Development of agricultural robots based on ROS

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

Abstract

The agricultural industry has become a major issue in many countries due to drastic weather changes and a decrease in the number of people willing to engage in agriculture. Therefore, this study attempts to develop a platform for agricultural robots so that robots can assist people in agricultural work. This study uses Robot Operating System (ROS) as the software foundation. Through this convenient software foundation, different agricultural robots can be developed quickly. This study uses this architecture to develop a lawn mower and leaf sweeper, which can be easily converted to different agricultural applications in the future. In this study, the control system architecture we used is different from the commonly used single-board computer or microcontroller architecture. Instead, we used an ultra-small x86 mini-PC platform, mainly because the x86 platform used to be larger in size, more power consumption and other issues. However, the platform used in this study can be as small as 88mm*88mm*50mm, which is very close to the common Raspberry Pi or Jetson Nano, but the computing power, expansion capability and subsequent related development issues can be well developed. Finally, we are also trying to develop a web-based development and human-machine interface (HMI), hoping to make the subsequent development and use of different robots easier.

Keywords: Agricultural robots, Robot Operating System (ROS), x86 mini-PC, Web base HMI

1. Introduction

Because the agricultural industry is a relatively hard job, fewer and fewer people are willing to work in it. Among them, GILLER, Ken E., et al. [1] mentioned the problems and future development of global agricultural production. KASSIM, Mohamed Rawidean Mohd [2] mentioned the trends and advantages of IoT technology in future agricultural development. CHUNJIANG, Z. H. A. O., et al. [3] proposed an agricultural robot architecture, while OLIVEIRA, Luiz FP; MOREIRA, António P.; SILVA, Manuel F. [4] proposed the technological progress and future of agricultural robots. From the above related research, it is found that there are actually several key issues in the development of agricultural robots. (1) The structure of agricultural robots is complex. Since the structures of various agricultural uses vary greatly, it is impossible to use one structural platform to meet all agricultural needs. (2) Technical difficulties: agricultural environments vary greatly, so the entire control and related technologies are very complex. (3) High cost and difficult operation: agricultural robots are expensive because there is no universal platform. In addition, it cannot provide a simple development or operation interface, so farmers have many problems in using it. Due to the above problems, robots cannot be widely used in agriculture.

In addition, in terms of the robot's structure and control system, this study uses an existing remote-controlled agricultural robot platform. The main reason for doing so is that such a platform can already be used in agriculture, but it lacks automation and intelligence functions. . In addition, we use the x86 mini-PC architecture in the control system. Most of the current mobile robots use single-board computers such as Raspberry Pi or Jetson Nano as the main controller. However, the computing and expansion capabilities of these single-board computers are insufficient, and the size and cost advantages that these single-board computers previously had have also disappeared. The authors of this study have used mini-PCs to develop many platforms and related robot teaching systems in the past, so we hope to apply such experience and architecture to agriculture, hoping to provide some help for agricultural automation.

2. System architecture

2.1. Hardware and Mechanism

In the past, we developed mobile robot platforms for teaching purposes. Despite this, we still have some higher-level development and use such as image processing, but in the process, we found that the commonly used control boards such as Raspberry Pi or Jetson Nano are relatively insufficient in computing power or scalability, so we Use x86 mini-PC as the control core of the robot. Fig. 1 is a block diagram of the robot system. Through this architecture, we make the development and modification of robots easier. This has always been our goal. We hope to use a similar platform as the control core of robots with different architectures.

Fig. 2(a) is the mobile robot teaching platform we developed. This robot is only 130mm*130mm*150mm, and its height is only 90mm without LiDAR (Light Detection and Ranging). After we develop this platform, we hope that it can be used not only in teaching but also in actual industry.

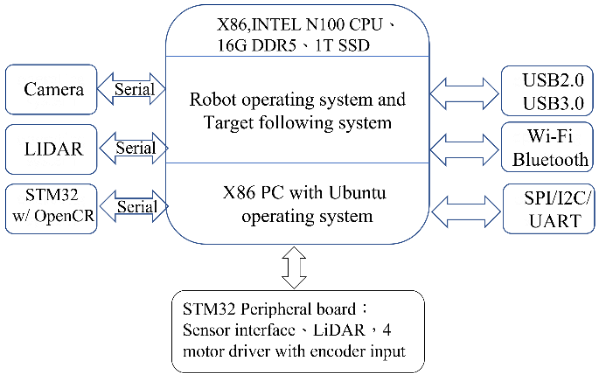

Fig.1 Block diagram of the robot control system.

a. Teaching robot.

b. Remote controlled lawn mower robot.

Fig. 2. Robotic platform.

There just happened to be a manufacturer of agricultural lawn mowers that wanted to transform their products from traditional remote-controlled to fully automatic lawn mowers. Fig. 2(b) shows the original remote control lawn mower. This original lawn mower uses a DC motor, but there is no encoder or related sensors. The lawn mower needs to be driven manually, so a considerable amount of work has been done on the mechanism and electronic control modification, Fig. 3 shows some of the modified areas. After the lawn mower modification was completed, the overall performance was very good after the commissioned manufacturer tested it. Therefore, the manufacturer commissioned us to modify the leaf sweeper. The general structure of this leaf sweeper is similar to that of the lawn mower, but the driving part uses 4 motors and wheels. Due to the systematization of the architecture and ROS parameters we use, the modification of the leaf sweeper was completed with only simple modifications. Fig. 4 shows the lawn mower and leaf sweeper after the modification. After completing the hardware modification of the two agricultural robots, we also developed the relevant human-machine interface and enabled the grass-cutting robot and the leaf-sweeping robot to cooperate with each other to complete the task. Next, we will explain the human-machine interface and software experience algorithm.

Fig.3 Block diagram of the robot control system.

a. Teaching robot.

b. Remote controlled lawn mower robot.

Fig. 4. Agricultural robots based on ROS.

2.2. *Group Task Algorithm*

Since ROS is a development system based on a single robot, it is relatively lacking in multi-robot interaction, data exchange, and task scheduling. We plan to adopt a sequential single-item auction [5]. We define each robot as R_i , and each job or path is defined as $W=\{W_1,W_2,......,W_n\}$, and the time each W takes the robot to take is defined as Tn, then we can define the cost of each robot as $P(R_i,W_{i,n}), i=1\cdots k$. In addition, we define an $Q=\{Q_1,Q_1,......,Q_n\}$ indicator to evaluate the efficiency of each W, on different robots, that is, different robots such as wheel type, carrying capacity, movement speed, etc. The higher the value of Q, the more suitable the robot is for this W. And the value of Q ranges from (1% to 100%), and the detailed definition of Q can be freely set according to the characteristics of the robot. Then we can rewrite the above robot cost definition as $P(R_i,W_n(R_i)Q_n)$. In addition, we also define a as the additional cost when the robot executes W under different conditions. Then we can define the time cost when the robot executes W without changing tasks midway and with changing tasks midway as Eq. (1);

$$\begin{cases} if\, change T\; P(R_i, W_n(R_i)Q_n) = P(R_i, W_n(R_i)Q_n) + E \\ if\, not\; change T\; P(R_i, w_n(R_i)Q_n) = P(R_i, w_n(R_i)Q_n) \end{cases} \quad (1)$$

Therefore, the evaluation function of the robot when performing work is:

$$MINSUM : \min_{W} \sum_{i=1}^{n} P(R_i, W_n(R_i)Q_n) \quad (2)$$

$$MINMAX : \min_{W} \max_{i} P(R_i, W_n(R_i)Q_n) \quad (3)$$

This function is also used to evaluate whether the robot should change or increase W during the execution of W. We define a collision space function $G = (C, N)$, where are all the task points, so $N = \{N_1, N_2, ..., N_n\}$, and we define an array S , which is the distance between two task points $S\{C_1, C_2, ..., C_n\}$. Then the optimal W allocation can be written as:

$$\min \sum_{i=1}^{n-1} C_i x_i \quad (4)$$

But subject to;

$$\sum_{i=1}^{n} x_i = 1 \quad (5)$$

$$x_i \in \{0,1\} \qquad (i = 1,2,......,n) \quad (6)$$

Because x_i is defined as whether each new W is the shortest path. Finally, we can rewrite the new MINSUM function as;

$$MINSUM : \min_{T} \sum_{i=1}^{n} P'(R_i, W_n(R_i)Q_n) \quad (7)$$

In this way, the various characteristics and paths of the robot can be taken into account, and the path or task can be updated quickly. Therefore, when necessary, a data center can be built to store the location of each robot in the environment, the environment map, the load The robot can update data at any time to monitor the status of the robot, and the data center can also transmit the calculation results to the robot. Through such a mechanism, each robot can complete all tasks quickly, and the task scheduling of agricultural robots can complete the tasks assigned by farmers to the robots in the simplest way.

2.3. *Human Machine Interface*

The agricultural robot in this study uses a web-based human-machine interface. The reasons for using this method are: (1) The operating interface is popular, and both WINDOWS, LINUX, and Mac OS have the ability to use web-based interfaces. Based on the application, and simple to use, almost everyone can do it. (2) 2D, 3D and multimedia technologies. In the past, due to the limitations of related web technologies, it was very difficult to display 3D images or multimedia and other complex visual effects on web interfaces. With the advancement of technology, the above All problems have been solved, so users can see the real-time working environment status, or simulate and operate in 3D simulation. (3) System integration is simple and has good security. Web-based is a standard network protocol application based on TCP/IP, so it is quite easy to connect with other systems. In terms of security, such as SSL (Secure Sockets Layer) , AES (Advanced Encryption Standard) can provide a very high level of data communication security protection, and is also widely used by government agencies, banks and other websites. Therefore, web-based is a pretty good platform.

This study uses Rosbridge API to develop the human-computer interface and converts graphical interface software such as rqt_graph and RViz in ROS to be displayed and operated on the Web. In the multi-robot path planning and intelligent task scheduling part, the robot can transmit the environment and robot status to the server at any time while moving, and then integrate the multi-robot environmental data into the database through the server. , and then perform dynamic path analysis and planning. In addition to expanding the robot's map data, it can also perform dynamic and more complete path planning and prediction, as well as group task scheduling for multiple robots.

For the web-based ROS human-machine interface, although ROS provides the Rosbridge API, which also has a websocket tool, how can the outgoing messages be integrated with the web interface, and how can the operations on the web interface be transmitted to ROS via Rosbridge? , how to display and integrate graphical tools in ROS, these are not done by Rosbridge alone, so this study uses C++, HTML, JavaScript and CSS to develop a web-based human-computer interface, among which C++ is used to Rosbridge compiles the function packages needed by the robot in the ROS workspace, and is also used to process the messages sent back by Rosbridge from ROS into data structures that can be used by web pages. HTML is the basis for writing the entire web page, including basic web page elements such as text, pictures, buttons, etc. JavaScript is used to process ROS data and allow web pages to have richer element control. CSS optimizes the color, layout, font size, etc. of a web page to make it more beautiful and user-friendly.

Fig. 5 shows the screen of a handheld device. Since the screen of the handheld device is relatively small, only a portion of the screen is displayed. However, other functions can still be switched through the window. Fig. 6 shows the screen as seen on a computer. Because the computer screen is larger, the robot's operation and related windows can be seen simultaneously. These functions are completed through a web browser.

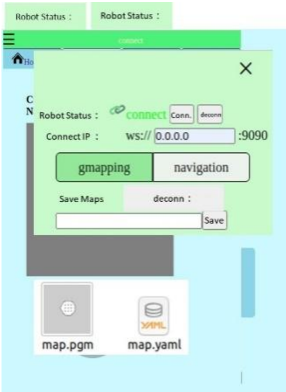

Fig.5 Block diagram of the robot control system.

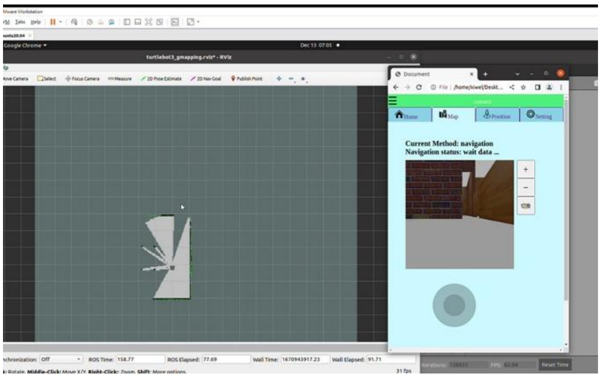

Fig.6 Block diagram of the robot control system.

3. Conclusion

We use ROS to convert lawn mowers and leaf sweepers that are already on the market into mobile robots that can operate autonomously. And also developed the Web base operation interface. After actual testing, the robots were able to operate normally and cooperate with each other to complete tasks. After this development experience, we also found that although ROS is very convenient, the related parameters and modifications are still too complicated. Therefore, we also developed a preliminary prototype so that different robot architectures and functions can be used in the future. This is done using the Web base method. In the future, we will bring this up again together with related tasks and map management for discussion.

Through this development, we have developed an agricultural robot that can really work on the farm. This result is also a beginning for us. We hope that in the future we can develop more agricultural robots to solve agricultural problems. The problem of insufficient manpower. We also try to simplify the development and use of robots. Finally, I hope that robots can truly serve people.

References

1. GILLER, Ken E., et al. The future of farming: Who will produce our food?. Food Security, 2021, 13.5: 1073-1099.
2. KASSIM, Mohamed Rawidean Mohd. Iot applications in smart agriculture: Issues and challenges. In: 2020 IEEE conference on open systems (ICOS). IEEE, 2020. p. 19-24.
3. CHUNJIANG, Z. H. A. O., et al. Agricultural robots: Technology progress, challenges and trends. *Smart Agriculture*, 2023, 5.4: 1.
4. OLIVEIRA, Luiz FP; MOREIRA, António P.; SILVA, Manuel F. Advances in agriculture robotics: A state-of-the-art review and challenges ahead. *Robotics*, 2021, 10.2: 52.
5. GUO, Jr-Hung. Applications of the Internet of Things technology in advanced planning systems. Sensors and Materials, 2018, 30.8: 1723-1734.

Authors Introduction

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.