Route planning of teleoperation mobile robot
Based on the Virtual Reality Technology

Jiwu Wang
School of Mechanical and Electronic Engineering, Beijing jiaotong University
Beijing, Haidian District, China

Xuechun Yuan
School of Mechanical and Electronic Engineering, Beijing jiaotong University
Beijing, Haidian District, China
E-mail: jwwang@bjtu.edu.cn, 18121262@bjtu.edu.cn
www.bjtu.edu.cn

Abstract

Mobile teleoperation robot is one of the effective methods to help operators to work in complex environments. However, the time delay by distance is a key factor that restricts its application. To solve this problem, the motion trajectory of the robot is simulated using virtual reality technology and the obtained optimization data are applied to control teleoperation robot.

Keywords: teleoperation robot ; virtual reality ;route planning

1. Introduction

Teleoperation robots are one of the effective methods to help operators to work in complex environments (such as nuclear power plants, underwater, space, etc). However, one of the main problems in teleoperation is the time delay. [1]Specifically, in actual work, due to the distance between the operator′s side and the slave side of the robot is far away, the communication will inevitably occur time delay, resulting in the fact that the operator′s side and the slave side of the robot cannot achieve synchronous action.

In order to solve this problem, the motion trajectory of the robot is simulated by virtual reality technology and the obtained optimization data are applied to control teleoperation robot. In order to improve the efficiency of simulation, a 3D simulation environment of the real world is built in the unity 3D platform, and the physical model and kinematic model of the robot are established.

At the same time, the environmental conditions of various adjustable parameters such as light and material are simulated in the platform. The physics engine for the robot is added in the simulation platform and the physical properties of the robot are configured.

Since there is no delay between the operator and the robot model, the robot model can immediately respond to the operator′s input, compensate for the influence of the delay, and assist the operator to complete the operation task smoothly and reliably.

2. The Construction of the Virtual Environment

The vraisemblance of virtual environment construction is an important guarantee to realize this project. Only by ensuring that the virtual world and the real world are as consistent as possible, can we ensure the accuracy of our entire operation. The construction of the virtual environment is divided into the construction of the robot

model and the construction of the virtual simulation environment.

2.1. Construction of the robot model

2.1.1. The construction of motion model

Wheeled mobile robot is an important branch of autonomous mobile robot, which has the advantages of simple structure and easy control.   It is often used as a platform to study robot positioning, navigation, path planning and other technologies.

The mobile robot with two wheels differential drive is composed of vehicle body, power supply, control system and positioning system. By controlling the speed of the two driving wheels, the robot can track different trajectories: when the speed and direction of the two wheels are the same, the robot moves forward in a straight line; when the speed of the two wheels is different, the robot turns. [2]The rear wheels are two independent driving wheels, while the front wheels are a universal wheel that can roll in any direction. The front wheel only plays a supporting role to keep the body stable. Its kinematic model is shown in figure 1.
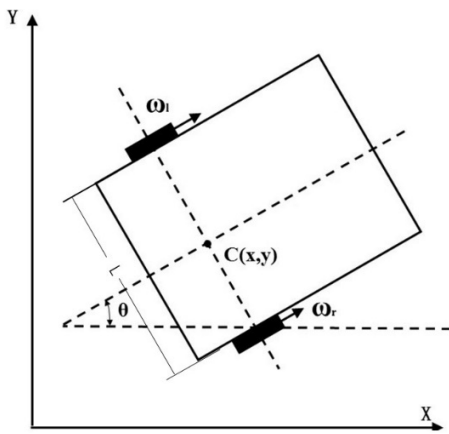

Fig. 1. Motion model of a mobile robot

Take the midpoint of the center line of the left and right wheels as the reference point. w_L and w_r are respectively the angular velocity of the left and right driving wheels. v and w are respectively the linear velocity and angular velocity of rotation of the robot. r and L are respectively the radius of the wheel and the distance between the two wheels. In the derivation process, $w_r > w_L$ is assumed, and its kinematic equation is shown as follows:

$$\begin{cases} x = v\cos\theta \\ y = v\sin\theta \\ \theta = w \\ v = r(w_L + w_r)/2 \\ w = r(w_L - w_r)/L \end{cases} \quad (1)$$

2.1.2. The construction of physics model

The robot is modeled by SolidWorks 2010. First, models of the parts of the robot such as head, arm and joint are built, then these parts are assembled. The assembly drawing is saved into the format of .stl and imported into 3d max, as shown in figure 2. After the model is output as .FBX format, it is imported into unity 3D for simulation, as shown in figure 3. After importing, set the physical properties for the robot in unity 3D, add the rigidbody component to the robot in unity 3D, and add the box collider component to each part.
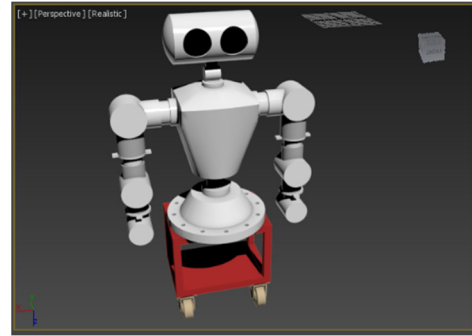

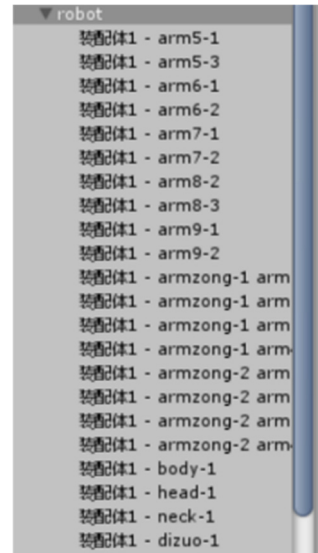Fig. 2. The robot model in 3d max imported from Solidworks

Fig. 3. The components of the robot in unity 3D imported from 3d max

2.2. Construction of the virtual simulation environment

Virtual environment modeling is realized by synthesizing two-dimensional environment images into three-dimensional models.

Firstly, multiple photos of the surrounding environment are collected from the vision system of the end-robot, and the photos are transmitted to the ground control end. Then a three-dimensional model is generated through Agisoft Photoscan software. Finally, the generated model is imported into unity 3D, and the light source is set for the whole scene, material, and texture are set for the objects in the scene. The Sunken Square of Beijing Jiaotong University is modeled as an example, and the model after importing unity 3D was shown in the figure 4.
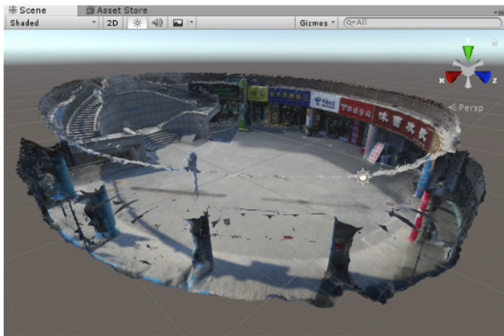

Fig. 4. The model in unity 3D imported from 3d max

3. Path Planning of Robots in Virtual Platform

Path planning is mainly implemented in unity 3D platform. First, several obstacles are set up in the scene. The four orange cuboids in the figure simulate obstacles. What we want to achieve is that by clicking on any position on the ground with the mouse, the robot can avoid obstacles and successfully reach the mouse click point. In this process, the walking track of the robot is displayed, and the three-dimensional coordinates of the walking track point of the robot are output to a. TXT file.

To realize the obstacle avoidance function of the robot, the pathfinding plug-in Navigation that comes with unity is used to realize automatic path finding through a series of calculations.

First, we baked the walking area of the robot, as shown in the blue area in figure 6, and added the Nav Mesh Agent component for the robot to set the speed and acceleration of the robot and other parameters.

Fig. 5. The walkable area of the robot

Write C# script and attach it to the robot to realize the function of mouse clicking on the ground robot to avoid obstacles and reach the target point, and add Line Render component to automatically draw the robot's walking track when the robot moves through the script. Use the StreamWriter class in C# to output real-time 3d coordinates of the robot as it moves to a.txt file, as shown in figure 6.

After the scene runs, the effect is shown in the figure 7 below. We can also zoom in and out the mobile scene through the middle mouse button.
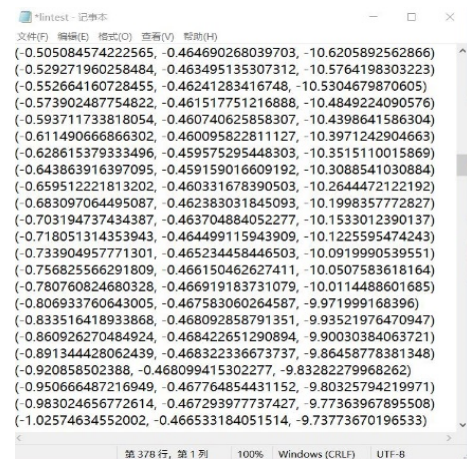

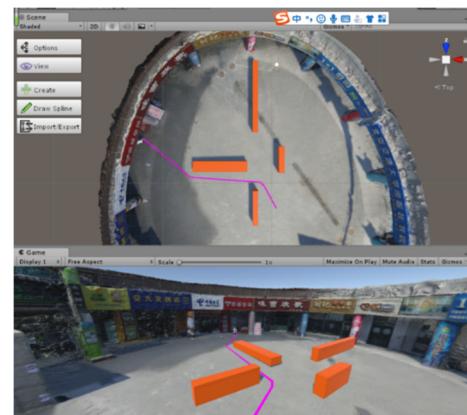Fig. 6. The three-dimensional coordinates of the robot in motion

Fig. 7. the operation result after improvement the scene runs

Draw the exported data points on the xz plane with Matlab, as shown in the figure 8.
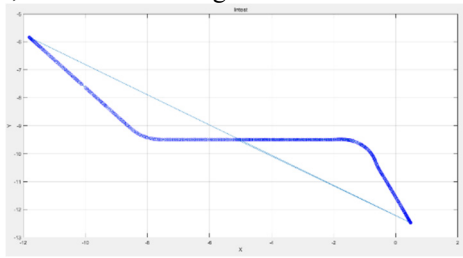

Fig. 8.   The trajectory of the robot in the xc plane

4. Conclusion

In this paper, the robot motion simulation platform is built in the virtual reality environment. With this platform, the robot's motion trajectory was simulated for any specified target points. At the same time, the obstacle avoidance strategy was also tested in that platform. The satisfactory results were obtained. After the simulation, the true moving trajectory in the true environments for the robot is gotten. Then the true robot can be controlled to move based on the above data. In the coming future, the robot will be verified with the simulation data.

References

1. Bejczy A K,Kim W S,Venema S C，The phantom robot：predictive display for teleoperation with time delay (IEEE International Conference on Robotics and Automation，1990)
2. Farzad P, Mattias P K, Adaptive control of dynamic mobile robots with nonholonomic constrains. (Computers and Electrical Engineering , 2002(28):241-253)