Transactions on Computer Science and Intelligent Systems Research
ISSN: 2960-1800, eISSN: 2960-2238 | Volume 5
AIDML 2024

Obstacle Avoidance Algorithm for Mobile Robots

Wenkai Xiao[1], Xinhao Ye[2, *]

[1]University of Shanghai for Science and Technology, Sino-German College, Shanghai, China.

[2]Shenzhen College of International Education, Shenzhen, China

*Corresponding author: s23803.Ye@stu.scie.com.cn

Abstract. Automatic car obstacle avoidance technology mainly utilizes advanced sensor technology to enhance the car's perception ability of the driving environment, feedback real-time information such as vehicle speed and position obtained by the perception system to the system, and judge and analyze potential safety hazards based on comprehensive information of road conditions and traffic flow. Mobile robots are increasingly used in applications, including in industrial production, agriculture, healthcare, rescue. However, mobile robots often face the challenge of avoiding obstacles while performing their tasks. Therefore, the development of intelligent obstacle avoidance algorithms is crucial to improve the autonomy, safety, and efficiency of mobile robots. In emergency situations, it automatically takes measures such as alarm prompts, braking or turning to assist and control the car to actively avoid obstacles, ensuring vehicle safety. At present, there are also many mature algorithms for local obstacle avoidance, and each algorithm has its own advantages and disadvantages. The current methods mainly include artificial potential field method (APF) and virtual force field method (VFF). This article aims to comprehensively review the latest research progress of obstacle avoidance algorithms for mobile robots. Chapter 2 introduces these methods in four different ways. The remaining part of the paper serves as the conclusion of the research and discusses future obstacle avoidance algorithms.

Keywords: Obstacle avoidance algorithm; A-star algorithm; DMA algorithm.

1. Introduction

In industrial production, mobile robots can replace workers to perform hazardous and repetitive tasks. In the field of agriculture, mobile robots can be used for automated planting, spraying pesticides, picking and other tasks. In healthcare, mobile robots can be used for tasks such as moving patients, delivering medication, and cleaning. In the rescue field, mobile robots can be used to perform tasks such as search and rescue. The intelligent obstacle avoidance algorithm needs to achieve three goals: environment perception, path planning and motion control. At present, domestic and international research on intelligent obstacle avoidance algorithms for mobile robots has achieved some results, but there are still some problems and challenges. Domestic research mainly focuses on sensor data processing, path planning and motion control. Foreign research focuses more on the use of machine learning and reinforcement learning methods to realize intelligent obstacle avoidance.

In terms of sensor data processing, researchers use sensors such as LIDAR and cameras to obtain environmental information, and process and analyze the data using image processing, point cloud processing and other technologies to extract the characteristics and location information of obstacles [1, 2].

In terms of path planning, researchers have proposed a series of algorithms, such as A* algorithm, D* algorithm, RRT algorithm and so on. These algorithms find the optimal obstacle avoidance path based on the environment information and the robot's dynamics.

In terms of motion control, researchers have proposed a number of control strategies, such as fuzzy control and PID control. These strategies control the robot's speed, direction, and other parameters according to the sensor data and path planning results, so that it can realize obstacle avoidance movement.

In conclusion, intelligent obstacle avoidance algorithms for mobile robots are of great significance to improve the autonomy, safety and efficiency of robots. At present, research on intelligent obstacle avoidance algorithms at home and abroad has achieved some results, but there are still some problems and challenges. Future research can be carried out in terms of sensor data processing, path planning and motion control to improve the performance and feasibility of intelligent obstacle avoidance algorithms [3].

Overall, the research on intelligent obstacle avoidance algorithms for mobile robots has made some progress, but there are still some problems that need to be solved. First, the accuracy and real-time nature of sensor data processing is the key to realizing intelligent obstacle avoidance. How to effectively process sensor data and accurately detect obstacles in the environment is an important research direction. Secondly, the efficiency and safety of path planning algorithms are also the focus of research. How to choose the optimal path to avoid collision and bypassing obstacles is a challenging problem. Finally, the accuracy and robustness of motion control algorithms also need further research. How to accurately control the robot's motion to cope with different environments and obstacles is a problem that needs to be solved.

Future research can be carried out in the following aspects: first, the sensor data processing algorithm can be further improved to enhance the accuracy and real-time detection. Second, more efficient and safe path planning algorithms can be investigated for more intelligent obstacle avoidance. Again, the application of methods such as machine learning and reinforcement learning in intelligent obstacle avoidance algorithms can be explored to improve the performance and robustness of the algorithms. Finally, more experiments and validations can be carried out to verify the feasibility and effectiveness of the intelligent obstacle avoidance algorithm. [4]. In order to solve the above problems, future research can be carried out in the following aspects: firstly, in-depth research on methods and techniques of sensor data processing to improve the accuracy and real-time performance of the data. Second, develop path planning algorithms adapted to complex environments, considering the robot's motion capability and the dynamic changes of the environment. Finally, research on methods and strategies to accurately control the robot's motion to achieve safer and more stable obstacle avoidance movement [3]. Research on intelligent obstacle avoidance algorithms for mobile robots at home and abroad has made some progress. Many research teams are committed to developing sensor-based obstacle avoidance algorithms. For example, LIDAR is utilized for environment sensing and obstacle detection, and then a safe path is selected for movement through path planning algorithms. In addition, there are researchers that utilize cameras for image processing and analysis to realize the perception of the environment and obstacle detection. In addition, obstacle avoidance algorithms based on machine learning methods have been widely studied. By training the dataset, the machine learning model can learn the association between the environment and the behavior to achieve intelligent obstacle avoidance. Recently, obstacle avoidance algorithms based on reinforcement learning have also become a hot research topic. Reinforcement learning can learn the optimal behavioral strategy through interaction with the environment, thus realizing intelligent obstacle avoidance.

2. Related Works

2.1. An Improved Algorithm base on A-Star Algorithm

This study proposes an improved A * algorithm that combines global and local path planning to solve the path planning and obstacle avoidance problems of mobile robots in dynamic environments. Simulation results show that the improved algorithm significantly improves convergence speed. This article combines global path planning with local path planning and discusses algorithms for mobile robots in dynamic environments. The improved A algorithm has been introduced, which improves the smoothness of the path and the convergence speed of the algorithm. This algorithm combines the improved A algorithm with the dynamic window method (DWA) to plan the global optimal path, provide guidance points, and use map information to plan the optimal path after obstacle avoidance. Through the fusion of two algorithms, the improved A_Star algorithm can extract key points on the global planning path as intermediate guidance points for the DWA algorithm, providing direction for the DWA algorithm in dynamic environments and avoiding getting stuck in local optima. The optimized DWA algorithm can also distinguish between dynamic and static obstacles, eliminate some known obstacles from interfering with the path, and further improve the algorithm's computational speed. The fusion algorithm can combine the advantages of two algorithms, with both obstacle avoidance function and the ability to plan the shortest path [5].

Assuming that v (t) and w (t) represent the linear and angular velocities of Turtlebot2 at time t in the world coordinate system, respectively. Within the sampling period ht, the displacement is small and can be considered as a uniform linear motion, the mathematical expression of the kinematic model is:

$$\begin{cases} x(t) = x(t-1) + v(t)\Delta t \cos[\theta(t-1)] \\ y(t) = y(t-1) + v(t)\Delta t \sin[\theta(t-1)]\theta(t) \\ \theta(t) = \theta(t-1) + w(t)\Delta t \end{cases} \tag{1}$$

The range expression for the search and solution of the DWA algorithm is:

$$V_t = \begin{Bmatrix} (v,w) \mid v_{\min} \le v \le v_{\max}, \\ w_{\min} \le w \le w_{\max} \end{Bmatrix} \tag{2}$$

The experimental platform is MATLAB 2016b environment, and a random grid diagram was created, where the white grid represents the unobstructed area and the black grid represents the obstructed area. By conducting comparative experiments with Dijkstra algorithm and BFS algorithm, the path length and time performance of A algorithm, Dijkstra's algorithm, and BFS algorithm were evaluated in different types of scenarios. The final experimental results show that compared with the traditional A algorithm, the improved A algorithm reduces convergence time by an average of 70% and slightly increases path length by 2.4%. The fusion algorithm can achieve local obstacle avoidance based on global path planning and can quickly converge to the global path. Table 1 showed the performance comparison of four path planning algorithms.

Table 1. Performance comparison of four path planning algorithms

Time / Type	1		2		3	
	Length/m	Time/s	Length/m	Time/s	Length/m	Time/s
A-Star	59.0	0.7	51.5	0.4	43.6	0.2
Dijkstra	59.0	1.3	51.3	1.3	43.6	3.0
BFS	66.0	0.1	54.9	0.1	46.5	0.1
New A-Star	60.4	0.2	53.1	0.1	45.1	0.06

2.2. The AGV Path Planning of Improved A* Algorithm

The manufacturing industry is the pillar of China's economy, and in order to achieve the transformation and upgrading of the manufacturing industry, the demand for AGV path planning capabilities in intelligent manufacturing has increased. In the past, traditional A * algorithms were used for AGV cars, but such algorithms had problems such as low search efficiency and multiple redundant nodes, which needed to be improved to meet the needs of intelligent manufacturing. In order to improve the efficiency and smoothness of AGV path planning, this study combines the improved A * algorithm with the dynamic window method to achieve a globally optimal smooth path planning. To address these issues, the traditional A algorithm has been improved as follows [6, 7]:

Extend the search neighborhood of the A * algorithm to 0.25 π of the search direction angle to enhance search efficiency.

Improve heuristic functions to reduce the difference between cost values and actual costs, in order to reduce path redundancy and robot diagonal crossing obstacles, thereby avoiding faults.

Using the vertical distance constraint method to eliminate redundant nodes, and finally optimizing the path with a 3-order uniform curve to make the path more in line with the kinematic requirements of the robot.

Compared to the traditional A * algorithm, the path planning time is longer and there are more turning points in the path. To optimize h (n), the value of Manhattan distance is appropriately increased to improve the efficiency of path planning. To avoid unnecessary turning, a turning penalty function is introduced:

$$Z(n) = (d_x + d_y) + k \times \min(d_x, d_y) \tag{3}$$

The improved cost function is shown in the formula:

$$f(n) = g(n) + h(n)(1.0 + \frac{\sqrt{2}}{R}) + Z(n) \tag{4}$$

In the experimental simulation, a 5x5 grid will be used to expand the search area to enhance search efficiency and remove unnecessary sub nodes. Improve the cost function by combining Manhattan distance and penalty function to reduce path planning time and unnecessary turns.

Simulated through experiments:

Compared with traditional A algorithm, the optimized A algorithm reduces path length by 12.49%, path planning time by 36.23%, and number of turns by 62.5%. The optimized A * algorithm combined with the DWA algorithm verified the dynamic obstacle avoidance ability in a dynamic environment, achieving safe navigation from the initial point to the target point, avoiding obstacles, and completing dynamic obstacle avoidance. By introducing dynamic and unknown obstacles, the improved A * algorithm combined with DWA algorithm reduced path length and time by 8.93% and 5.96%, respectively. These results validate the feasibility and efficiency of the algorithm in AGV operations. Figures 1 and 2 showed the traditional A * algorithm and improved A * algorithm results, respectively.

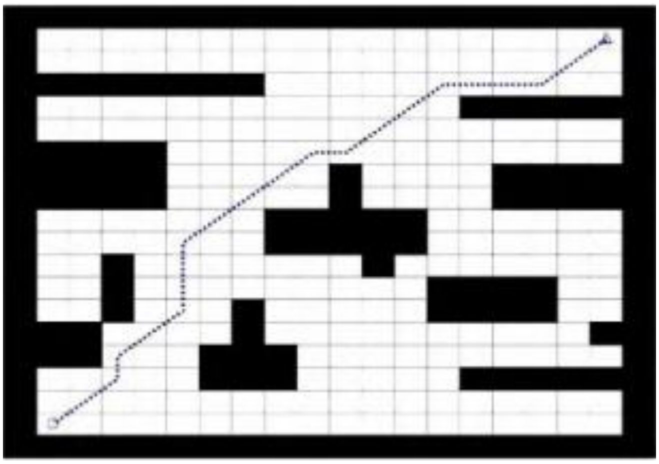

Fig. 1: Traditional A * algorithm [8]

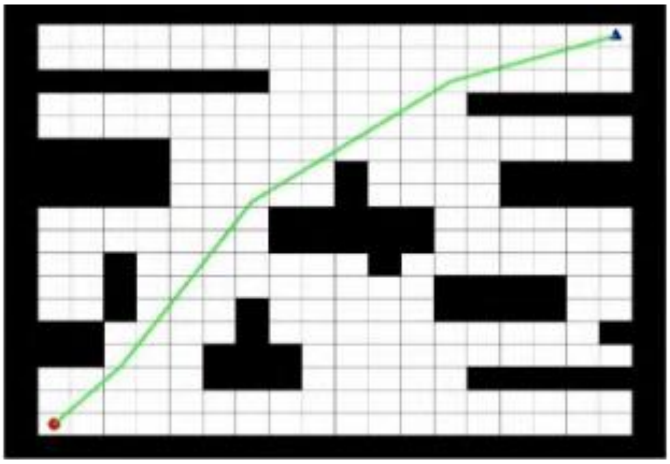

Fig. 2 Improved A * algorithm [8]

2.3. DWA Algorithm

This study proposes an obstacle avoidance path planning method for distribution network operation robots based on the DWA algorithm. By constructing the robot's kinematic model, designing the velocity vector space, and evaluating the function, precise obstacle avoidance is achieved in different operating environments and the shortest path is obtained. In recent years, power supply enterprises have continuously strengthened their power grid operation capabilities to improve the domestic business environment and economic benefits. However, manual operation intensity is high, and the risk coefficient is high, which may pose a threat to the safety of employees. Therefore, intelligent technology is needed to improve work efficiency.

This study aims to solve the problems of poor obstacle avoidance ability and path redundancy of distribution network operation robots in different working environments, and proposes a method for obstacle avoidance path planning of distribution network operation robots based on the DWA algorithm. Introduced the obstacle avoidance path planning method for distribution network operation robots based on DWA algorithm. This method involves constructing a robot kinematic model based on mass, wheel radius, and wheel spacing information to describe the robot's posture at different times. It also determines the workspace of the robot, constructs a local map that combines the starting and ending points, and limits the robot's range of motion.

The typical evaluation function for DWA algorithm is:

$$G(V,\omega) = \kappa(\alpha \cdot h(V,\omega) + \beta \cdot d(V,\omega)) + \gamma \cdot v(V,\omega) + \kappa(V,\omega) \tag{5}$$

In the formula: α、 β and γ The weights of the evaluation functions for direction, obstacle clearance, and speed; κ Is the smoothing coefficient; H (V, ω), D (V, ω) and v (V, ω) The evaluation functions are direction, obstacle distance, and speed, respectively.

1) Direction evaluation function.

Function h (V, ω) It is to simulate a forward path and calculate the distance between the tangent line at the end of the path and the target point, with a known sampling speed. The expression is:

$$h(V,w) = 180° - \frac{\varphi}{2} \tag{6}$$

In the formula: φ The angle difference between the actual path and the ideal path of the robot. φ The smaller the value, the function h (V, ω) The higher the score. Since the estimated position depends on speed, h (V), ω) It is also related to speed.

2) Obstacle distance evaluation function.

Function d (V, ω) Describe the distance between the motion path and obstacles. The smaller the value, the more likely the robot is to collide with obstacles; On the contrary, the larger the spacing, the more d (V, ω) The higher the score, the safer the robot path will be. If the function score is determined by a certain threshold, then:

$$d(V,w)=\begin{cases} D,(D<L) \\ L,(D\geq L) \end{cases} \tag{7}$$

In the formula: D is the distance between a certain point on the path and an obstacle; L is the set threshold.If the distance is far, d (V, ω) The output is L. It can avoid considering only the distance of obstacles and ignoring other factors when choosing the optimal path.

3) Speed evaluation function.

Function v (V, ω) Can reduce the time it takes for the robot to reach the finish line, expressed as:

$$v(V,w)=\left|V_g\right| \tag{8}$$

In the formula, V_g is the forward linear velocity.

In the evaluation function G (V, ω) Among them, the above three sub functions are indispensable in jointly constraining the robot to move towards the target point. Robots can only avoid obstacles and reach the finish line when driving under these constraints.

This method designs a velocity vector space for the DWA algorithm, generates multiple paths for selection within the velocity constraint area, and evaluates the paths using direction, obstacle distance, and velocity sub functions. Choose the path with the highest evaluation value as the optimal obstacle avoidance planning path. This article also discusses the obstacle avoidance path planning process of the DWA algorithm in the robot operating environment, including creating a local map based on known starting and ending points, setting obstacle vertices, connecting vertices according to specific rules, sampling the robot's speed, simulating different trajectories, and selecting the highest rated obstacle avoidance path. The key role of the DWA algorithm is to use sliding windows in the search space to transform obstacle avoidance problems into velocity constrained problems. This article also describes the design of evaluation functions for speed sampling, speed constraints, feasible collision range, and selecting the optimal path.

2.4. APF Method

The traditional artificial potential field method is a commonly used path planning algorithm, which has the advantages of small computational complexity, fast response speed, and smooth path. This study proposes an improved artificial potential field (APF) method for robot obstacle avoidance path planning, which solves the problem of unreachable targets in traditional APF methods by improving the repulsion field function, introducing dynamic distance parameters and adjusting factor coefficients. Meanwhile, by introducing the adaptive tangent deviation angle method and virtual obstacle points, the problem of robots easily getting stuck in local minima in complex obstacles has been solved. The effectiveness of the improved APF method in path planning was verified through simulation and physical demonstrations.

Introduce the dynamic influence distance parameter between the robot and the target point, and substitute it into the repulsive field function within the repulsive range of each obstacle:

$$U_{rep}(j)=\begin{cases}\frac{1}{2}mtR_d(\frac{1}{\|j-j_{obs}\|}-\frac{1}{d_i})^2, \|j-j_{obs}\|\leq d_i^0\\0, \|j-j_{obs}\|\geq d_i^0\end{cases} \tag{9}$$

Among them, Rd represents the distance between the robot and the target point, t is the adjustment factor of the obstacle repulsion field, with a value range of 1 ≤ t ≤ 2, generally taken as 2. The repulsion function is obtained as shown in equation:

$$F_{rep}(j)=-\nabla U_{all}(j)=\begin{cases}F_m(j)+F_y(j), \|j-j_{obs}\|\leq d_i^0\\0, \|j-j_{obs}\|\geq d_i^0\end{cases} \tag{9}$$

$$\begin{cases}F_m(j)=mt(\frac{1}{\|j-j_{obs}\|}-\frac{1}{d_i^0})\frac{1}{(j-j_{obs})^2}R_d\frac{\partial\|j-j_{obs}\|}{\partial j}\\F_n(j)=\frac{1}{2}mt(\frac{1}{\|j-j_{obs}\|}-\frac{1}{d_i^0})^2\frac{\partial\|j-j_{obs}\|}{\partial j}\end{cases} \tag{10}$$

Among them, Fm (j) indicates that the robot is facing opposite to the obstacle, and Fn (j) indicates that the robot is facing in the same direction as the obstacle.

Considering the improvement of the repulsive function to break the balance of forces, the coordination force Fx is introduced, which is defined as:

$$F_x=\frac{1}{2}\eta(j-j_{obj})(\frac{1}{\|j-j_{obs}\|}-\frac{1}{d_i^0})^2 \tag{11}$$

In the formula, η For the coordination coefficient, η The range of values for is 0 ≤ η ≤ 1. Generally, the value is 0.5. Therefore, the improved artificial potential field resultant force function is:

$$F(j)=F_{all}(j)+F_{rep}^i(j)+F_x \tag{12}$$

Meanwhile, by introducing the adaptive tangent deviation angle method and virtual obstacle points, the problem of robots easily getting stuck in local minima has been solved. The effectiveness of the improved APF method in path planning was verified through simulation and physical demonstrations.

This article proposes an improved artificial potential field method for robot obstacle avoidance path planning. By introducing dynamic distance parameters between robots and obstacles, the traditional repulsive field function is enhanced to solve the problem of unreachable targets. In order to solve the problem of local minima caused by oscillations around complex obstacles, an adaptive tangent deviation angle method is introduced. This method shows better performance in optimizing path smoothness and reducing computation time. The effectiveness of the improved active filter algorithm was verified through MATLAB simulation and physical demonstration. And an improved APF algorithm was introduced to determine virtual obstacle points using adaptive tangent deviation angle. This algorithm enhances the repulsion function by considering the dynamic influence distance between the robot and the target point, as well as adjusting the factor coefficients.The specific steps are as follows:

Step 1: Initialize the grid map and algorithm parameters.

Step 2: Determine the initial position of the robot in the grid map and the target point of the robot.

Step 3: Use the traditional APF method to calculate the repulsive and gravitational fields of the robot at each step during its movement.

Step 4: Calculate the repulsive force and the gravitational force of the target point on the obstacles around the robot to obtain the resultant force, determine whether the resultant force is zero, and then determine whether one of the two major problems of the traditional APF method has been encountered. Use an improved repulsive field function to find a new local path; By using robots to determine, when oscillation occurs in the simulation and falls into local minimum problems, adaptive deflection angle is utilized ζ Identify the virtual obstacle point located on the other side of the farthest obstacle and add additional repulsion to help the robot overcome local minima.

Step 5: Calculate the robot's forward position at the next moment using the APF method, and finally reach the endpoint. If the above problem occurs again, return to step 4 to search for a new path for the next step.

From Table 2-4, it can be seen that the TAPF algorithm often falls into local minima and dead ends when facing complex obstacles, while the A-APF algorithm and the algorithm proposed in this paper are both successful in planning complete paths. It can be seen that in a 20x20 grid map environment, due to complex obstacles, the TAPF algorithm cannot plan a complete path, while the algorithm proposed in this paper performs better.

Table 2. 20×20 Simulation data in complex environmental maps

Algorithm	Step	Time/s
TAPF	/	/
A-APF	28	2.43
This article's algorithm	24	2.11

Table 3. 25×25 Simulation data in complex environmental maps

Algorithm	Step	Time/s
TAPF	35	2.81
A-APF	29	2.46
This article's algorithm	23	2.07

Table 4. 30×30 Simulation data in complex environmental maps

Algorithm	Step	Time/s
TAPF	/	/
A-APF	62	7.2
This article's algorithm	56	6.3

3. Discussion

The A * algorithm is a commonly used path finding and graph traversal algorithm that can find a solution within a limited search space. If a solution exists, it can definitely be found, and under certain conditions, A * can find the optimal solution, that is, the shortest path. The application range of A * algorithm is also very wide, which is suitable for various types of problem fields, including path planning, game AI, etc. At the same time, suitable heuristic functions can be selected based on specific problems to balance search efficiency and optimality.

However, the performance of A * highly depends on the design of heuristic functions, and different heuristic functions may lead to different search efficiency and optimality. And A * needs to store the nodes that have already been accessed and their status information, which may require a large amount of memory for large-scale problems. Finally, if the state of the environment changes during the search process, A * may not be able to update and find the optimal solution in a timely manner.

The Dijkstra algorithm is a classic algorithm used to solve the single source shortest path problem, which can find the shortest path from the starting node to all other nodes, making it very effective in solving the shortest path problem. And the algorithm has a simple idea, is easy to understand and implement, and is suitable for graphs of various scales. Like the A * algorithm, it is not only applicable to directed graphs, but also to undirected graphs, and can handle graphs with weights.

At the same time, the Dijkstra algorithm needs to maintain an auxiliary array to record the shortest path length and path information of each node, so it may occupy more memory space in large-scale graphs. In each iteration, the Dijkstra algorithm needs to traverse all unreachable nodes to find the next node in the current shortest path, so its time complexity is high. And the Dijkstra algorithm can only be used for static graphs. When the structure of the graph or the weight of edges changes, the entire shortest path needs to be recalculated, so it is not suitable for dynamic network scenarios.

The DAW algorithm is a dynamic weight A * algorithm that combines the advantages of A * algorithm and Dijkstra algorithm to solve path planning problems. Inheriting the completeness and optimality of the A * algorithm, it can flexibly balance the trade-off between search speed and optimal solution during the search process. And due to dynamically adjusting weights, it is more suitable for solving situations with changes or uncertainties in problems, such as path planning in dynamic environments.

However, its heuristic function design is very complex, and due to the fact that weight parameters may have a significant impact on the performance of the algorithm, it is necessary to continuously adjust the weights during the search process, which may result in high computational costs for the DAW algorithm, especially in complex search spaces.

In the future, research on intelligent obstacle avoidance algorithms for mobile robots can be further deepened through the following aspects. Firstly, sensor data processing methods can be further optimized to improve the accuracy and real-time performance of obstacle avoidance algorithms. Secondly, more machine learning and reinforcement learning methods can be explored to improve the adaptability and intelligence of obstacle avoidance algorithms.

4. Conclusion

The research on intelligent obstacle avoidance algorithms for mobile robots is a constantly developing field. With the wide range of robot applications and the increasing demand, higher requirements have been put forward for the research of intelligent obstacle avoidance methods. Future research should focus on the following directions and priorities: Nowadays, only a single sensor is used, which may result in issues such as noise and blind spots in the received information. Therefore, in future research, it can be considered to fuse data from multiple sensors to improve the accuracy and reliability of obstacle avoidance algorithms. Deep learning can efficiently learn and classify features of image and video data. Therefore, applying deep learning technology to intelligent obstacle avoidance algorithms for mobile robots can further improve the performance and adaptability of the algorithms. In addition, environmental modeling and path planning are also very important for the research of robot obstacle avoidance. Future research can explore how to use high-precision maps and advanced path planning algorithms to improve the obstacle avoidance and navigation capabilities of robots. Finally, mobile robots usually need to avoid obstacles in situations with high real-time requirements, so when studying intelligent obstacle avoidance algorithms, the real-time and effectiveness of the algorithms need to be considered. During the operation, there may be some unexpected situations that require the algorithm to complete calculations and respond in a very short time. Future research can explore how to improve the real-time performance and efficiency of algorithms by optimizing their structure and implementation.

In summary, future research on intelligent obstacle avoidance algorithms for mobile robots should focus on the application of multi-sensor data fusion, deep learning and reinforcement learning, optimization of environment modeling and path planning, and improvement of real-time performance and efficiency. These research directions and focuses will contribute to improving the obstacle avoidance ability and autonomy of mobile robots in complex environments, promoting the development and application of mobile robots.

Authors Contribution

The authors of this paper are all co-authors and have made the same contributions in this paper.

References

[1] Liao B, Wan F, Hua Y, Ma R, Zhu S, Qing X. F-RRT*: An improved path planning algorithm with improved initial solution and convergence rate[J]. Expert Systems With Applications, 2021, 158-160.

[2] Deng X, Li R, Zhao L, Wang K, Gui X. Multi-obstacle path planning and optimization for mobile robot[J]. Expert Systems With Applications, 2021.

[3] Zong C, Han X, Zhang D, Liu Y, Zhao W, Sun M. Research on local path planning based on improved RRT algorithm[J]. Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering, 2021.

[4] Wei L, Pan D, Liao M. Application of Reverse Car-seeking in Large Underground Parking Lot Based on A Star Algorithm: A Real Case. Scientific Research and Reviews, 2020. 51-60.

[5] Khalid H, Naveed I, Tanzila S, Amjad R, Zahid M. LSDAR: A light-weight structure based data aggregation routing protocol with secure internet of things integrated next-generation sensor networks[J]. Sustainable Cities and Society, 2020.

[6] Siregar B, et al. Food Delivery System with the Utilization of Vehicle Using Geographical Information System (GIS) and A Star Algorithm[J]. Journal of Physics: Conference Series, 2017.

[7] Patience I. Adamu;;Hilary I. Okagbue;;Pelumi E. Oguntunde.Fast and Optimal Path Planning Algorithm (FAOPPA) for a Mobile Robot[J]. Wireless Personal Communications, 2019.

[8] Zhao Y, Wang X, Chen K. An AGV path planning's mix improved A* algorithm [J]. Si Chuang university of Engineering and science (Natural Science), 2024, 37(01):71-78.