Proceeding of the IEEE
International Conference on Robotics and Biomimetics
Dali, China, December 2019

Research on Robot Navigation Technology in Emergency Events*

Chao Wang[1)], Yueyang Bian[2)], Bin Zhao[1)], Guodong Qi[2)] and Weijun Zhang[#2)]

[1)] *Kunming Power Supply Bureau*
Yunnan Power Grid Co.,Ltd.
Kunming, China
1349979306@qq.com

[2)]*School of Mechanical Engineering*
Shanghai Jiao Tong University
Shanghai, China
sjtubyy@sjtu.edu.cn

***Abstract*—This paper presents a method for quickly building navigation maps using building fire evacuation maps and doorplates. This paper uses image processing, robot navigation and other related technologies to realize the rapid conversion of fire evacuation maps into grid maps that can be applied to robot positioning and navigation. In addition, by extracting the doorplate in a fire evacuation map, a semantic map based on the doorplate is constructed. The global position of a robot is associated with the doorplate information on a floor. For the difficulty of robot positioning, this paper proposes a method of using doorplate semantics to realize robot positioning and navigation, combined with improvement in the A-star algorithm. Finally, the effectiveness and practicability of this research were verified by a simulation environment and experimental verification.**

Keywords—Emergency Events, Doorplate Identification, Map Construction, Positioning Navigation

I. INTRODUCTION

In urban buildings, emergencies such as fires, explosions, and chemical spills occur from time to time, endangering the safety of life and property. To ensure the safety of rescue workers, explosive ordnance disposal (EOD) robots were created. A navigation map needs to be built when a robot moves autonomously in an environment. However, due to the suddenness of an emergency, it is impossible to construct a navigation map of the disaster site in advance. Existing navigation map construction methods require entering the disaster environment and cost a large amount of time. Therefore, we must make full use of known building information to quickly build navigation maps of unfamiliar building environments. A fire evacuation diagram is a kind of building information that is easily obtained for a building environment. Doorplates are also the most common information in a building and can provide important assistance for the positioning of robots.

Robot positioning means that the robot acquires information about itself and the environment through various sensors, transforms the sensor data into a navigation map, and finally calculates its position and posture in the map. In view of the positioning problem of robots, researchers have put forward various ideas and methods. According to different positioning principles, they can be divided into three types: dead reckoning (DR) [1], map matching and beacon-based. Tsai, Ching-Chih developed a method to determine the posture of a mobile robot [2]. R. B. Noland used map matching to realize vehicle navigation in complex urban road networks [3]. Kajioka realized the indoor positioning of a robot based on a Bluetooth LE beacon [4]. This paper uses doorplates to assist positioning, which is a beacon-based method. Since a fire evacuation map usually contains doorplate information, a navigation map can be constructed from a fire evacuation map.

This paper mainly studies the positioning and navigation of robots in emergency events. To meet the challenges of an unknown environment in an accident scene, this study proposes a method of quickly constructing a map using a fire evacuation map and semantic navigation based on doorplates. This paper first studies the use of image processing technology to obtain doorplate information and quickly identify the doorplate. Second, a realization method of transforming the fire evacuation map into the robot navigation map is studied. Finally, robot positioning is realized by using the doorplate semantics, and the A* algorithm is improved for navigation in combination with the actual situation.

II. DOORPLATE IDENTIFICATION

Doorplate identification includes two parts: doorplate positioning and character recognition. Doorplate positioning needs to determine whether the doorplate appears in view. If it appears, it is necessary to determine the specific position of each character on the doorplate. Then, the robot filters through a series of restrictions to determine the number on the doorplate. In the process of identification, the current doorplate is identified by feature extraction and a support vector machine (SVM) classifier.

A. Doorplate positioning

To better locate the doorplate, this paper combines maximally stable extremal regions (MSER), the width and height ratios, the transformation of the stroke width, and the text continuity hypothesis to obtain a fast and effective algorithm that eliminates reflection, slight jitter and even many other nondoor number text interferences. The effect is very significant. The specific steps are as follows:

Step 1: Use the MSER algorithm to extract the candidate text area. At this point, although the text can be extracted, a large number of nontext areas are also extracted;

Step 2: Aggregate overlapping boxes. Take the minimum box as the candidate box after aggregation.

Step 3: Select boxes with a length-to-width ratio of 0.5-5.

* This work is partially supported by China Southern Power Grid (YNKJXM20170208)
# Corresponding author, zhangweijun@sjtu.edu.cn

Step 4: Use the character width-based stroke width transform (SWT) algorithm to filter the remaining candidate boxes [5].

Step 5: Look for a set of numbers in a row, which is considered to be the doorplate.

(a) MSER (b) SWT (c) Selection

Fig. 1 Image processing of a doorplate

B. Character recognition

Then, we need to identify the characters extracted and segmented in the previous steps. First, Histogram of Oriented Gradient (HOG) feature extraction is performed on the extracted characters. The HOG is performed on the local elements of the image and is therefore unaffected by geometric distortion and illumination. The subtle interference of the local image has little influence on the detection result. The steps of HOG feature extraction are as follows:

Step 1: Image preprocessing: The image is grayscale and corrected by gamma.

$$I(x,y) = I(x,y)^{gamma} \quad (1)$$

Step 2: Find the gradient of each point in the image:

$$G_x(x,y) = H(x+1,y) - H(x-1,y) \quad (2)$$

$$G_y(x,y) = H(x,y+1) - H(x,y-1) \quad (3)$$

Step 3: Find the gradient size and direction of each point:

$$G(x,y) = \sqrt{(G_x(x,y)^2 + G_y(x,y)^2} \quad (4)$$

$$\alpha(x,y) = tan^{-1}\frac{G_x(x,y)}{G_y(x,y)} \quad (5)$$

Step 4: Divide the entire image into several cell units, perform histogram statistics on each unit, and obtain a feature vector for each unit.

Step 5: Combine the cell units and normalize them to form the HOG descriptor. Each image with the same size can have a feature vector with the same dimension that can be used for subsequent classification.

A support vector machine (SVM) refers to an algorithm widely used in the field of machine learning. Here, the recognition range is limited to numbers. When the certainty is more than 50%, the recognition is considered correct.

The previously obtained candidate sequences are identified one by one and judged for whether they are in the doorplate list. For example, 303 is in the list and is determined to be the doorplate, while 30273051 is not in the list and is excluded.

III. MAP CONSTRUCTION

SLAM is a classic problem in the field of robotics. Existing SLAM solutions require the use of 2D radar to establish a grid map in advance [6]. However, in emergency situations, it is often impossible to build maps in advance. In addition, there is no time or condition to allow robots to map accident scenes. Many SLAM solutions do not work in these situations.

Based on a traditional robot navigation map, this chapter uses a fire evacuation map to construct the navigation map of the robot and adds semantic information as a reference for navigation and positioning. First, through image processing, the fire evacuation map is simplified, leaving only the routes that the robot can pass and the obstacles that it cannot pass. Then, the map is rasterized for navigation positioning of the robot. Finally, doorplate information is attached to the map.

A. Rasterize the map

1) Remove redundant information

Apart from the required building structure information, a fire evacuation map provides a large amount of other redundant information, such as text, signs, and arrows. Before the map is rasterized, it must be processed to remove text and logos from the image.

The character information in the image can be processed by the method of text recognition mentioned in the previous chapter. The doorplate information in the map is extracted by the method of maximally stable extremal regions (MSER). After confirming that the text is text information, the corresponding characters are removed. For other symbols, selection and discrimination are performed using various conditions, such as color, aspect ratio, and symbol classification.

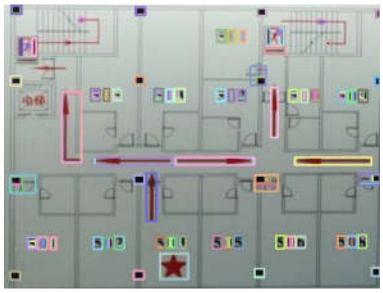

Fig. 2 Extracted region by MSER

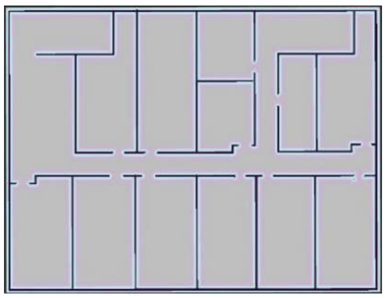

Fig. 3 A plain map without redundant information

2) Rasterize the map

To save the environmental feature information as much as possible and to facilitate navigation planning calculations, this paper uses a two-layer grid map to describe the basic navigation information of a building. A layer of a raster map containing the information of all of the maps is called the parent map. The parent map is obtained by direct rasterization of the fire evacuation map. On the basis of the parent map, a subgraph is obtained by adjusting the resolution of the parent map. The fire evacuation map rasterization process is as follows:

Fig. 4 Rasterization steps

The fire evacuation map is binarized, where all obstacles are marked as 0 and the barrier-free area is marked as 1. This results in a binarized raster map containing only obstacle information. To more fully characterize obstacles and pass ability, the raster map is further processed and converted into a grayscale raster map as follows:

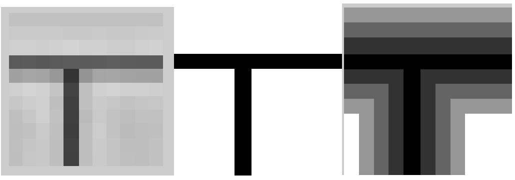

Fig. 5 Original, binarized, and grayscale images

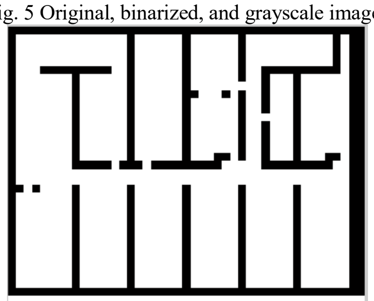

Fig. 6 Binarized map

B. *Add semantic information*

1) Extract the doorplate location

The doorplate information is obtained by extracting the room number in the image and then matching the room number with the door. Using the doorplate extraction technology, the doorplate can be extracted from the fire evacuation map.

To determine the specific location of the doorplate, a template matching method is used. The sector in the map represents the location of the doorplate. With the help of template matching, the location of each doorplate is determined as shown below.

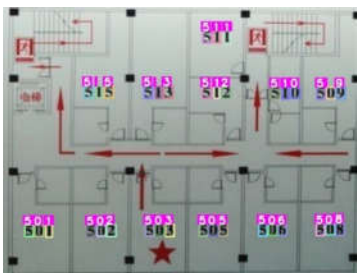

Fig. 7 Doorplate number extraction

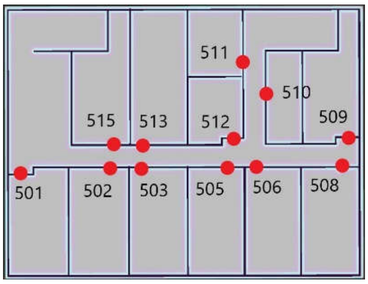

Fig. 8 Corresponding doors and numbers

2) Doorplate semantics

The position of the doorplate extracted from the fire evacuation map is inaccurate. Affected by factors such as the difference in the hanging position of the doorplate and the accuracy of the fire evacuation map, the identified door position is only an approximate position. Therefore, the role of doorplate semantics is to provide a reference for the robot in global positioning. In a two-layer raster map, the subgraph is used for global positioning and navigation. Therefore, the semantic layer, in which the map resolution is the same as that in the subgraph. All semantic items are indexed, and the indexes are stored at the corresponding locations in the semantic raster map.

IV. Positioning and Navigation

Since the fire evacuation map is not accurate, this article extracts structural nodes and semantic nodes from the map to assist positioning and navigation. In this paper, a semantic node refers to doorplate information, and a structural node refers to the actual position obtained by the sensor. If many semantic nodes can be obtained from the fire evacuation map, the problem is degraded to the identification of the route signs in the topology map, and only the doorplate needs to be identified in the walking process to complete the positioning. However, if enough semantic nodes cannot be extracted from the fire evacuation map, the structural nodes can be very helpful.

In this paper, an absolute positioning method and a relative positioning method are comprehensively applied. Absolute positioning methods include doorplate semantic positioning, structural feature positioning, and compass positioning. The relative positioning method uses a driving wheel encoder as an odometer to obtain the moving distance of the robot.

A. *Doorplate semantic positioning*

The semantic positioning of a doorplate is used to detect the doorplate information in the environment through the visual sensor, and then the semantic map is retrieved to obtain the orientation of the doorplate in the map, thereby obtaining the global position of the current robot and achieving absolute positioning.

The use of doorplate semantic positioning involves two steps: global positioning and local positioning. Global positioning finds the location of the detected doorplate in the navigation map by matching semantics. Local positioning refers to estimating the position of the robot relative to the doorplate through data provided by the vision sensor.

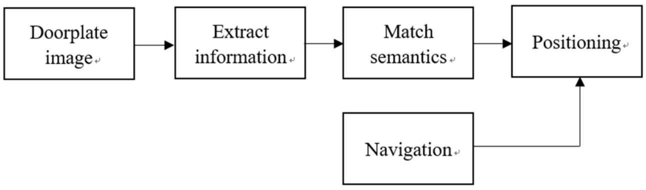

Fig. 9 The global positioning process by doorplate semantics

The principle of local positioning is to use the parallax method combined with the odometer to obtain the local position of the robot relative to the doorplate. After the robot detects the doorplate through the camera, the platform posture is appropriately adjusted to position the doorplate in the center of view, which will ensure that the doorplate number is still in the camera's field of view for the next shot. After the robot moves a certain distance, an environmental image is taken, and the doorplate information is extracted. By calculating the position of the doorplate and the shooting angle in the image, the parallax of the two shots can be calculated, and the position of the robot relative to the doorplate can be calculated in combination with the odometer data.

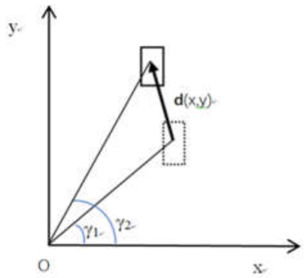

Fig. 10 Calculation of local position by the parallax method

There are three main sources of error in semantic positioning:

(1) The location of the doorplate. In this paper, it is assumed that the doorplate is installed in the middle of the door (horizontal direction). After detecting the doorplate in the map by means of template matching, the center of the door is taken as the door position. The actual doorplate installation position may be to the left or right. This factor causes an error of approximately 0.5-1.0 m.

(2) Map accuracy. There is a certain error in the drawing of the fire diagram, and there is a deviation in the obtained position information of the doorplate based on the fire diagram.

(3) Local positioning error. The systematic error in this method is very small on the premise that the two image sampling intervals are large enough. Therefore, the calculation deviation caused by the odometer deviation is mainly considered. Theoretical analysis and experiments show that under the premise of the robot moving straight ahead, if the robot moves a distance of more than 10 cm, the mileage error can be reduced to less than 5%.

B. Path planning and navigation

In this paper, the A* algorithm is used to complete global path planning based on the raster map. The basic idea of the A* algorithm is to estimate the path cost in the raster map and select the grid with the least path cost as the path passing point.

It is noted that the planned path is often not the optimal running path due to the size and direction of the grid. As shown in the following figure, the optimal path should be the path indicated by the yellow arrow, not the blue grid.

The main reason for this phenomenon is that in the grid map, there are only 8 kinds of movement directions for the robot, that is, 0°, 45°...315°, and this is not the case in actual walking. To solve this problem, this paper proposes a navigation method. The basic idea of this method is that when the robot walks, its moving target is not the current mesh but a certain grid in front of the path. Replan a path between the current raster and the target raster.

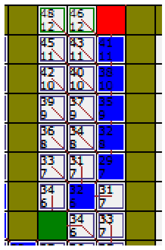

Fig. 11 Difference between the A* path and optimal path

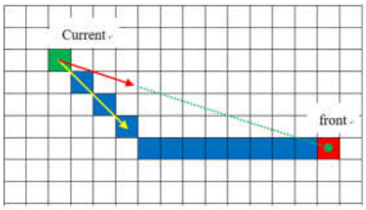

Fig. 12 The principle of front navigation

As shown in the above figure, the green grid is the current point, and the red grid is the front point. According to the navigation method, the moving direction of the robot is shown by a red arrow, and the moving path is a path indicated by a green dotted line.

The basic steps based on the navigation method are as follows:

Step 1: Select the point closest to the current position in the A* track as the current point, and select the appropriate number of previous steps.

Step 2: Use the line between the current point and the front point as the path of the robot motion.

Step 3: Check whether the moving motion path passes through an obstacle, and if it is unobstructed, it acts as the moving path of the robot.

Step 4: If the path passes through an obstacle, reduce the number of previous steps and replan.

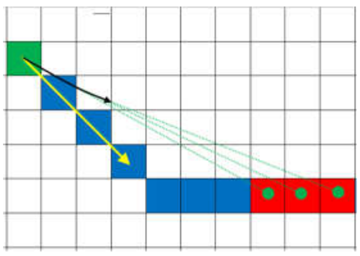

Fig. 13 Front navigation track

V. Simulation And Experiment

A. *Software platform*

The experimental system software consists of three main parts: robot real-time control, ROS-based robot navigation control, and CUDA-based graphics processing software. The main work of this paper focuses on robot navigation control and graphics processing.

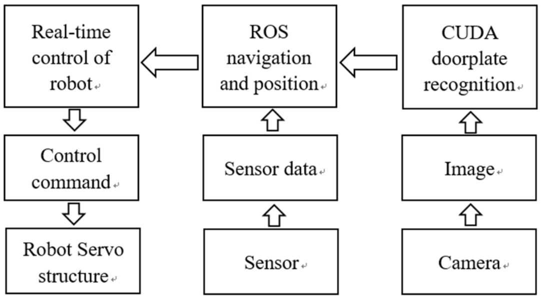

Fig. 14 System software structure

Robot motion control: Control the robot movement according to navigation commands. Position control is realized by a position interpolation algorithm so that the robot can move the specified distance according to the command.

Positioning and navigation: Collect and process sensor data to obtain the environmental information of the robot. Self-positioning and navigation of the robot are achieved by positioning algorithms and maps. The path planning of the robot is implemented according to the task requirements.

Doorplate number identification: Using the graphics processing function provided by CUDA, the camera image is processed to realize the positioning and recognition of the doorplate number.

B. *Simulation*

This part used Gazebo (a 3D dynamic simulator) to establish a simulation environment. The black dots in the figure indicate the mobile robot, which is located in the corridor. Gazebo simulates the necessary sensors for robot navigation and sends the data to the corresponding subject of ROS. Additionally, it receives the walking command sent by the ROS node and walks the corresponding displacement or rotates the corresponding angle in the environment. The data sent by the sensor contain the distance, the direction of movement, the obstacle data of the radar, and the like. To simulate a real sensor situation, some noise is added to the data.

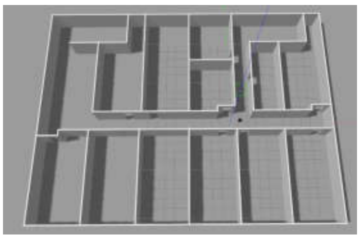

Fig. 15 Indoor environment simulation in Gazebo

Then, a cost map is generated based on the raster map so that the A* algorithm can be used to generate the planned path on the cost map. To perform local obstacle avoidance, the robot also generates a local cost map. The cost estimate of the local cost map is based on the actual observed obstacles, which avoids the inaccuracy of the original raster map or the occurrence of dynamic obstacles. The visualization of the cost map is as follows:

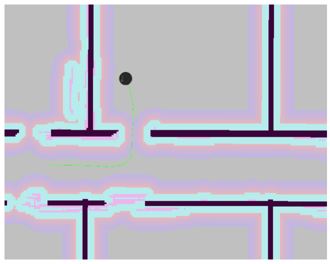

Fig. 16 Navigation in the cost map

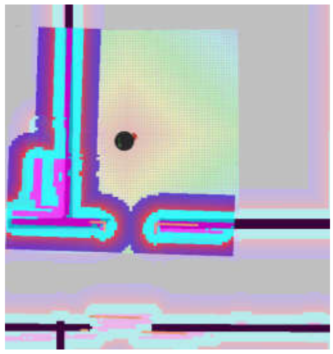

Fig. 17 The local cost map

The simulation results show that when the grid map is consistent with the simulated environment and the initialization position is basically accurate, the robot's self-positioning and navigation effect are good in the whole process, and the algorithm can meet the navigation requirements.

C. *Experiment*

To verify the validity of the research content, this paper conducted a test experiment on the floor in a laboratory environment.

1) *Doorplate number recognition*

The doorplate number of the experimental site is relatively simple, and the color contrast is obvious. Therefore, there is a very high recognition success rate, and no recognition failure occurs in the experiment. The average recognition time is 80 ms, the longest recognition time is 340 ms, and the shortest recognition time is 38 ms.

Fig. 18 Doorplate in the experiment

2) *Navigation*

The main test is based on the feasibility and reliability of semantic map robot navigation. The experimental process is as follows:

Step 1: The robot is placed in the experimental environment and has no starting position information. In addition, the robot is given a command to move to the designated room door.

Step 2: The robot moves in one direction by means of the obstacle avoidance module and detects the doorplate.

Step 3: After the robot detects the doorplate semantics, it combines that information with the compass information to complete its own positioning.

Step 4: The robot reaches the target position through path planning.

A total of three sets of test experiments were conducted to test the performance: the robot near the doorplate, in a corner, and near abnormal walls (targets that are difficult to detect with laser sensors such as glass). The experiments showed that the reliability of using the doorplate semantic positioning method is high, which can effectively eliminate interference.

Fig. 19 Scenarios (in a corner, beside the doorplate, and next to glass)

CONCLUSION

This paper studies the robot navigation problem in the emergency rescue process. This paper proposes a robot positioning method based on doorplate semantics, which enables the robot to realize positioning and navigation under a lack of environmental features and maps. This paper elaborates on a method of quickly constructing a robot navigation map using a fire evacuation map and the specific process of using doorplate semantics to navigate and locate.

REFERENCES

[1] Jimenez A R, Seco F, Prieto C, et al. A comparison of Pedestrian Dead-Reckoning algorithms using a low-cost MEMS IMU[C]// IEEE International Symposium on Intelligent Signal Processing. 2009.

[2] Velaga N R, Quddus M A, Bristow A L. Developing an enhanced weight-based topological map-matching algorithm for intelligent transport systems[J]. Transportation Research Part C, 2009, 17(6):672-683.

[3] Kajioka S, Mori T, Uchiya T, et al. Experiment of indoor position presumption based on RSSI of Bluetooth LE beacon[C]// Consumer Electronics. 2014.

[4] Shah S F A, Srirangarajan S, Tewfik A H. Implementation of a directional beacon-based position location algorithm in a signal processing framework[J]. IEEE Transactions on Wireless Communications, 2010, 9(3):1044-1053.

[5] Epshtein B, Ofek E, Wexler Y. Detecting text in natural scenes with stroke width transform[C]// Computer Vision & Pattern Recognition. 2010.

[6] Chen Y C, Chou J H. Mobile Robot Localization by RFID Method[C]//ICDT 2012, The Seventh International Conference on Digital Telecommunications. 2012: 33-38.