Application of Genetic Algorithm and Convolutional Neural Network in Mobile Robot Path Planning

Tao Di, Xiaodong Zhang*

Department of Mechanical and Electrical Engineering, Lanzhou Vocational Technical College, China
* Correspondence author: Xiaodong Zhang

Abstract: This paper proposes a method that integrates genetic algorithm and neural network to complete the path planning of mobile robot in unknown dynamic environment. In this method, the movement of the robot is controlled by a neural network, and the environment information is obtained through five sensors mounted on the front end. The obtained obstacles, pose and target information are used as the input of the neural network, and then the genetic algorithm is used to train and adjust the neural network. The weight, and finally the output of the neural network after training and adjustment is used as the driving control force of the robot. The simulation results show that the robot can find a smooth obstacle avoidance path in a short time, and complete a smooth and efficient movement from the starting point to the target point.

Keywords: neural networks; genetic algorithm moving robot; path planning

1. Introduction

Traditional calculation methods are simple and easy to implement, but the adaptability of these methods is not strong enough. For example, the model may need to be reconstructed when the environmental space situation changes [1]. As the dimensionality of the environment increases and the number of obstacles increases, the computational complexity will also increase significantly, the operation cycle will be lengthened, and the processing efficiency will also be greatly reduced. This article studies the application of intelligent algorithm-neural network method in mobile robot path planning [2].

A moving robot is a highly intelligent system that integrates functions such as information perception, dynamic decision-making, and behavior control. Moving robots have very ideal research prospects in many fields, among which representative ones include home services, space exploration, and medical rehabilitation [3]. For these fields, mobile robots mainly face dynamic and unknown environments. In order to complete the tasks in the above-mentioned complex environments, mobile robots should have basic autonomous navigation capabilities, and path planning is important for mobile robots to achieve autonomous navigation [4]. The problem is also a necessary prerequisite for the realization of other tasks [5]. Path planning is to move from the starting point to the target point, and ensure the safety of the robot and the objects in the environment during the movement. At this stage, path planning in a known static environment is relatively complete. However, in actual situations, mobile robots are often faced with an unknown dynamic environment. The existing path planning methods in an unknown dynamic environment often only focus on "Feasibility", but ignores the performance target requirements such as the "length" and "smoothness" of the planned path, which results in the robot's movement not being smooth and efficient [6]. Mobile robot model and motion space definition. This article is based on the AS-R capability storm robot [7]. The front end of the robot is equipped with 5 sonar ranging sensors to determine obstacles and direction information.

Obstacles in the working environment are represented by polygons, and whether an obstacle is detected is judged by whether the sensor line segment and the polygon line segment intersect [8]. Because the robot needs to fuse the information obtained by the sensor to determine the next movement direction during the movement, there must be a return value when using the sensor to obtain environmental information, which is expressed as follows:

$$\mathrm{f(sensor)} = \begin{cases} -1(\text{No obstacles detected}) \\ \mathrm{l/d}(\text{Obstacles detected}) \end{cases}$$

Among them, d is the perception distance, and l is the distance between the intersection of the line segments and the center of the robot. If the return value is positive (obstacles are detected) and the return value is less than the previously set safety distance (set the safety distance as the ratio of the shape width to the perceived distance, the size is 3/20), in order to achieve safe obstacle avoidance, this When the mobile robot should stop moving and rotate a certain angle to achieve safe movement [9].

2. Neural Network Controller Structure

Artificial Neural Network (Artificial Neural Network, or ANN), started research in the 1980s, has developed rapidly, and has attracted the attention of many scholars [10]. Path planning reflects the connection relationship between environment modeling and function space. This relationship is relatively complicated and it is difficult to directly connect them with accurate mathematical concepts. In neural network path planning, the main advantage of this method is that the environment space is easy to represent. In the neural network diagram, the data collected by the sensor is input through the input layer of the neural network, and the expected direction angle increment of the mobile robot movement is used as the network. The output of the final position information is used as a sample set. The definition of artificial neural network can be summarized as: Artificial neurons composed of a large number of simple electronic components or optical components are connected to form a complex structure called a neural network [11]. Each component has input and output functions, and the connection between the components is the synapse (connection Bond), the size of neuron interaction is determined by the connection bond. The interaction between them is divided into three types: excitatory effect, inhibitory effect and no effect [12]. After setting the parameters of each level, a complex neural network has the function of learning, training or self-organization.

This article uses neural network as the motion controller of the mobile robot. The entire network structure includes three layers: input layer, hidden layer and output layer: The input layer has 11 neurons, which are used to correspond to environmental obstacle information (5), target position information (5) and target position information (1). The hidden layer has 20 neurons, and the output layer has only 2 neurons, which are used to control the movement of the left and right wheels of the mobile robot. Using a unipolar sigmoid function as the excitation function, assuming that the current coordinates of the robot in the working environment are (x_i, y_i), the updated coordinates of the robot at the next moment can be calculated as:

$$(x_{i+1}+y_{i+1}) = \left(x_i + v_{i+1}\times(-\sin\theta_{i+1})\right), y_i + v_{i+1}\times\cos\theta_{i+1}$$

Among them, v_{i+1} is the moving speed of the robot, and θ_{i+1} is the moving direction angle of the robot. In order to display the robot's planned movement path in real time, an array is used to save the coordinate information of the robot at different time points, and it is displayed as a small red square [13].

3. Genetic Manipulation

3.1. Chromosome Coding

Encode the connection weights of all neurons in the neural network controller into chromosomes. The length of the chromosome is the number of weights in the neural network. Before starting iterative training using genetic algorithms, set the weights to (-1,1) Random real numbers in the interval [14].

3.2. Fitness Function Design

The route planning of a mobile robot is to move from the starting point to the target point, and to ensure the safety of the robot and the objects in the environment during the movement. Under normal circumstances, multiple such paths can be planned. In research applications, an optimal path is usually selected, and the optimal evaluation indicators include: The length of the path, the length of the planning time, the smoothness of the path and whether it is safe to avoid obstacles. Therefore, when designing the fitness function, these several evaluation indicators are taken into consideration, and the fitness function is determined as:

$$f = (7000 - t) + (1200 - l) + c/30 + s/20$$

Among them, t is the time required to reach the target point, l is the distance from the target point, s and c are rotation compensation and collision compensation, respectively. If the robot does not collide with obstacles during the movement, and the rotation angle is less than the set minimum when the value is 0.03, its value is increased by 1.

3.3. Select Operation

The genetic algorithm completes the survival of the fittest among population individuals through selection operations. The purpose of selection operations is to use a certain method to select some excellent individuals from the parent population and pass them to the next generation population, so that the excellent performance can be preserved. This paper adopts the tournament selection method. First, select 5 individuals in the population arbitrarily, and then take the individual with the highest fitness value as the parent individual, and repeat the process until the individual selection is completed. Because the optimal individual may become inferior due to the cross-mutation operation in the iterative process, in the newly generated population, for individuals with a smaller fitness value due to cross-mutation, use individuals with a relatively larger fitness value. Replacement, to ensure that the number of good individuals in the population increases, and the iterative process continues to evolve in a better direction [14].

3.4. Cross Mutation Operation

The crossover operation is to exchange part of the content of two matched chromosomes with a certain probability Pc in a certain way, thereby generating two new chromosomes. Crossover operation is an important method for generating new chromosomes in genetic algorithms. The weights of each neuron in the neural network controller are combined into a whole, represented by a small rectangular block, as shown in Figure 1, and then these weights are rectangular The position between the blocks is used as the intersection position. This paper adopts the two-point crossover method, assuming that the number of neurons in the controller is N, first calculate the possible N-1 crossover positions, and then randomly determine the two crossover positions to complete the crossover operation to generate new individuals, and ensure the completeness of the population.

Before crossing:

Figure 1. Crossover process

At the same time, in order to maintain the individual diversity of the population and avoid entering the local optimal solution in the iterative evolution process, non-uniform mutation is used, and the maximum value of the mutation is set to 0.3. Algorithm design and implementation of genetic control algorithm flow chart shown in Figure 2.

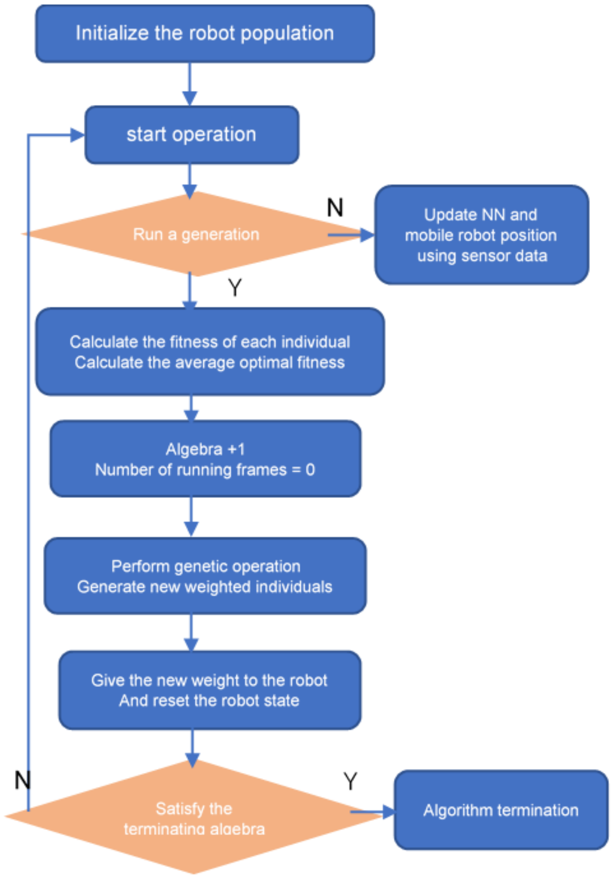

Figure 2. Flow chart of genetic control algorithm

4. Simulation Experiment and Results

The population size is set to 50, the crossover probability is 0.65, the mutation probability is 0.075, the number of running frames per generation is 1 200, and the maximum evolution generation number is 300. Simulation experiments are carried out in an unknown static environment and an unknown dynamic environment, respectively, as shown in Figure 3. Figure 4 is the simulation result.

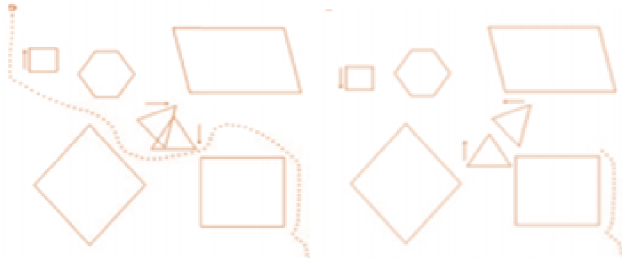

Figure 3. The planned path in an unknown static environment

It can be seen from Figure 3 that in an unknown static environment, a smooth and efficient safe path can be searched when the iteration reaches about 60 generations.

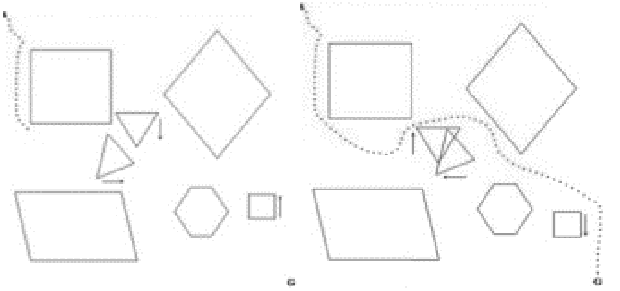

Figure 4. Planning path in an unknown dynamic environment

Where: R_F is the defect detected by low-rank matrix decomposition, RR is the defect manually marked, and R_{FR} is the intersection of R_F and R_R. Precision is used to evaluate the accuracy of the low-rank matrix factorization model. It is the proportion between the real defects detected by the algorithm in this paper and all the defects detected by the algorithm. Recall is used to evaluate the ability of the algorithm to detect all defects and is the detection algorithm. F-Measure is used to evaluate the overall defect detection capability of the detection algorithm, which is the ratio between the real defect part of the detected defects and the artificially marked defects (all real defects). It is a combination of Precision evaluation and Recall evaluation. By selecting 100 different types of bearings (no defects, surface rust spots, scratches, inclusion defects) for low-rank matrix decomposition defect detection, the precision, recall and F-Measure statistical results are shown in Table 1. It can be seen that the low-rank matrix decomposition detection algorithm for bearing surface defects has a low missed detection rate and high accuracy, which basically meets the defect detection requirements.

Table 1. Accurate results of algorithm detection

method	Precision	Recall	F-Measure
Algorithm	0.854	0.815	0.834

5. Conclusion

From the simulation, this method has successfully realized the obstacle avoidance function of the mobile robot in square obstacles and circular obstacles. Compared with other methods, this method has the advantages of shorter iteration cycle, faster calculation speed, and easier implementation on hardware. But this method also has the following disadvantages:

(1) When some parameters are set improperly or obstacles are more complicated, it is easy to form a local optimal route, and the optimal route for the entire operating environment cannot be obtained.

(2) Learning new obstacle patterns during neural network training may interfere with previous samples.

The space operating environment studied in this paper is relatively simple. In the future research, we can increase the diversity of space obstacles and seek better ways to solve the problem of local optimization.

In order to effectively improve the accuracy of bearing surface defect detection, according to the characteristics of bearing surface defects, a low-rank matrix decomposition-based bearing surface defect detection method is proposed, which decomposes the bearing defect image into a low-rank background image and a sparse foreground Image, the size and location of bearing surface defects are judged by thresholding the defect foreground image. Experimental research results show that compared with existing algorithms, this method can accurately and effectively detect bearing surface defects and meet the requirements of bearing surface defect detection.

Acknowledgment

The authors wish to thank the 2019 Gansu Province Colleges and Universities Innovation Capability Improvement Project: "Research on Application of Multi-robot Scheduling for Flexible Handling System" (No.: 2019A-243)

References

[1] Chen, J.G.; Chen, Hao; Zhang, Ben. Detection of bearing roller surface defects based on improved Niblack algorithm. Modular Machine Tool and Automatic Processing Technology, **2018** (12): 82-85+97.

[2] Luo, T.B.; Chen, S.; Zhao, Z.Y.; et al. Research on defect detection of bearing sleeves based on machine vision algorithm. Mechanical and Electrical Engineering, **2018**, 35(2): 148-152.

[3] Lu, M.H.; Fan, S.; Tang, Q.T. Bearing ring inspection system based on machine vision. Bearing, **2017** (5): 39-44.

[4] Zhou, X.J.; Tu, H.B.; Hu, G.L. An improved FCM clustering detection algorithm for bearing defect images. Foundry Technology, **2006** (12): 1395-1399.

[5] Cai, W.W. Research on detection technology of bearing appearance surface defects. Luoyang: Henan University of Science and Technology, **2011**.

[6] Zheng, Y. Research on the surface defect detection system of bearing inner ring based on machine vision. Shen Yang: Shenyang University of Technology, **2017**.

[7] Zhang, L.N.; Cen, Y.G. Low-rank matrix decomposition defect detection of rail images. Signal Processing, **2019**, 35(4): 667-675.

[8] Tang, X.; Huang, J.W. Low-rank robust principal component analysis for occlusion face recognition. Journal of Nanjing University of Science and Technology, **2017**, 41(4): 460-465.

[9] Li, C.L.; Wang, J.P.; Liu, Z.F.; et al. Fabric defect detection method based on dual norm low-rank decomposition. Cotton Textile Technology, **2019**, 47(1): 5-10.

[10] Fujimori, A.; Nikiforuk, P.N.; Gupta, M.M. Adaptive navigation of mobile robots with obstacle avoidance. IEEE Transactions on Robotics and Automation, **1997**, (134): 596-601.

[11] Liang, Y.M.; Xu, L.H. Research status and development trend of mobile robot path planning technology. Mechatronics, **2009**, 15(3): 35-38.

[12] Wang, Y.X.; Ding, R.; Yao, L.Q. Blind path finding algorithm for mobile robot global path planning. Computer Simulation, **2010**, 27(5): 157-161.

[13] Zhang, X.L. Robot path planning method based on differential evolution algorithm. Dalian University of Technology, **2006**.

[14] Liao, S.H.; Zhang, L.D. The status quo and development trend of robot path planning technology. Mechanical Engineer, **2007**(7): 13-16.