Research on Intelligent Car PID Autonomous Navigation System Based on ROS and Lidar

Hanqing Hu, Zhengxun Li

School of Economics and Management, Beijing Information Science & Technology University, Beijing, China

Abstract: Aiming at the error and inefficiency of the existing smart car navigation system, a new smart car PID autonomous navigation system based on ROS and lidar is proposed. The basic hardware management of the smart car is based on the ROS robot control system, and the laser radar is used to scan the environment and build a map, and realize the visualization through Rviz. Integrate PID algorithm into ROS, and improve the accuracy of deviation correction through filtering. Realize obstacle avoidance and autonomous navigation of smart cars. After field testing, the smart car has a good effect on map construction and autonomous navigation, which is better than traditional amcl navigation in terms of autonomy and efficiency.

Keywords: ros; pid; smart car; autonomous navigation

1. Introduction

Smart car is the product of the combination of electronic computer with the modern automobile industry. It integrates many cutting-edge disciplines such as sensors, computers, artificial intelligence, communication, navigation and positioning. With the continuous breakthrough of key technologies of smart cars and the popularization of the Internet of Things, unmanned driving and automatic navigation are an important development direction of smart cars in the future.

When a smart car is working, it often faces many complex environments with unknown states. In complex environments, how to identify obstacles, build maps and navigate autonomously has become a key issue. To solve the above problems, the smart car first needs to be equipped with suitable sensors and software systems. There are two main ways for smart cars to recognize the surrounding environment, one is computer vision recognition, and the other is lidar detection. Computer vision recognition often has more stringent requirements for light, so in the face of complex operating environments, lidar is usually used for the construction of intelligent vehicle environmental maps and autonomous navigation. As one of the popular open source robot operating systems, ROS can realize the realization of common functional modules such as hardware interaction and package management. The ROS-based lidar sensor can determine the surrounding environment information through laser triangulation, so as to construct a map and provide a good data foundation for the autonomous navigation of smart cars [1]. Zhao Jianwei et al [2]. used lidar to build a map and applied it to the visual navigation of small robots; Zhang Xu et al. [3]. combined lidar and IMU to calculate the pose and improved the accuracy and accuracy of the map through filtering; Li Linchen et al. [4]. proposed a PID-based robot path tracking algorithm, which improved the system stability and dynamic adaptability of robot path tracking; Meng Shaonan et al. [5]. used ROS to realize virtual control of the manipulator, and realized visualization through Rviz; Li Yeqian et al. [1]. established a robot automatic navigation system based on ROS and lidar, which reduced the lateral deviation in the navigation process. In this study, map information is collected on the basis of software and hardware based on ROS and lidar, and the mainstream control algorithm PID is applied to the autonomous navigation of smart cars. The precision and accuracy are improved by filtering, so as to construct a new autonomous navigation of smart cars. The system enables smart cars to navigate autonomously without human intervention.

In this study, map information is collected on the basis of software and hardware based on ROS and lidar, and the mainstream control algorithm PID is applied to the autonomous navigation of smart cars. The precision and accuracy are improved by filtering, so as to construct a new autonomous navigation of smart cars. The system enables smart cars to navigate autonomously without human intervention.

2. System Hardware Architecture

Smart car hardware design needs to meet the overall functional requirements. Therefore, this study mainly uses IMU, industrial computer, encoder, driver board, lidar (rplidar), lithium battery, etc.to configure the hardware system of the smart car. And load the ROS system on the industrial computer to perform hardware abstraction, underlying device control, communication, and function package management for the smart car. The finished smart car assembly is shown in Figure 1.

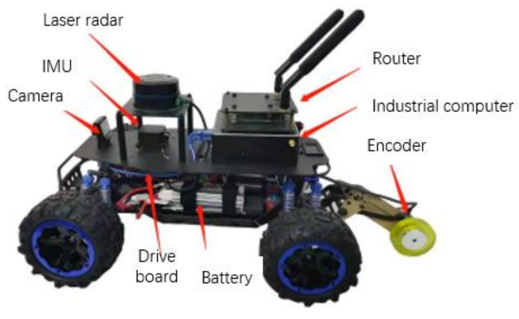

Figure 1. Assembled finished product of smart car

2.1. Lidar Sensor Principle

Lidar is responsible for collecting map environment information in the whole system, including three parts: laser transmitter, visual acquisition system and rotating motor. The laser transmitter emits infrared laser signals on the surface of the surrounding environment, scattering and reflection of light occurs, and the reflected laser light converges at the acquisition system to form an image. When the lidar moves, the light spot formed by the reflection will also move together. The geometric triangle theorem can be used to The actual distance is obtained, so it is called the triangular ranging method. The triangulation ranging method is generally divided into direct type and oblique type according to the incident laser angle. The oblique type has higher resolution than the direct type. The laser radar selected for assembly in Figure 2 also adopts the oblique-shooting ranging method [2].

Figure 2. Lidar

The principle of triangular ranging of oblique laser radar is shown in the Figure 3 [6]:

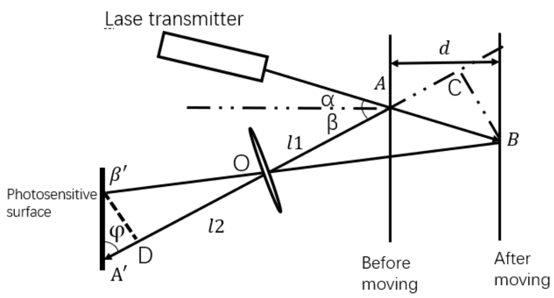

Figure 3. Principle of oblique laser radar

The angle between the laser beam and the normal of the moving object surface is α, the angle between the scattered laser beam and the discovered laser beam is β, the angle between the scattered laser beam AA' and the photosensitive surface is φ, the distance between the incident point A and the lens is $l1$, and the distance between the imaging point A' and the lens is $l2$.The vertical intersection of the extension line of AA' over point B is AA' at point C, and the vertical intersection of AA' over point B is AA' prime at point D.When the moving distance of the object is d, the moving distance of the light spot on the photosensitive surface is y = $A'B'$.

There are:

$$\frac{B'D}{BC}=\frac{OD}{0C}=\frac{OA'-DA'}{OA+AC} \tag{1}$$

Parameters:

$$BD'=y\,sin(\varphi) \tag{2}$$

$$BC=ABsin(\alpha+\beta) \tag{3}$$

$$OA'=l2, OA=l1 \tag{4}$$

$$DA'=ycos(\varphi) \tag{5}$$

$$AC=ABcos(\alpha+\beta) \tag{6}$$

$$AB=\frac{d}{cos(\alpha)} \tag{7}$$

Insert equality:

$$\frac{y\sin\varphi}{\frac{d}{\cos a}\sin(\alpha+\beta)}=\frac{l2-y\cos\varphi}{l1+\frac{d}{\cos a}\cos(\alpha+\beta)} \tag{8}$$

$$d=\frac{yl1\,sin\,\varphi\,cos\,a}{l2sin(\alpha+\beta)-ysin(\alpha+\beta+\varphi)} \tag{9}$$

The lens focal length is f, according to Gauss imaging theorem:

$$\frac{1}{l1}+\frac{1}{l2}=\frac{1}{f} \tag{10}$$

Surface moving distance d:

$$d=\frac{y(l1-f)\,sin\,\varphi\,cos\,a}{fsin(\alpha+\beta)-y(1-f/l1)sin(\alpha+\beta+\varphi)} \tag{11}$$

2.2. ROS Operating System

The first step of software installation is to install the ubuntu system on the industrial computer, and then update the system mirror source based on the Ubuntu system to install the ROS operating system.

ROS is a robot-oriented Linux-like open source operating system, which supports C++ and python, and can provide common sensors and motor drivers. ROS also provides a distributed framework, which takes nodes as a unit, loosely couples each node when running, realizes the comprehensive management of basic hardware, and processes multi-threaded tasks in parallel. It can also provide common software packages, algorithm packages and compilers [7].

3. Modeling and Visualization of Laser Radar Map Data

3.1. Environmental Information Acquisition

The local map of the intelligent vehicle is constructed by collecting the original environmental data by laser radar during driving. The odometer information and the position and posture of the intelligent vehicle are obtained by IMU and encoder. The position information of the intelligent vehicle is obtained by data association between the three, and the local map is updated continuously to obtain the global map [3].

3.2. Rviz Visualization

Rviz is one of the core functions of ROS system, which can complete the establishment of sensor data coordinates and coordinate the conversion relationship between each component coordinates [5].

The gmaping package under ROS framework provides the Slam (simultaneous localization and mapping) of the lidar. As shown in the Figure 4, according to the input and attitude data of the lidar, a grid-based 2D map is established, which is the input of Rviz.

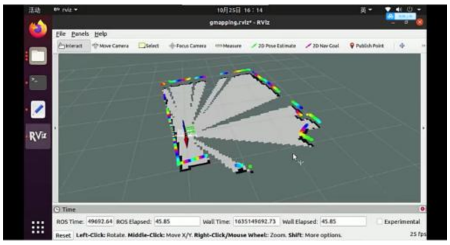

Figure 4. Rviz map interface display

However, in the actual operation, due to the limitation of laser radar ranging and the existence of obstacles, it is necessary to continuously control the autonomous movement of the smart car and traverse the map. This study will improve the autonomous navigation by PID algorithm.

4. Autonomous Navigation Based on PID

4.1. PID Algorithm

PID is the most widely used closed loop controller in engineering practice, which has strong stability. Its characteristic is that the output of the controlled object will affect the output of the controller. PID is often used in the control of sensors, drone and simulation models. Therefore, the analysis and design of PID control system and its relationship with other systems have been a research hotspot in the control field [8].

In this study, PID is applied to the position and angular velocity control of intelligent vehicle to correct the deviation generated in the process of autonomous navigation. The structure of PID is shown in Figure 5 [4].

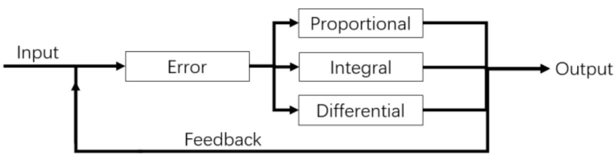

Figure 5. Structure diagram of PID controller

$$u(t) = kp(e(t) + \frac{1}{T_i}\int e(t)dt + T_d\frac{de(t)}{dt}) \qquad (12)$$

The discretization form is:

$$u(t) = kp * e(t) + \frac{kp * T}{T_i}\sum_{n=0}^{t} e_n + \frac{kp * T_d}{T}\big(e(t) - e(t-1)\big) \qquad (13)$$

where $u(t)$ is the controller output, $e(t)$ is the output deviation, kp is the proportional constant, T is the sampling time, T_i is the integral time constant, and T_d is the differential time constant.

PID includes three parts: proportion, integral and differential. In the iterative process of PID program of smart car, this study uses the obstacle information of the surrounding environment detected by laser radar to take the tangent of the midpoint of the obstacle on both sides of the road as the target of the next position and direction of the car. Then based on the current linear coordinates and angular velocity, there will be a distance deviation and direction deviation, which is $e(t)$ in PID.

First use P proportional control:

$$u(t) = kp * e(t) \qquad (14)$$

Where kp is the proportional coefficient, $u(t)$ is the output of the intelligent vehicle controller based on error in each iterative modification process. However, in fact, there will be resistance loss caused by motor friction and steady-state error. Therefore, the concept of integral is introduced.

In order to eliminate steady-state error, I integral control is introduced to eliminate steady-state error:

$$u(t) = kp * e(t) + \frac{kp * T}{T_i}\sum_{n=0}^{t} e_n \qquad (15)$$

The integral control is that the cumulative historical error is multiplied by the integral constant. As long as the deviation exists, the integral will continue to accumulate. When the integral term no longer accumulates and becomes a constant, the steady-state error can be offset. The introduction of integral can eliminate steady-state error, but will increase overshoot.

In order to eliminate overshoot, differential control is introduced:

$$u(t) = kp * e(t) + \frac{kp * T}{T_i}\sum_{n=0}^{t} e_n + \frac{kp * T_d}{T}\big(e(t) - e(t-1)\big) \qquad (16)$$

Differential can reflect the variation trend of deviation and play a correction role before overshoot [9].

4.2. PID Autonomous Navigation

Firstly, it is necessary to install and compile the package of each hardware function module according to the ROS operating system. So that ROS system can manage all hardware modules as a whole.

The main program introduces the PID structure, defines the sampling time, and connects the environment and pose data collected by lidar and IMU to the PID program. At the same time, the navigation starting point is marked, and the linear velocity and angular velocity of the smart car are released. The original pose of the smart car is determined according to the data collected by IMU, encoder and lidar. After the main program runs, it will be visualized on the Rviz interface, as shown in Figure 6.

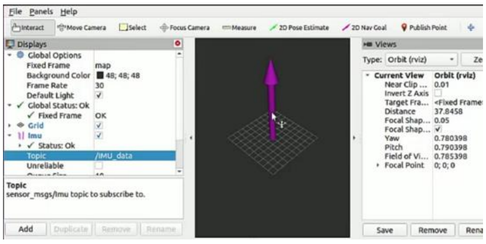

Figure 6. Rviz display interface for IMU pose of smart car

According to the ROS operating system, the vehicle speed object is introduced. According to the laser radar scanning results, the position of the obstacle is displayed, and the target pose in the next stage is determined according to the position of the obstacle. As shown in Figure 7, it is the process of correcting the direction of the smart car.

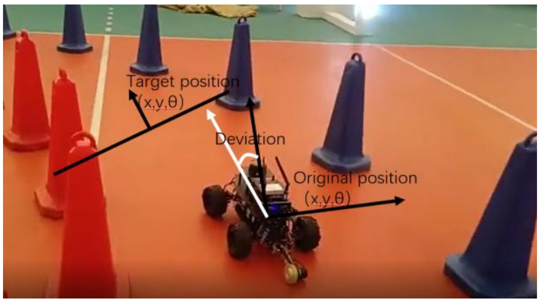

Figure 7. Pose control diagram

The ROS system ROS-info is used to open the control node of the car, publish the x, y angle parameters and distance parameters of the initial smart car, and publish the linear velocity and angular velocity. In order to adapt to the steering actuator performance, the sampling time interval is set to 0.5s–1s to obtain the data of radar, IMU and encoder in real time. Figure 8 is the update process of the location information of the smart car displayed on the computer. The deviation is calculated according to the target position information, and the position deviation and angle deviation are calculated respectively.

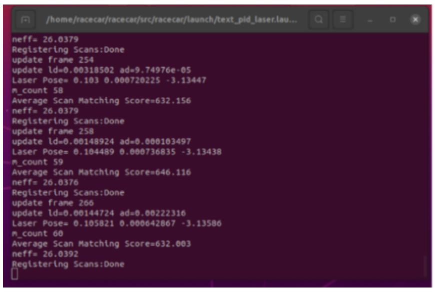

Figure 8. Iteration process of automatic navigator node

At the same time, the radar data is filtered to enhance the safety and stability of automatic navigation, and the angle of error less than the filtering threshold is controlled by PID to output the final control direction.

5. Test Results and Comparative Analysis

In order to verify the authenticity and feasibility of the autonomous navigation method in this study, the PID intelligent vehicle navigation system is tested and verified, and the obstacle runway is built on the indoor site of the school. Figure 9 shows the construction process of the obstacle course. The barrier is 30 * 30 * 60 plastic cone barrel, the cone barrel spacing is about 1m, the runway spacing is 1.5m, the total length of the runway is about 25m, including linear area, right angle bend, ring bend and other different runway environment. The selection of the length-width ratio of the runway environment is mainly to adapt to the size of the smart car. In the actual scene, the expansion coefficient of the smart car can be modified according to the different surrounding environment to adapt to the environment, so as to better construct the map and autonomous navigation.

Figure 9. Construction of indoor obstacle runway

Firstly, install the ubuntu system for the intelligent vehicle industrial computer, and update the domestic mirror source to improve the interaction speed, install the ROS control system. Then the laser radar, encoder, IMU debugging, installation function package, compile. The PID autonomous navigation mentioned above is written in C++ and compiled into the system. The software environment is built by gmapping, base-controller and amcl function package under ROS system, and the map construction and automatic navigation of intelligent vehicles are observed by Rviz visual operation interface.

After the navigation starts, the terminal is opened to run the PID autonomous navigation program, and then gmapping is opened, which will make the radar and rivz modules run. Finally, the smart car motor is opened, and the smart car starts to observe.

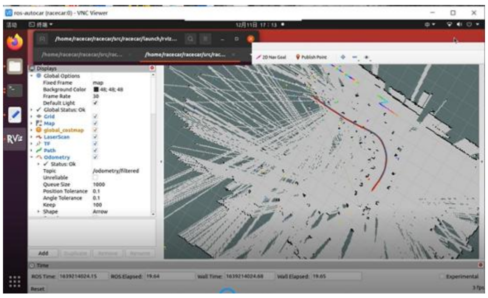

Figure 10. Rviz automatic navigation map building

As can be seen in Figure 10, the system can draw a grid map. The red line part is the automatic navigation trajectory of the car, and the black spots around the red line are obstacles identified by lidar. The specific map parameter information is displayed on the left side of the Rivz interface. If all the above problems are not found, it indicates that the Slam function meets the requirements. By comparing the constructed map with the physical map, it is found that the two structures are basically the same. Through the test, the system can be in the obstacle track and no collision state of unmanned autonomous navigation, basically reached the experimental expectation.

In order to evaluate the performance of PID autonomous navigation system based on laser radar, this study adopts the open source amcl (adaptive Monte Carlo localization method) of ROS system for comparative analysis [10]. The results show that compared with amcl, PID autonomous navigation system does not need to build the map in advance, and can independently explore the map, which is more suitable for unmanned environment. On the other hand, PID autonomous navigation system is superior to the traditional amcl method in mapping accuracy and navigation efficiency.

6. Conclusions and Outlook

In this study, the map environment data were collected by the intelligent vehicle model equipped with lidar built by self-assembly. The PID algorithm was realized in the ROS system through python and c++, and the intelligent vehicle identification, mapping and autonomous navigation of obstacles were completed. In order to verify the accuracy of the laser radar mapping and the effectiveness of the pid method navigation, the obstacle runway is built in the real environment, and the intelligent vehicle equipped with the program is simulated and tested. The test results show that the smart car can traverse the surrounding environment by itself under the condition of unmanned control, use laser radar to build a clear and complete map, and realize autonomous obstacle avoidance navigation through PID algorithm, which is superior to the traditional amcl method in efficiency and autonomy. It provides a new idea for solving the autonomous navigation of intelligent vehicles, which can be applied to the autonomous navigation of intelligent vehicles.

In the technical realization of the current PID algorithm, the steering of the car will be frequently adjusted, which will cause the loss of steering gear. In further research, it can be considered to turn the autonomous navigation into a continuous process to increase the predictability of the target point.

References

[1] Yeqian Li, Chunmiao Chen.Design of mobile robot automatic navigation system based on ROS and laser radar.Modern Electronics Technique, 2020, 43(10):176-178, 183.

[2] Jianwei Zhao, Hongjing Zhang, Hongyan Wang, et al.Study on Building Indoor Environment Map with Laser Radar.Machinery Design & Manufacture, 2017(5):135-137.

[3] Xu Zhang, Zhi Wang, Can Cui, et al.Design and implementation of 2Dlidar positioning and mapping system.OPTICAL TECHNIQUE, 2019, 45(5):596-600.

[4] Linchen Li, Xiaoping Jiang.Path Tracking of Mobile Robot Based on PID Control.LASER JOURNAL, 2016, 37(2):110-112.

[5] Shaonan Meng, Yan, Heng Shi.The Motion Planning of a Six DOF Manipulator Based on ROS Platform.JOURNAL OF SHANGHAI JIAO TONG UNIVERSITY, 2016, 50(S1):94-97.

[6] Yi Cai, Changjie Liu.Parameters Optimization of Laser Range Sensor Based on Triangulation.CHINESE JOURNAL OF SENSORS AND ACTUATORS, 2017, 30(10):1472-1477.

[7] Peng Zhang, Fang Gao, Feng Shuang.Design and Implementation of Omni-directional Mobile Robot Control System Based on ROS.Modular Machine Tool & Automatic Manufacturing Technique, 2018(7): 89-92, 96.

[8] Xujun Lyu, Zongli Lin.PID Control of Planar Nonlinear Uncertain Systems in the Presence of Actuator Saturation.JOURNAL OF AUTOMATICA SINICA, 2022, 9(1):90-98.

[9] Xiao Yang, Guozhu Wang.The Improved Particle Swarm Optimization Algorithm Based on PID Control. Control Engineering of China, 2019, 26(8): 1497-1502.

[10]Kaihong Zhang, Shengnian Chen, Guangyu Fan.INTELLIGENT ROBOT POSITIONING ALGORITHM BASED ON PARTICLE FILTER.Computer Applications and Software, 2020, 37(2):134-140, 199.