2021 International Conference on Computer Information Science and Artificial Intelligence (CISAI) | 978-1-6654-0692-5/21/$31.00 ©2021 IEEE | DOI: 10.1109/CISAI54367.2021.00159

Research on Localization and Path Planning of Indoor Robot Based on ROS

XiaoHong Wang
School of Information Science and Engineering
Hebei University of Science and Technology
Shijiazhuang, China
feifeiwangxiaohong@126.com

XuePei Ma
School of Information Science and Engineering
Hebei University of Science and Technology
Shijiazhuang, China
mxp_studio@163.com

ShengLan Zhu
School of Information Science and Engineering
Hebei University of Science and Technology
Shijiazhuang, China
zhushenglan0521@163.com

Kun Zhang*
School of Information Science and Engineering
Hebei University of Science and Technology
Shijiazhuang, China
* zhangkun@hebust.edu.cn

YaNan Sun
School of Information Science and Engineering
Hebei University of Science and Technology
Shijiazhuang, China
synhbkd@163.com

***Abstract*—To meet the needs of monitoring the indoor environment, a mobile robot that patrols indoors is designed. The mobile robot uses Raspberry Pi equipped with a ROS system as the control core, uses lidar to collect environmental information, and uses various function packages to develop algorithms under the ROS distributed framework. It is based on the Karto algorithm to build maps and uses the A* algorithm and DWA algorithm to realize autonomous navigation. The robot uses a two-wheel independent drive robot chassis to cope with the complex environment. After the indoor environment test, the goal of indoor map construction and path planning can be completed, which is of great significance to the research in the indoor environment in the future.**

Keywords: ROS; SLAM; Lidar

I. INTRODUCTION

An intelligent mobile robot [1] is a system that integrates communication technology, electric power technology, microelectronic technology, and mechanical technology. In recent years, the rapid development of mobile robots has attracted widespread attention. Moreover, mobile robots are also widely used in people's daily life, such as house cleaning, smart express delivery, medical services, catering services, military, smart transportation, and entertainment.

The prerequisite for mobile robots to work autonomously in these fields is that the robot must realize the two core technologies of mapping positioning and path planning. Mapping and positioning are realized by robot sensors and computer units, aiming to replace human perception systems. The main problems that need to be solved in mapping and positioning are as follows: (1) Where am I (location) and (2) What is around me (map)? Because the positioning accuracy of the robot directly affects the mapping accuracy, the two problems are intertwined. So they have to be solved at the same time.

SLAM [2] (simultaneous localization and mapping) is a widely used method for fully autonomous intelligent robot navigation. It essentially allows a mobile robot to navigate in an unknown environment while avoiding obstacles, and add it to the memory by drawing a map of the environment. SLAM technology considers the real topological structure of the environment, can directly and effectively obtain the environment map and can update the map as the environment changes. SLAM based on lidar can accurately measure the angle and distance of obstacle points and is not affected by light when working. It is widely used in unmanned vehicles, indoor and outdoor robot navigation, three-dimensional reconstruction, and other fields.

Since the SLAM issue was raised in 1988, it has been nearly 30 years. Early SLAM research focused on the use of filter theory to minimize the noise of moving body poses and landmark points on the map. After the 21st century, scholars began to learn from SFM [3] (Structure from Motion) to solve SLAM problems based on optimization theory. Depending on the sensor type and installation method, the implementation and difficulty of SLAM will vary greatly. According to sensors, SLAM is mainly divided into two categories: laser and vision. Laser sensors [4] can directly obtain direct distance information relative to the environment, thereby realizing direct relative positioning. On the basis of relative positioning, the absolute positioning and trajectory optimization of laser sensor can be carried out; it is difficult for the visual sensor to directly obtain the direct distance information relative to the environment, and it must estimate its pose change through two or more frames of images, and then calculate the current position by accumulating the pose change. In recent years, the number of people concerned in the field of intelligent robots has increased rapidly, and there are currently several open-source SLAM algorithms for learning

This work was supported in part by Key projects supported by the Department of Education of Hebei Province under Grant ZD2020176.

978-1-6654-0692-5/21/$31.00 ©2021 IEEE
DOI 10.1109/CISAI54367.2021.00159

and reference. Among them, laser SLAM has been studied for a long time, and the theory is relatively mature. The vision solution is still in the laboratory research stage, and practical product applications are almost invisible on the market. In recent years, the rapid development of mobile computing and mobile Internet has made low-cost, high-performance computing and communication networks on the mobile side a reality, giving intelligent robots that were previously forced by computing performance and communication costs a broad space for development.

In SLAM, to autonomously navigate and perform useful tasks, a mobile robot needs to know its accurate position and direction. Therefore, robot positioning is a very critical issue in providing autonomous capabilities for mobile robots. The key requirement in the field of mobile robots is whether the robot can be placed in an unknown location in an unknown environment, and whether it is possible for the robot to gradually construct a map of the environment and determine its location on the map. Therefore, the use of SLAM technology to study indoor robot positioning and path planning is a research topic of practical significance, which will help promote the development of intelligent robots and unmanned driving technologies.

II. Materials and Methods

A. Robot motion model

The model shown in Figure 1 is the solution used in this design. The chassis is connected to four wheels. The front two are independent driving wheels, the rear two are independent casters, and the front two wheels are changed by differential speed. The movement mode and state of the robot. The mathematical model of the differential motion chassis is simpler than that of an omnidirectional chassis, a multi-leg chassis, and a chassis with a differential.

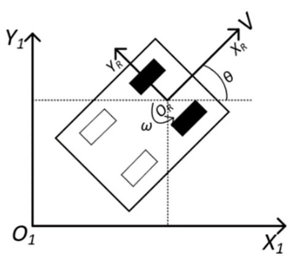

Figure 1. Robot kinematics model

The actual controllability is better than that of an omnidirectional chassis composed of omnidirectional wheels. It is better than a chassis with a differential. The omnidirectional chassis requires higher ground gloss, while the excellent off-road capabilities of crawler and leg-foot robots are useless. Therefore, the design uses the front two independent driving wheels and the rear This solution with two casters.

It is known that the position of the robot can be represented by an orthogonal rotation matrix. As shown in Figure 1, θ is the attitude angle of the robot, that is, the angle between the longitudinal axis of the robot and the X-axis, $R(\theta)$ is the orthogonal rotation matrix, and ξ_R The movement along the global reference frame, ξ_I is the movement along the local reference frame, and the relationship between ξ_R and ξ_I is shown in formula 1:

$$\xi_R = R(\theta)\xi_I = \begin{bmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \mathrm{x} \\ y \\ \theta \end{bmatrix} \tag{1}$$

For the characteristics of the two-wheel differential mobile robot, the motion equation of the mobile robot under the local reference frame can be obtained, as shown in formula 2, where l is the length between the two driving wheels, r is the radius of the driving wheels, φ_1 and φ_2 They respectively indicate the rotational angular velocity of the left and right driving wheels.

$$\xi_I = \begin{bmatrix} \mathrm{x} \\ y \\ \theta \end{bmatrix} = f\left(l, r, \theta, \varphi_1, \varphi_2\right) \tag{2}$$

As shown in equation 3 and equation 4, for a robot with differential drive in the global reference frame, the contributions of the two driving wheels to X_R are:

$$\mathrm{X}_{\mathrm{R}1} = \frac{1}{2} r\varphi_1 \tag{3}$$

$$\mathrm{X}_{\mathrm{R}2} = \frac{1}{2} r\varphi_2 \tag{4}$$

The robot in this design does not have a driving wheel that can provide lateral motion, so the contribution of the two driving wheels Y_R =0 to θ is shown in formula 5 and formula 6 respectively:

$$\omega_1 = \frac{r\varphi_1}{2l} \tag{5}$$

$$\omega_2 = -\frac{r\varphi_1}{2l} \tag{6}$$

Combined equations, the kinematics model of the mobile robot in this design is obtained as formula 7:

$$\xi_I = R(\theta)^{-1}\xi_R = \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \frac{r\varphi_1}{2} + \frac{r\varphi_2}{2} \\ 0 \\ \frac{r\varphi_1}{2l} - \frac{r\varphi_2}{2l} \end{bmatrix} \tag{7}$$

B. Hector and Karto algorithm

Hector's [5] algorithm is a robust 2D SLAM method and navigation technology using an inertial sensing system. It does not need an odometer, but it has a higher standard for lidar, which requires a fast update frequency and low noise lidar. A probabilistic grid map is a way for lidar to describe the real world, and it has been proven to be able to graphically describe any environment. Karto[6] SLAM is an open-source version of the Karto algorithm based on graph optimization launched by SRI Internationals. In the graph-based algorithm, the nodes represent the pose and sensor data sets of the robot trajectory, which are connected by edges, and the edges represent the motion between the robot poses at adjacent moments, which are constraints between consecutive posts. The optimization method is used to minimize the error between nodes, that is, to optimize the pose and trajectory of the robot.

C. *Robot system*

The robot used in the real scene is a small mobile robot developed based on ROS. In addition to the mechanical body, the robot hardware structure mainly includes a controller (PC end, remote visual control), drivers and sensors. The robot control system is based on the upper computer (Raspberry Pi) and the lower computer STM32 control board. Raspberry Pi is a microcomputer based on Linux[7] system. The lower computer is a control board based on STM32F103, which realizes the delivery of driving signal and collect the sensor data information.

The sensors configured by the robot include inertial measurement unit [8] and lidar. The inertial measurement unit (IMU) is used to measure the three-axis attitude angle of an object. Generally, an inertial measurement unit has three single-axis accelerometers and three single-axis gyroscopes. This article uses mpu6050. The chip contains three single-axis accelerometers and three single-axis gyroscopes to directly measure angular velocity and linear acceleration. It has high angular velocity measurement accuracy, high measurement frequency, and low linear acceleration accuracy. The AHRS algorithm fuses the attitude of the system to obtain relatively high-precision attitude data. The parameters of the Shanchuan Delta2A lidar used are shown in Table 1. Although the configuration of this radar is not very high, it is completely achievable for realizing indoor environment modeling and navigation.

TABLE I. PARAMETERS OF SUGIKAWA DELTA2A LIDAR

Parameter Type	**Lidar**
Range	0.13m-8m (white wall)
scanning frequency	6.2HZ
Laser power	3mW
Laser wavelength	780nm
volume	107 mm×76 mm×53mm

The robot installed on the PC is based on the ROS kinetic version under Ubuntu 16.04.1 LTS. Considering the compatibility of hardware, this article installs the ROS kinetic version under Ubuntu16.04.2 LTS on the robot side.

The upper computer of the robot is controlled through the PC. Visualization software such as rviz must run on the PC and subscribe to the ROS node on the robot. After the robot is powered on, wait for the Raspberry Pi to start, check and connect to the open wifi of the robot host computer through the host, change the host network adaptation mode, and establish a local area network between the Ubuntu host and the robot host computer. After the networking of the robot control system is complete, we use the Ping command to check the network communication. As shown in Figure 2, you can see that the Raspberry Pi and the PC have successfully communicated.

```
PING 192.168.3.104 (192.168.3.104) 56(84) bytes of data.
64 bytes from 192.168.3.104: icmp_seq=1 ttl=64 time=5644 ms
64 bytes from 192.168.3.104: icmp_seq=2 ttl=64 time=4728 ms
64 bytes from 192.168.3.104: icmp_seq=3 ttl=64 time=3786 ms
64 bytes from 192.168.3.104: icmp_seq=4 ttl=64 time=2939 ms
64 bytes from 192.168.3.104: icmp_seq=5 ttl=64 time=1990 ms
64 bytes from 192.168.3.104: icmp_seq=6 ttl=64 time=1071 ms
64 bytes from 192.168.3.104: icmp_seq=7 ttl=64 time=179 ms
64 bytes from 192.168.3.104: icmp_seq=8 ttl=64 time=56.7 ms
```

Figure 2. Communication between host and Raspberry Pi

III. RESULTS & DISCUSSION

A. *Mapping experiment*

The experimental scene in this article is an indoor environment, using a four-wheel-drive differential SLAM trolley to conduct mapping experiments with two different algorithms, Hector and Karto. The laboratory site is shown in Figure 3, and the mapping results of the two algorithms are shown in Figure 4.

Figure 3. Construction site

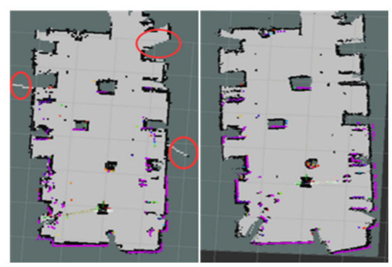

Figure 4. Build the experimental comparison results

Through comparison, we can find that among the two SLAM mapping algorithms, the Hector SLAM algorithm has higher performance requirements for lidar, and the single-line lidar used in this experiment is caused by pose jitter during the mapping process. Individual points are misplaced, the map is blurred, and the map boundary outline and fixed obstacle information cannot be effectively displayed. The lidar signal scanning feedback effect is slightly poor. Comparing the Karto algorithm, it can be found that the map boundaries and obstacle outlines built by the Karto algorithm are the clearest. Therefore, based on the above comparison, the accuracy of mapping in the same environment is used as the evaluation standard, and the Karto algorithm is the best, followed by Hector SLAM.

B. Path planning experiment

This paper uses the Matlab platform to simulate the path planning process of the Dijkstra algorithm, and the simulation results are shown in Figure 5. In the simulation program, the map size is set to a 12×12 network grid, in which white and gray areas are barrier-free areas, black areas are obstacles, dark gray areas are the best path calculated by the algorithm, and light gray areas are the algorithm's for or the area to be traversed, the point (0, 0) is the starting point, and the point (12, 12) is the ending point.

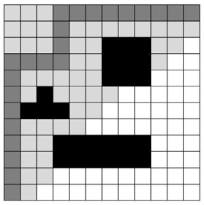

Figure 5. Dijkstra simulation results

From the above simulation results, it can be found that the Dijkstra algorithm lacks constraint conditions in path planning, which results in high time and space complexity of the algorithm and low work efficiency. The A* algorithm is also simulated on the Matlab platform, and the same simulation map as the above-mentioned Dijkstra algorithm is used to verify the simulation results, as shown in Figure 6.

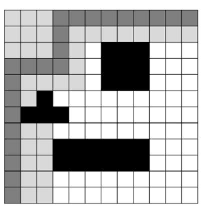

Figure 6. A* simulation results

It can be seen from the simulation results that the A* algorithm traverses the map area much smaller than the Dijkstra algorithm, and the planning time used by the A* algorithm is 0.14s, and the planning time used by the Dijkstra algorithm is 0.52s. The A* algorithm is more efficient and more suitable. Global path planning algorithm.

The global path planning algorithm under ROS includes the Dijkstra algorithm and A* algorithm. The map construction in the simulation environment has been completed before. This experiment uses the A*+DWA algorithm for path planning. The planning results are shown in Figure 7. The path planning results are more Smooth.

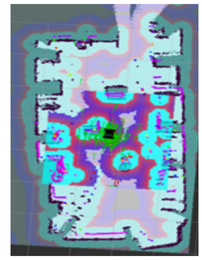

Figure 7. Path planning results

IV. CONCLUSION

In terms of hardware, a practical and reliable mechanical structure was selected, and appropriate controllers, sensors, etc. were selected according to functional requirements. In terms of software, the ROS system runs the Hector algorithm, Karto algorithm, A* algorithm, Dijkstra, and DWA algorithm to achieve map construction and path planning functions. The goals of map construction and autonomous navigation have been achieved, and a mobile robot with low cost, modularity, convenient expansion, and reliable performance has been obtained. This lays the foundation for future applications in larger and more complex indoor environments.

ACKNOWLEDGMENT

Kun Zhang acknowledges partial financial support in part by Key projects supported by the Department of Education of Hebei Province under Grant ZD2020176.

REFERENCES

[1] C.Y.Lee,M.H.A.Sharif,S.Vinayagan,W.A.F.W.Othman,S.S.N.Alhady,A. A.A.Wahab. "Design and development of a quadruped shuffling mobile robot," Journal of Physics: Conference Series, vol. 1969, no.1, pp. 012012, March 2021.

[2] Z.Yan, S.Chu, L.Deng. "Visual SLAM based on instance segmentation in dynamic scenes," Measurement Science and Technology, vol. 32, no. 9, pp. 095113, June 2021.

[3] H. Zhang,M. A. Ikbal. "Unmanned vehicle dynamic obstacle detection, tracking and recognition method based on laser sensor," International Journal of Intelligent Computing and Cybernetics, vol. 14, no. 2, pp. 239-251, February 2021.

[4] J.An, H.Mou, R.Lu, Y.Li. "Localization and Navigation Analysis of Mobile Robot Based on SLAM," Journal of Physics: Conference Series, vol. 1827, no. 1, pp. 012089-,January 2021.

[5] H. C. H.El, K.Audun. "Combining Hector SLAM and Artificial Potential Field for Autonomous Navigation Inside a Greenhouse" Robotics, vol. 7, no. 2, pp. 5694-5709, May 2018.

[6] M.O.Georgewill,J.C.Ezeofor."Development of the intelligent Mobile Robot for Service Use," Journal of Research in Science and Engineering, vol. 2, no. 4, pp. 1656-1996, April 2020.

[7] W. Sirigool , R. Kesvarakul. "Particle Filter for Hector SLAM to Improve the Performance of Robot Positioning by Image Processing Based," International Journal of Machine Learning and Computing, vol. 10, no. 3, pp. 490-494, May 2020.

[8] D. Han,Y.Li,T.Song,Z.Liu."Multi-Objective Optimization of Loop Closure Detection Parameters for Indoor 2D Simultaneous Localization and Mapping," Sensors, vol. 20, no. 7, pp. 1906-1906, March 2020.