13].

Research Reproducibility Note

The numerical values reported in Table 2 were generated specifically for this paper using a deterministic Python grid-search test with 8-connected motion, 0.25 m map resolution, identical start/goal cells, identical obstacles, and 30 repeated timing runs. The figure and table therefore serve as reproducible methodology outputs. They should not be represented as measurements from a physical robot until the same protocol is executed on hardware.

References

- 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. DOI: 10.1109/TSSC.1968.300136.
- O. Khatib, "Real-Time Obstacle Avoidance for Manipulators and Mobile Robots," *The International Journal of Robotics Research*, vol. 5, no. 1, pp. 90–98, 1986. DOI: 10.1177/027836498600500106.
- L. E. Kavraki, P. Svestka, J.-C. Latombe, and M. H. Overmars, "Probabilistic Roadmaps for Path Planning in High-Dimensional Configuration Spaces," *IEEE Transactions on Robotics and Automation*, vol. 12, no. 4, pp. 566–580, 1996. DOI: 10.1109/70.508439.
- D. Fox, W. Burgard, and S. Thrun, "The Dynamic Window Approach to Collision Avoidance," *IEEE Robotics & Automation Magazine*, vol. 4, no. 1, pp. 23–33, 1997. DOI: 10.1109/100.580977.
- S. Karaman and E. Frazzoli, "Sampling-Based Algorithms for Optimal Motion Planning," *The International Journal of Robotics Research*, vol. 30, no. 7, pp. 846–894, 2011. DOI: 10.1177/0278364911406761.
- B. K. Patle, L. Ganesh Babu, A. Pandey, D. R. K. Parhi, and A. Jagadeesh, "A Review: On Path Planning Strategies for Navigation of Mobile Robot," *Defence Technology*, vol. 15, no. 4, pp. 582–606, 2019. DOI: 10.1016/j.dt.2019.04.011.
- J. Gao, W. Ye, J. Guo, and Z. Li, "Deep Reinforcement Learning for Indoor Mobile Robot Path Planning," *Sensors*, vol. 20, no. 19, art. 5493, 2020. DOI: 10.3390/s20195493.
- J. Yu, Y. Su, and Y. Liao, "The Path Planning of Mobile Robot by Neural Networks and Hierarchical Reinforcement Learning," *Frontiers in Neurorobotics*, vol. 14, art. 63, 2020. DOI: 10.3389/fnbot.2020.00063.
- B. Song, Z. Wang, and L. Zou, "An Improved PSO Algorithm for Smooth Path Planning of Mobile Robots Using Continuous High-Degree Bezier Curve," *Applied Soft Computing*, vol. 100, art. 106960, 2021. DOI: 10.1016/j.asoc.2020.106960.
- L. Xu, B. Song, and M. Cao, "A New Approach to Optimal Smooth Path Planning of Mobile Robots with Continuous-Curvature Constraint," *Systems Science & Control Engineering*, vol. 9, no. 1, pp. 138–149, 2021. DOI: 10.1080/21642583.2021.1880985.
- K. Zhu and T. Zhang, "Deep Reinforcement Learning Based Mobile Robot Navigation: A Review," *Tsinghua Science and Technology*, vol. 26, no. 5, pp. 674–691, 2021. DOI: 10.26599/TST.2021.9010012.
- G. Chen, L. Pan, Y. Chen, P. Xu, Z. Wang, P. Wu, J. Ji, and X. Chen, "Deep Reinforcement Learning of Map-Based Obstacle Avoidance for Mobile Robot Navigation," *SN Computer Science*, vol. 2, art. 417, 2021. DOI: 10.1007/s42979-021-00817-z.
- Muhammad, M. A. H. Ali, and I. H. Shanono, "A Review: On Intelligent Mobile Robot Path Planning Techniques," in *2021 IEEE 11th IEEE Symposium on Computer Applications & Industrial Electronics (ISCAIE)*, pp. 53–58, 2021. DOI: 10.1109/ISCAIE51753.2021.9431819.