Realization of Autonomous Mapping and Path Planning in a Lower Limb Rehabilitation Robot Moving Platform

Shi-wen TONG\textsuperscript{1,2,*}, Yong-jie MA\textsuperscript{2}, Ying-gang KE\textsuperscript{2}, Shi-bo FU\textsuperscript{3} and Guang CHENG\textsuperscript{4}

\textsuperscript{1}Beijing Key Laboratory of Information Service Engineering, Beijing Union University, Beijing 100101, China
\textsuperscript{2}College of Robotics, Beijing Union University, Beijing 100101, China
\textsuperscript{3}College of Mechanical and Electrical Engineering, Beijing University of Chemical Technology, Beijing 100029, China
\textsuperscript{4}College of Urban Rail Transit and Logistics, Beijing Union University, Beijing 100101, China

*Corresponding author

Keywords: Lower limb rehabilitation robot, Mobile platform, Autonomous mapping, ROS, Gazebo.

Abstract. Map construction is a key technique for the indoor navigation and positioning, which is widely used in the robot mobile platform. This paper has designed and realized an autonomous mapping and path planning method for the lower limb rehabilitation robot moving platform. The method is completed by a Python program and functional packages in ROS. Simulations in Gazebo environment are carried out and the experiments oriented to the real scenarios are also presented. It is proved that the proposed method is very effective.

Introduction

With increases of stroke, sports injuries, disabilities and advances in technology, lower limb rehabilitation robots has become very popular [1], which involves navigating [2], positioning [3], biological information recognition [4], rehabilitation training [5], and other researches [6-7]. Unfamiliar environments, different venues could increase joy of rehabilitation training and stimulate hope of life, however, which will heavily rely on the navigating and positioning. Map construction is an unavoidable core technology, especially for the scenarios that GPS signals are not available. Many methods are proposed [8-10], manual or automatic operation, laser-based, camera-based or combination of them. But, these methods seem to be either too cumbersome or too complicated. In this paper, we proposed a very simple autonomous mapping and path planning method. In addition, this method is based on ROS, a popular robot operation system, which makes it practical and portable.

The paper is organized by five sections: Section 2 presents the design of the lower limb rehabilitation robot moving platform. Section 3 illustrates the autonomous mapping and path planning method. Simulations and experiments are given in Section 4 and conclusions are obtained in Section 5.
Design of the Lower Limb Rehabilitation Robot Moving Platform

Figure 1 presents the structure of the lower limb rehabilitation control system. The ROS-based lower limb rehabilitation robot uses a microcomputer running ROS on the Linux as the central control unit. ROS is an open source software platform for robots, which make it flexible to design all kinds of robots. Sensors such as lidar, camera and pickup etc. are connected to the microcomputer through USB. The robot chassis adopts a three-wheel omnidirectional moving structure (See Figure 2), forming an equilateral triangle between the wheels which are made up of Mecanum wheels and are driven by a motor.

According to Figure 2 and the established coordinates we can build the inverse Kinematic equations for the robot, where \( V_x, V_y, \theta \) stands for the line speed and angular velocity of the chassis, respectively. \( V_a, V_b, V_c \) represents the line speed of each wheel. \( L \) is the distance of the wheel from the center of the chassis.

\[
V_a = \frac{\sqrt{3}}{2} V_x + \frac{1}{2} V_y + L \theta \\
V_b = -\frac{\sqrt{3}}{2} V_x + \frac{1}{2} V_y + L \theta \\
V_c = -V_x + L \theta
\]

Then we can obtain the positive kinematics equations.

\[
\begin{align*}
V_x &= \frac{\sqrt{3}(V_a - V_b)}{3} \\
V_y &= \frac{V_a + V_b - 2*V_c}{3} \\
\theta &= \frac{V_a + V_b + V_c}{3* L}
\end{align*}
\]

Based on the kinematics equations of the robots, we can realize the control of the chassis by using an arduino controller.

Autonomous Mapping and Path Planning

In the uncertain environment, especially in an indoor that the GPS signal is not available, robot moving heavily relies on the map, which is constructed from a laser sensor. With this map, the robot can navigate and locate. In ROS, the following three functions can be used to achieve autonomous navigation.

- gmapping: Build a map based on laser data.
move_base: combines global planning and local planning to complete its navigation tasks. Global planning is used to establish a path to the final target on the map. Local planning is used to establish a path to a local target and avoid obstacles.

amcl (Adaptive Monte Carlo localization): An upgraded version of the Monte Carlo positioning method that updates particles using the adaptive KLD method.

When we wanted to do a map construction and path planning, we usually use the SLAM function package and the keyboard or other control device to artificially control the wheeled robot movement to construct a two-dimensional grid in a known environment. Then the function package of path planning is started to do some simple path planning based on the constructed map. This work is very cumbersome, and it also has some obvious shortcomings. For example, in an unknown environment, the process of human-controlled robots will undergo a purposeless walking and mapping process is very long. This long and boring process even drives people frustrating. Furthermore, it is almost impossible to complete the work of building a map in this way for environments with severe radio interference.

In fact, after studying the path planning function package in ROS, we find that a known map is not necessary for its core component move_base node. In other words, we don't have to wait until the artificial control robot completes the map creation process. It is straightforward to combine the two at the same time. In this way, we only need to arbitrarily specify a target point in the Rviz, the upper monitoring interface. The robot independently plans a path to the target point, constructs the surrounding map in real time and dynamically avoids the obstacles during the process of going to the target point.

Naturally, we can use the mouse to point the robot to any point in the house, and it will dynamically avoid all obstacles along the way, which seems to be much better than the original method. But it still has certain flaws, for example, it still needs human intervention and cannot be completely autonomous.

Based on this, we designed and wrote a Python script that has the following features:

1. Initialize a set of target positions in the map.
2. Randomly select the next location.
3. Send the appropriate MoveBaseGoal target to the move_base operation server.
4. Record the success or failure of the navigation to the target, and the elapsed time.
5. Statistics of the total travel time and distance.

The core of the entire script program is to preset a set of target points, randomly select these target points through the program and send them to the server of move_base, and the move_base server processes and issues the corresponding speed control message.

Through the above design, the wheeled robot is completely autonomous in the unknown environment for map construction, and can dynamically avoid obstacles.

It should be pointed out that the path planning including global planning and local planning can be achieved by using the standard functional packages in ROS.

Simulations and Experiments

To verify the autonomous mapping and path plan strategy, simulations and experiments are carried out, respectively. Conclusions
Simulations
The simulation platform is established based on the Gazebo simulation environment and the Rviz monitoring interface. The function structure of the simulation platform can be seen in Figure 3. Gazebo is an open source excellent physical simulation platform that supports a wide range of sensors and controllers. Here we built the gazebo model of the wheeled robot and added a laser radar sensor and an Arbotix controller to it. After the assembly is completed, the gazebo model of the wheeled robot will walk and collect the surrounding environmental obstacle information in the simulation environment like a real car. And the collected environmental information will be released to the upper displayer for further processing.

The information scanned by the laser radar returned by gazebo is processed by the slam function package to construct a two-dimensional raster map through the corresponding algorithm, and the map planning node receives the target information to make a path plan.

The biggest function of Rviz is to display the various sensor data of the robot, including the robot's respective postures, on the PC in 3D visualization. Rviz itself is an OGRE-based 3D display environment that integrates a large number of robot motion control and sensor data display plug-ins. Robot developers can use these plug-ins to display data such as Kinect and Lidar directly on the Rviz interface, while enabling rapid data fusion. Through the complete establishment of the robot model by Rviz, programmers can perform off-line simulation programming for the robot without using the robot itself, which increases the safety and improves the developing efficiency.

Based on the software system architecture discussed in the previous sections and related algorithms, the autonomous map construction, real-time path planning and obstacle avoidance functions of the robot under unknown environment are tested in the gazebo physics simulation scenario. Figure 4 shows the constructed scenario and Figure 5 presents the autonomous mapping results. It can be seen that they are agreed very well. Through the analysis of experimental phenomena and algorithm effects, the performance of the visual navigation system and the effectiveness of the automated construction algorithm of the wheeled robot in the simulation environment are verified.

Experiments
To test the mapping and path planning strategy, we also carried out experiment for real scenario. In the experiment, the corridor road on the 9th floor of the experimental building of BUU (see Figure 6) was selected as the scene for map construction and path planning. The actual construction effect is shown in Figure 7.
It can be seen that the degree of restoration of the map is high, and the autonomous navigation and path planning algorithms work well during the construction process. Combined with the above simulation results, the ROS-based wheeled robot autonomous navigation and path planning algorithm are very effective. Map construction and path planning in an indoor unknown environment have good performance. Through the perfect operation of simulation experiments and hardware experiments, it can be proved that the system has good portability and applicability, and can be perfectly applied to the expansion and development of autonomous navigation functions of other robots.

Conclusions

In this paper, an autonomous mapping and path planning, based on ROS, method is proposed. The method can be easily realized by executing a Python script program in ROS. The mapping progress is stimulated and supervised by Gazebo and Rviz, respectively. Simulations and experiments verified the feasible of the method.

Acknowledgement

This work is supported by the National Natural Science Foundation of China(51205025, 51775048, 61602041), the Science and Technology Program of Beijing Municipal Education Commission (KM201811417001), the Premium Funding Project (BPHR2017CZ08) for Academic Human Resources Development in BUU, the Beijing Natural Science Foundation-Beijing Municipal Education Commission Joint Fund (KZ201811417048) and the Project of 2018-2019 Basic Research Fund of BUU, the Beijing Advanced Innovation Center for Intelligent Robots and Systems Open Fund (2018IRS17).

References


