Proceedings of The 2021 IEEE International
Conference on Real-time Computing and Robotics
July 15-19, 2021, Xining, China

Research on Graph-Based SLAM for UVC Disinfection Robot

Xuan Tan[1], Hui Zhang[2], Xidong Zhou[1], Hang Zhong[2] and Li Liu[2]

2021 IEEE International Conference on Real-time Computing and Robotics (RCAR) | 978-1-6654-3678-6/21/$31.00 ©2021 IEEE | DOI: 10.1109/RCAR52367.2021.9517506

***Abstract*— With the emergence of the COVID-19 pandemic, more and more non-contact mobile disinfection robots have appeared in the medical field, which have made great contributions to the fight against the epidemic. Aiming at the problems of single disinfection method, single application scenario, low degree of intelligence and lack of autonomous mobile disinfection in existing disinfection robots, this paper proposes and designs a disinfection and epidemic prevention intelligent robot called Aimi-Robot UVC, which is based on graph-optimized slam algorithm to complete the localization and map creation functions of the robot in the unknown environment. After testing in the isolated single ward of the hospital, the real-time localization accuracy reaches 0.04m, which provides high-precision and high-reliability localization for the disinfection robot in the hospital scene and has great practical significance for the application of intelligent disinfection robots in epidemic prevention and control.**

I. INTRODUCTION

The outbreak of COVID-19 has now become a pandemic, and mobile robots quickly participated in the fight against the epidemic. From disinfection, transportation of medicines and medical equipments, waste cleaning for temperature measurement, mobile robots have been widely used in many places [1]. Among them, mobile robots have made major breakthroughs in the promotion and application of remedial systems. On the one hand, mobile robots can protect medical staffs and reduce the risk of infection of the medical staffs. On the other hand, they can complete high-standard disinfection and sterilization tasks, thereby alleviating the pressure on medical staffs [2]. Despite the rapid development of intelligent robot technology in recent years, epidemic prevention robots need to complete specific tasks in complex unstructured scenarios (for example, disinfection mobile robots need to complete disinfection tasks autonomously in complex environments in public places), which requires epidemic prevention robots to have strong autonomy, real-time and high flexibility. The biggest technical challenge is the ability to perceive the environment and obtain perceptual information, so as to realize autonomous localization, navigation and control in a complex unstructured environment [3]. In order to ensure the smooth progress of the disinfection task, the real-time position information of the disinfection robot system needs to be tracked at all times, and the autonomous localization technology is particularly important. Autonomous localization technology is a technology for precise localization of the robot's position, and it is also the basis for the robot to realize autonomous mobile navigation [4].

Although there are many kinds of localization technologies, most of them either have many limitations in practical applications or are too expensive to be popularized. For example, GPS cannot be used in indoor and heavily blocked outdoor environments, and the localization accuracy is low; the cost of high-precision inertial navigation systems is too high; localization schemes based on wireless signals (WiFi, Bluetooth, UWB) need to be arranged in advance for use scenarios, etc. [5]. The laser radar-based simultaneous localization and mapping (SLAM) technology accurately measures the angle and distance of obstacles, does not require pre-arrangement of the scene, can integrate multiple sensors, work in poor light environments, and can easily generate the advantages of navigation environment map and other advantages have become an indispensable new technology in the current localization scheme [6]. SLAM systems are increasingly deployed on robots that operate in unstructured environments, or robots that cannot access reliable external localization infrastructure [7]. In order to achieve high intelligence and complete specific epidemic prevention and control tasks in complex unstructured scenarios such as hospitals, slam technology is a necessary condition for epidemic prevention robots. The earliest laser slam filter localization algorithm is based on the Kalman Filter (KF) mobile robot localization algorithm, but due to its filter has many restrictions, such as the application system must be linear, it is difficult to meet the application requirements of most non-linear practical environments. Therefore, on this basis, slam localization algorithms such as Extended Kalman Filter (EKF) [8], Monte Carlo Localization [9], and Adaptive Kalman Filter Localization have emerged. However, because filtering algorithms are based on recursive calculations, only the robot pose and map information at two adjacent moments are estimated, and the update efficiency decreases with the increase of the map scale, linearization, lack of closed-loop detection, and poor adaptability, which are difficult to apply. In a complex environment with large-scale, multiple loops, long distances and other wide. Therefore, a novel

*This work was supported in part by the National Key RD Program of China under Grant 2018YFB1308200, in part by the National Natural Science Foundation of China under Grants 61971071, 6202780012, Changsha science and technology project under Grant kq1907087, Special funds for the construction of innovative provinces in Hunan Province under Grant 2020SK3007, Postdoctoral innovative talent support program under Grant BX20200122.

[1]Xuan Tan, Xidong Zhou are with the College of Electrical and Information Engineering, Changsha University of Science and Technology, 410012, Hunan Province, China tanxuanxf@126.com; xidong_zhou@163.com

[2]Hui Zhang is with School of Robotics, also with Visual Perception and Control Technology National Engineering Laboratory, Hunan University, 410082, Hunan Province, China zhanghuihby@126.com

[2]Hang Zhong, Li Liu are with the College of Electrical and Information Engineering, Hunan University, 410082, Hunan Province, China zhonghang@hnu.edu.cn; liuli@hnu.edu.cn

method based on graph optimization is proposed to solve the slam problem of large-scale scene mapping. In robotics, most graph-based slam techniques depend on local search methods for nonlinear optimization to estimate poses of the autonomous agent and positions of observed landmarks [10]. Graph-based SLAM is usually divided into the front-end and back-end. The front-end constructs the nodes and edges of the graph depending on the observation values and system constraints. The back-end applies optimization techniques to complete the graph optimization [11]. The advantage of this is that incorrect data associations can be rectified, and pose errors and map drift can be lowered. Since the graph-based slam method utilizes all the observations information to optimize the robots accomplish trajectory and environment, we can get a globally consistent trajectory and map [12]. Among them, the most representative google open source cartographer [13], in the realization of 2D slam, it can generate a two-dimensional grid real-time map with an accuracy of 5cm, use loop detection to optimize the pose of the submap, and eliminate the problem in the process of mapping. Accumulate errors while achieving a balance between the amount of calculation and real-time performance. The lidar-based localization method is usually used for the localization of mobile robots due to its high accuracy and robustness to illumination and viewpoint changes. Among them, the most widely used method is amcl localization based on particle filtering [14]. However, the amcl algorithm still has a problem: when the robot is in a complex and unstructured environment with dynamic obstacles and uneven ground, the best pose estimation algorithm is the center of the particle swarm instead of the best match between the map and laser scanning. Even after convergence, localization accuracy and robustness cannot be guaranteed. In addition, once the error between the particle swarm and the actual attitude is large, it takes a certain amount of time to perform self-calibration, which may even cause serious localization deviation [15].

In addition to providing a slam framework, cartographer can also be used as a map self-location technology similar to amcl. By studying the pure-localization algorithm of cartographer and optimizing the localization algorithm parameters, the accuracy and robustness of the autonomous localization of the disinfection robot in the complex non-structural environment are solved, which provides high precision and high reliability for the disinfection robot to operate in the hospital scene.

II. ALGORITHM STRUCTURE ANALYSIS

The primary idea of cartographer [12] is to use closed-loop detection to reduce the cumulative error in the mapping process. The algorithm as a whole can be divided into two parts: the first part is called local slam, which establishes and maintains a series of submaps through a frame of laser scan, and the submap is a series of grid maps. When there is a new laser scan, it will be inserted into the best position in the submap through ceres scan matching. However, submap will have the problem of error accumulation. Therefore, the second part of the algorithm called global slam, is to perform closed-loop detection through loop closure to eliminate accumulated errors. When the construction of a submap is completed, no new laser scan will be inserted into the submap, the algorithm will add the submap to the closed-loop detection. The algorithm structure diagram is given in Figure 1.

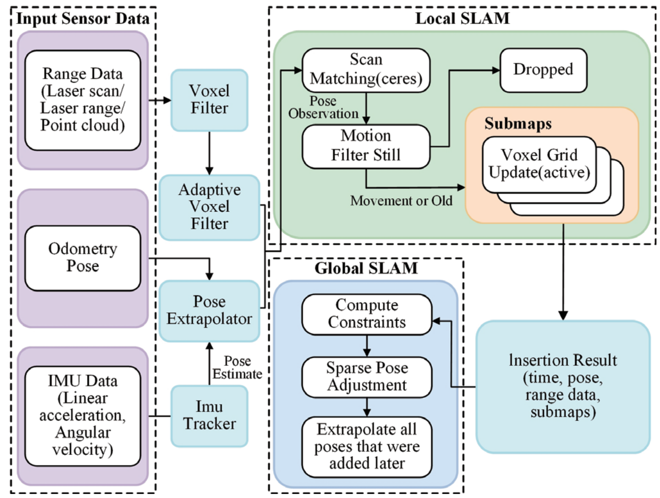

Fig. 1. Cartographer algorithm structure block diagram.

A. Local Slam

In 2D SLAM, the three parameters of translation (x,y) and rotation ξ_θ obtained by lidar scanning can determine the pose of mobile robot $\xi = (\xi_x, \xi_y, \xi_\theta)$. Record the data measured by the lidar sensor as $H = \{h_k\}_{k=1,\ldots,K}, h_k \in \mathbb{R}^2$. The initial laser point is $\mathbf{0} \in \mathbb{R}^2$. The pose transformation of the laser radar scanning data frame mapped to the submap is denoted as T_ξ, which can be mapped to the submap coordinate system by equation (1):

$$T_\xi p = \underbrace{\begin{pmatrix} cos\xi_\theta & -sin\xi_\theta \\ sin\xi_\theta & cos\xi_\theta \end{pmatrix}}_{R_\xi} p + \underbrace{\begin{pmatrix} \xi_x \\ \xi_y \end{pmatrix}}_{t_\xi} \tag{1}$$

Some continuous scans form a submap, and the submap takes the form of a probability grid. When new scan data is inserted into the probability grid, the state of the grid will be calculated, and each grid has two states: hit and miss. The grid points in each hits are assigned the initial value $M = P_{hit}$, and the grid points in each misses are assigned the initial value $M = P_{miss}$. If the grid point already has a P, use the following equation to update:

$$odds(p) = \frac{p}{1-p} \tag{2}$$

$$M_{new}(x) = clamp(odds^{-1}(odds(M_{old}(x)) \cdot odds(p_{hit}))) \tag{3}$$

Before inserting scan into the submap, use the scan matcher of the ceres library to optimize the pose ξ of scan, and turn the problem of solving scan pose into a problem of solving nonlinear least squares.

$$\underset{\xi}{argmin} \sum_{k=1}^{K}\left(1-M_{smooth}\left(T_{\xi}h_k\right)\right)^2 \tag{4}$$

In the above equation, T_{ξ} represents the pose transformation in the scan conversion to the corresponding submap frame, and the pose conversion converts the scan point h_k from the scan to the submap frame. Smoothing function $M_{smooth}:\mathbb{R}^2\rightarrow R$ smoothly maps the probability value of each scan point to the local subgraph, the goal is to maximize the probability value of all scan points inserted into the scan, and construct a nonlinear least square multiply the objective function.

B. Global Slam

Since the lidar scan frame is only matched with the current submap, the environment map is composed of a series of submaps. As the number of submaps increases, the cumulative error in the scanning matching process will become larger and larger, so the sparse attitude adjustment (SPA) to optimize the pose of all lidar data frames and submaps. The pose of the lidar data frame when inserted into the submap will be cached in the memory for closed-loop detection. When the submap no longer changes, all scanned frames and submaps will be used for closed-loop detection.

$$\underset{\Xi^m,\Xi^s}{argmin} \ \frac{1}{2}\sum_{ij}\rho\left(E^2(\xi_i^m,\xi_j^s;\Sigma_{ij},\xi_{ij})\right) \tag{5}$$

In the above equation, ρ is a loss function, which can reduce the influence of outliers added to the optimization problem on the system. $\Xi^m=\{\xi_i^m\}_{i=1,\ldots,m}$,$\Xi^s=\left\{\xi_j^s\right\}_{j=1,\ldots,n}$, respectively represent the pose of the submap and pose of the scan frame under certain constraints. The relative pose ξ_{ij} represents the matching position of the scanned frame j in the submap i, and the associated covariance ξ_{ij} uses the ceres library for feature estimation. The residual E of the constraint can be calculated by equation (6):

$$E^2\left(\xi_i^m,\xi_j^s;\Sigma_{ij},\xi_{ij}\right)=e(\xi_i^m,\xi_j^s;\xi_{ij})^T\sum_{ij}^{-1}e(\xi_i^m,\xi_j^s;\xi_{ij}),$$

$$e(\xi_i^m,\xi_j^s;\xi_{ij})=\xi_{ij}-\begin{pmatrix}R_{\xi_i^m}^{-1}(t_{\xi_i^m}-t_{\xi_j^s})\\ \xi_{i;\theta}^m-\xi_{j;\theta}^s\end{pmatrix}. \tag{6}$$

In addition, the branch and bound scan matching algorithm is used to accelerate the process of closed-loop detection and relative pose solving, determine the search window, use the search method to construct the loop, and use the equation (7) to search:

$$\xi^{\star}=\underset{\xi\in W}{argmax}\sum_{k=1}^{K}M_{nearest}(T_{\xi}h_k) \tag{7}$$

In the above equation, W represents the search window, $M_{nearest}$ is the extension from the nearest grid point of the parameter in M to the corresponding pixel ($\mathbb{R}^2$), and using the branch and bound method can efficiently calculate the value of $\xi^{\star}$.

III. Aimi-Robot UVC

The overall system structure of the Aimi-Robot UVC, an ultraviolet disinfection robot developed by us, is shown in Figure 2. The robot system structure is mainly composed of mini PC, underlying embedded controller (motor, encoder, ultrasonic, imu, infrared sensor, etc.), sensors/other ROS devices (depth camera, lidar, etc.), human-computer interaction

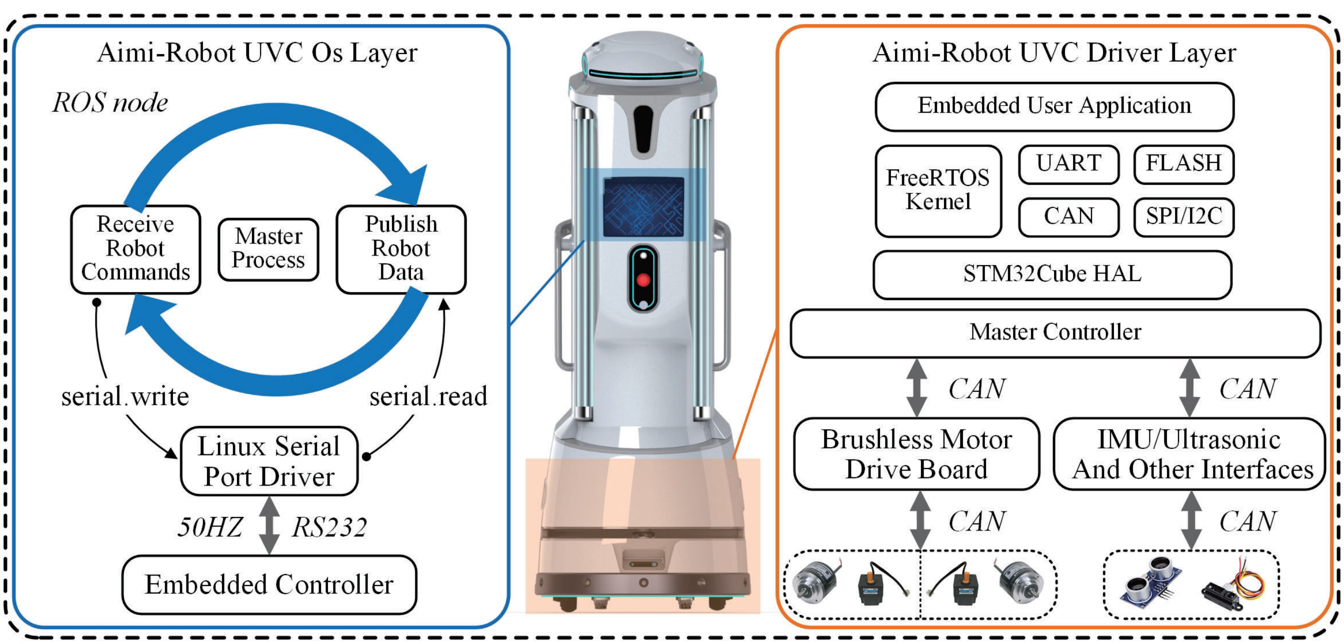

Fig. 2. Aimi-Robot UVC.

interface and other components. The entire robot adopts the mode of distributed control system, which is divided into two parts: the upper control platform and the lower controller. The bottom controller uploads all kinds of sensor information (position, attitude, etc.) collected by the bottom layer to the upper control platform through RS232 communication at the frequency of 50 Hz. The upper layer relies on ROS [16] to transmit the control command at the same rate through data analysis, so as to realize the accurate control of the robot [17]. Aimi-robot UVC can move flexibly and autonomously in all directions, and has core technologies such as accurate navigation and localization, autonomous planning of itinerant disinfection path, automatic charging, etc. It has the functions of temperature, carbon dioxide concentration, air quality detection and multi-point infrared human body sensing. It is equipped with ultraviolet disinfection lamp and disinfectant spraying device to complete two kinds of disinfection in the hospital ward at the same time. Workers only need to complete the task configuration of the robot through the human-computer interface before starting the robot, and set the time, route, content, walking frequency and task planning of the robot, the separation of human and machine is realized, the contact of personnel is reduced and the risk of virus infection is effectively reduced.

IV. EXPERIMENT AND ANALYSIS

In order to verify the effectiveness of the algorithm, the test site of this experiment is an isolated single ward of the hospital, and the physical picture of Aimi-Robot UVC is shown in Figure 3. The hardware of the upper-level control platform of this experiment is mainly Intel NUC, and the system on the NUC is Ubuntu 16.04 and robot operating system (ROS). The experimental program is mainly run through ROS, and all experimental data are saved through the rosbag command mechanism in ROS.

Fig. 3. Aimi-Robot UVC in hospital ward

A. Map Construction

Aimi-Robot UVC equipped with lidar, odometer, and IMU will respectively use gammping algorithm and cartographer algorithm to build maps in a single isolation ward of the hospital, and the mapping effects are shown in Figure 4 and Figure 5 respectively.

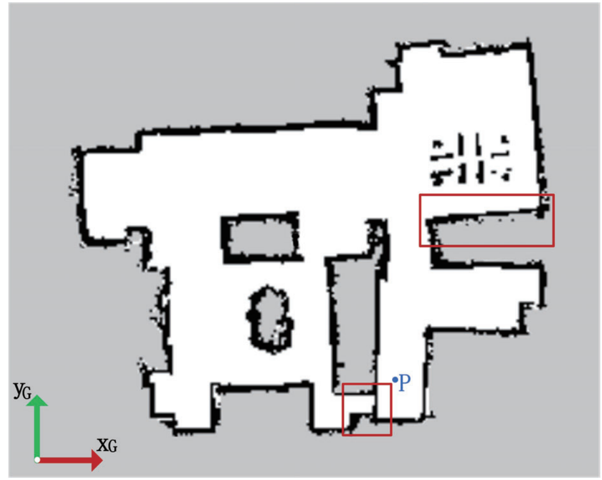

Fig. 4. Gammping algorithm mapping results.

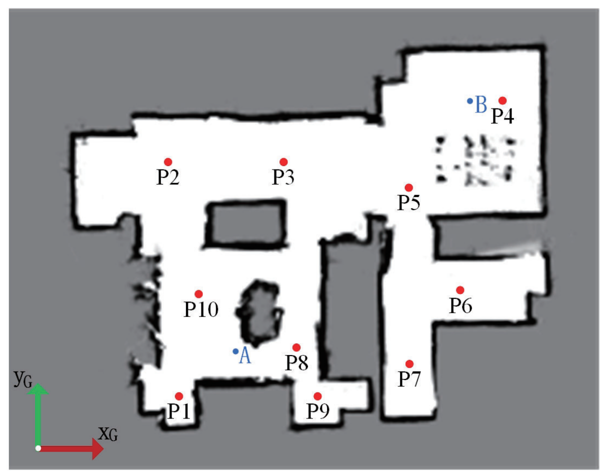

Fig. 5. Cartographer algorithm mapping results.

In the process of mapping construction, Aimi-Robot UVC was manually operated to rotate at high speed at P and P7 points. It can be seen from Figure 4 that the rotation at point P lead to the obvious drift of some maps in the red box, while Figure 5 is relatively stable. It can be seen from the above that cartographer slam algorithm is better than gammping slam algorithm in the same environment and situation.

B. Global Localization Experiment

As shown in the Figure 5, 10 way points are randomly selected on the map to evaluate the global localization robustness. AMCL and cartographer were used to test each way point for 10 times. The global localization robustness is evaluated by the cumulative localization success rate (the average success rate of 10 way points). In order to verify the global localization effect of amcl algorithm, the amcl algorithm is used to carry out localization experiments on the occupation grid constructed by cartographer. The global

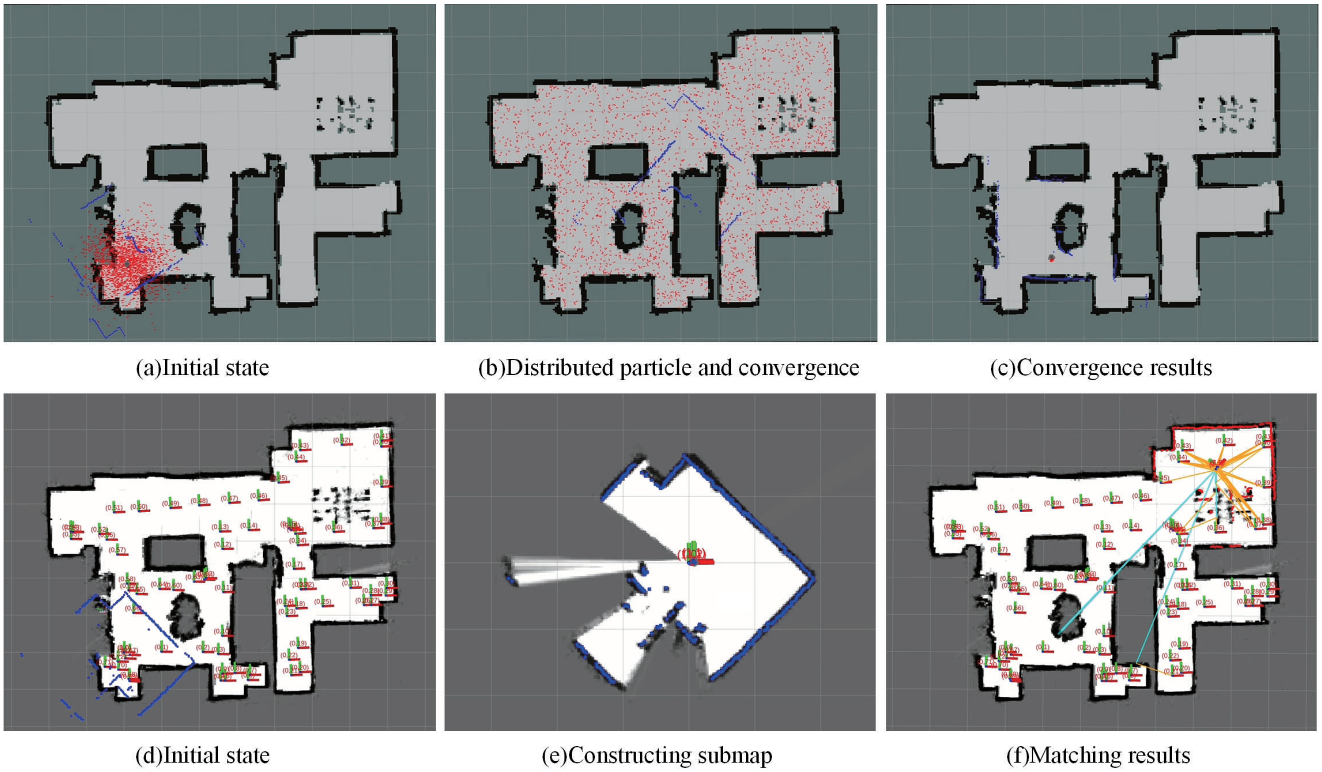

Fig. 6. Global localization process.

localization process is shown in Figure 6. In order to speed up the convergence rate, an initial particle swarm is randomly generated near the actual initial posture. The small red arrow in Figure 6 represents an attitude particle, and the blue point cloud is a laser scanning point cloud. Figure 6(a) shows the initial state of the particle swarm. In order to accelerate the convergence of localization, an approximate estimation of the initial attitude of the robot is given. It can be seen from the figure that many particles are generated near the initial attitude estimation. Figure 6(b) and Figure 6(c) show that when the robot moves, the particles gradually converge, and the convergence result is point A in Figure 5. It can be seen from the blue point clouds in Figure 6(b) and Figure 6(c) that the laser scanning points are not well aligned with the obstacles. This is because amcl algorithm uses the weighted average pose of particle swarm optimization as the estimated pose, resulting in a certain deviation between the estimated pose and the actual pose.

In order to verify the global localization effect of cartographer algorithm, the map file based on .Pbstream format shown in Figure 5 is loaded, and the global localization experiment is carried out by using the pure-localization of cartographer algorithm. The global localization process is shown in Figure 6. Figure 6(d) shows the initial state, and the blue point cloud is the laser scanning point cloud. Figure 6(e) shows the process of creating a local submap. After the submap is created, it is matched with the global reference map by scanning matching algorithm. Figure 6(f) shows the scanning matching result, which is represented as point B in Figure 5.

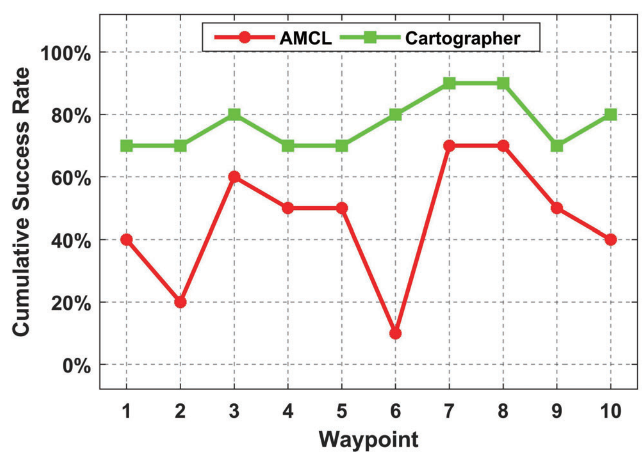

Fig. 7. Global localization robustness comparison.

The cumulative success rate is shown in Figure 7. The figure shows, the localization success rate of amcl algorithm is very unstable. In the experiment of 10 way points, the highest localization success rate is 70%, and the lowest localization success rate is 10%. However, the localization success rate of cartographer pure-localization mode remains between 70% and 90%, which is relatively stable. Cartographer global localization effect is better than amcl.

C. Error Analysis of Localization Trajectory

In order to test the real-time localization accuracy of Aimi-Robot UVC after global localization, the experiment is carried out on the map shown in Figure 5, and Aimi-Robot UVC is remotely controlled to move in an irregular track. As shown in Figure 8, the red generation represents the theoretical trajectory, while the blue represents the real-time localization trajectory.

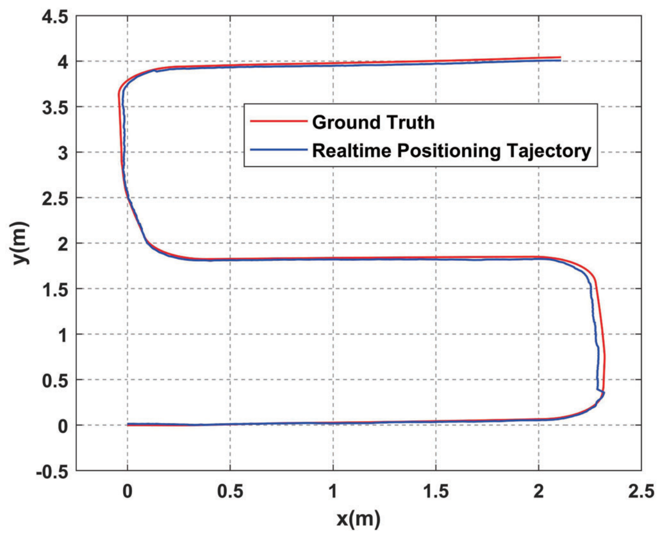

Fig. 8. Real-time localization track.

Analyzing the real-time localization error during the movement, as shown in Figure 9, the real-time localization accuracy of Aimi-Robot UVC is about 0.04m, which provides more accurate real-time localization information for subsequent autonomous robot navigation.

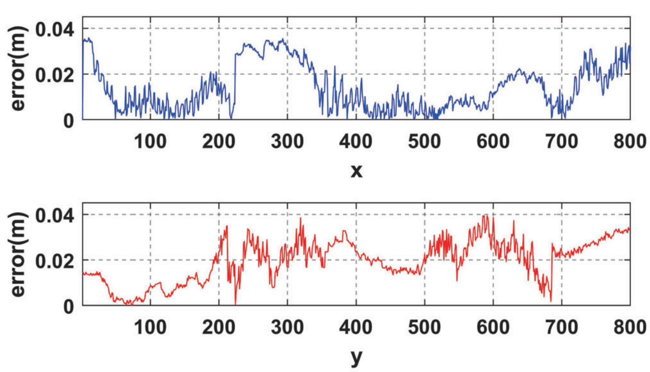

Fig. 9. Real-time localization error curve.

V. CONCLUSIONS

In this paper, Aimi-Robot UVC is used as the carrier, and the graph-based slam algorithm is combined with lidar sensors, gyroscopes, and odometers to obtain unknown environmental information and complete the construction of unknown environmental maps. At the same time, by optimizing the localization algorithm parameters, the accuracy and robustness of autonomous localization of the intelligent disinfection robot during operation are improved. Experimental results show that compared with amcl, the relocation success rate reaches 70%∼90%. At the same time, after the global localization is successful, the real-time localization accuracy is restored to 0.04m, which provides an important guarantee for the autonomous navigation of the mobile disinfection robot. At present, the semantic information of sensors has not been used. In the future, research can be combined with deep learning to promote the rapid development of intelligent medical service robots.

REFERENCES

[1] G. Z. Yang, B. J. Nelson, R. R. Murphy, et al. "Combating COVID-19–The role of robotics in managing public health and infectious diseases," *Science Robotics*, vol. 30, no. 1, pp. eabb5589-eabb5589, 2020.

[2] V. Chamola, V. Hassija, V. Gupta and M. Guizani, "A comprehensive review of the COVID-19 pandemic and the role of IoT, Drones, AI, Blockchain, and 5G in managing its impact," *IEEE Access*, vol. 8, pp. 90225-90265, 2020.

[3] F. Rubio, F. Valero, C. Llopis-Albert. "A review of mobile robots: Concepts, methods, theoretical framework, and applications," *International Journal of Advanced Robotic Systems*, vol. 16, no. 2, 2019.

[4] E. Rubies, et al. "Mobile robot self-localization with 2D push-broom LIDAR in a 2D map," *Sensors*, vol. 20, no. 9, pp. 2500, 2020.

[5] D. Liu, B. Sheng, F. Hou, et al. "From wireless localization to mobile localization : An overview of recent advances," *IEEE Systems Journal*, vol. 8, no. 4, pp. 1249-1259, 2014.

[6] Z. Alsayed, G. Bresson, A. Verroust-Blondet, and F. Nashashibi. "2D SLAM correction prediction in large scale urban environments," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2018, pp. 5167-5174.

[7] C. Cadena, L. Carlone, H. Carrillo, et al. "Past, present, and future of simultaneous localization and mapping: Toward the robust-perception age," *IEEE Transactions on Robotics*, vol. 32, no. 6, pp. 1309-1332, 2016.

[8] S. Huang, G. Dissanayake, et al. "Convergence and consistency analysis for extended Kalman Filter based SLAM," *IEEE Transactions on Robotics*, vol. 23, no. 5, pp. 1036-1049, 2007.

[9] A. Hornung, D. Maier, et al. "Monte Carlo localization for humanoid robot navigation in complex indoor environments," *International Journal of Humanoid Robotics*, vol. 11, no. 2, pp. 1441002, 2014.

[10] T. Fan, H. Wang, M. Rubenstein, et al. "CPL-SLAM: Efficient and certifiably correct planar Graph-Based SLAM using the complex number representation," *IEEE Transactions on Robotics*, vol. 36, no. 6, pp. 1719-1737, 2020.

[11] A. Dine, B. Vincke, et al. "Graph-based simultaneous localization and mapping: Computational complexity reduction on a multicore heterogeneous architecture," *IEEE Robotics Automation Magazine*, vol. 23, no. 4, pp. 160-173, 2016.

[12] H. Gao, X. Zhang, J. Wen, et al. "Autonomous indoor exploration via polygon map construction and graph-based SLAM using directional endpoint features," *IEEE Transactions on Automation Science and Engineering*, vol. 16, no. 4, pp. 1531-1542, 2018.

[13] W. Hess, D. Kohler, H. Rapp and D. Andor, "Real-time loop closure in 2D LIDAR SLAM," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2016, pp. 1271-1278.

[14] W. P. N. dos Reis, O. Morandin and K. C. T. Vivaldini, "A quantitative study of tuning ROS Adaptive Monte Carlo Localization parameters and their effect on an AGV localization," in *2019 19th International Conference on Advanced Robotics (ICAR)*. IEEE, 2019, pp. 302-307.

[15] G. Peng, W. Zheng, Z. Lu, et al. "An improved AMCL algorithm based on laser scanning match in a complex and unstructured environment," *Complexity*, 2018, 2018.

[16] B. Dieber, B. Breiling, S. Taurer, et al. "Security for the robot operating system," *Robotics and Autonomous Systems*, vol. 98, pp. 192-203, 2017.

[17] A. Santos, A. Cunha, N. Macedo, R. Arrais and F. N. dos Santos, "Mining the usage patterns of ROS primitives," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2017, pp. 3855-3860.