Research on dynamic obstacle avoidance and complex path planning strategies based on ROS robots

Yi-Wei Chen

Bachelor's Degree Program in Intelligent Robotics, National Yunlin University of Science & Technology.123 University Road, Section 3, Douliou, Yunlin 64002, Taiwan, R.O.C

Jr-Hung Guo*

Department of Electrical Engineering, National Yunlin University of Science & Technology,123 University Road, Section 3, Douliou, Yunlin 64002, Taiwan, R.O.C

Kuo-Hsien Hsia

College of Future National Yunlin University of Science and Technology, 123 University Road, Section 3, Douliou, Yunlin 64002, Taiwan, R.O.C

Email: b10811150@yuntech.edu.tw, jrhung@yuntech.edu.tw, khhsia@yuntech.edu.tw

Abstract

Robot Operating System (ROS) is a software system framework used by many robot systems. Although ROS provides a good development environment and related frameworks, ROS is not suitable for public places such as restaurants because of the coming and going of people. Dynamic obstacle avoidance is often handled by stopping the robot, or when there are frequent environmental map changes or when sensors such as optical radar fail, the stop action is also used. However, this often causes path obstructions or delays in completing tasks. Therefore, this study attempts to use images, auxiliary sensors, and various path avoidance strategies to solve the problem of the robot stopping and waiting for the obstacles to disappear. The problem of rapid changes in map paths.

Keywords:*Robot Operating System (ROS), Dynamic obstacle, Dynamic obstacle avoidance.*

1. Introduction

Robot Operating System (ROS) [1] is a very commonly used development platform for developing and researching robot systems in recent years. Because this software uses open-source code and uses resources provided by many robot developers. Let this ROS system grow rapidly. Whether mobile robots can actually operate in actual environments is also a very important research topic for mobile robots. Because the mobile robot is in the actual environment, cannot avoidance dynamic or static obstacles. Or without the ability to re-plan the correct path, the mobile robot well cannot actually operate in a real environment. Of course, this also limits the application of mobile robots. Therefore, this study uses ROS as a research platform to focus on how mobile robots dodge dynamic obstacles in the environment. And re-chart the path to try to find a solution. This can reduce the chance of the mobile robot stopping when encountering obstacles and improve the efficiency of the mobile robot. It also enables mobile robots to be applied in real environments. In previous research, dynamic obstacle avoidance is an important research topic for many mobile robots. For example, CHEN, Chin S., et al. [2] use the ROS architecture to evaluate the path cost of autonomous mobile robots (AMR) for obstacles in the environment. Then choose the better path and let the mobile robot run. CHOI, Jaewan., et al. [3] use reinforcement learning to improve the reliability of dynamic window approach (DWA) [4] and timed elastic band (TEB) [5]. Make mobile robot obstacle avoidance more efficient.

It can be found from previous related research that many mobile robots mostly use small controllers such as Raspberry Pi [6] due to cost and volume. And it uses LiDAR Light Detection and Ranging (LiDAR) as the main sensor. Although such an architecture is lower in cost, it results in insufficient computing power. And because when detecting obstacles, lidar can only detect obstacles and measure distances on the plane that light or laser can scan. It often happens that when an obstacle is not in the LiDAR light scanning area, the obstacle is misjudged or cannot be detected. Therefore, the development and application of mobile robots are hindered. Therefore, this research will use the architecture of x86 computers. In addition to using LiDAR, it will also use imaging technology. Through this architecture, the mobile robot's computing power and obstacle detection can be faster and more accurate. Allowing mobile robots to truly overcome dynamic obstacles in the environment and enable practical applications.

2. System Architecture

The architecture diagram of the mobile robot is shown in Fig. 1. The robot is equipped with cameras, LiDAR and STM32 control boards, etc. It also has many various interfaces for expansion of different hardware. It can be widely used in actual or simulated various task development and applications. This robot architecture uses ROS as the main core and adds modules such as targets following systems. The system can be adjusted according to different applications to provide a diverse robot development environment. At the same time, the development results can also be quickly transferred to different mobile robot platforms.

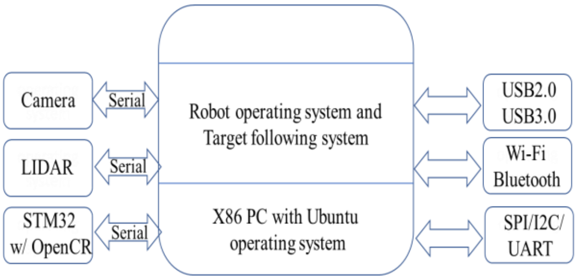


Fig.1 ROS mobile robot architecture block diagram.

In the ROS system, the sensors commonly used for object distance measurement are mainly LiDAR and imaging lenses. In order to consider the calculation amount and get the object distance, many robot platforms are based on LiDAR. However, general LiDAR is only 2D, that is, it can only measure one plane, and this also causes LiDAR to be unable to detect objects outside the 2D plane it scans. Therefore, in order to avoid obstacles that cannot be detected using LiDAR, this study uses images to detect obstacles and calculate distances.

There are two algorithms commonly used in ROS for object detection and distance calculation. One is KCF [7] (Kernelized Correlation Filter), which is a method to find the image features being tracked and then use this feature to track the object. Therefore, the operation speed of this method can be very fast. However, when obtaining object features and image tracking, this algorithm will lose the image data, making it impossible to perform accurate tracking. The other is DSST [8] (Discriminative Scale Space Tracker). DSST mainly improves the shortcomings of KCF that only uses gray scale to obtain object features and does not calculate the object scale. It makes the judgment of objects more accurate and enables distance estimation. The most important thing is that it will not have a big impact on the overall calculation amount. Although the above two algorithms are sufficient for tracking and following general mobile robots. However, because this research hopes to use images to process more complex, even more obstacles and more targets, the TLD [9] (Tracking-Learning-Detection) algorithm is also added. This algorithm can accurately identify the target, even when the object reappears after being obscured, it can still correctly identify the target. And if the target object rotates, TLD can also correctly identify it. Of course, the number of calculations required for this algorithm will be relatively large.

2.1 Research methods

In this study, in order to make ROS have a higher object recognition rate and calculate a more accurate object distance, we integrated KCF and DSST. As the main object recognition and distance calculation core, the integrated operation structure diagram is shown in Fig. 2. In this processing block, KCF is mainly used to quickly identify targets and establish preliminary samples for subsequent identification. The main purpose is to identify possible obstacles or targets as quickly as possible, and then use DSST to further process the identification model and identification samples. Because KCF has a faster processing speed, it can first determine which objects may be closer to the mobile robot. Then, when necessary, it will be handed over to LiDAR and DSST for confirmation. Once a possible obstacle or path impact on the mobile robot is confirmed, it will continue to be tracked with KCF.

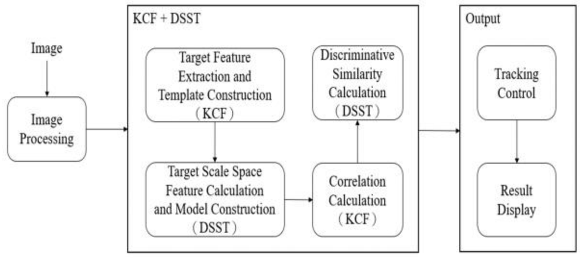


Fig.2 Block diagram of KCF+DSST target recognition system.

In the TLD algorithm part, because the preset requirements are relatively high for the computing power of the system, the calculation will only be performed when KCF+DSST determines that the object is closer to the mobile robot. Its block diagram is shown in Fig. 3. Moreover, the frequency of calculation can be set in the system, such as once per second, 10 times per second, etc. In addition, the maximum number of tracking can also be set. This prevents the CPU from consuming too much computing power in object tracking. Next, we will conduct experimental tests on the above methods.

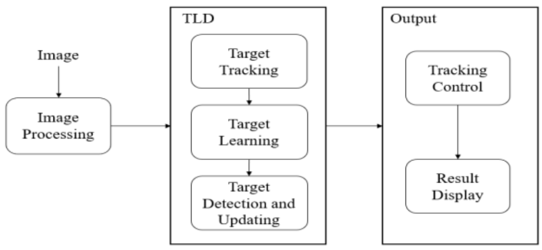
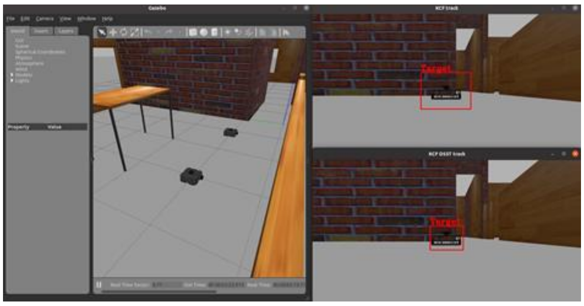


Fig.3 TLD target identification system block diagram.

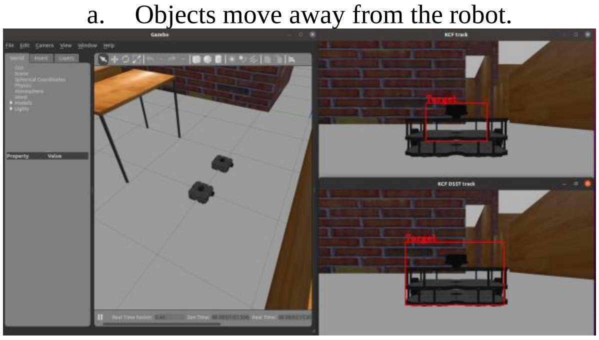
3. Experimental Results

We use an x86 computer as the main control core, and the specifications used are: CPU, Intel® Processor N-series

N100, 8GB DDR4 DRAM, 512GB SSD. And directly install Ubuntu, and then install Gazebo and ROS in the system. First, we first confirm the object detection capabilities of the KCF+DSST and TLD algorithms. Fig. 4 is our test results on ROS. We use the same obstacles to identify objects and calculate scale distances using the KCF+DSST and TLD algorithms. According to the experimental results, both algorithms can operate normally. The operation speed of KCF+DSST is about 20% faster than the TLD algorithm. The calculation results of the scale distance between the two are almost the same.



a. Objects move away from the robot.



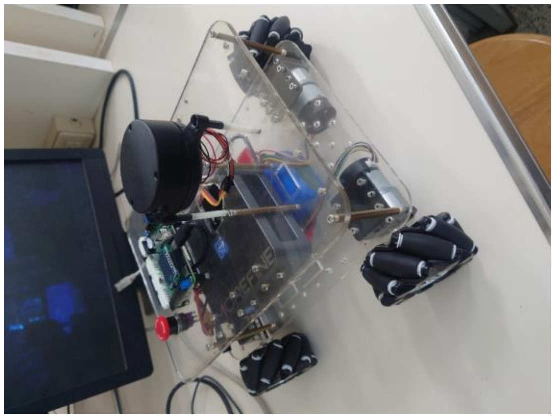
b. Object approaches the robot

Fig.4 Object tracking and recognition results

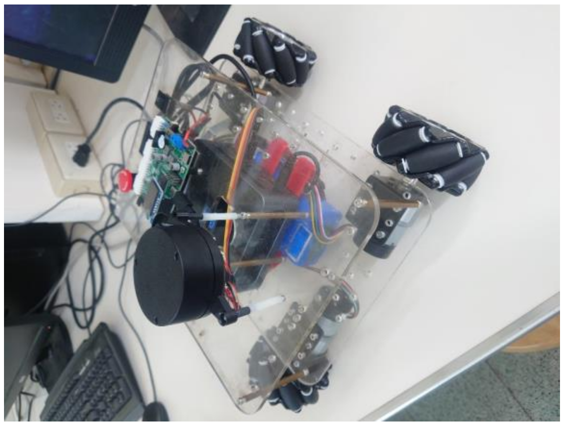
But when we dynamically generate multiple objects with different sizes, distances, and positions. Although KCF+DSST can still find objects quickly. However, when objects are interlaced or have large displacement or rotation, KCF+DSST will lose the tracked object. As a result, the risk of collision of mobile robots becomes higher. When using TLD, this algorithm can quickly and correctly handle the interlacing of objects, as well as rapid displacement, rotation, etc., and can also correctly calculate the distance of objects. But when more than 150 objects appear, there will be a short pause based on the hardware we are using. Although it does not affect the operation of the system, there will be obvious pauses. KCF+DSST will only pause for a short time when there are about 300 objects.

4. Conclusions

This paper uses imaging technology to deal with the detection problem of multi-obstacle objects, and develops a mobile robot platform using an x86 CPU computer. The actual photo of this platform is shown in Fig. 5. We verified on ROS that KCF+DSST and the TLD method were used to verify the path re-planning that can be used to solve multi-dynamic obstacles. That is to say, obstacle avoidance can be performed when there are many dynamic obstacles in the environment. Of course, the current structure still cannot completely allow mobile robots to pause. However, based on the current experimental results, more than 90% of suspension situations can be avoided. The other 10% are almost all dynamic obstacles that completely block the path of the mobile robot. We will also try to solve this part in the future so that mobile robots can overcome the obstacles of dynamic emergence and movement in the environment and serve people in real environments.



a. Mobile robot left view.



b. Mobile robot right view

Fig.5 Mobile robot platform

References

1. ROS Website, “https://www.ros.org/ “,2023/12/02”
2. CHEN, Chin S., et al. “Velocity estimation and cost map generation for dynamic obstacle avoidance of ROS based AMR.”, Machines, 2022, 10.7: 501.
3. CHOI, Jaewan; LEE, Geonhee; LEE, Chibum. “Reinforcement learning-based dynamic obstacle avoidance and integration of path planning. Intelligent Service Robotics.”, 2021, 14: 663-677.
4. OX, Dieter; BURGARD, Wolfram; THRUN, Sebastian. “The dynamic window approach to collision avoidance.”, IEEE Robotics & Automation Magazine, 1997, 4.1: 23-33.
5. RÖSMANN, Christoph; HOFFMANN, Frank; BERTRAM, Torsten. “Timed-elastic-bands for time-optimal point-to-point nonlinear model predictive control.”, In: 2015 european control conference (ECC). IEEE, 2015. p. 3352-3357.

6. Raspberry_Pi Wikipedia Website, "https://en.wikipedia.org/wiki/Raspberry_Pi ",2023/12/02
7. HENRIQUES, João F., et al. "High-speed tracking with kernelized correlation filters.", IEEE transactions on pattern analysis and machine intelligence, 2014, 37.3: 583-596.
8. DANELLJAN, Martin, et al. "Discriminative scale space tracking.", IEEE transactions on pattern analysis and machine intelligence, 2016, 39.8: 1561-1575.
9. KALAL, Zdenek; MIKOLAJCZYK, Krystian; MATAS, Jiri.,"Tracking-learning-detection.",IEEE transactions on pattern analysis and machine intelligence, 2011, 34.7: 1409-1422.

Authors Introduction

Mr. Yi-Wei Chen

Yi-Wei Chen, a graduate of National Yunlin University of Science and Technology, Taiwan, holds an engineering degree with a focus on robotics-based technologies. His academic pursuits are deeply rooted in the exploration and advancement of robotic applications.

Dr. Jr. Hung Guo

He is an Assistant Professor at the Department of Electrical Engineering, National Yunlin University of Science & Technology, Douliou, Taiwan. He received his Ph.D. Degree from National Yunlin University of Science & Technology, Taiwan in 2012. His research interests include sensor network, intelligent systems, intelligent robot. and technical and vocational education

Dr. Kuo-Hsien Hsia

He received the Ph.D. degree in electrical engineering from the National Sun Yat-Sen University, Taiwan, in 1994. He is currently an Associate Pofessor of National Yunlin University of Science and Technology in Taiwan. His research interests are in the area of mobile robotics, fuzzy control and image processing.