Journal of Computers Vol. 32 No. 3, 2021, pp. 210-221
doi:10.3966/199115992021063203015

Train Carriage Disinfection Robot Based on Visual SLAM

Dezhao Chen, Hezhi Lin*, Chen Zhao, Jingyi Lei, Jian Zou, Lianfen Huang

School of Information, Xiamen University, Xiamen, China 361005
22920182204137@stu.xmu.edu.cn, linhezhi@xmu.edu.cn, 35320182200287@stu.xmu.edu.cn
22920182204394@stu.xmu.edu.cn, 22920182204209@stu.xmu.edu.cn, lfhuang@xmu.edu.cn

Received 18 March 2021; Revised 17 May 2021; Accepted 18 May 2021

Abstract. Under the trend of normalization of the prevention and control of the CoviD-19 epidemic, robots have gradually received attention in the field of assisting people in daily disinfection work. In this paper, a solution is presented for the localization, navigation and obstacle avoidance of the disinfection robot for train carriage. Integrated the vision and IMU sensors, the VIO localization scheme based on visual SLAM is adopted to make the scheme more adaptable and robust to the train carriage. In terms of navigation and obstacle avoidance, the global navigation and local navigation which using A* algorithm and DWA algorithm separately are combined to achieve the function. And on this basis, the method of obstacle expansion and limiting the obstacle avoidance range are proposed to improve the algorithm, so that the robot can move smoothly in the special environment of the train carriage. In the experiment, we built our disinfection robot and a simulated carriage environment on the spot, and completed the test of the algorithm. The results show that the VIO localization scheme and the improved algorithm can better adapt to the environment of the train carriage, and the robot can successfully realize the functions of localization, navigation and obstacle avoidance in the carriage.

Keywords: disinfection robot, SLAM, localization, route plan

1 Introduction

At the beginning of 2020, the world was confronted with an unprecedented challenge by the COVID-19. According to available information, SARS-CoV-2 can be transmitted by droplets and contact with contaminated surfaces [1]. In a relatively closed environment, prolonged exposure to high concentrations of aerosols is likely to spread through the air.

There are two main ways to prevent the virus. Firstly, in order to prevent the spread of a high prevalence virus, person-to-person contact should be minimized. Secondly, in contaminated or densely populated environments, it is necessary to disinfect all surfaces [2]. As an important mode of travel, high-speed trains and other passenger trains are characterized by dense personnel, large mobility, tight space and long transportation distance. In the fight against the epidemic, high-speed trains and other important means of transport have become the forefront of epidemic prevention and control. In order to ensure the safety of passengers, the railway departments in every part of the train carriage disinfection have become a very important routine epidemic prevention work. According to the requirements of epidemic prevention, railway departments have been sterilizing the trains returning to the warehouse and departing soon. According to the data of China Railway Shanghai Bureau, since the summer rush, the Yangtze River Delta Railway had put in a total of 55,000 person-times of disinfection personnel, and more than 30,000 teams of sterilized high-speed train.

Currently, CoviD-19 is mainly sanitized by humans using chemicals in complex indoor environments. This disinfection method cannot guarantee the disinfection quality completely, and poses a great risk to human health. Therefore, unmanned disinfection is a good choice to improve the quality of disinfection and reduce the risk of exposure [3]. This is where robots come into play. Robots have been introduced to

* Corresponding Author

regularly disinfect subway stations in places like Beijing and Hong Kong. London's Heathrow Airport has deployed disinfection robots as an epidemic prevention measure and Wuhan's intelligent field hospital has introduced them to reduce the spread of the virus. However, due to the small movement space in the carriages of important transportation vehicles such as high-speed trains, these disinfection robots aimed at large public places cannot be directly applied. Therefore, based on the visual SLAM, the algorithm is improved to adapt to the narrow space in the carriage and meet the requirements of flexible movement in this special environment. Finally, we implemented the improved algorithm on the self-designed robot platform to complete the automatic disinfectant dispersion covering disinfection of the key disinfecting parts of the train carriage. The results show that the designed robot can flexibly realize the functions of localization, navigation and obstacle avoidance in the carriage, and achieve the purpose of intelligent disinfection.

2 Related Work

Since the outbreak of Covid-19, it has become common sense to reduce offline contact. But medical care, transportation, manufacturing and other work are in urgent need of people to deal with. With the development of artificial intelligence, robots have emerged in the front lines of the fight against the epidemic. A large number of related research projects and commercial products have come forth.

Researchers found that in the disinfection operation, the combination of automatic disinfection and manual disinfection of disinfection robot can achieve the best disinfection effect [4], which demonstrates the effectiveness of intelligent disinfection. A method of extracting image semantic descriptors based on knowledge graph was proposed by Baofu Fang et al., which effectively dealt with SLAM problems for scenarios with dynamic objects, and solved the navigation problem of robots in medical institution scenarios [5]. Qingchun Feng et al. designed a disinfection spray system for epidemic prevention to realize intelligent disinfection management of livestock and poultry houses [6].

In the industry, a number of robotics companies have also launched a variety of disinfection robots for different places. A collaboration between Danish company Blue Ocean Robotics and the University Hospital of Odense has introduced a UVD Robots disinfection robot that uses class C ultraviolet light (UV-C) to kill viruses and bacteria in the air and on surfaces. It helps hospitals kill 99.99 percent of all bacteria and microorganisms, thereby reducing the spread of disease. Because ultraviolet light is harmful to the human body, Singapore-based DISA Pte Ltd has launched a disinfection robot that avoids people while working. However, if there are more objects in the room, it may be because some areas are covered with objects that cause the dead angle of disinfecting to appear. The Westin Houston Medical Center in the United States has introduced Lightstrike Germ-Zapping Robot for guest room cleaning, which uses ultraviolet light and ozone to disinfect the room and sense if there is a person in the area to disinfect safely. With the support of 3D laser fence, multi-sensor scene adaptation and other technologies, the disinfection robot of UV lamp + atomizing disinfectant developed by the Fudan University team can set the disinfection plan according to the needs, realize automatic disinfection, and undertake the daily disinfection work of the NPC and CPPCC.

After reviewing relevant studies, we find that there is no disinfection robot for passenger train carriages. Due to the characteristics of closed, narrow space and many obstacles, the general disinfection robot cannot complete the disinfection work well in the train carriage. At present, there is no navigation algorithm specifically for the specific narrow space in the carriage, and the general algorithm does not perform well in the carriage.

3 Solution of Localization, Navigation and Obstacle Avoidance

The width of the corridor in the train carriage is usually 50cm~70cm. In order to achieve the automatic disinfection of the carriage, it is necessary to solve the problem of localization, navigation and obstacle avoidance in the narrow space. In this paper, the localization scheme of Visual-Inertial Odometry and global-local path planning are used to complete the automatic navigation in narrow space.

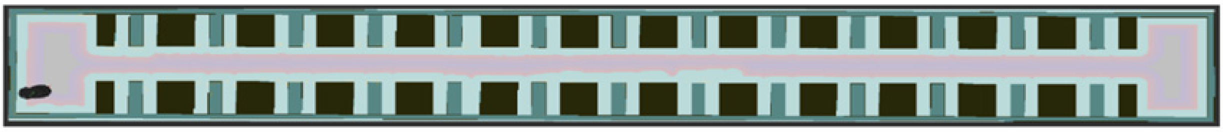

Fig. 1. Schematic diagram of high-speed railway carriage [7]

3.1 VIO Localization Scheme

VIO (Visual Inertial Odometer) is an algorithm that integrates camera and IMU data to realize SLAM (simultaneous localization and mapping). It can be divided into two frameworks: loose coupling and tight coupling. The loose coupling is the fusion calculation of the output of two independent modules, vision and IMU. While the tight coupling is to use the original data of two sensors to estimate a set of variables together. The noise would affect each other, but the full use of sensor data can make the better effect. This paper adopts the tightly coupled framework [8].

Using stereo cameras, the information of environmental depth can be directly calculated by binocular parallax matching. Harris corner is used to extract feature points in binocular vision. Harris corner detection has the advantage of high stability, especially for L-shaped corner detection.

In this localization scheme, the pose of the current frame is calculated by the pose of the previous frame, so there must be cumulative error in the estimation of the current pose in front-end computing. In order to eliminate the influence of cumulative error, DBOW2 can be used for loop detection [12]. Loop detection is to determine whether the robot has returned to the previous position. It has a more compact and accurate constraint than that of the backend. A trajectory map with a consistent topology is then created. If it can detect a closed loop, that is, determine that two frames are in the same position, then optimize it, the error can be eliminate greatly. The difference between the two frames can be calculated by (1):

$$s(v_A - v_B) = 2\sum_{i=1}^{N} |v_{Ai}| + |v_{Bi}| - |v_{Ai} - v_{Bi}|. \quad \textbf{(1)}$$

v_A and v_B are the sparse vectors representing images A and B respectively. v_{Ai} and v_{Bi} represent words only in A or B. $v_{Ai} - v_{Bi}$ represents the intersection of A and B, the words that A and B share. The larger the score s is, the more similar the two frames of pictures are. If the score is large enough, it can be judged that the two frames of pictures may be loopback. In addition to using the Harris corner feature, a certain number of FAST corners are added and the BRIEF descriptors are used for better recall rates in loop back detection. When the loop is detected, the corresponding relationship between images can be found through the matching of BRIEF descriptor, then the connection between the local slide window and the candidate frame of the loop is established.

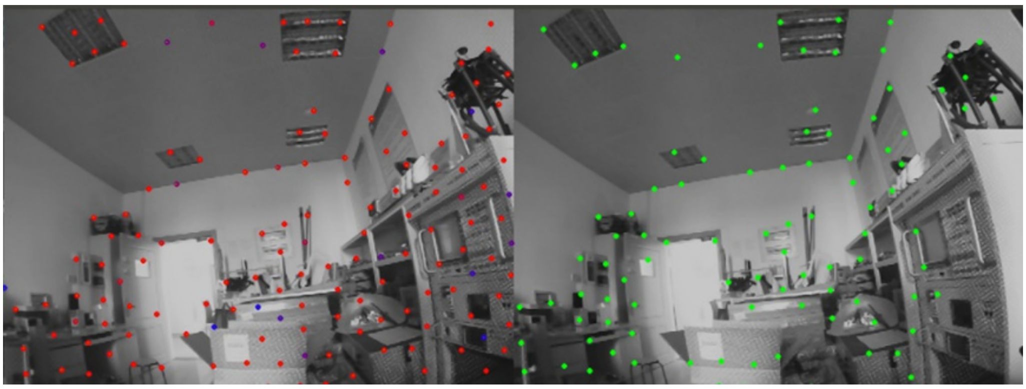

Fig. 2. Corner detection

Visual sensor works well in most richly textured scenes, but it does not perform well in scenes with fewer features, such as glass or white walls. And the localization tracking is easy to lose when moving fast. However, because the vision would not cause drift, the rotation and translation can be measured directly.

Due to the existence of zero bias and noise, IMU has a large cumulative error after long-time use. However, the IMU has the advantages of high output frequency and 6DOF measurement information, and its relative displacement data has high accuracy in a short time. Therefore, the Visual-IMU fusion localization scheme is complementary to some extent. When the vision sensor is invalid due to rapid movement in a short time, the integration of IMU data can provide a short period of precise localization. And the visual localization information is used to estimate the zero bias of the IMU to reduce the divergence and cumulative error. The fusion of the two can optimize the problem of low output frequency of visual pose estimation, meanwhile, the accuracy of pose estimation is improved. The whole system is more robust.

3.2 Automatic Navigation Algorithm

During the disinfection process, the robot needs to automatically map out a feasible route to its destination. Firstly, the robot should find out the global optimal path to the target point in the global-cost-map through global path planning. Then, it conducts local path planning in the local-cost-map. According to the environmental changes, local path planning and navigation can be carried out in real time to avoid obstacles.

3.2.1 Global Path Planning

The A* algorithm is used to implement global path planning. A* algorithm is the most effective direct search method to solve the shortest path in the static road network, which has good performance and accuracy.

The global-cost-grid-map should be established on a two-dimensional plane firstly. A* algorithm uses the evaluation function to estimate the distance between any point on the map and the target position, and select the optimal direction for global searching. The evaluation function is defined as:

$$f(n) = g(n) + h(n). \tag{2}$$

where $f(n)$ is the comprehensive priority of node n. $g(n)$ is the actual cost of an optimal path from s to n. $h(n)$ is the actual cost of an optimal path from n to a preferred goal node of n [13]. The smaller the value of $f(n)$ is, the better the path is. A* algorithm uses two sets, $open\_set$ and $close\_set$, to represent the nodes to be traversed and the nodes that have been traversed. When the algorithm starts, firstly, the starting point should be put in the $close\_set$. Then, the reachable nodes around the starting point would be put into the $open\_set$, pointing to the starting point as their parent node. After calculating the $f(n)$ value of the node in the $open\_set$, it needs to select the node with the smallest $f(n)$ value as the next node to traverse. And take the node out of the $open\_set$ and put it in another one. After that, put the nodes around this node into the $open\_set$ and point to that node. Finally, the value of $f(n)$ for each node in the $open\_set$ would be updated. The process is repeated until the next node is the endpoint.

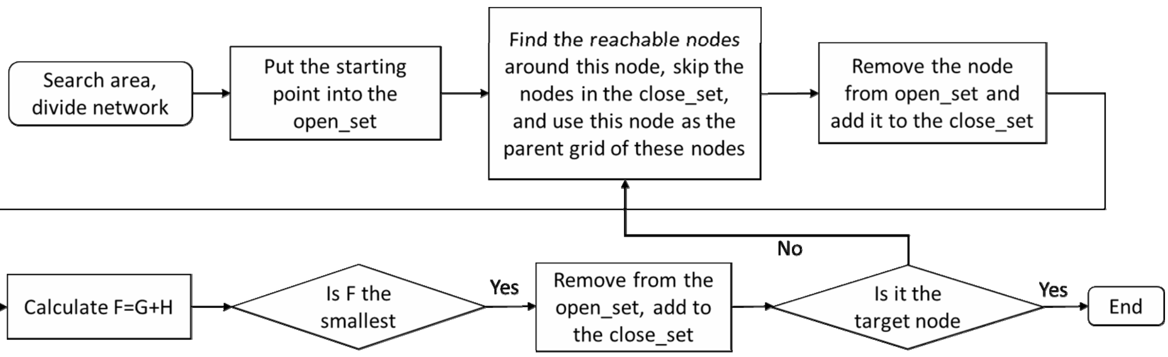

Fig. 3. Specific implementation steps of A* algorithm

3.2.2 Local Path Planning

The dynamic window is used for local path planning and obstacles avoidance.

The dynamic window method (DWA algorithm) is to sample multiple groups of velocity data in the space of velocity (v, w), and simulate the trajectory of the robot in the next time at these speeds. These trajectories can be evaluated by the evaluation function. The speed corresponding to the optimal trajectory is selected to drive the robot until it reaches the target position [14].

Here, considering the particularity of the space of the train carriage, the motion model can be simplified as that the robot can only move forward or rotate. Because the moving distance of the robot at the adjacent times is relatively short, the trajectories of adjacent times can be approximated to a short straight line. Get the motion trajectory of the robot in the XY plane coordinate system over a period of time as follows:

$$\begin{cases} x = x + v\Delta t\cos(\theta_t) \\ y = y + v\Delta t\sin(\theta_t) \\ \theta_t = \theta_t + w\Delta t \end{cases}. \tag{3}$$

In the formula, v represents the given velocity, θ_t represents the angle of rotation, and w represents the angular velocity.

The velocity vector space V_r in the dynamic window algorithm is obtained from the intersection of three constrained velocity vector spaces. These three velocity vectors are the velocity vector set V_s controlled by the maximum and minimum velocities, the allowable velocity vector set V_a based on collision consideration and the velocity dynamic window V_d limited by acceleration. In the selection of the permitted speed, a principle should be followed: at the current speed, the robot will not collide with the surrounding obstacles at the next moment in its reachable acceleration. So, the V_r can be expressed as $V_r = V_s \cap V_a \cap V_d$. Each velocity vector in this space can enable the robot to avoid obstacles under the condition of satisfying the own acceleration ability.

The speed vector set V_a is used to constrain the speed of the robot, so as to achieve local obstacle avoidance. This set of speeds is considered feasible if the robot can stop before hitting an obstacle. Equation (4) gives the definition of the set of V_a, where $dist(v, w)$ represents the distance between the nearest obstacle and the curve determined by the velocity pair (v, w).

$$V_a = \{v, w \mid v\} \le \sqrt{2 \cdot dist(v, w) \cdot v_b}, w \le \sqrt{2 \cdot dist(v, w) \cdot v_b}\}. \tag{4}$$

Considering that the acceleration provided by the robot motor is limited, and excessive acceleration will make the robot more likely to hit obstacles in the train carriage with multi obstacles, the V_d is used to further constrain the velocity vector space. In this set, the velocity vector space is constrained to a dynamic window. This window is centered on the current actual speed of the robot and expands based on the given acceleration. Speeds exceeding the robot acceleration limit will not appear in the window. The position and width of this window will change with the current speed and allowable acceleration of the robot. Each simulated trajectory can be evaluated as follows:

$$\cos t = \alpha \cdot P + \beta \cdot L + \gamma \cdot O. \tag{5}$$

Where, P denotes the distance to the path from the endpoint of the trajectory in meters, L denotes the distance to the local goal from the endpoint of the trajectory in meters, and O denotes the maximum obstacle cost along the trajectory in obstacle cost (0-254) [15]. α, β and γ were the weighted coefficients of the related terms.

Choosing the path with the least cost enables the robot to keep away from obstacles and move towards the target near the optimal path. Because of the particularity of the carriage environment, a better γ parameter needs to be chosen in the actual movement. In such a narrow space with many obstacles, when γ is too large, the robot is too sensitive to the judgment of the surrounding obstacles, which leading to great changes in the calculated trajectories each time and unable to move smoothly. When γ is too small, the robot is more likely to collide with obstacles due to its poor judgment of obstacles.

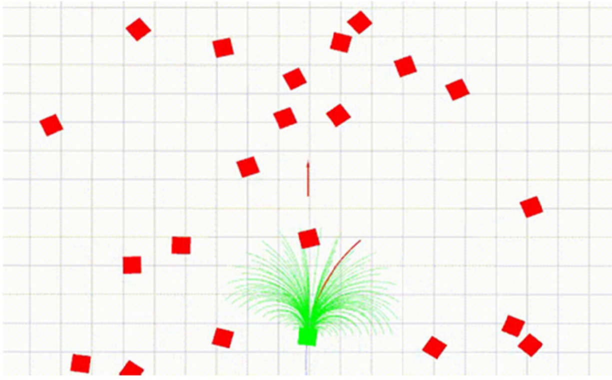

Fig. 4. DWA path planning

3.3 Algorithmic Optimization of Narrow Space

In the local obstacle avoidance, the traditional navigation algorithm regards the robot as a particle, and the velocity calculated by the dynamic window algorithm will not cause the robot to collide with the obstacle in theory. The width of the train carriage passage is about 60cm. The movement space is small. Moreover, the stereo camera used in this paper is placed at the front of the robot, so it is difficult to judge the obstacles on the side and back well. Therefore, we adopted the obstacle expansion method and limited the obstacle avoidance range to improve the algorithm to adapt to the train carriage.

3.3.1 Obstacle Expansion Method

In order to solve the problem of obstacle avoidance in the narrow space, the edge of the obstacle point identified by the camera should be expanded outward, so as to increase the obstacle cost in the calculation of the dynamic window algorithm. The expansion radius is determined by the width of the channel and the shape of the robot. Excessive expansion radius produces over-sufficient-expansion phenomenon, which makes the robot too sensitive to the perception of obstacles, and it will pause constantly when moving. Too small expansion radius will produce insufficient-expansion phenomenon, which will make the robot easily hit obstacles in such a narrow space. The expansion radius selected after the test should enable the robot to pass smoothly through the narrow channel without colliding with obstacles.

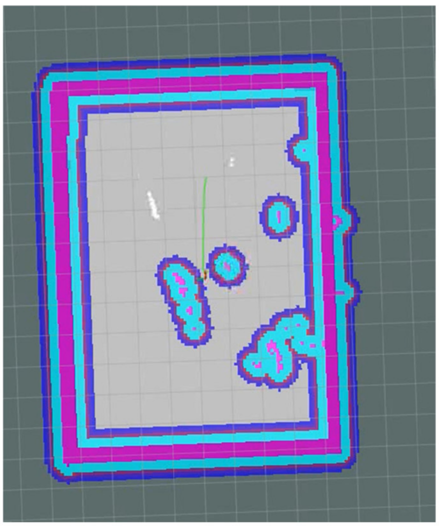

Fig. 5. Obstacle expansion: The light blue area is the obstacle area after expansion

After enlarging the area of the obstacles by obstacle expansion method, the areas that the obstacles expand will be considered impassable in the calculation (in fact, there is no obstacle). In this way, the buffer space is left for the avoidance of the disinfection robot in train carriage. The disinfection robot will not collide the obstacles because of the location of the camera. When the robot judges that the obstacle is too close to the front (<20cm from the camera) or the obstacle is too large to pass, it will trigger the emergency stop mechanism and stay in place until the obstacle is removed. It can avoid unnecessary collision due to blind motion.

3.3.2 Limit the Range of Obstacle Avoidance

Since the corridor of the carriage is generally a rectangular two-dimensional narrow space, the shape of the disinfection robot is better to make into a cuboid. When sterilizing, the robot only needs to move forward and slightly adjust the direction of its advance, rather than perform a large rotation. Therefore, the focus only needs to consider the adjustment of the moving position and posture in the avoidance of obstacles on the left and right sides, and the emergency stop or avoidance when obstacles appear ahead. The obstacles behind it can be ignored.

Since the camera is placed at the front of the robot, the point cloud range extracted by the stereo camera that needs to be calculated can be limited to the 45° fan-shaped area centered on the camera. As long as the calculations are performed successfully in this area, the body can pass aisle safely without the need for further inspection outside this area. Point clouds in this region are extracted for obstacle detection and calculation, while others are ignored. In this way, the local path planning can not only avoid large changes in a short time, but also greatly improve the speed of calculation.

4 Experimental Results and Analysis

4.1 Experimental Scheme

4.1.1 Hardware

(1) Body structure

The body of the disinfection robot adopts the rectangular structure to adapt to the corridor of the train. The stereo camera is placed in the front of the robot, the disinfection spray device is placed on the rear of the body, and the visual processor and servo control module are placed on the bottom.

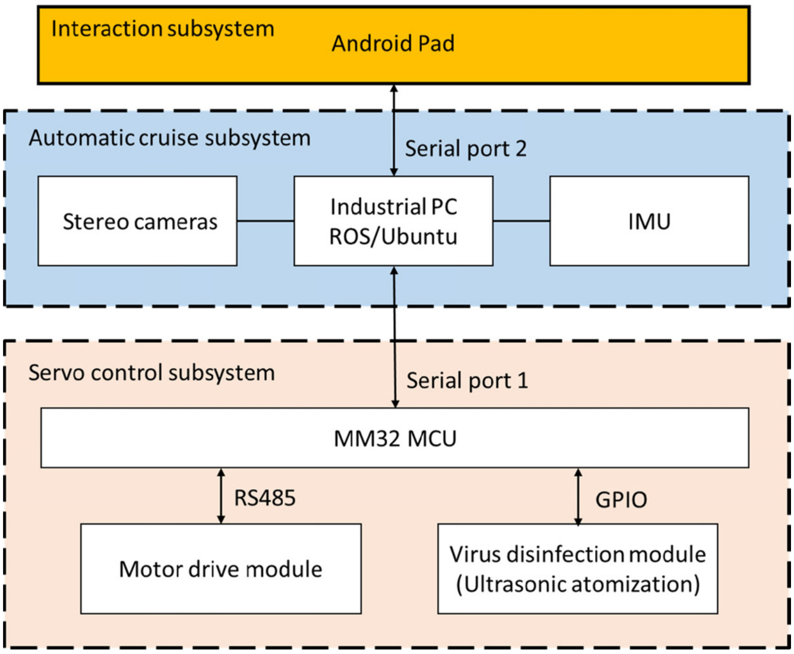

Fig. 6. Diagram of the disinfection robot system

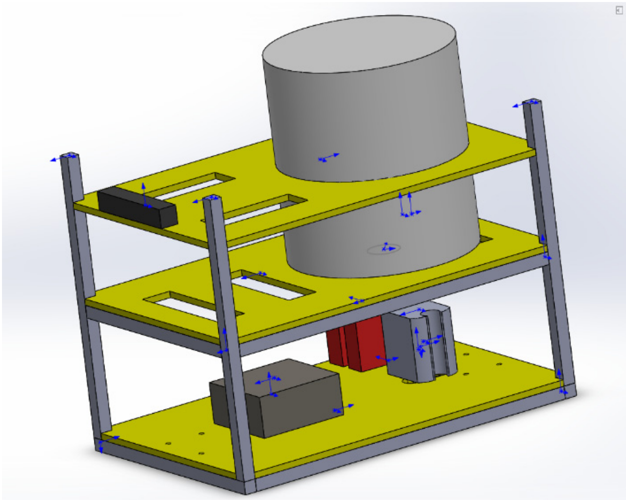

Fig. 7. Circuit for computing *P'*(*x*)

(2) Visual processing module

The stereo camera produced by the company of MYNTAI is used in this paper, with built-in six-axis IMU sensor and active infrared detector (AIR). The supporting SDK can obtain binocular raw images and IMU raw data.

(3) Servo control module

The servo control subsystem realizes the motion servo and other control of the disinfection robot through MM32 single chip microcomputer, mainly the motor drive and disinfectant spray. The core of MM32L373PF is ARM®Cortex®-M3, which is responsible for receiving the motion control instruction issued by the processor. The single chip computer transmits the data to the two motor drives at the interval of 20ms, so as to control the movement process of the robot. It'll also outputs PWM signal which will be converted into voltage signal to control the virus disinfection module.

Virus disinfection module (ultrasonic atomization) adopts the PWM signal generated by MM32 controller to control the oscillator circuit of three-point. The 1.7MHz sinusoidal wave would be generated to drive the piezoceramics to convert electrical energy into mechanical energy and atomize the disinfectant. Finally, the centrifugal fan will spray the disinfection mist to every corner of the carriage to realize the diffusion covering disinfection.

The motor drive module is responsible for receiving the control instructions of the machine in real time, so as to drive the motor to move according to the command transmitted by the upper layer, and feedback the running status of the motor.

A tablet is also put on the robot as a human-computer interaction device.

4.1.2 Software

(1) ROS

ROS is an open-source operating system for robots. It is a distributed framework. Based on the ROS framework, the VIO Localization Scheme which integrates camera and IMU data is used to realize the SLAM algorithm for indoor localization. Then we've improved the automatic navigation algorithm suitable for the narrow space in the carriage.

(2) VINS-Fusion

VINS-Fusion is an optimization-based multi-sensor state estimator from Hong Kong University of Science and Technology on January 12, 2019, which can achieve accurate self-localization for autonomous applications [8-11]. VINS-Fusion supports a variety of visual inertial sensor types (including the stereo camera + IMU used in this paper), supports online spatial and temporal calibration, and has the function of closed-loop detection. There are three parts of VINS-Fusion: front-end, initialization and back-end nonlinear optimization.

In the experiment, the raw images obtained from the stereo camera and the raw IMU data were transmitted to the nodes in the VINS-Fusion of the visual processor for data fusion. After the localization was realized, the position information was transferred to the navigation node through the transformation of TF tree coordinate. Then the improved navigation algorithm was run and the calculated motion data would be transmitted to the control node on the MCU in real time. The two-wheel differential speed model was established on MCU, and the data of angle and speed were converted into the rotating speed of the two wheels to achieve the precise control of the robot.

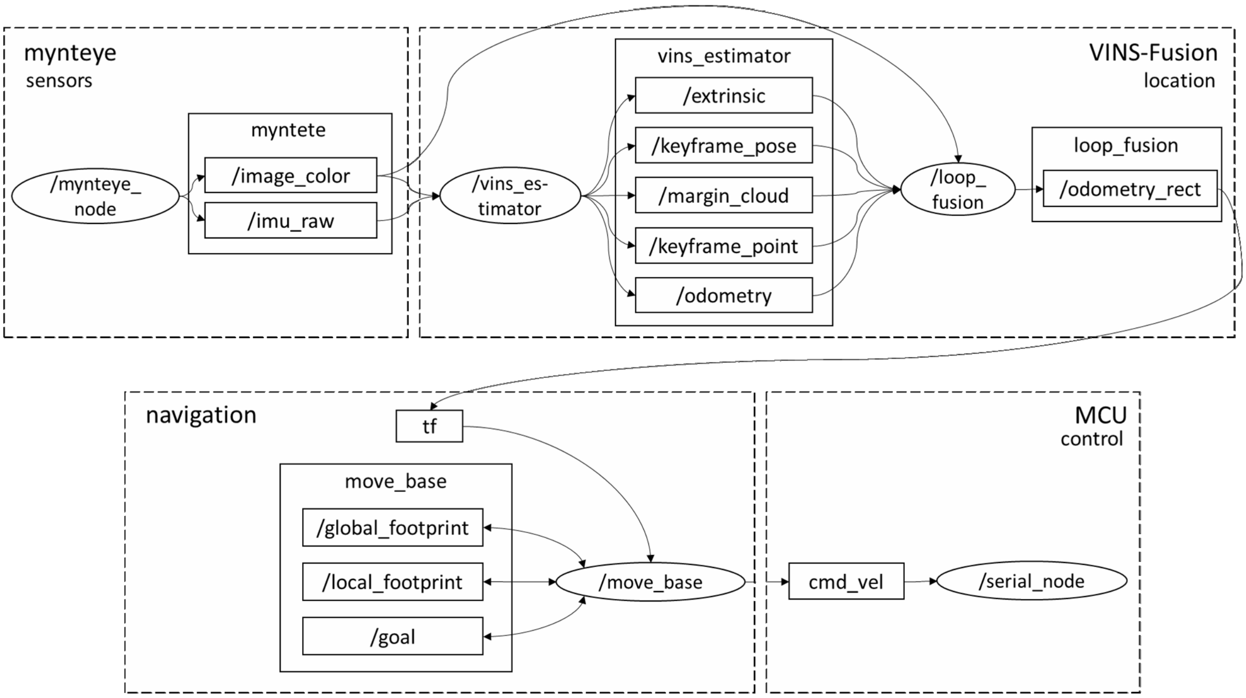

Fig. 8. Brief diagram of ROS node

4.2 Results and Analysis

On the experiment, we set up our disinfection robot. The Ubuntu 18.04 was used in the visual processor. The algorithm code was run on the basis of ROS and VINS-Fusion. Two rows of armchairs were placed in the indoor space, and the width of the middle aisle was about 65cm, to simulate the environment inside the carriage. The disinfection robot was placed at the starting point of the aisle and the target point was set at the end of the aisle. The program was started to realize autonomous movement, localization, navigation and obstacle avoidance.

Fig. 9. Set up the simulated carriage environment and carry out the test

The first test adopted the original algorithm. In the test, when the robot entered the narrow aisle, it would overspin and constantly pause, and was unable to move in the aisle smoothly. When obstacles appeared in the front side (such as the operator suddenly appeared), the robot couldn't avoid correctly or stop in an emergency, resulting in collision between the body and obstacles in front or seats on both sides.

While the second test made used of the improved version. In the local path planning mentioned above, the maximum and minimum speeds set and the allowable range of acceleration of the robot would affect the position and width of the velocity vector dynamic window, thus affecting the movement of the robot. Since the space that the robot could pass through in this scene was small, the maximum speed of the robot in the X-axis direction and rotation were set to a smaller value. Acceleration was set to 1.3 (unit: m/s. It's a smaller value for the normally movement of a robot). Because the MCU sent the rotation speed of the wheels to the driver every 20ms, excessive acceleration would cause the robot to collider with obstacles before it reacts.

The expansion radius parameter mentioned above was set as 0.03-0.07-0.09 (unit: m) respectively. And the obstacle avoidance range was set as a 45° fan-shaped area centered on the camera for testing. The runtime debugging window is shown in the Fig. 10. The light blue areas in the figure represent the expansion range of the obstacle. When different expansion radii were set, the width of the passageway marked as passable in the middle would change accordingly.

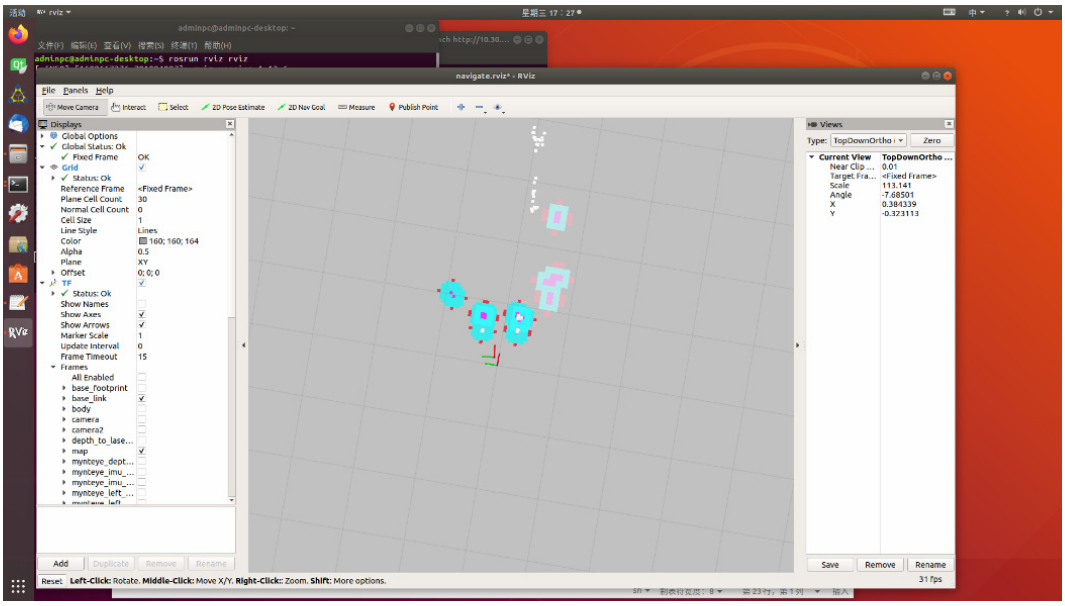

Fig. 10. Algorithm running debugging window

After the field test, when the expansion radius parameter of the obstacle was set around 0.03 and 0.09, the effect was poor, insufficient-expansion and over-sufficient-expansion phenomena appeared respectively, and collision and stagnation would occur during passage. When the parameter was set around 0.07, the disinfection robot could pass through the simulated narrow passageway successfully. The 45° fan-shaped obstacle avoidance range had a better effect, and would not result in over-detection of obstacles on both sides, which would lead to stagnation in operation. Moreover, when the operator suddenly appeared in front of the robot, the robot using the improved algorithm could successfully complete the avoidance of obstacles while moving forward, or trigger the emergency stop mechanism when the obstacles could not be avoided.

Based on the test, the following conclusion can be drawn that the disinfection robot could realize the localization, navigation, obstacles avoidance well in narrow space of the train by the using of VIO localization scheme and the improved navigation algorithm. The algorithm improved by obstacle expansion method and limiting the range of obstacle avoidance could make the disinfection robot more suitable for the application scenarios. The robot scheme could meet the application demand of the disinfection robot for the train carriage.

Fig. 11. The emergency stop mechanism was triggered when the obstacles couldn't be avoided

5 Conclusion

Under the trend of the normalization of epidemic prevention and control of CoviD-19, the emergence of disinfection robots has reduced a lot of pressure for people's daily disinfection and cleaning work. This paper presents a scheme of localization, navigation and obstacle avoidance for the disinfection robot in the train carriage. The using of the VIO positioning scheme combining vision and IMU sensors can provide more accurate localization of disinfection robots, and make it more adaptive and robust to the special environment of the train carriage. The combination of global and local path planning is used to achieve navigation and obstacle avoidance. The global path planning realizes the automatic navigation from the robot's position to the target point on the global cost map. The local path planning implements dynamic obstacle avoidance in local cost map. Targeted at the narrow aisle space in train carriage, based on the dynamic window algorithm and A* algorithm, the algorithm is improved by obstacle expansion method and limiting the range of obstacle avoidance, to adapt to the special environment.

Finally, we set up the simulated environment of the carriage aisle, completed the construction and debugging of the disinfection robot, so as to conduct field test of our algorithm. The ROS framework was used to complete code implementation of the improved algorithm on the basis of VINS-Fusion. In field test, after adjusting parameters such as acceleration, expansion radius and obstacle avoidance range, the robot could pass smoothly in the narrow corridor, and the improved algorithm had a good effect.

Acknowledgements

This work was supported by the College Students' Innovative Entrepreneurial Training Plan Program in Xiamen University (202110384002, S202110384330); the Xiamen University President fund (20720212009); the Fujian Science and technology project (2020H6001).

References

[1] N.V. Doremalen, T. Bushmaker, D.H. Morris, M.G. Holbrook, A. Gamble, B.N. Williamson, A. Tamin, J.L. Harcourt, N.J. Thornburg, S.I. Gerber, J.O. Lloyd-Smith, E. de Wit, V.J. Munster, Aerosol and surface stability of SARS-CoV-2 as compared with SARS-CoV-1, New England journal of medicine 382(16)(2020) 1564-1567.

[2] E. Ackerman, Autonomous robots are helping kill coronavirus in hospitals, IEEE Spectrum (2020).

[3] G.Z. Yang, B.J. Nelson, R.R. Murphy, H. Choset, H. Christensen, S.H. Collins, P. Dario, K. Goldberg, K. Ikuta, N. Jacobstein, D. Kragic, R.H. Taylor, M. McNutt, Combating COVID-19—The role of robotics in managing public health and infectious diseases, Science Robotics 5(40)(2020) eabb5589.

[4] K. Cresswell, A. Sheikh, Can Disinfection Robots Reduce the Risk of Transmission of SARS-CoV-2 in Health Care and Educational Settings?, Journal of Medical Internet Research 22(9)(2020) e20896.

[5] B. Fang, G. Mei, X. Yuan, L. Wang, Z. Wang, J. Wang, Visual SLAM for robot navigation in healthcare facility, Pattern Recognition 113 (2021) 107822.

[6] Q. Feng, X. Wang, Q. Qiu, C. Zhang, B. Li, R. Xu, L. Chen, Design and test of disinfection robot for livestock and poultry hous, Smart Agriculture 2(4)(2020) 79-88.

[7] J. Zhao, Research on positioning and navigation technology of autonomous food delivery robot in high-speed rail environment, Master's thesis, Southeast University, 2019.

[8] T. Qin, P. Li, S. Shen, Vins-mono: A robust and versatile monocular visual-inertial state estimator, IEEE Transactions on Robotics 34(4)(2018) 1004-1020.

[9] T. Qin, J. Pan, S. Cao, S. Shen, A general optimization-based framework for local odometry estimation with multiple sensors, <https://arxiv.org/abs/1901.03638>, 2019.

[10] T. Qin, S. Cao, J. Pan, S. Shen, A general optimization-based framework for global pose estimation with multiple sensors, <https://arxiv.org/abs/1901.03642>, 2019.

[11] T. Qin, S. Shen, Online temporal calibration for monocular visual-inertial systems, in: Proc. 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2018.

[12] D. Gálvez-López, J.D. Tardos, Bags of binary words for fast place recognition in image sequences, IEEE Transactions on Robotics 28(5)(2012) 1188-1197.

[13] P.E. Hart, N.J. Nilsson, B. Raphael, A formal basis for the heuristic determination of minimum cost paths, IEEE transactions on Systems Science and Cybernetics 4(2)(1968) 100-107.

[14] D. Fox, W. Burgard, S. Thrun, The dynamic window approach to collision avoidance, IEEE Robotics & Automation Magazine 4(1)(1997) 23-33.

[15] B.P. Gerkey, K. Konolige, Planning and control in unstructured terrain, in: ICRA Workshop on Path Planning on Costmaps, 2008.